

LVDC Equation Defining Document (EDD) for the Saturn IB Flight Program

Parts III and IV

(Complete Reprinting: ASTP Mission)

Revision K

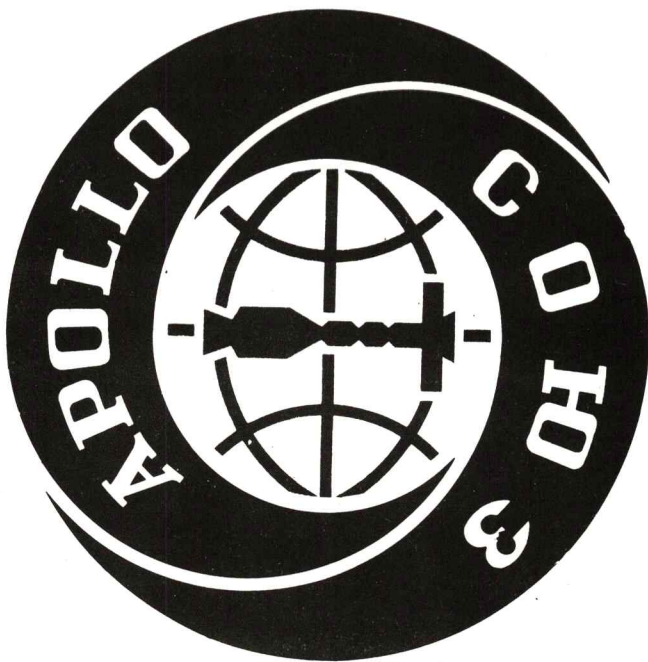
25 March 1975

Contract No. NAS8-14000

ASTP-DRL-686

Line Item 020

IBM No. 74W-00288



IBM

Federal Systems Division, Space Systems/Huntsville, Alabama 35810

Contract No. NAS8-14000
ASTP-DRL-686
Line Item 020
IBM No. 74W-00288

Prepared for the
GEORGE C. MARSHALL
SPACE FLIGHT CENTER
Huntsville, Alabama

LVDC Equation Defining Document (EDD) for the Saturn IB Flight Program

Parts III and IV
(Complete Reprinting: ASTP Mission)

Revision K

25 March 1975

Classification and Content Approval

Robert W. Drury

Data Manager Approval

Henry Kocbowski

Program Office Approval

J-E LQ

PART III

TABLE OF CONTENTS

Paragraph	Title	Page
N/A	EDD CONFIGURATION CONTROL PAGE LISTING	iv
SECTION 1:	FLIGHT PROGRAM STRUCTURE AND ORGANIZATION	--
1.1	STRUCTURE AND ORGANIZATION	III-1-1
1.2	CONTROL PROGRAM	2
1.2.1	<u>Mission-Phase Initialize</u>	4
1.2.1.1	Mission Executive	4
1.2.1.2	Phase Initializer	7 *
1.2.2	<u>Basic Iterative Cycle</u>	7
1.2.2.1	Non-Interrupt Sequencer	9
1.2.2.2	Periodic Processor	9
1.2.3	<u>Interrupt Processor</u>	12
1.2.3.1	Timer 1 Scheduler	14
1.2.3.2	Timer 2 Scheduler	16
1.2.4	<u>Queue/Control Tables</u>	18
1.2.4.1	Sequential Queue Table	19
1.2.4.2	Periodic and Timer 2 Tables	19
1.2.4.3	Phase Selection Table	23
1.2.4.4	Sequential Control Tables	23 *
1.3	COMMON COMMUNICATIONS AREA	26
1.4	APPLICATION MODULES	27
1.5	FLIGHT PROGRAM CONFIGURATION	28
1.5.1	<u>Program Macro Definition</u>	28
1.5.1.1	Mission Macros	29
1.5.1.2	Phase Macros	29
1.5.1.3	Interrupt Macros	29
1.5.2	<u>General Flowchart</u>	35
SECTION 2:	APPLICATION MODULES DESCRIPTION	--
2.1	INTRODUCTION	III-2-1
2.1.1	APS ATTITUDE CONTROL TEST (AC)	4a
2.2	ACCELEROMETER PROCESSING (AP)	5
2.3	ACCELEROMETER READ (AR)	14
2.3.1	ALTERNATE SEQUENCE REQUESTER (AS)	19a
2.4	ARCTANGENT UTILITY (AT)	20
2.5	CHI COMPUTATIONS (CC)	23
2.6	PHASE II CONTROL (CM)	26
2.7	CUTOFF LOGIC (CO)	31
2.8	COMPRESSED DATA TELEMETRY (CT)	34
2.9	DISAGREEMENT BIT PROCESSING (DG)	38
2.10	DISCRETE OUTPUT REGISTER SET/RESET UTILITY (DO)	43

2.11	DISCRETE PROCESSOR (DP)	46
2.12	DISCRETE INPUTS 14 AND 15 (DP1)	49
2.14	DISCRETE INPUT 9 (DP3)	56
2.15	DIGITAL COMMAND SYSTEM (DCS) (DS)	59
2.16	MEMORY DUMP TELEMETRY (DT)	81
2.17	F/M CALCULATIONS (DV)	85
2.18	ETC/BTC (EB)	89
2.19	MINOR LOOP ERROR MONITOR (EC)	94
2.20	EVENTS PROCESSOR (EP)	100
2.21	FIXED POINT DIVIDE UTILITY (FD)	103
2.22	ABSOLUTE/COMMAND FREEZE UTILITY (FR)	106
2.23	PREFLIGHT G&C STEERING LINKAGE (GC)	109
2.24	GIMBAL ANGLE RTC TEST UTILITY (GF)	113
2.25	ACQUISITION GAIN/LOSS (GL)	116
2.26	GRAVITATION ACCELERATION (GR)	120
2.27	PREFLIGHT DCS LINKAGE (HD)	124
2.28	S-IVB CUTOFF PREDICTION (HS)	129
2.29	ITERATIVE GUIDANCE MODE (IG)	134
2.30	PHASE INITIALIZATION (IN)	152
2.31	NATURAL LOGARITHM UTILITY (LN)	165
2.32	LIFTOFF SEARCH (LO)	168
2.33	LADDER RAMP PROCESSOR (LR)	171
2.34	MINOR LOOP (ML)	176
2.35	MATRIX MULTIPLY UTILITY (MM)	196
2.36	ORBITAL GUIDANCE MANEUVER PROCESSOR (MP)	199
2.37	MINOR LOOP SUPPORT (MS)	202
2.38	MATRIX TRANSPOSE UTILITY (MT)	209
2.39	BOOST NAVIGATION (NE)	212
2.40	NAVIGATION EXTRAPOLATION (NI)	218
2.41	NAVIGATION UPDATE IMPLEMENTATION (NU)	221
2.42	ORBIT NAVIGATION (NZ)	225
2.43	ORBIT ACCELERATIONS (OA)	236
2.44	DIGITAL OUTPUT MULTIPLEXER (DOM) TELEMETRY (OD)	241
2.45	ORBITAL GUIDANCE (OG)	244
2.45.1	ORBIT VELOCITY CALCULATIONS (OV)	256a *
2.46	PHASE ACTIVATOR (PA)	257

2.47	ONE SECOND DATA COMPRESSION (PC01)	260
2.48	FIFTY SECOND DATA COMPRESSION (PC50)	263
2.49	SIXTY SECOND DATA COMPRESSION (PC60)	267
2.50	DATA COMPRESSION POSITION UTILITY (PC70)	270
2.51	DATA COMPRESSION STORE UTILITY (PC80)	273
2.52	ONE HUNDRED SECOND DATA COMPRESSION (PC99)	276
2.53	SIMULATED PLATFORM GIMBAL ANGLES (PG)	280
2.54	ROLL GUIDANCE (RG)	285
2.55	SIMULATED ACCELEROMETERS (SA)	288
2.56	SINE-COSINE UTILITY (SC)	293
2.57	SCALED DIVIDE UTILITY (SD)	297
2.58	STEERING MISALIGNMENT CORRECTION (SM)	300
2.59	SQUARE ROOT UTILITY (SR)	303
2.60	SWITCH SELECTOR PROCESSOR (SS)	306
2.61	M/F SMOOTHING (ST)	330
2.62	TIME BASE SET (TB)	333
2.63	TIME BASE 1 (TB1)	336
2.64	TIME BASE 2 (TB2)	339
2.65	TIME BASE 3 (TB3)	343
2.66	TIME BASE 4 (TB4)	346
2.66.1	TIME BASE 5 (TB5)	350a
2.67	REAL TIME TELEMETRY DELAY UTILITY (TR)	351
2.68	SIMULTANEOUS MEMORY ERROR RECOVERY (TS)	354
2.69	TIME TILT GUIDANCE (TT)	357
2.70	PHASE II TIME UPDATE (TU)	361
2.71	PHASE I TIME UPDATE (UM)	364
2.72	VECTOR CROSS PRODUCT UTILITY (VC)	367
2.73	VECTOR DOT PRODUCT UTILITY (VD)	370
2.74	WATER VALVE PROCESSOR (WM)	373
SECTION 3:	VARIABLE CROSS REFERENCE	III-3-1
SECTION 4:	REQUIREMENTS/MODULE CROSS REFERENCE	III-4-1

PART III

EDD CONFIGURATION CONTROL PAGE LISTING, REVISION H

NOTE: The following Flight Program Change Requests (FPCR's) are authorized and incorporated in Revision K:

FPCR's 210-1 thru -14

(Revision K incorporates a complete reprinting of Parts III and IV of the EDD; pages revised are identified with a "Rev. K" notation in the lower-right corner of each page.

SECTION 1

FLIGHT PROGRAM STRUCTURE AND ORGANIZATION

1.1 STRUCTURE AND ORGANIZATION

The flight program which is structured modularly, is composed of two basic subsystems: the control subsystem and the application subsystem. The control subsystem controls the sequence and order of execution of all programmed functions in the application subsystem.

The application subsystem provides a master pool of all defined LVDC functions which can be used to generate a total flight program configuration for any given mission task. It is composed of a collection of relatively independent, closed program modules. Each application module is designed to perform one or a number of related functions; e.g., accelerometer readings, chi computations, orbital navigation. As new requirements are defined, new modules may be added to the application subsystem. The operational linkage, sequence, and execution of the application modules for a given mission are governed by the Control Program.

The control subsystem consists of the Control Program and a common communications area which allows for centralization of inter-module communications, common data and indicators, and mission or vehicle dependent parameters. The Control Program, composed of sub-programs and tables, controls the execution of application modules, services interrupts and routes control to the appropriate application module on a priority basis, and provides utility operations.

At flight program assembly time, the application modules and their mode of operation, sequence, and time frame of execution are specified via system macros. The LVDC Assembler automatically selects the specified modules from the application subsystem and appropriately links them to the Control Program and the common communications area. As requirements are changed and new modules are defined, the flight program configuration may easily be changed at assembly time by identifying different or additional modules.

Hence, the modular construction and control concepts of the program provide a high degree of flexibility and ease of maintenance in order to minimize overall impact and response time for modifications due to varying mission requirements.

1.2 CONTROL PROGRAM

The Control Program is a group of interacting sub-programs whose primary purpose is to control the sequencing and scheduling of the execution of individual application program modules to meet the operational requirements of a given LVDC mission. It examines interrupts and priority control tables to determine order of module execution and provides a means of transferring operational control to and from the modules. The design and organization of the Control Program is based on the use of sets of master queue/control tables by which the application requirements of any given mission can be specified.

The flight program task is divided into distinct mission segments or phases according to different overall vehicle modes of operation throughout the mission. A phase includes all functions required to be accomplished within a certain time frame of the mission such as powered flight or orbital operation. The number of phases and functions within a phase are defined via system macros independent of the Control Program. Functions within a mission phase must be executed at different frequency rates with different timing precisions. Thus, modules, according to their particular timing requirements, are assigned via system macros to operate under a specific sub-program of the Control Program. The structure (see Fig. 1-1) of the Control Program consists of the following sub-programs:

- Mission Executive: Governs phase-to-phase transition throughout the entire mission.
- Phase Initializer: Initializes master queue/control tables to proper status at the start of each defined mission phase.

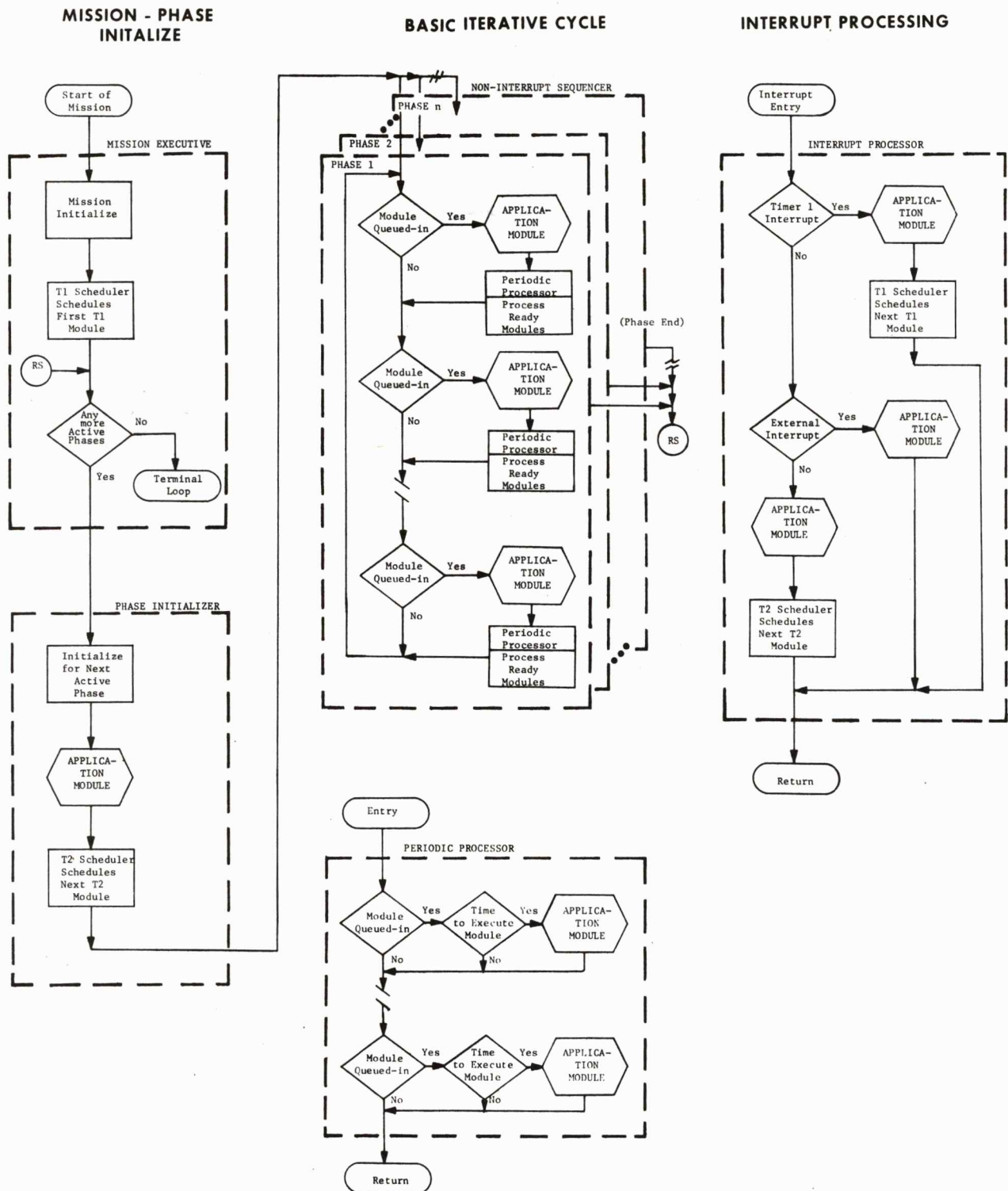


Figure 1-1 Control Program Structure

- Non-Interrupt Sequencer: Controls the execution of those modules which make up the basic iterative computation cycle of any phase, i.e., those functions of a non-priority nature which are to be operated as a part of a repetitive, ordered sequence.
- Periodic Processor: Controls the execution of those modules of a non-priority nature but which are to be executed a given number of times within a specified, but not precisely rigid, time frame.
- Interrupt Processor: Services both hardware and program controlled interrupts and routes control to the appropriate program module on a priority basis.
- Timer 1 Scheduler: Schedules the use of the Timer 1 interrupt to control the execution of priority modules which require operation at an exact time or at a precisely cyclic rate.
- Timer 2 Scheduler: Schedules the use of the Timer 2 interrupt to control the execution of priority modules of a lower order than those under Timer 1 but which also require operation at a specific time or rate as precisely as possible.

1.2.1 Mission-Phase Initialize

The Mission Executive performs general system initialization, determines which phases of the specified mission are to be executed, selects required control words for phase initialization, gives control to the Phase Initiator to initiate phase execution, and performs all required system terminal housekeeping at mission end. The Phase Initiator sets the module queue status for the selected phase, executes the phase initialization module, and schedules the next Timer 2 operation before passing control to the Non-Interrupt Sequencer.

1.2.1.1 Mission Executive

Upon initial entry, the Mission Executive (see Fig. 1-2) inhibits all interrupts, initiates real time clock readings and initializes accumulated time

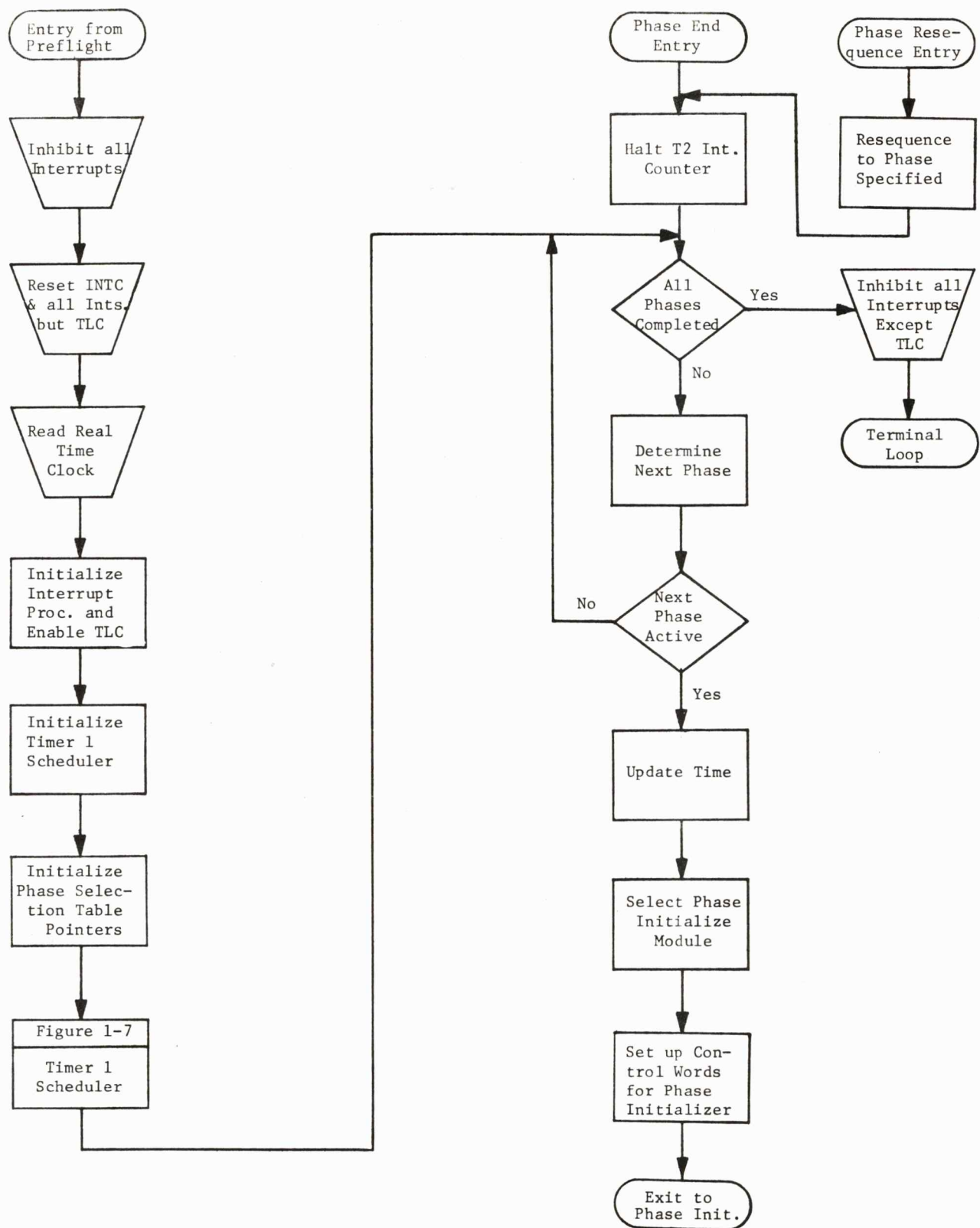


Figure 1-2 Mission Executive

in mission and time base reference. The program then performs all initialization required for the proper operation of the Interrupt Processor and the Timer 1 Scheduler.

The Timer 1 Scheduler is called to initiate operation of the Timer 1 interrupt counter. Control is then returned to the Mission Executive for phase selection. The Timer 1 interrupt inhibit is not removed until the completion of first phase initialization.

The Mission Executive determines the next phase to be executed by means of a special active phase indicator word. This indicator consists of individual bits arranged in chronological order for each phase defined. Each bit indicates whether or not the corresponding phase is to be executed. At program generation time, all phases defined are set active. If there are no more active phases to be executed, the Mission Executive will perform all required terminal housekeeping functions and enter a terminal loop. If any active phases remain to be executed, the next in order is selected. Control words associated with the phase that is to be executed are obtained from the Phase Selection Table and supplied to the Phase Initializer. The Mission Executive then gives control to the Phase Initializer and is not re-entered again until phase execution is completed. Since phase termination will normally be dependent upon a particular function or calculation performed by some application module, it will be the responsibility of that application module to accomplish Mission Executive re-entry at phase end.

Two entry points are provided to terminate a phase. One entry point allows a specific phase to be selected whereas the other entry automatically selects the next chronological phase. Upon phase end entry, the Mission Executive halts the Timer 2 interrupt counter to avoid any Timer 2 interrupt scheduled but not completed in the previous phase. Timer 1 interrupts are allowed to operate during next phase selection and initialization. The program then proceeds to select the next phase or to enter the terminal loop.

1.2.1.2 Phase Initializer

Upon entry from the Mission Executive, the Phase Initializer (Fig. 1-3) uses control information supplied to it to set the master queue tables to the initial queue status specified for the selected phase. The information also provides the linkage to the appropriate application initialize module and to the Sequential Control Table for that phase. After all queue tables are initialized, the Phase Initializer executes the user application initialize module which returns control upon completion. The Timer 2 Scheduler is then called to initiate operation of the Timer 2 interrupt counter. At completion of Timer 2 scheduling, control is passed to the Non-Interrupt Sequencer to control the execution of the application modules which comprise the basic iterative cycle.

1.2.2 Basic Iterative Cycle

The basic iterative cycle consists of the Non-Interrupt Sequencer and Periodic Processor sub-programs, both of which are essentially executable tables. These sub-programs control the sequence and time of execution for the majority of the application modules.

The Non-Interrupt Sequencer determines if a module under its control is to be executed, and if so, accesses the module and the Periodic Processor. The Periodic Processor determines for modules under its control whether they are to be executed, and if so, if it is time for their execution. When the Non-Interrupt Sequencer has examined each module under its control, it recycles to the first module and repeats the process; hence, the name Basic Iterative Cycle. The iterative cycle, or computation cycle, is terminated when some application module returns control to the Mission Executive at phase end.

The Non-Interrupt Sequencer and Periodic Processor determine whether a module is to be executed by means of its queue status. The system queue tables specify for each application module whether the module is queued-in or queued-out. A module which is queued-in will be executed at the proper time. A module which is queued-out will not be executed until it is

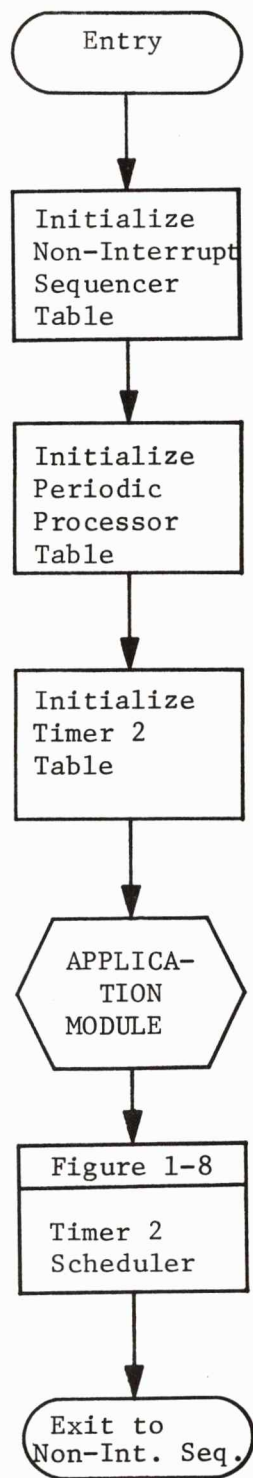


Fig. 1-3 Phase Initializer

queued-in. Changing a module's executable status is known as queuing. System macros are provided so that a module at any time may change its own or another module's queue status.

1.2.2.1 Non-Interrupt Sequencer

Essentially, the Non-Interrupt Sequencer (Fig. 1-4) consists of a number of tables: a Sequential Control Table for each phase of the mission and a single Sequential Queue Table for the total mission. Each control table is an executable table; i.e., it is made up of executable instructions. It is actually one of these tables that is given control at completion of phase initialization. The selected table will remain in control executing associated application program modules in an iterative loop until some user application module detects phase end conditions and effects re-entry to the Mission Executive.

Each entry in the Sequential Control Table picks up a specific entry from the Sequential Queue Table and checks to see if it is queued-in. If the entry is queued-in, the corresponding application module is executed, and control is passed to the Periodic Processor. When the entry is not queued-in or when control is returned at the completion of periodic processing, the next control table entry in sequence is processed. After the last entry in the phase Sequential Control Table has been processed, a transfer to the beginning of the table is made and the processing procedure is repeated. This iterative loop will continue until some application module returns control to the Mission Executive at phase end.

1.2.2.2 Periodic Processor

The Periodic Processor (Fig. 1-5) is entered from the Non-Interrupt Sequencer after each application module execution. The Periodic Processor is basically an executable queue/control table, the Periodic Table. This table consists of entries to time tests which provide for servicing time-dependent, periodic application modules.

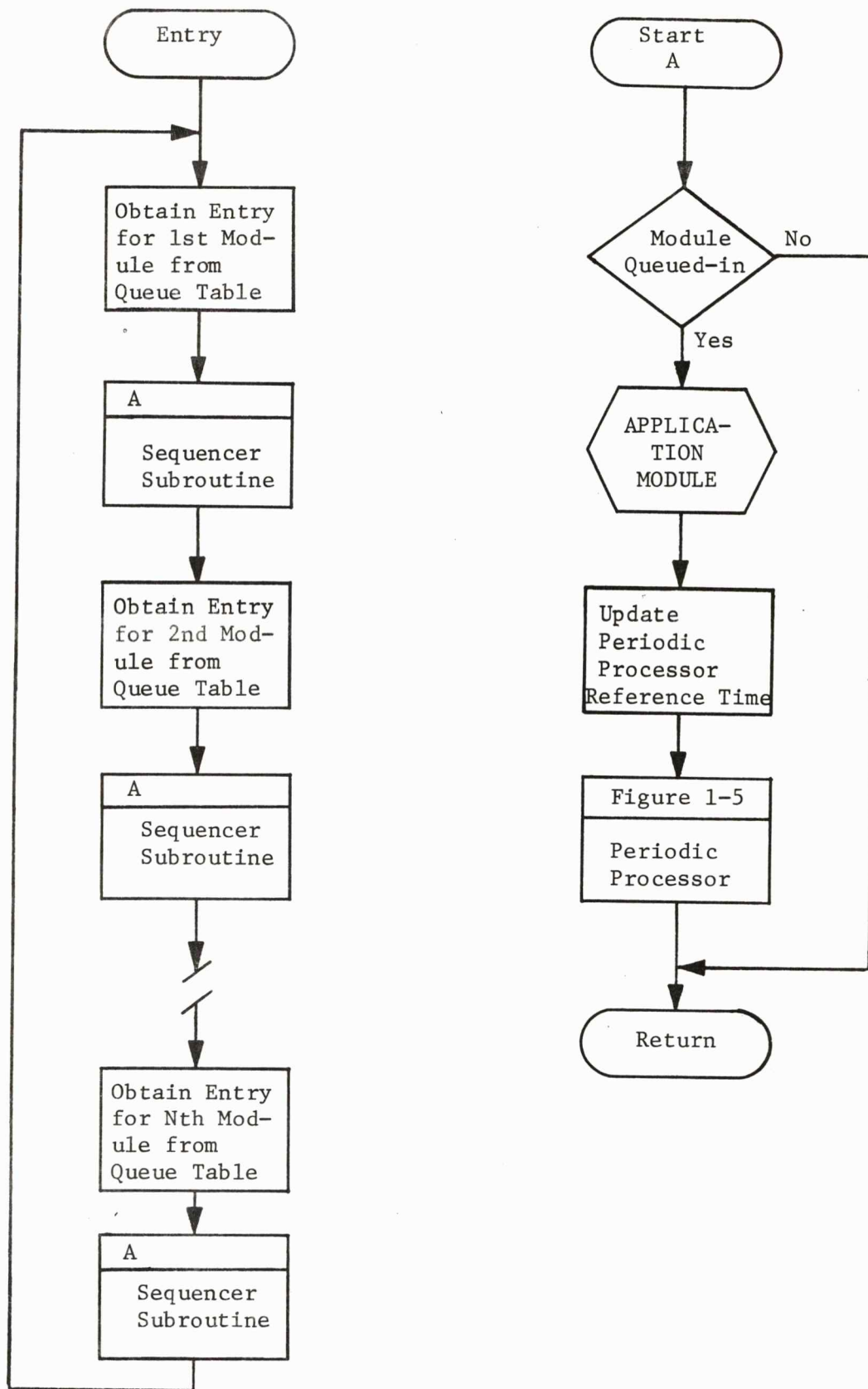


Fig. 1-4 Non-Interrupt Sequencer

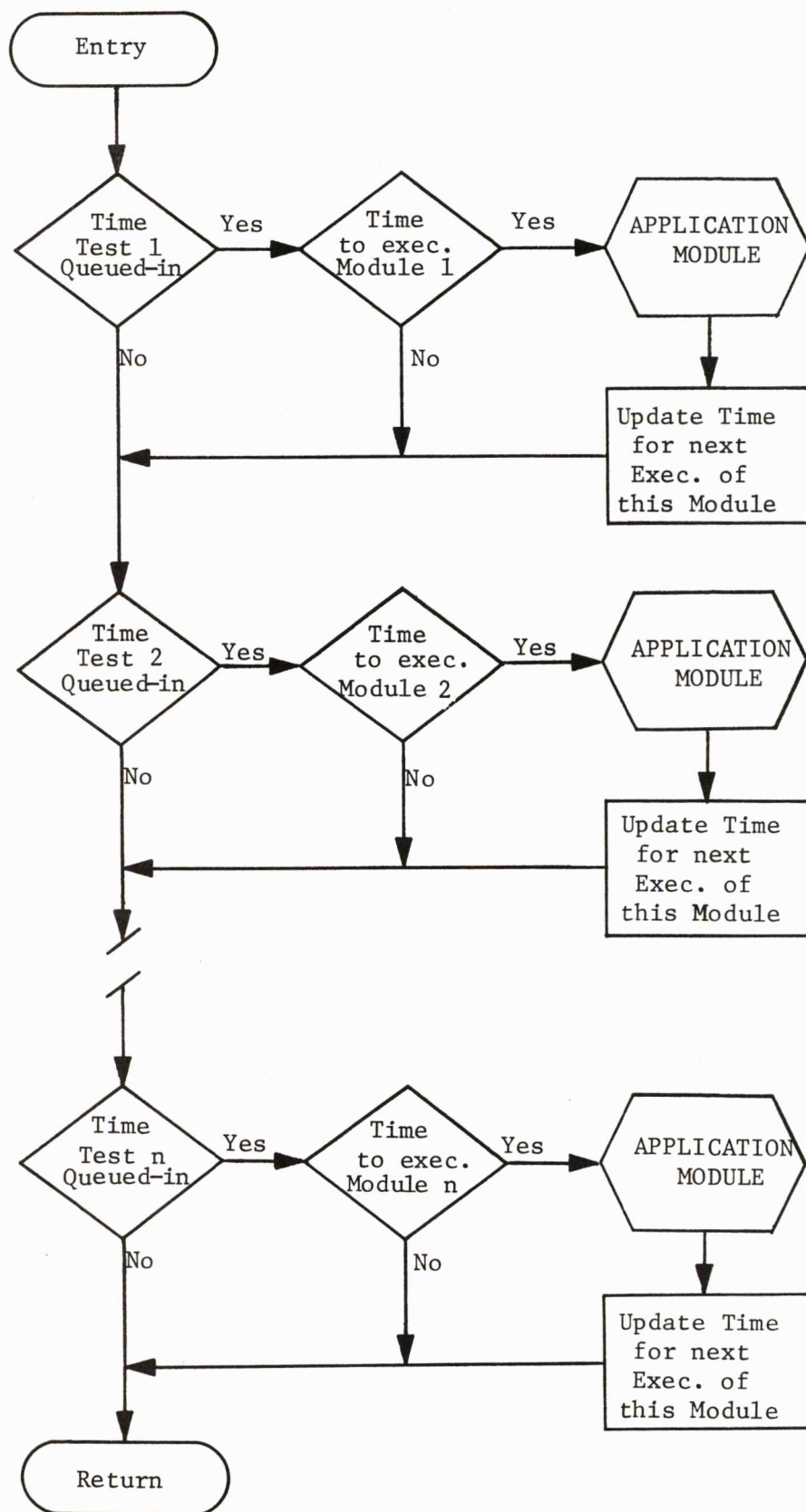


Fig. 1-5 Periodic Processor

Upon entry, the first item in the queue/control table is processed. If the entry is queued-in, an associated time test is made to determine whether or not a corresponding application module is to be called. If the time test is passed, the application module is executed and time for its next execution updated. The next entry in sequence in the queue/control table is then processed in the same manner. If a table entry is queued-out, no time test or application module will be executed before proceeding to the next entry. Likewise, if a time test is not passed, the corresponding application module will not be executed before proceeding to the next table entry. Control is returned to the Non-Interrupt Sequencer after all Periodic Table entries have been processed.

1.2.3 Interrupt Processor

The Interrupt Processor services all LVDC/LVDA interrupts under four basic levels of priority as follows:

Level 4: TLC (LVDC Simultaneous Memory Failure) interrupt.

Level 3: Timer 1 interrupt.

Level 2: External interrupts by order of assignment to the LVDA Interrupt Storage Register (ISR).

Level 1: Timer 2 interrupt.

These levels determine the order in which processing will occur whenever more than one interrupt requires servicing at the same time. They also indicate the degree of priority provided for associated application module processing (Level 4 is the highest priority.) See Fig. 1-6. Processing at any level may be interrupted and delayed until all required processing at the higher level is completed. All processing performed on a non-interrupt basis is effectively at level 0, the lowest priority level.

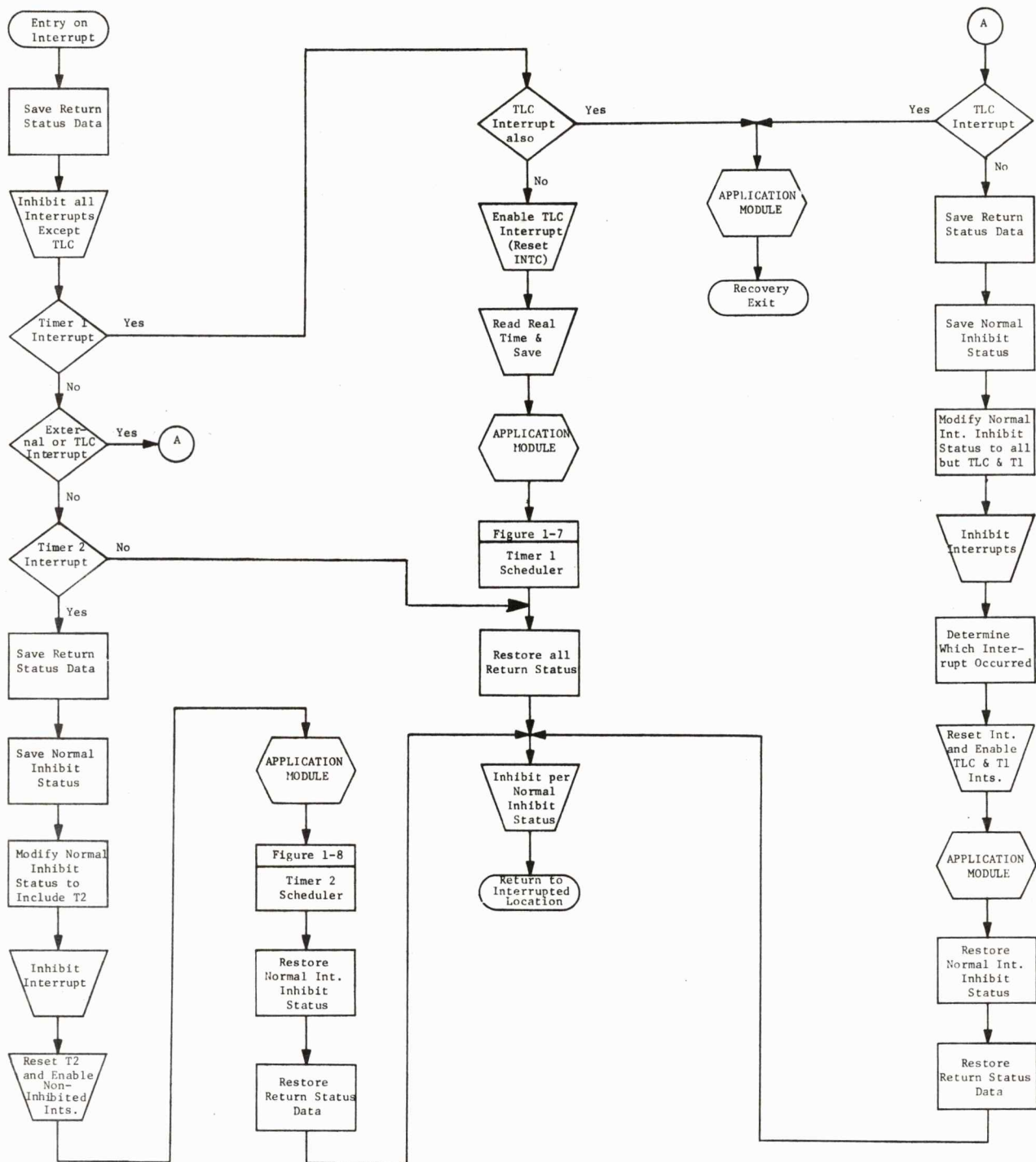


Figure 1-6 Interrupt Processor

The timer schedulers also make use of a priority scheme in selecting the next application modules to be serviced by the timer interrupts. This priority scheme, however, is only in effect when at least two application modules specify identical times for use of the same timer interrupt. In this case, the latter application module will be selected.

There are two basic classes of interrupts serviced by the Interrupt Processor: program controlled interrupts and external interrupts.

There are two program controlled interrupts: Timer 1 (formerly the minor loop counter) and Timer 2 (formerly the switch selector counter). Both of these counters operate at a 492.2 microsecond rate. Timer 1 is a continuously operating or wrap-around counter which will automatically generate an interrupt every 4.032 seconds (maximum counter value) if not under program control. Timer 2 is not a wrap-around counter. It will generate an interrupt only if initiated under program control.

These timers are used to service application modules requiring high timing accuracy on a requested or cyclic basis. The timer counters are controlled only by the system scheduler routines (one is provided for each timer) which determine when the next timer interrupts are to occur and the application modules to be serviced. The Interrupt Processor allows Timer 2 scheduling and the application modules serviced by Timer 2 to be interrupted by Timer 1. Timer 1, therefore, is intended for use by high priority functions while Timer 2 is for functions of a lower priority.

All LVDC/LVDA interrupts other than the timer interrupts are externally controlled; i.e., the interrupts generated are a result of the occurrence of an external event or hardware malfunction. All application modules serviced by these interrupts are required to return control to the Interrupt Processor upon their completion. The Interrupt Processor will then return control to the last interrupted location.

1.2.3.1 Timer 1 Scheduler

The Timer 1 interrupt is reserved for modules whose operational accuracy requirements demand stringent timing. The Timer 1 Scheduler (Fig 1-7) is

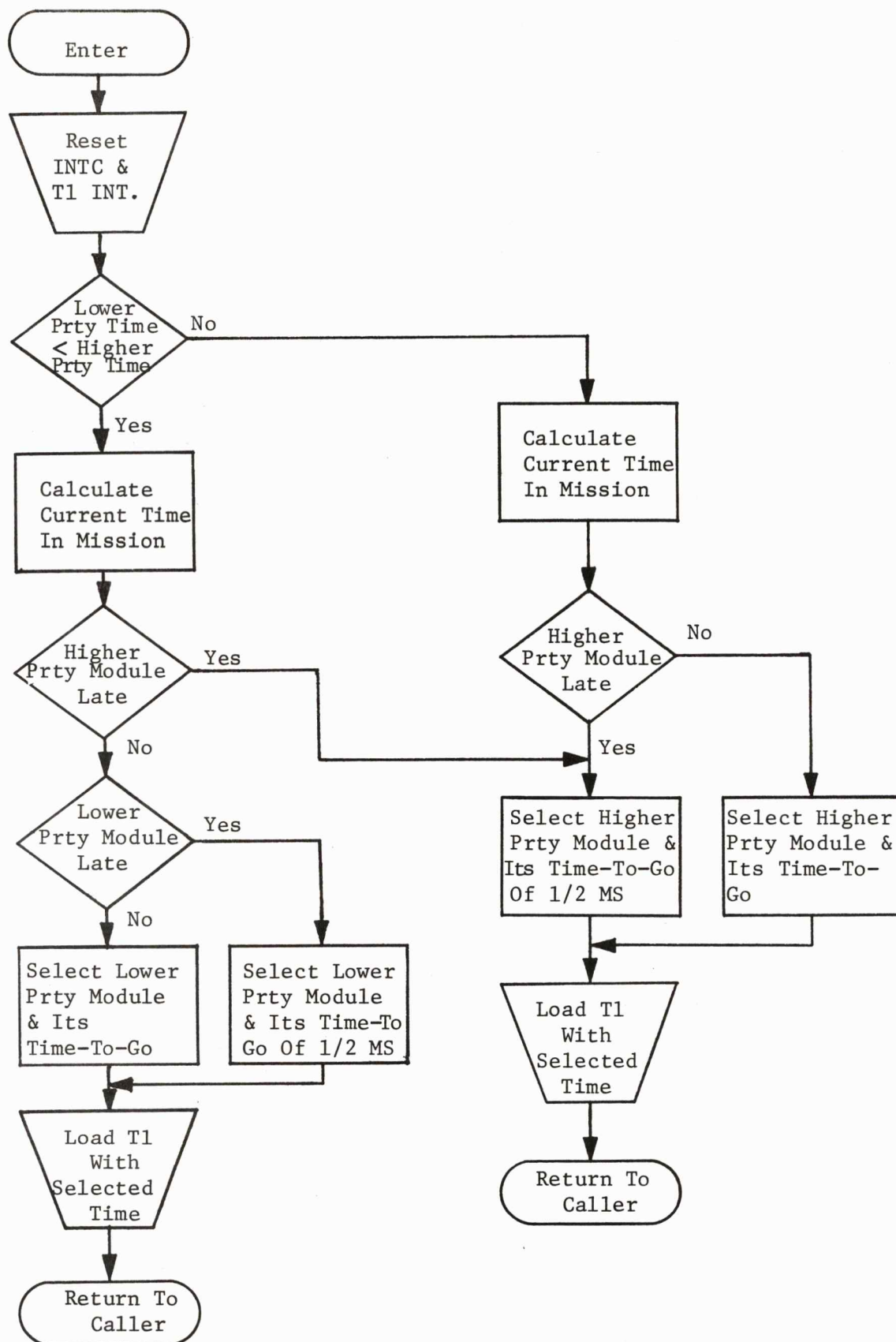


Fig. 1-7 Timer 1 Scheduler

designed to determine which of two modules is to execute next and to schedule the Timer 1 interrupt.

The Timer 1 Scheduler gives a higher priority to one module when both modules are scheduled for the same time or are late. The higher priority module (the Program Macro Definition sub-section describes the macro by which the priority is specified) is predominantly used for minor loop operations. The lower priority module is predominantly used for switch selector operations. The two application modules and/or their entry points may be changed by queuing operations from other modules.

The Timer 1 Scheduler is entered from the Interrupt Processor following the execution of a Timer 1 module or from an application module whenever a Timer 1 queuing operation is performed. Upon entry, the scheduler will decide which module requires the use of the next Timer 1 interrupt. Proper linkages for the execution of the selected module by the next Timer 1 interrupt are set-up, the Timer 1 counter is loaded with the associated time-to-go, and control is returned to the Interrupt Processor.

1.2.3.2 Timer 2 Scheduler

The Timer 2 Scheduler is automatically entered from the Interrupt Processor whenever any module serviced by the Timer 2 interrupt completes processing. See Fig. 1-8. It is also automatically entered whenever any Timer 2 queuing operation is performed. The Timer 2 Scheduler is basically an executable table, the Timer 2 Table. This table consists of entries to time tests which provide for servicing time-dependent priority application modules on a requested basis.

Upon entry, the scheduler first selects itself as a module to be executed four seconds later. This is done to maintain control of scheduling in case no application module requires the use of the Timer 2 interrupt within the maximum time delta (4.032 seconds) which the timer can accommodate. The scheduler then steps through the entries in the Timer 2 Table. Each queued-in time test determines whether or not the associated application module requires use of the interrupt before the currently selected module.

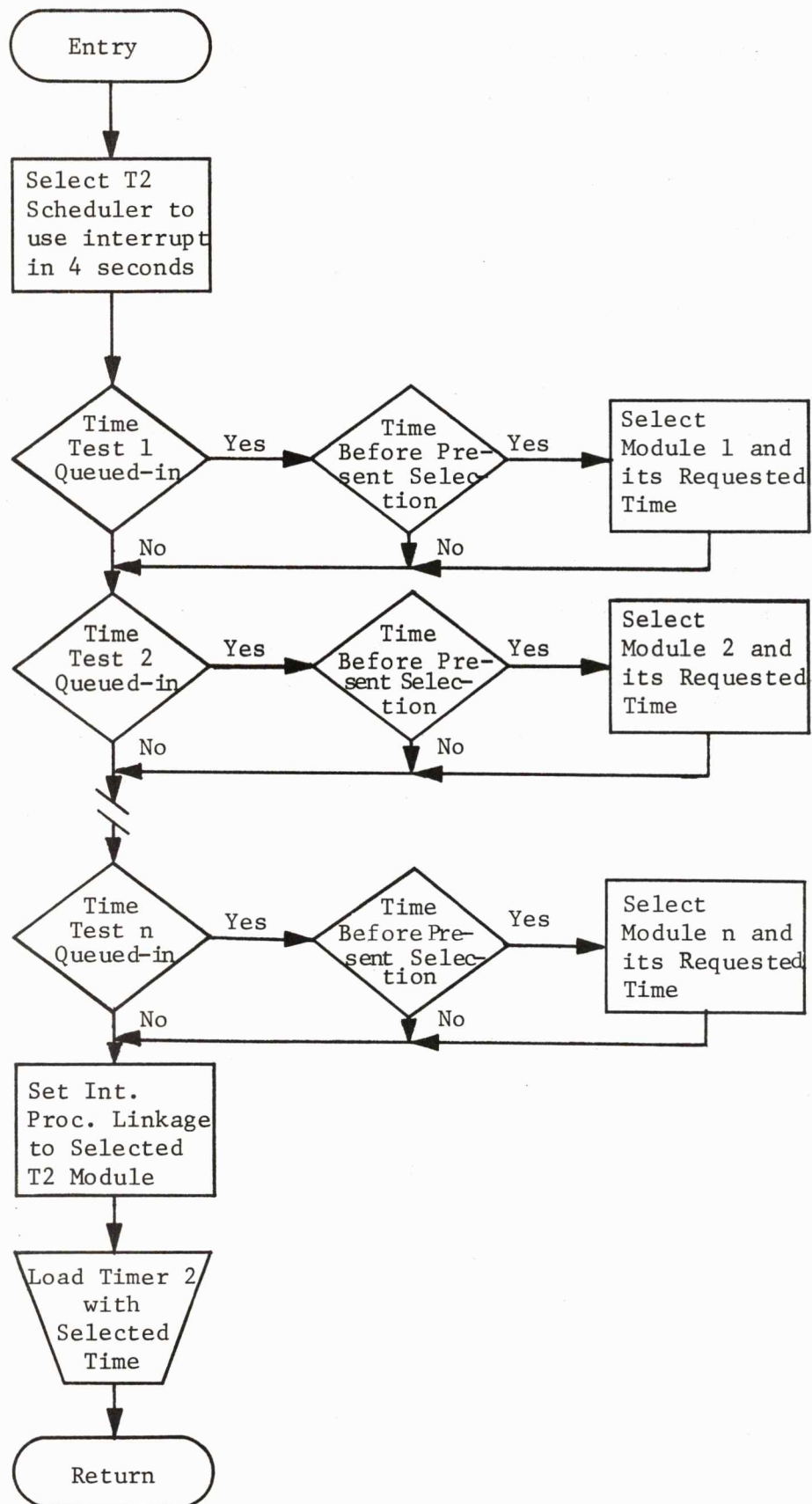


Fig. 1-8 Timer 2 Scheduler

If so, that module and its associated request time becomes the current selection. This process is repeated until all table entries have been checked. Linkage to the last selected module is set up for execution on the next Timer 2 interrupt. The Timer 2 counter is loaded with the associated request time, and control is returned to the calling routine.

1.2.4 Queue/Control Tables

The primary tasks of the Control Program consist of sequencing and/or scheduling the various application modules that make up the flight program. These functions are accomplished by queue/control tables set up by the programmer with the aid of system macros. (These macros are described in the Program Macro Definition sub-section). Through these tables, a mission and its operational phases can be completely defined to the Control Program. They provide the identification of functions, their associated time requirements, and initial queue (active or inactive) status. Furthermore, certain of these tables can be dynamically altered (queuing) during the program execution to reflect changing requirements due to phase-to-phase transition or as the result of external events or program calculations. The Control Program, then, directs and controls the operation of the total flight program basically through the use of master queue/control tables.

There are essentially two classes of system tables: mission tables, which are used throughout a mission, and phase tables, which are only used for a single mission phase.

There are four major mission tables:

- The Sequential Queue Table for the Non-Interrupt Sequencer
- The Periodic Table for the Periodic Processor
- The Timer 2 Table for the Timer 2 Scheduler
- The Phase Selection Table for the Phase Initializer

The first of these is a queue table and the next two are queue/control tables. This distinction is made because of the manner in which the tables are used. Queuing may be performed in these three tables but only the Periodic and the Timer 2 Tables are directly executed to provide control. The Sequential Queue Table is used as data by the phase Sequential Control Tables and is not directly executed. The last table, the Phase Selection Table, is neither queued nor executed but provides control variables and constants for each mission phase. *

There is only one type of phase table, the Sequential Control Table. However, there is a Sequential Control Table for each phase of the mission. Note that the term "control table" is used in keeping with the previous nomenclature convention. These tables cannot be queued but are directly executed to provide control.

1.2.4.1 Sequential Queue Table


The Sequential Queue Table (Table 1-1) consists of instructions which address the various sequential application modules to be executed. One entry is required for each module to be sequenced regardless of the number of phases in which sequencing is to be performed. The system provides for a maximum of 200 entries in this table. The order of the entries is irrelevant. Queuing is performed by altering the instruction operation code in the desired table entry.

1.2.4.2 Periodic and Timer 2 Tables

The queue/control tables for the Periodic Processor (Table 1-2) and the Timer 2 Scheduler (Table 1-3) are essentially alike. They differ only in the form of their respective time tests and will therefore be discussed together. Each entry consists of a linkage to a corresponding time test for the scheduling of time dependent application modules. One entry is required for each module to be scheduled. The system allows a maximum of 25 entries in each table. Again, queuing is accomplished by the alteration of the operation code in the desired table entry. Note that in these tables only a time test is being queued, not an actual application module.

Table 1-1 Sequential Queue Table


NSQTB

HOP/ADD*	to HPC for Module A
HOP/ADD*	to HPC for Module B
HOP/ADD*	to HPC for Module C
HOP/ADD*	to HPC for Module D
	
HOP/ADD*	to HPC for Module X
HOP/ADD*	to HPC for Module Y
HOP/ADD*	to HPC for Module Z

*Note: Queuing is accomplished by altering the operation code field in this table. A "HOP" indicates an active module and an "ADD" indicates an inactive module.

One word is required per entry.
A maximum of 200 entries is allowed for this table.


Table 1-2 Periodic Table

TRA/CLA*	to test for Periodic Module 1
TRA/CLA*	to test for Periodic Module 2
	
TRA/CLA*	to test for Periodic Module N
HOP	return to Sequencer Subroutine

*Note: Queuing is accomplished by altering the operation code field in this table. A "TRA" indicates an active test and a "CLA" indicates an inactive test.

One word is required per entry. A maximum of 25 entries is allowed for this table.

Table 1-3 Timer 2 Table

TRA/CLA*	to test for Timer 2 Module 1
TRA/CLA*	to test for Timer 2 Module 2
TRA/CLA*	to test for Timer 2 Module 3
	
TRA/CLA*	to test for Timer 2 Module N
HOP	Return to Timer 2 Scheduler

*Note: Queuing is accomplished by altering the operation code field in this table. A "TRA" indicates an active test and a "CLA" indicates an inactive test.

One word is required per entry. A maximum of 25 entries is allowed for this table.

A distinction regarding the ordering of entries does exist between the Periodic and the Timer 2 Tables. In the Periodic Processor, an application module is called directly by its associated time test (assuming that test was queued-in and that the time test was passed). This imparts a priority to the entries in the Periodic Table; the order of the entries in the table implies their priority of execution. In the Timer 2 Scheduler, the order of entries is somewhat irrelevant. The queued-in time tests select the application module (or the scheduler itself) to be executed next, strictly as a function of time. However, if two or more modules request simultaneous execution, the last module tested will be serviced first.

1.2.4.3 Phase Selection Table

The Phase Selection Table (Table 1-4) is generated as a by-product of the system macros and provides information for the initialization of each mission phase. Entries are made in the order in which the corresponding phases are to be executed.

1.2.4.4 Sequential Control Tables

These tables (see Table 1-5 for table format) consist of two instructions for each entry. The first instruction addresses a particular entry from the Sequential Queue Table. The second instruction is identical for all entries; it is linkage to a system subroutine. Essentially, the subroutine uses the entry from the queue table to determine whether or not the associated application module is to be executed.


A Sequential Control Table for any particular mission phase will contain at least one entry for each application module to be handled by the Non-Interrupt Sequencer during that phase. Multiple entries for a particular module can be used to provide a higher frequency (relative to the other modules) of execution for that module. Note that a priority is implied by the order of the entries in the Sequential Control Tables. The system will permit a maximum of 50 entries in each of these tables.

Table 1-4 Phase Selection Table

Phase 1	HPC to Application Program Initialization Module
	HPC to Sequential Control Table
	Pointer/counter for Sequential Control Table
	Optional queue word for Sequential Queue Table
	Queue word for Sequential Queue Table
	Queue word for Periodic Table
	Queue word for Timer 2 Table
Phase 2	(same as Phase 1)
Phase n	(same as Phase 1)

Note: One word is required for each entry in this table.
 Total size of this table is variable depending upon
 the number of phases defined and the occurrence of
 the optional queue word in each set of phase entries.

Table 1-5 Sequential Control Table

CLA	NSQTAB + n*
HOP	to Sequencer Subroutine
CLA	NSQTAB + k*
HOP	to Sequencer Subroutine
CLA	NSQTAB + r*
HOP	to Sequencer Subroutine
	
CLA	NSQTAB + x*
HOP	to Sequencer Subroutine
TRA	to beginning of table

*Note: NSQTAB is the address of the first location in the Sequential Queue Table (see Table 1-1) and n, k, r ... x are pseudo-variables indicating the relative position of the particular entry within that table.

Two words are required per entry which represents an application module to be sequenced during the phase. A maximum of 50 entries is allowed for this table.

One such table is required for each mission phase defined.

Flexibility is the primary reason for the multiple-table design of the Non-Interrupt Sequencer. The control tables provide the capability for altering the sequence of execution of application modules from phase to phase while the master queue tables provide the uniform queuing base for all phases.

1.3 COMMON COMMUNICATIONS AREA

A common communications area, the data pack, is provided in order to centralize all data communications between application modules and between the Control Program and application modules. The data pack identifies and defines all data used by the flight program. It is organized into groups segmenting telemetry identification, pseudo variables, data definitions, and table definitions.

Data which are to be telemetered are identified in a telemetry table via the TELD macro stating the data name, the mode setting, and the PIO tag. Actual data definition and memory address assignments are not done at this point but are included in the appropriate function data definition group.

All vehicle or mission dependent data parameters are defined by means of the "pseudo-variable" capability of the LVDC Assembler. This allows module operation to be independent of these change prone variables and constants. Also, status words, conversion factors, and certain control words are defined as pseudo variables. All left-hand labels for pseudo-variables begin with 'P.'. For example, P.M1 is the label of the pseudo-variable representing the mass of the vehicle at liftoff used to compute $(F/M)_c$.

Data definitions include targeting parameters, utility data, control data, and module associated data. Assignments to residual storage are based on number of references in order to conserve storage. Module associated data are grouped to identify data common to a small group of modules. Data definitions unique to individual modules are also isolated. Data definition labels begin with 'D.', 'V.', 'L.', 'F.', 'G.', or 'H.'. *

Table definitions include such tables as Switch Selector Tables, Event Processor Tables, and Ladder Ramp Tables.

1.4 APPLICATION MODULES

An application module is defined as a program unit exclusively designed and tailored to perform one or a number of related functions, i.e., each application module is function-oriented. The application module concept is intended to provide a source pool of program modules which can be used to generate different flight program configurations for any number of missions.

Module execution is governed by the Control Program either directly (the module is called by the Control Program) or indirectly (the module is called as a subroutine by another module already under direct control of the Control Program). When called, each module will perform its unique function and return control to the calling source. All inter-module communication, and all queuing of modules is to be accomplished through the system macros.

Two system macros are provided for modules to specify the queue status of a module and, if applicable, the time or rate of execution for that module. The E.QT macro specifies the time or rate of execution, and the E.Q macro sets the queue status of a module.

The E.QT macro is not valid for queuing Non-Interrupt Sequencer modules since their operation is cyclic rather than time associated. The time given in the E.QT macro for Periodic Processor modules is an execution rate. A module that is queued-in will be executed at the time interval specified. The time specified for Timer 1 and Timer 2 modules indicates a specific time the module is to be executed. When a Timer 1 or Timer 2 module gains control, it must reschedule itself or queue itself out before returning control to the Control Program.

For queuing Timer 1 modules only, the E.QT macro also queues-in the specified module. Timer 1 modules are queued-out indirectly by queuing-in another module via the E.QT macro or by specifying a time to execute which will never be reached.

The E.Q macro is used for queuing Non-Interrupt Sequencer, Periodic Processor, and Timer 2 modules. A time or rate of execution must be

specified for Periodic Processor and Timer 2 modules before the E.Q macro is used. The E.Q macro is invalid for Timer 1 modules. Table 1-6 summarizes the queuing functions for the different classes.

Table 1-6 Queuing Macros

Macro	Non-Interrupt Sequencer Modules	Periodic Processor Modules	Timer 2 Modules	Timer 1 Modules
E.QT	Invalid	Specifies rate of execution	Specifies time of execution	Specifies time of execution and queues-in
E.Q.	Queues-In/Out	Queues-In/Out	Queues-In/Out	Invalid

1.5 FLIGHT PROGRAM CONFIGURATION

The flight program is composed of the Control Program, the common communications area, and all of the application modules described in the Application Modules Description section. The configuration of the flight program is defined by the flight programmer via system macros. The macros define those modules which will execute under direct control of the Control Program and the controlling component under which each will execute.

1.5.1 Program Macro Definition

There are three categories of program definition macros. Mission macros provide module definitions and certain control information for the entire mission. Phase macros define the functions for each program phase. Interrupt macros provide the control information for interrupt processing.

The macro program definition is expected to remain relatively consistent from flight to flight, but new requirements may result in timing differences or in changing modules. Part IV will define the individual program definition when it varies from that specified in Table 1-7. The following paragraphs describe the three macro categories.

1.5.1.1 Mission Macros

There is one group of mission macros beginning with the E.MIN macro and ending with the E.MEND macro. This group must precede the other groups. Macros in this group define all modules (their entry points) which will execute under direct control of the Control Program during the mission and the controlling sub-program under which each will execute. In addition, the execution rate for those modules which will be controlled by the Periodic Processor are defined.

The queue status of all Timer 2 modules will be set as queued-in. The queue status can be overridden by the phase definition macros for the phase(s) in which the module is to execute or by another module. The queue status of the Non-Interrupt Sequencer modules is not defined via the mission macros since there is a separate control table for each phase. The Sequential Queue Table, the Periodic Table, and the Timer 2 Table are allocated as a result of these macros.

1.5.1.2 Phase Macros

There are two operational phases of flight recognized by the flight program:

- o Phase 1 Boost to Orbit From GRR to preset time in TB4
- o Phase 2 Parking Orbit From Phase 1 end to End of Mission (EOM)

Each phase definition begins with the E.PIN macro which specifies the phase initialization module and the number of Non-Interrupt Sequencer modules which will be allowed to execute in the phase. The E.PEND macro terminates each phase definition.

1.5.1.3 Interrupt Macros

This group consists of two major types which are concerned with interrupt processing. These macros may be specified anywhere between the E.MEND macro and the last E.PEND macro. The E.INTS macro specifies those external system interrupts which are to be active and the module which is to gain

Table 1-7 Program Macro Definition

1	SYMBOL 2	OPERATOR 8	ADDRESS 16

*	MISSION DEFINITION		
	E.MIN	2	number of phases defined
*	NON-INTERRUPT SEQUENCER MISSION TABLE		
		module entry point	
	E.MNS	M.AR00	ACCELEROMETER READ MODULE
	E.MNS	M.SA00	SIMULATED ACCELEROMETERS MODULE
	E.MNS	M.DP00	DISCRETE PROCESSOR MODULE
	E.MNS	M.AP00	ACCELEROMETER PROCESSING MODULE
	E.MNS	M.DV00	F/M CALCULATIONS MODULE
	E.MNS	M.NE00	BOOST NAVIGATION MODULE
	E.MNS	M.PA00	PHASE ACTIVATOR MODULE
	E.MNS	M.TT00	TIME TILT GUIDANCE MODULE
	E.MNS	M.CC10	CHI COMPUTATIONS MODULE (GUIDANCE CHI TELEMETRY ENTRY)
	E.MNS	M.IG00	ITERATIVE GUIDANCE MODE MODULE
	E.MNS	M.OG00	ORBIT GUIDANCE MODULE
	E.MNS	M.TB47	TIME BASE 4 MODULE
	E.MNS	M.MS00	MINOR LOOP SUPPORT MODULE
	E.MNS	M.AC00	APS ATTITUDE CONTROL TEST MODULE
	E.MNS	M.PG00	SIMULATED PLATFORM GIMBAL ANGLES MODULE
	E.MNS	M.EB00	ETC/BTC MODULE
	E.MNS	M.HS00	S-IVB CUTOFF PREDICTION MODULE
	E.MNS	M.CT00	COMPRESSED DATA TELEMETRY MODULE
	E.MNS	M.DT00	MEMORY DUMP TELEMETRY MODULE
	E.MNS	G.P212	SERVICE PERIODIC PROCESSOR (CONTROL PROGRAM)

*

Table 1-7 Program Macro Definition (Continued)

1	SYMBOL 2	OPERATOR 8	ADDRESS 16
* PERIODIC PROCESSOR MASTER TABLE			
		module entry point	rate of execution in seconds
	E.MPP	M.PC50,50	FIFTY SECOND DATA COMPRESSION MODULE
	E.MPP	M.PC60,60	SIXTY SECOND DATA COMPRESSION MODULE
	E.MPP	M.PC99,100	ONE HUNDRED SECOND DATA COMPRESSION MODULE
* TIMER 2 MASTER TABLE			
		module entry point	
	E.MT2S	M.UM00	PHASE I TIME UPDATE MODULE
	E.MT2S	M.LR10	LADDER RAMP PROCESSOR MODULE
	E.MT2S	M.EP00	EVENTS PROCESSOR MODULE
	E.MT2S	M.NU00	NAVIGATION UPDATE IMPLEMENTATION MODULE
	E.MT2S	M.CM00	PHASE II CONTROL MODULE (NAVIGATION ENTRY)
	E.MT2S	M.CM20	PHASE II CONTROL MODULE (GUIDANCE ENTRY)
	E.MT2S	M.CM10	PHASE II CONTROL MODULE (MIDPOINT ENTRY)
	E.MT2S	M.WM00	WATER VALVE PROCESSOR MODULE
	E.MT2S	M.MP50	ORBITAL GUIDANCE MANEUVER PROCESSOR MODULE (NOMINAL MANEUVER)
	E.MT2S	M.MP55	ORBITAL GUIDANCE MANEUVER PROCESSOR MODULE (GENERALIZED MANEUVER)
	E.MT2S	M.TB50	TIME BASE 5 START MODULE
	E.MT2S	M.AS00	TIME BASE 5 ALTERNATE SEQUENCE REQUEST MODULE
	E.MEND		TERMINATION OF MISSION MASTER TABLES
* INTERRUPT DEFINITION			
		Minor Loop entry point	external interrupt servicing modules for
		and execution rate in ms.	ISR positions 10-1
	E.INTS	M.ML00,40,P.N.,M.T,M.D,P.N,M.T2A,M.T3,M.T4,P.N,M.T2,P.N	

Table 1-7 Program Macro Definition (Continued)

1	2	SYMBOL	OPERATOR	ADDRESS
			8	16
				module entry point
				Timer 1 priority where P.IL3 is higher than P.IL3A
		E.INT	M.SS05,P.IL3A	SWITCH SELECTOR (CHECK ENTRY)
		E.INT	M.SS30,P.IL3A	SWITCH SELECTOR (HUNG STAGE TEST ENTRY)
		E.INT	M.SS40,P.IL3A	SWITCH SELECTOR (STAGE & ADDRESS ISSUANCE ENTRY)
		E.INT	M.SS50,P.IL3A	SWITCH SELECTOR (VERIFY ADDRESS ENTRY)
		E.INT	M.SS55,P.IL3A	SWITCH SELECTOR (READ TIME CHECK ENTRY)
		E.INT	M.SS60,P.IL3A	SWITCH SELECTOR (ISSUE READ ENTRY)
		E.INT	M.SS70,P.IL3A	SWITCH SELECTOR (RESET READ ENTRY)
		E.INT	M.SS80,P.IL3A	SWITCH SELECTOR (STAGE & COMPLEMENT ADDR ENTRY)

		*PHASE DEFINITIONS		
		* PHASE 1		
			phase initialization	number of non-interrupt sequencer modules
			module	defined in this phase
		E.PIN	M.IN00,17	PHASE 1 TABLES
		* NON-INTERRUPT SEQUENCE CONTROL TABLES		
			module entry point	initial queue status which will be set by
				Phase Initializer
		E.PNS	M.AR00,P.IN	ACCELEROMETER READ MODULE ACTIVE
		E.PNS	M.SA00,P.OUT	SIMULATED ACCELEROMETERS MODULE INACTIVE
		E.PNS	M.DP00,P.IN	DISCRETE PROCESSOR MODULE ACTIVE
		E.PNS	M.AP00,P.IN	ACCELEROMETER PROCESSING MODULE ACTIVE
		E.PNS	M.DV00,P.OUT	F/M CALCULATIONS MODULE INACTIVE
		E.PNS	M.NE00,P.IN	BOOST NAVIGATION MODULE ACTIVE
		E.PNS	M.PA00,P.OUT	PHASE ACTIVATOR MODULE INACTIVE
		E.PNS	M.TT00,P.OUT	TIME TILT GUIDANCE MODULE INACTIVE

Table 1-7 Program Macro Definition (Continued)

1	SYMBOL 2	OPERATOR 8	ADDRESS 16		
	E.PNS	M.CC10,P.IN	CHI COMPUTATIONS MODULE ACTIVE		
	E.PNS	M.IG00,P.OUT	ITERATIVE GUIDANCE MODE MODULE INACTIVE		*
	E.PNS	M.HS00,P.OUT	S-IVB CUTOFF PREDICTION MODULE INACTIVE		
	E.PNS	M.OG00,P.OUT	ORBIT GUIDANCE MODULE INACTIVE		
	E.PNS	M.TB47,P.OUT	TIME BASE 4 MODULE INACTIVE		
	E.PNS	M.AC00,P.OUT	APS ATTITUDE CONTROL TEST MODULE INACTIVE		*
	E.PNS	M.MS00,P.IN	MINOR LOOP SUPPORT MODULE ACTIVE		
	E.PNS	M.PG00,P.OUT	SIMULATED PLATFORM GIMBAL ANGLES MODULE INACTIVE		
	E.PNS	M.EB00,P.IN	ETC/BTC MODULE ACTIVE		
*	PERIODIC PROCESSOR INITIAL STATUS				
	All modules inactive				
*	TIMER 2 CONTROL TABLE INITIAL STATUS				
		module entry point			
	E.PT2S	M.NU00	NAVIGATION UPDATE IMPLEMENTATION MODULE INACTIVE		
	E.PT2S	M.CM00	PHASE 2 CONTROL MODULE INACTIVE		
	E.PT2S	M.CM10	PHASE 2 CONTROL MODULE INACTIVE		
	E.PT2S	M.CM20	PHASE 2 CONTROL MODULE INACTIVE		
	E.PT2S	M.LR10	LADDER RAMP PROCESSOR MODULE INACTIVE		
	E.PT2S	M.EP00	EVENTS PROCESSOR MODULE INACTIVE		
	E.PT2S	M.WM00	WATER VALUE PROCESSOR MODULE INACTIVE		
	E.PT2S	M.MP50	ORBITAL GUIDANCE MANEUVER PROCESSOR MODULE INACTIVE		*
	E.PT2S	M.MP55	ORBITAL GUIDANCE MANEUVER PROCESSOR MODULE (GEN. MANEUVER)		*
			INACTIVE		*
	E.PT2S	M.TB50	TIME BASE 5 START MODULE INACTIVE		*
	E.PT2S	M.AS00	TIME BASE 5 ALTERNATE SEQUENCE REQUEST MODULE INACTIVE		*
	E.PEND		END OF PHASE 1 TABLE DEFINITION		

Table 1-7 Program Macro Definition (Continued)

1	2	8	16
*	PHASE 2		
		phase initialization module	
			number of non-interrupt sequencer modules defined in this phase
	E.PIN	M.IN10,3	PHASE 2 TABLES
*	NON-INTERRUPT SEQUENCE CONTROL TABLE		
		module entry point	initial queue status which will be set by Phase Initializer
	E.PNS	M.CT00,P.OUT	COMPRESSED DATA TELEMETRY MODULE INACTIVE
	E.PNS	M.DT00,P.OUT	MEMORY DUMP TELEMETRY MODULE INACTIVE
	E.PNS	G.P212,P.IN	SERVICE PERIODIC PROCESSOR ACTIVE
*	PERIODIC PROCESSOR INITIAL STATUS		
		module entry point	
	E.PPP	M.PC50	FIFTY SECOND DATA COMPRESSION MODULE ACTIVE
	E.PPP	M.PC60	SIXTY SECOND DATA COMPRESSION MODULE ACTIVE
	E.PPP	M.PC99	ONE HUNDRED SECOND DATA COMPRESSION MODULE ACTIVE
*	TIMER 2 CONTROL TABLE INITIAL STATUS		
		module entry point	
	E.PT2S	M.UM00	PHASE 1 TIME UPDATE MODULE INACTIVE
	E.PT2S	M.NU00	NAVIGATION UPDATE IMPLEMENTATION MODULE INACTIVE
	E.PT2S	M.TB50	TIME BASE 5 START MODULE INACTIVE
	E.PT2S	M.AS00	TIME BASE 5 ALTERNATE SEQUENCE REQUEST MODULE INACTIVE
	E.PEND		END OF PHASE 2 TABLE DEFINITION
*	END OF PROGRAM CONFIGURATION DEFINITION		

control on occurrence of a given interrupt. This macro also specifies the Timer 1 controlled Minor Loop module and its execution rate. Minor Loop is implicitly declared as the higher priority module and is initiated by the Control Program during initialization.

The remaining Timer 1 controlled modules and their relative priority (P.IL3 specifies higher priority modules and P.IL3A, lower priority) are specified via the E.INT macro. These modules must be queued-in at the appropriate time by application modules.

1.5.2 General Flowchart

A general flow of the Saturn IB Flight Program is shown in Fig. 1-9. The modules which are directly controlled by the Control Program are shown in the sequence defined by the macro definition given in the preceding paragraphs. Commentary on each module's queue status is included beside each module.

A number of utility modules are provided as subroutines for general use. These modules, given in Table 1-8 can only be used on a non-priority level; i.e., by Non-Interrupt Sequencer or Periodic Processor modules.

Indirectly controlled modules are described in Table 1-9 which accompanies each Control Program component. Some indirectly controlled modules are multiply listed. This occurs when a module is under direct control of the Control Program as well as under control of another module, and when a module is accessed from modules operating under different Control Program components. This is primarily a result of different timing requirements between the boost and non-powered phases of flight.

Table 1-8 Utility Modules

	MODULE
ID	NAME
AT	Arctangent Utility
DO	Discrete Output Register Set/Reset Utility
FD	Fixed Point Divide Utility
FR	Absolute/Command Freeze Utility
GF	Gimbal Angle RTC Test Utility
LN	Natural Logarithm Utility
MM	Matrix Multiply Utility
MT	Matrix Transpose Utility
PC70	Data Compression Position Utility
PC80	Data Compression Store Utility
SC	Sine-Cosine Utility
SD	Scaled Divide Utility
SR	Square Root Utility
TR	Real Time Telemetry Delay Utility
VC	Vector Cross Product Utility
VD	Vector Dot Product Utility

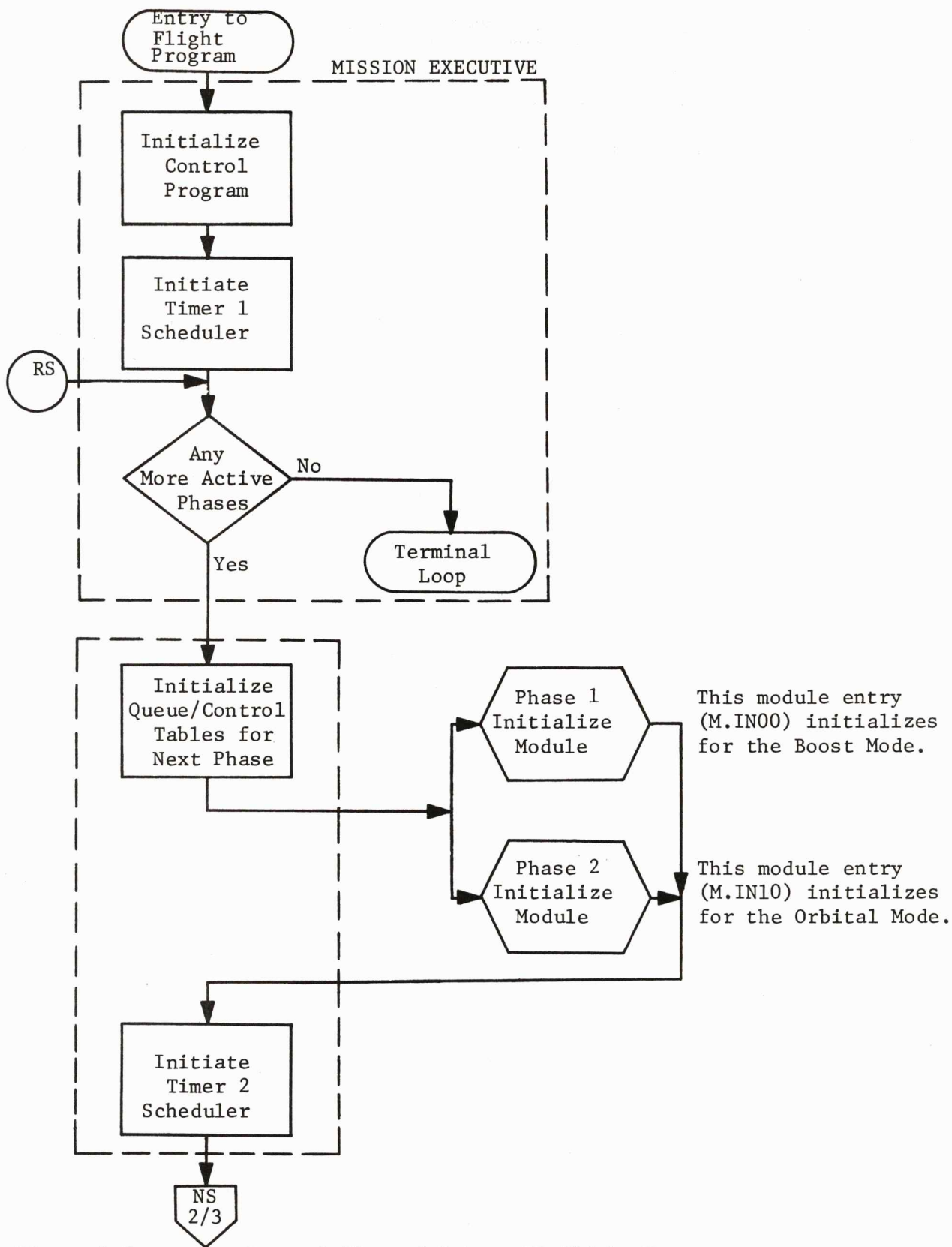


Figure 1-9a General Flow of Saturn IB Flight Program

Table 1-9a Indirectly Controlled Modules

Controlled by Non-Interrupt Sequencer Modules:

ID	MODULE NAME	CONTROL COMMENTS
CC	Chi Computations	Accessed by IGM and by Orbit Guidance.
CC	Chi Computations (Telemetry Entry)	Accessed by Time Tilt and by Orbit Guidance.
DG	Disagreement Bit Processor	Accessed by Minor Loop if a gimbal disagree- ment bit is detected.
DP1	Discrete Inputs 14 & 15	Accessed only during Phase I by Discrete Processor if DI's are detected.
DP3	Discrete Input 9	Accessed by the Discrete Processor for T4+5* to EOM when DI 9 is detected.
GR	Gravitation Acceleration	Accessed by Boost Navigation.
MP	Maneuver Processor	Accessed by Orbit Guidance.
OD	Digital Output Multiplexer (DOM)	Accessed by ETC/BTC each execution.
SM	Steering Misalign- ment Correction	Accessed by IGM at a preset time in TB3.
ST	M/F Smoothing	Accessed by F/M Calculations at preset times.
RG	Roll Guidance	Accessed by Time Tilt Guidance when active.

*

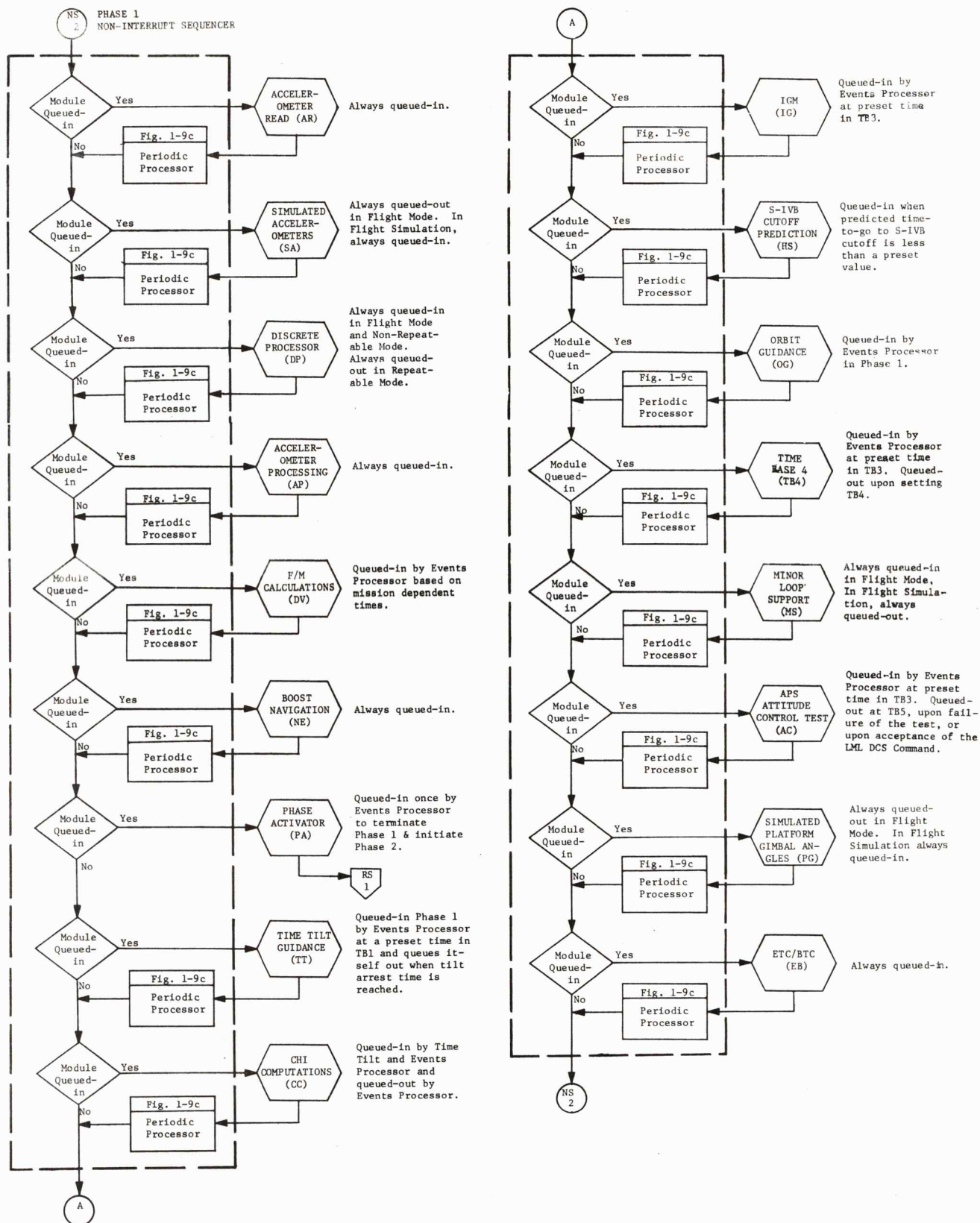


Figure 1-9b General Flow of the Saturn IB Flight Program (Continued)

Table 1-9b Indirectly Controlled Modules

Controlled by Interrupt Processor modules:

ID	MODULE NAME	CONTROL COMMENTS
CO	Cutoff Logic	Accessed by Minor Loop in Phase 1 when time-to-go to issuance of S-IVB Cutoff Command is less than preset value.
DP	Discrete Processor	Accessed every second by Phase II Control during Phase 2. Not accessed during Repeatable Flight Simulation Mode.
EB	ETC/BTC	Accessed every second by Phase II Control during Phase 2.
EC	Minor Loop Error Monitor	Accessed every 0.8 seconds by Phase I Time Update during Phase 1. Accessed every second by Phase II Control during Phase 2.
GL	Acquisition Gain/Loss	Accessed every 8 seconds by Phase II Control during Phase 2.
MS	Minor Loop Support	Accessed every second by Phase II Control during Phase 2 only in Flight Mode.
NI	Navigation Extrapolation	Accessed every second by Phase II Control during Phase 2.
NZ	Orbit Navigation	Accessed every 8 seconds by Phase II Control during Phase 2.
OA	Orbit Accelerations	Accessed by Orbit Navigation every 8 seconds during Phase 2.
OG	Orbit Guidance	Accessed every second by Phase II Control during Phase 2.

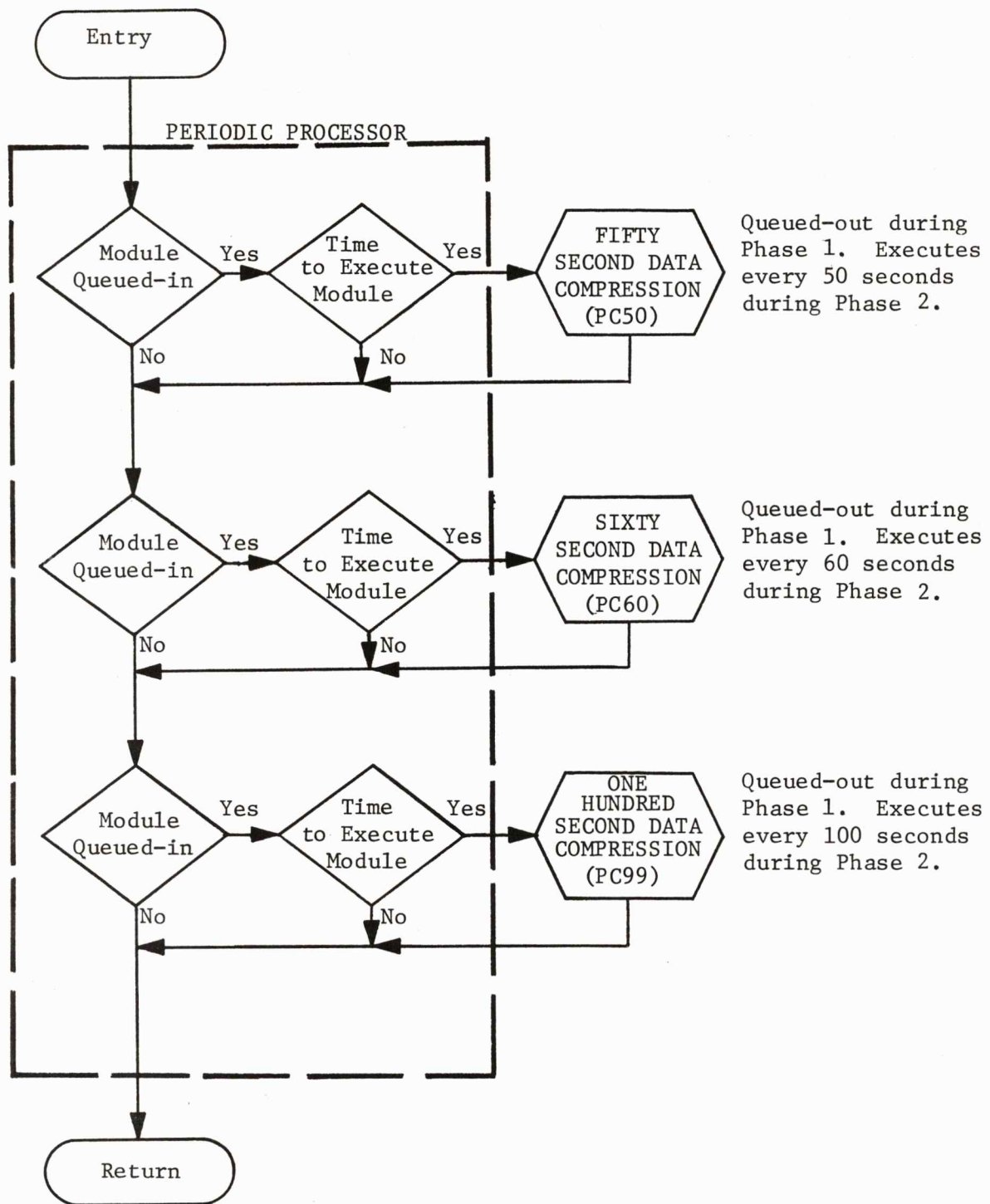


Figure 1-9c General Flow of Saturn IB Flight Program (Continued)

Table 1-9b Indirectly Controlled Modules (Continued)

ID	MODULE NAME	CONTROL COMMENTS
PC01	Once Second Data	Accessed every second by Phase II Control during Phase 2.
PG	Simulated Platform Gimbal Angles	Accessed every second by Phase II Control during Phase 2 only in Flight Simulation Mode.
TU	Phase II Time Update	Accessed every second by Phase II Control during Phase 2.
AC	APS Attitude Control Test	Accessed every second by Phase II Control from T4+15* to T5+0.
GR	Gravitation Acceleration	Accessed by Orbit Navigation every 8 seconds during Phase 2.
OV	Orbit Velocity Calculations	Accessed every second by Phase II Control from T5+0 to EOM.

*

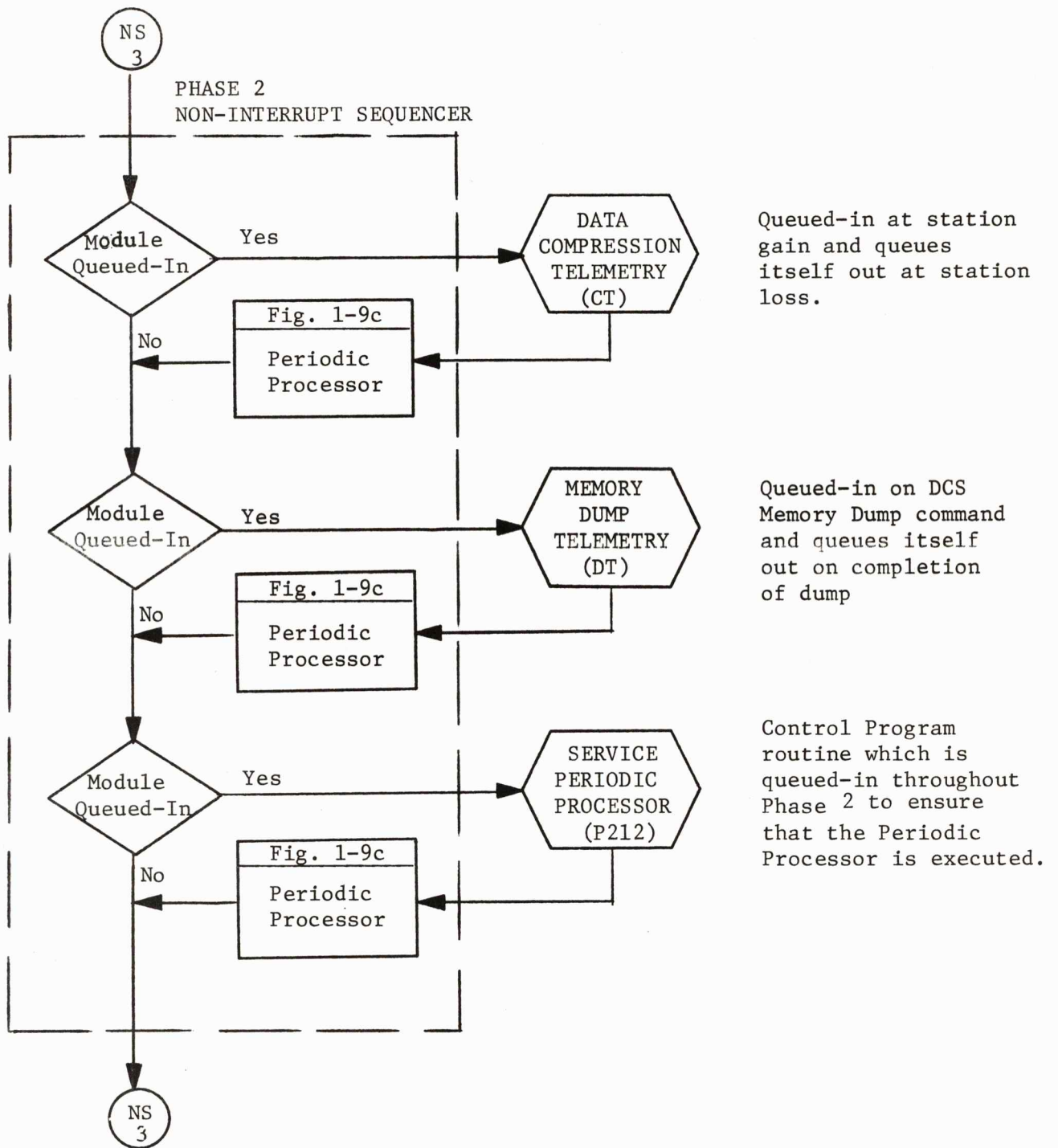


Figure 1-9d General Flow of Saturn IB Flight Program (Continued)

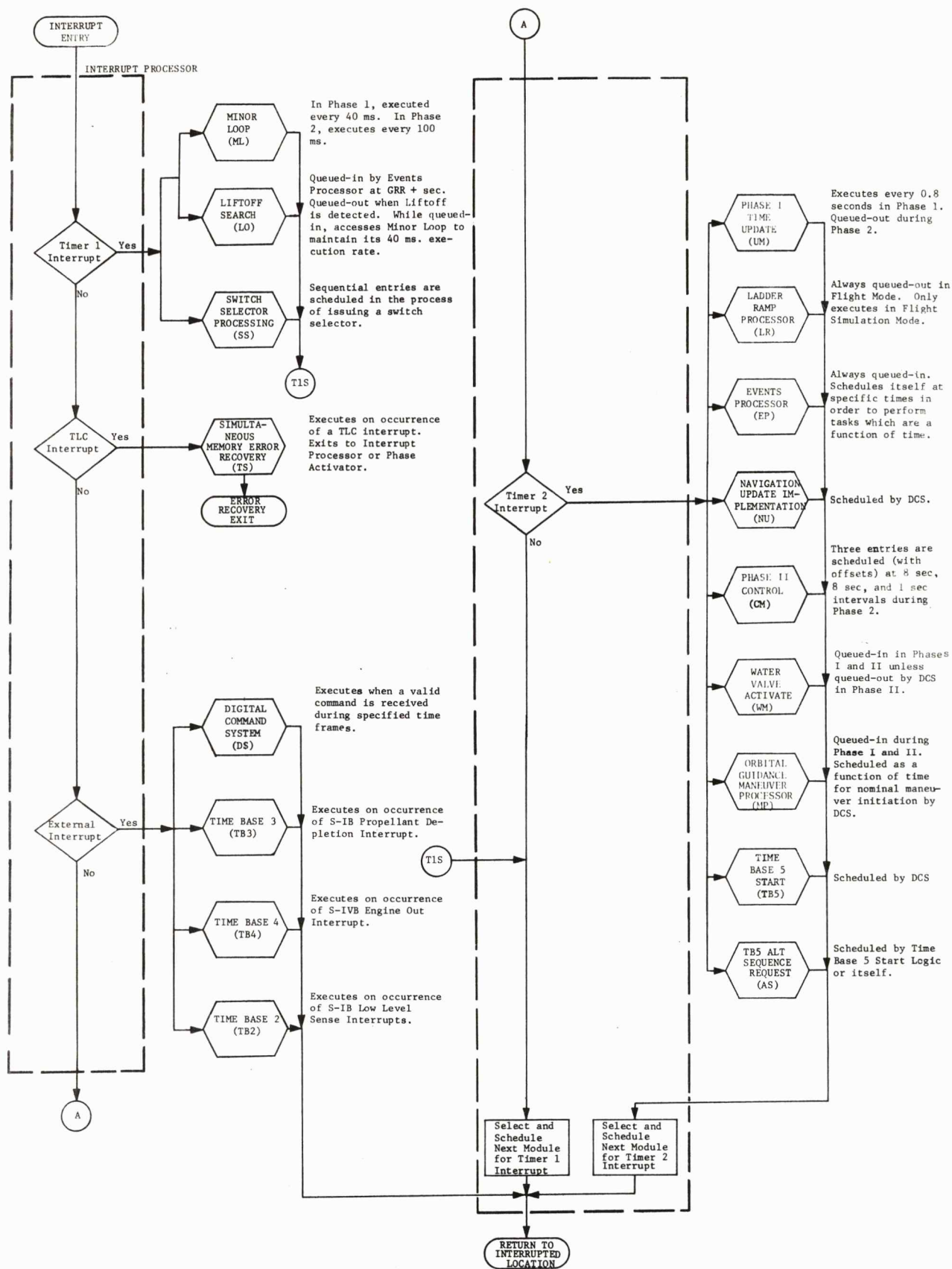


Figure 1-9e General Flow of Saturn IB Flight Program (Continued)

SECTION 2

APPLICATION MODULES DESCRIPTION

2.1 INTRODUCTION

This section describes the implementation of each application module in the Saturn IB Flight Program. If the implementation of a module varies because of a specific flight's requirements, that module will be described in the appropriate section of Part IV.

The module flowcharts are written in prose for better understanding of the operations performed and decisions made. Since these flowcharts are generated from the program listing, it is very easy to correlate the programming instructions to the flowchart. The hierarchy of flowcharts allows a reader to gain an overview of the module's operation from the first level and more detail from the subsequent levels.

The programming notes which support the flowcharts define all intermodule linkages, telemetry data, and initial conditions. The flowcharts show all external module linkages except for certain utility routines used in computing equations. The programming notes include all external linkages as well as system macros. All system macros begin with "E." as well as Control Program subroutines which can be accessed by modules. In their expansion, the following system macros access Control Program subroutines:

E.ENBL System macro to restore the normal LVDC interrupt status by providing linkage which transfers control to the executive interrupt enable subroutine which, in turn, reloads the inhibit register with the normal inhibits associated with the processing level in progress and returns control to the user.

E.MPP	System macro for generating an entry in the Periodic Processor queue table.
E.MT2S	System macro for generating an entry in the Timer 2 Scheduler queue/control table.
E.PEND	System macro for the termination of the phase table information.
E.PIN	System macro for initialization required prior to generation of Phase Table information.
E.PNS	System macro for generation of an entry in the Non-Interrupt Sequencer control table.
E.PPP	System macro for indicating the initial queue status of the Periodic Processor queue/control table for a particular phase.
E.PT2S	System macro for indicating the initial queue status of the Timer 2 Scheduler queue/control table for a particular phase.
E.Q	System macro for queuing entries into or out of the system queue tables.
E.QT	System macro for specifying the execution time for a module which is to be queued active by the E.Q macro.

System macros executed to perform a function in support of the calling routine are referred to as utility macros. The following system macros are used for support functions:

U.BITO	System macro to control and insure correct bit being set, reset, or monitored in a status word.
--------	---

U.DIV System macro that inhibits interrupts for 8 machine cycles during a DIV instruction.

U.ENTR System macro to be called when the entry for a module, routine, subroutine, or program is to be called. It saves the HOP return to Interrupt Processing for any of these cases.

U.EXIT System macro to be called when the exit for a module, subroutine, routine, or program is to be called. It supplies the correct HOP return for any of these cases.

U.MPY System macro that inhibits interrupts for 5 machine cycles during a MPY instruction.

U.MOVE System macro to move one data location to another data location.

U.MOV2 System macro to move data from one data location to two data locations.

U.OR System macro for logically ORing two arguments.

U.SSRQ System macro to request any alternate Switch Selector sequence to be issued.

U.TPL System macro to transfer if the contents of the accumulator is plus.

U.TZE System macro to transfer if the contents of the accumulator is zero.

2.1.1 APS ATTITUDE CONTROL TEST (AC)

Function

The APS Attitude Control Test Module tests for vehicle control during periods of flight when the vehicle is under APS attitude control. If the test indicates loss of attitude control, the module will open the ladder magnitude limits for roll, pitch, and yaw, then queues itself out. *

Entries

M.AC00 Entered from the non-interrupt Sequencer once per BML when the module is active in boost mode and once per second during orbital mode, unless queued-out by the Ladder Magnitude Limit DCS Command, or test failure. *

External

E.Q APS Attitude Control Test (M.AC00)

Input Data

D.VTHX Total roll gimbal angle *

D.VTHZ Total yaw gimbal angle *

D.VCCX Commanded Chi X (Rate limited gimbal angle) *

D.KT4 Time Base 4 start time from GRR *

Output Data

D.VM06 Pitch and yaw ladder magnitude limit *

D.VM16 Roll ladder magnitude limit *

G.AC00 Variable HOP Constant to M.AC00 or C.M40

F.CLML Ladder magnitude limit change word

Status Words

D.VMC4 Mode Code 24

Telemetry

None.

Initial Conditions

Before entering M.ACOO, V.ACTX and V.ACTZ must be initialized to perform the appropriate test for the particular axis.

Notes

This module will queue itself out after detection of a failure of any channel. This module will be queued-out upon receipt of the Ladder Magnitude DCS Command.

*
*

ATTITUDE CONTROL TEST (AC)

1

M. AC00

```

START
M. AC00
    
```

V

```

PERFORM X
ATTITUDE
ERROR TEST
EQ. 8.3.11
    
```

V

```

LOSS OF APS
ATTITUDE
CONTROL
INDICATED
    
```

YES

V

```

AC20      A0
CHANGE LADDER
MAGNITUDE
LIMITS
    
```

V

```

EXIT
RETURN
    
```

V

AC10

```

PERFORM APS
CONTROL TEST
ON Z CHANNEL
    
```

YES

V

```

PERFORM Z
ATTITUDE TEST
EQ. 8.3.12
    
```

V

```

LOSS OF APS
ATTITUDE
CONTROL
INDICATED
    
```

YES

V

```

AC20      A0
CHANGE LADDER
MAGNITUDE
LIMITS
    
```

V

AC99

```

EXIT
RETURN
    
```



2.2 ACCELEROMETER PROCESSING (AP)

Function

The Accelerometer Processing module detects velocity measurement errors through disagreement, zero, and reasonableness tests, accumulates velocity measured by the platform accelerometers or calculated by the flight program when an accelerometer fails, and computes a performance factor based on measured acceleration and computed acceleration to be used in computing backup acceleration. *

Entries

M.AP00 Entered from the Non-Interrupt Sequencer once per computation cycle during Phase 1.

External

U.TR00 Real Time Telemetry Delay Utility
U.SD00 Scaled Divide Utility

Input Data

D.VBDX Change in velocity along X_s (backup)
D.VBDY Change in velocity along Y_s (backup)
D.VBDZ Change in velocity along Z_s (backup)
D.VDAX Optisyn A change in X velocity between accelerometer reads
D.VDBX Optisyn B change in X velocity between accelerometer reads
D.VDAY Optisyn A change in Y velocity between accelerometer reads
D.VDBY Optisyn B change in Y velocity between accelerometer reads
D.VDAZ Optisyn A change in Z velocity between accelerometer reads
D.VDBZ Optisyn B change in Z velocity between accelerometer reads
D.VRCX X accelerometer reasonableness test constant
D.VRCY Y accelerometer reasonableness test constant
D.VRCZ Z accelerometer reasonableness test constant

D.VDX	X component of the backup velocity accumulation bias vector	
D.VDY	Y component of the backup velocity accumulation bias vector	
D.VDZ	Z component of the backup velocity accumulation bias vector	
D.VFX	Expected X platform velocity change	
D.VFY	Expected Y platform velocity change	
D.VFZ	Expected Z platform velocity change	
D.VDT	Elapsed time between current and previous computation cycle	*
	accelerometer reads in seconds	*
D.VDTP	Intermediate parameter (20*D.VDT)	
D.FZER	Zero test enable flag	
D.VSND	Constant used to compute V.ACZR	*
D.VFOM	Total vehicle acceleration determined from accelerometer readings	
F.PFC	Compute performance factor enable flag	
D.V20F	$20 * D.VDTP * D.VFMC$	*
V.MBA1	MBS11A component of MBSA matrix	
V.MBA4	MBS21A component of MBSA matrix	
V.MBA7	MBS31A component of MBSA matrix	
D.VPFO	Computed performance factor (previous cycle)	*

Output Data

D.VXDM	X component of measured velocity	*
D.VYDM	Y component of measured velocity	*
D.VZDM	Z component of measured velocity	*
D.FSMC	Steering Misalignment Correction inhibit flag	*
D.VVSQ	Sum of the squares of X, Y and Z accelerometer changes	*
D.VPF	Performance factor to be used in computing backup acceleration	
D.PSMC	Steering misalignment past computation cycle inhibit flag	
F.ACCF	Accelerometer failure flag	

Status Words

D.VMC4 Mode Code 24

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.VXDM	1024	X component of measured velocity
D.VYDM	1030	Y component of measured velocity
D.VZDM	1020	Z component of measured velocity

Initial Conditions

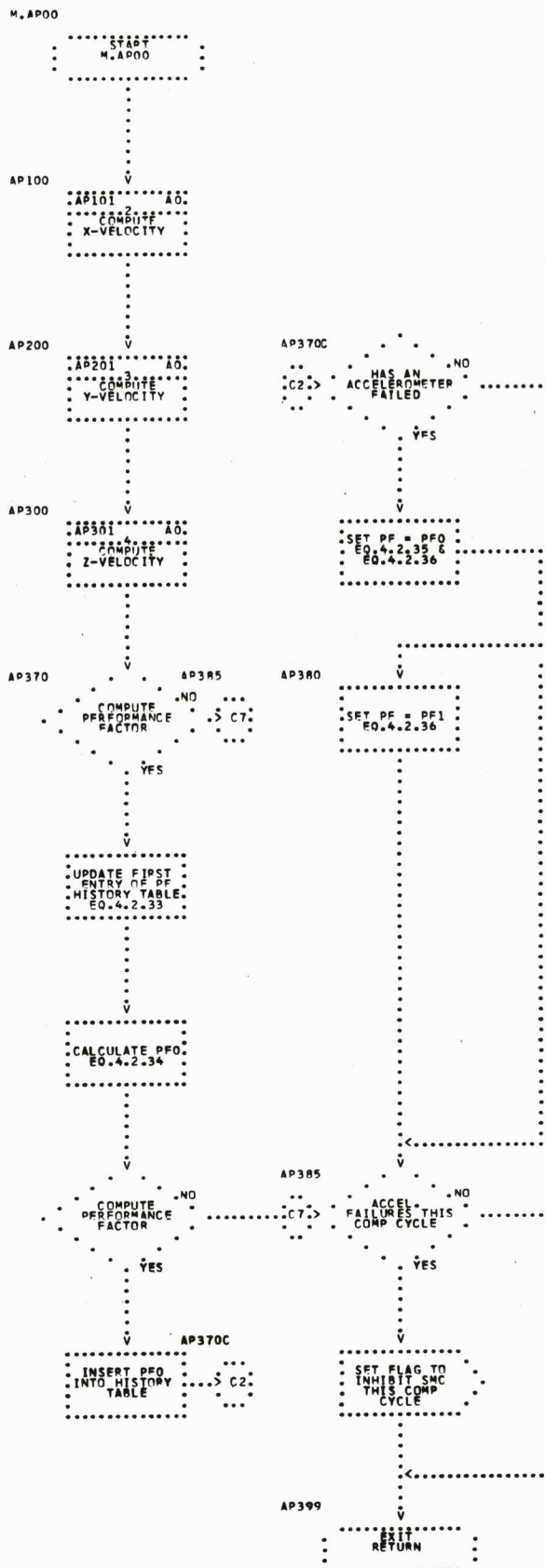
The previous velocity components (V.POXV, V.POYV, and V.POZV) must be initially zero. The reasonableness test constants (D.VRCX, D.VRCY, and D.VRCZ) must be initialized at mission start.

The zero test will be enabled and disabled at mission dependent times.

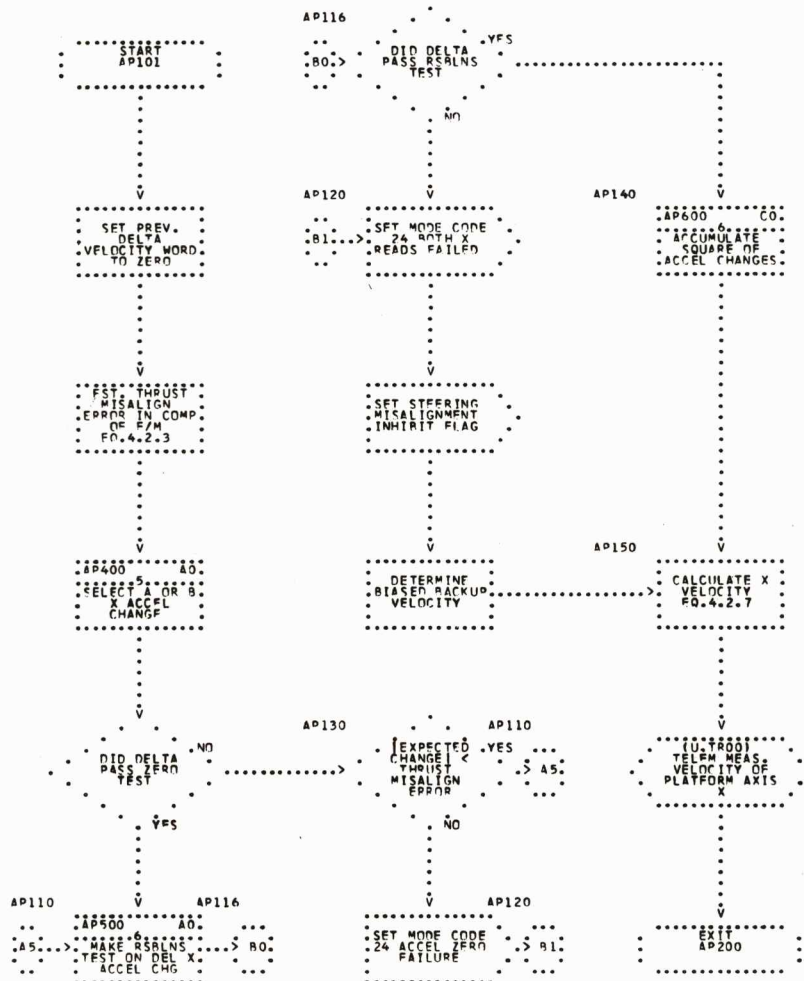
The performance factor computations will be enabled at mission dependent times.

Notes

Accelerometer Processing accumulates the square of the accelerometer changes each computation cycle for use as a backup for initiation of Time Base 4.

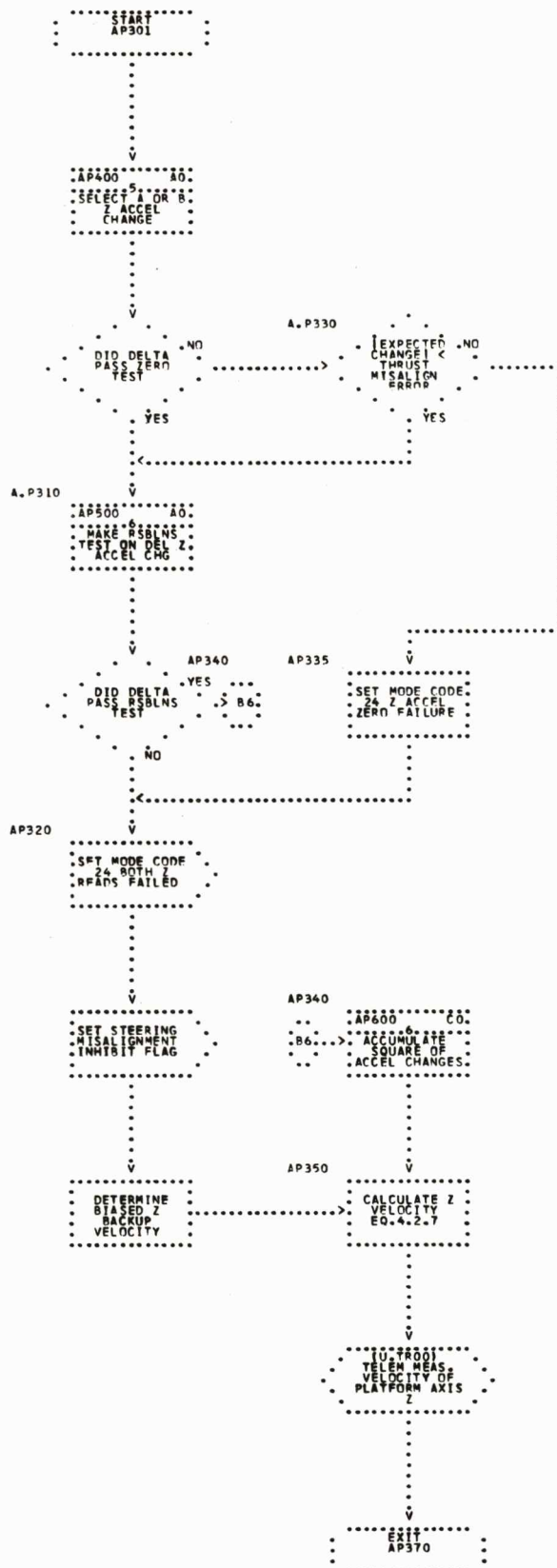


ACCELEROMETER PROCESSING (AP)

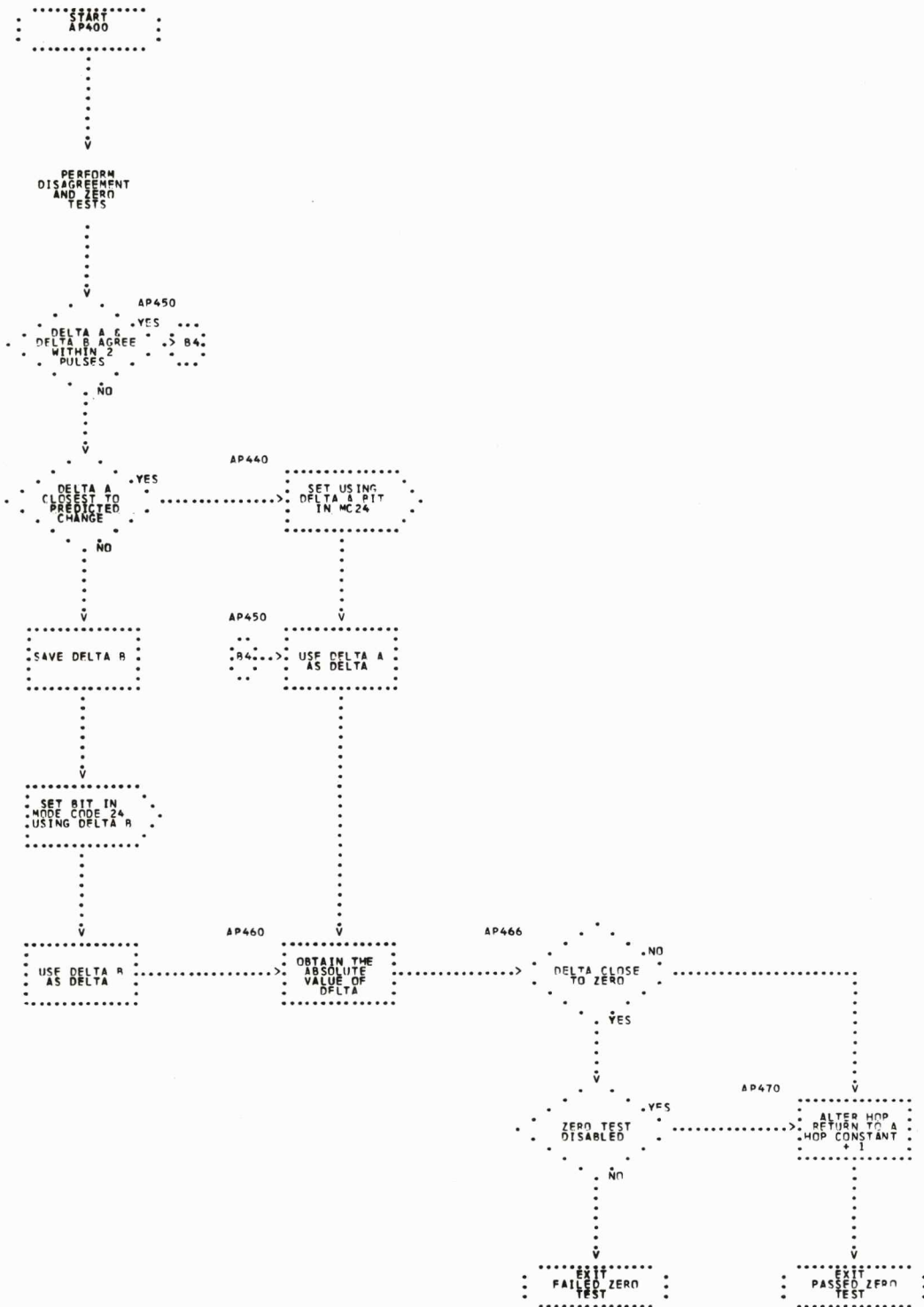




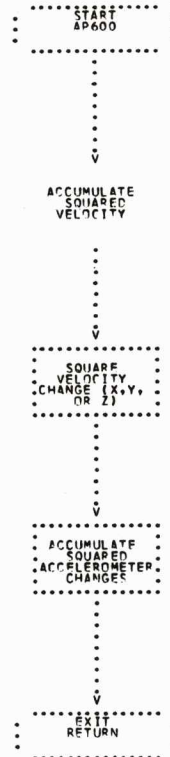
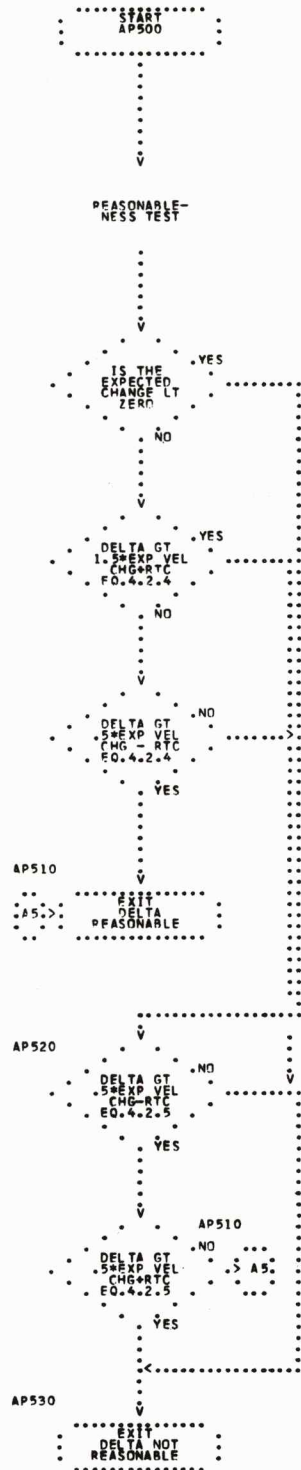
ACCELEROMETER PROCESSING (AP)



ACCELEROMETER PROCESSING (AP)



ACCELEROMETER PROCESSING (AP)



2.3 ACCELEROMETER READ (AR)

Function

The Accelerometer Read module reads platform accelerometers, formats these reads for use in the Accelerometer Processing module, and computes backup acceleration for use in case of accelerometer failures.

Entries

M.AR00 Entered from the Non-Interrupt Sequencer once per computation cycle during Phase 1.

External

G.TVNT Change thrust event parameters
G.ARHC Change thrust misalignment vector (M.AR20)
U.SC00 Sine-Cosine Utility
U.SD00 Scaled Divide Utility
U.TR00 Real Time Telemetry Delay Utility
U.MM00 Matrix Multiply Utility
E.ENBL Enable interrupts
E.IHBT Inhibit interrupts

Input Data

D.VCCY	Commanded Chi Y (Rate limited pitch gimbal angle)	*
D.VCCZ	Commanded Chi Z (Rate limited yaw gimbal angle)	*
D.VFOR	Thrust of vehicle (backup)	
D.VMAS	Mass of vehicle (backup)	
D.VMFR	Mass flow rate of vehicle (backup)	
D.VTHX	Total roll gimbal angle	*
D.VTHY	Total pitch gimbal angle	*
D.VTHZ	Total yaw gimbal angle	*
D.VFOM	Total vehicle acceleration determined from accelerometer readings readings	
D.VTAS	Mission time from GRR at accelerometer read	*

D.VTMM	Elapsed time in mission from GRR to last time update	*
D.VRTC	Real time clock reading at last time update	
D.VEOF	Engine out multiplication factor for backup F/M calculation	*
D.VERT	Time error associated with last time update	*
D.VTI	Time from GRR that the current time base was set	
D.VDTE	Fraction of a BML prior to a thrust change event (seconds)	
D.VEFP	Engine out factor in use prior to a thrust change event (seconds)	
T.TEVT	Time of a thrust change event (seconds)	
D.VPFP	Performance factor in use prior to a thrust change event	
D.VFMP	Mass flow rate in use prior to a thrust change event	
D.VFRP	Force in use prior to a thrust change event	
D.VPF	Performance factor to be used in computing backup acceleration	*
D.ARST	Time to change thrust misalignment vector	
D.VGX	X component of gravity acceleration	
D.KT1	Time Base 1 start time from GRR	*
F.TEVT	Thrust change event flag	
D.FSMC	Steering Misalignment Correction inhibit flag	*
D.KT3	Time Base 3 start time from GRR	*
D.VGCX	Guidance Chi X (Desired roll gimbal angle)	*
D.VGCY	Guidance Chi Y (Desired pitch gimbal angle)	*
D.VGCZ	Guidance Chi Z (Desired yaw gimbal angle)	*
D.FLT	Flight mode indicator	*
V.MPC1	New vehicle mass at thrust change event	*

Output Data

D.VDAX	Optisyn A change in X velocity between accelerometer reads	*
D.VDBX	Optisyn B change in X velocity between accelerometer reads	*
D.VDAY	Optisyn A change in Y velocity between accelerometer reads	*
D.VDBY	Optisyn B change in Y velocity between accelerometer reads	*
D.VDAZ	Optisyn A change in Z velocity between accelerometer reads	*
D.VDBZ	Optisyn B change in Z velocity between accelerometer reads	*
D.VACX	X accelerometer reading	*
D.VACY	Y accelerometer reading	*
D.VACZ	Z accelerometer reading	*
D.VBDX	Change in velocity along X_s (backup)	*
D.VBDY	Change in velocity along Y_s (backup)	*
D.VBDZ	Change in velocity along Z_s (backup)	*

D.VFX	Expected X platform velocity change	
D.VFY	Expected Y platform velocity change	
D.VFZ	Expected Z platform velocity change	
D.VTB	Time elapsed in current time base	*
D.VDX	X component of the backup velocity accumulation bias vector	
D.VDY	Y component of the backup velocity accumulation bias vector	
D.VDZ	Z component of the backup velocity accumulation bias vector	
D.VDT	Elapsed time between current and previous computation cycle	*
	accelerometer reads in seconds	*
D.VTAS	Total time elapsed in mission	*
D.VFMC	Generated acceleration (backup)	
D.VCZA	Average of present and past Minor Loop yaw command chi at the time of major computer cycle accelerometer read	
D.VCYA	Average of present and past Minor Loop pitch command chi at the time of major computer cycle accelerometer read	
D.VDTP	Intermediate parameter (20* D.VDT)	*
D.VACT	Real Time Clock reading associated with D.VTAS	*
V.MBA1	MBS11A component of MBSA matrix	
V.MBA4	MBS21A component of MBSA matrix	
V.MBA7	MBS31A component of MBSA matrix	
D.V20F	20* D.VDTP* D.VFMC	*
F.ACCF	Accelerometer failure flag	
D.PSMC	Steering Misalignment past computation cycle inhibit flag	

Status Words

D.VMC4 Mode Code 24

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>	
D.VACZ	0004	Z accelerometer reading	
D.VACX	0010	X accelerometer reading	
D.VACY	0014	Y accelerometer reading	
D.VACT	0174	Real Time Clock reading associated with D.VTAS or D.VNMT	*
D.VTB	0031	Time elapsed in current time base	*
D.VTI	0561	Time from GRR when current time base was set	
D.VTAS	0000	Total time elapsed in mission	*

Initial Conditions

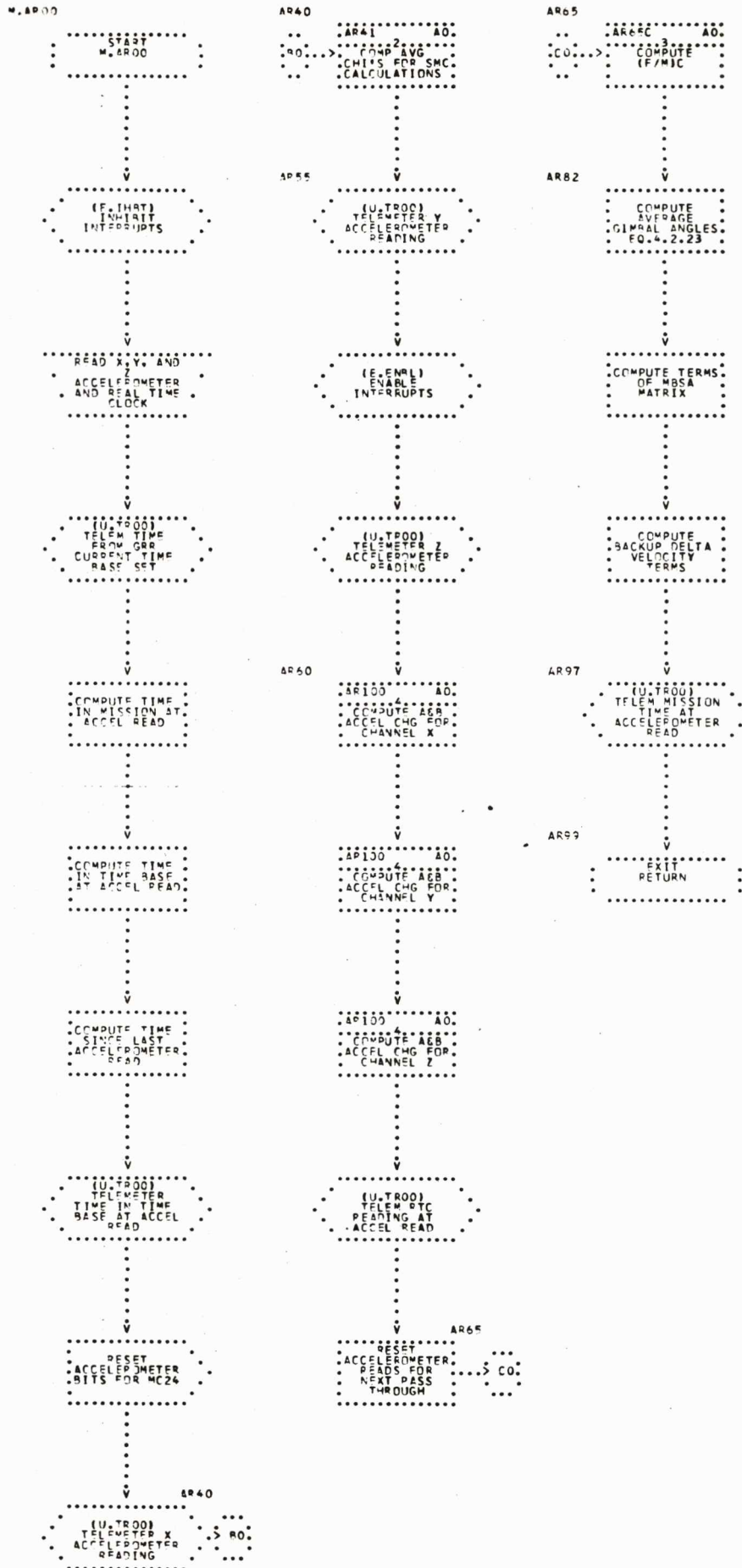
The previous accelerometer readings (V.OACX, V.OACY, and V.OACZ) will be initialized by the Control Program at GRR. D.VFOM is initialized to (P.FOVO).

Notes

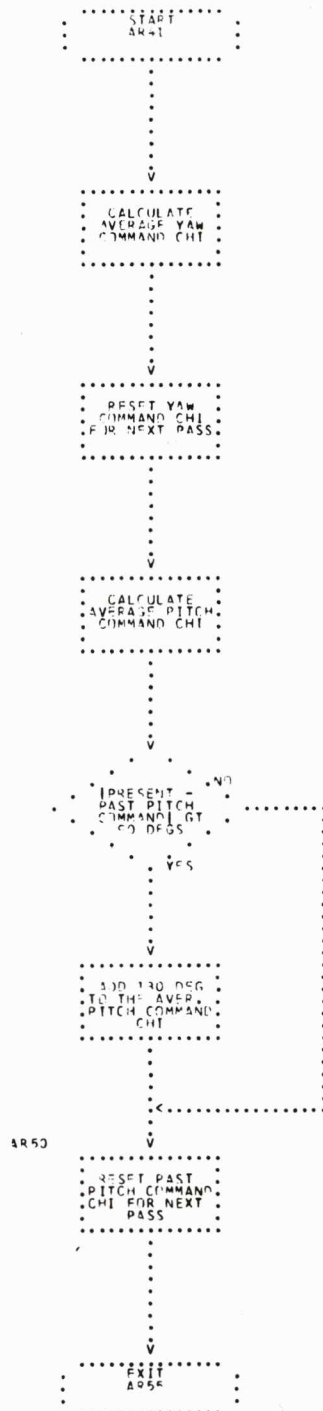
Interrupts are inhibited across accelerometer and real time clock reads, calculations of time in mission (D.VTAS), time in time base (D.VTB), computation cycle length (D.VDT), the resetting of Mode Code 24 (D.VMC4) and the use of D.VFOR (Force), D.VMAS (Mass), D.VMFR (Mass flow rate) and D.VPF (performance factor) in the backup acceleration computations.

Average pitch and yaw Minor Loop command chi's are calculated in the Accelerometer Read module for use in computing the steering misalignment correction terms. These angles are calculated at this time to synch them with the accelerometer readings.

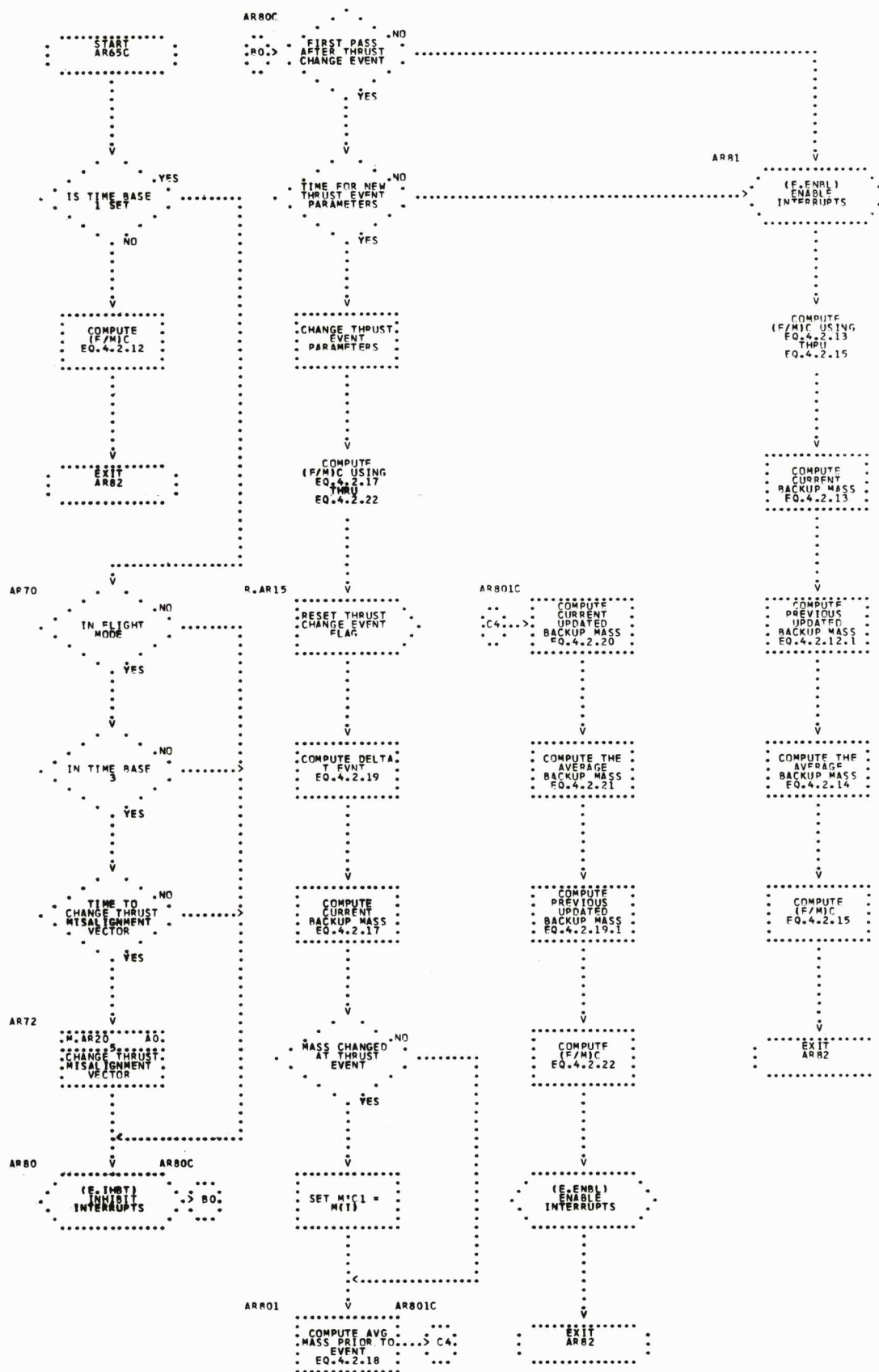
(THIS PAGE LEFT BLANK INTENTIONALLY)



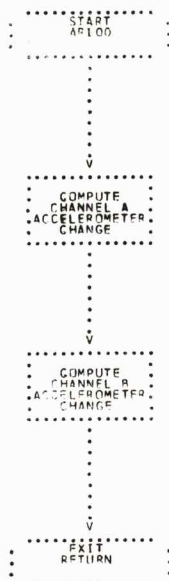
ACCELEROMETER READ (AP)

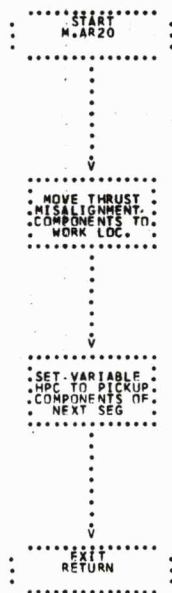


ACCELEROMETER READ (AR)



ACCELEROMETER READ (AR)





2.3.1 ALTERNATE SEQUENCE REQUESTER (AS)

Function

This module requests the two Time Base 5 alternate sequences for the S-IVB/IU De-Orbit propellant dumps at the intervals included in the S-IVB/IU De-Orbit DCS command.

Entries

M.AS00 Entered from Timer 2 when scheduled by either the Time Base 5 start logic or itself.

External

U.SSRQ Request Propellant Dump Switch Selector Sequences (M.SS00)

E.Q Alternate Sequencer Requester (M.AS00)

E.QT Alternate Sequencer Requester (M.AS00)

Input Data

D.VTLD Duration time of LOX dump included in De-Orbit command

D.VTHD Duration time of LH2 dump included in De-Orbit command

Output Data

D.ELVR Delta velocity check allow flag

*

Status Words

None.

Telemetry

None.

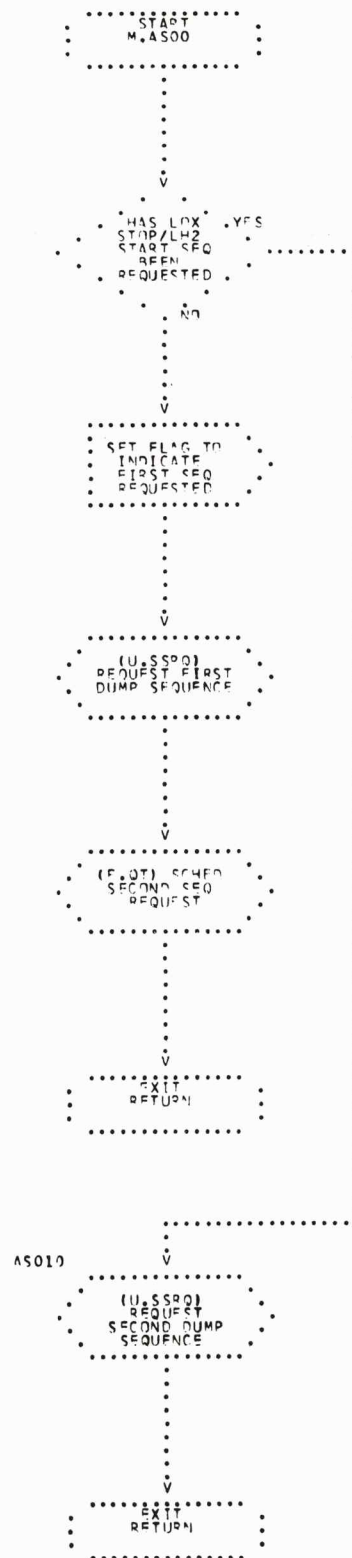
Initial Conditions

None.

Notes

F.SRW must be zero on first entry to obtain proper order of dump sequences. *

M. AS00



(THIS PAGE INTENTIONALLY LEFT BLANK)

2.4 ARCTANGENT UTILITY (AT)

Function

The function of the Arctangent Utility module is to compute the arctangent of the quotient of two specified numbers by means of an algorithm which utilizes a polynomial approximation to the arctangent function.

Entries

U.AT00 Entered from any module.

External

None.

Input Data

See Notes.

Output Data

See Notes.

Status Words

None.

Telemetry

None.

Initial Conditions

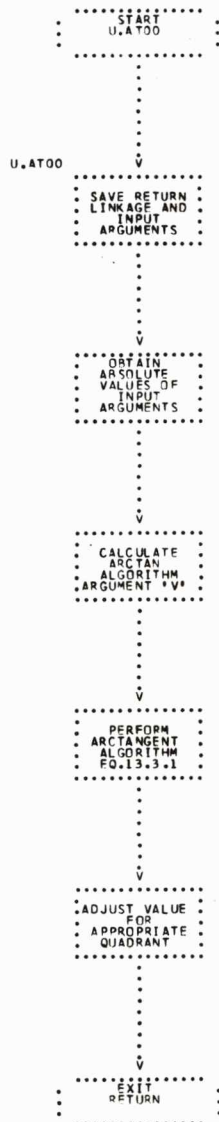
None.

Notes

On entry the accumulator must contain the sine of the desired angle (ARG1) and the PQ register must contain the cosine of the desired angle (ARG2).

Upon exit the desired angle, $\arctan (ARG1/ARG2)$, will be in the accumulator in pirads scaled B0.

ARCTANGENT UTILITY (AT)



2.5 CHI COMPUTATIONS (CC)

Function

The Chi Computation module computes the pitch and yaw commands in the S-system and telemeters the desired attitude values in both the boost and orbital modes.

Entries

- M.CC00 Entered once per computation cycle from IGM while IGM is active during Phase 1. Entered 1/sec from the Orbit Guidance module during Phase 2.
- M.CC10 Entered from Time Tilt Guidance module from liftoff until start of IGM. Entered once per computation cycle from Orbit Guidance during an attitude freeze.

External

- U.AT00 Arc tangent Utility
- U.MM00 Matrix Multiply Utility
- U.SR00 Square Root Utility
- U.TR00 Real Time Telemetry Delay Utility

Input Data

- | | | |
|--------|---|---|
| D.VTP1 | Cosine Chi Y in 4-system | * |
| D.VTP3 | Sine Chi Y in 4-system | * |
| D.VTP4 | Cosine Chi Z in 4-system | * |
| D.VSCY | Pitch steering misalignment correction | * |
| D.VSCZ | Yaw steering misalignment correction | * |
| V.XSP | X component of desired unit thrust vector in S-system | * |
| V.YXP | Y component of desired unit thrust vector in S-system | * |
| V.ZSP | Z component of desired unit thrust vector in S-system | * |

Output Data

D.VGCY	Guidance Chi Y (Desired pitch gimbal angle)	*
D.VGCZ	Guidance Chi Z (Desired yaw gimbal angle)	*

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>	
D.VGCX	0005	Guidance Chi X (Desired roll gimbal angle)	*
D.VGCY	0011	Guidance Chi Y (Desired pitch gimbal angle)	*
D.VGCZ	0001	Guidance Chi Z (Desired yaw gimbal angle)	*

Initial Conditions

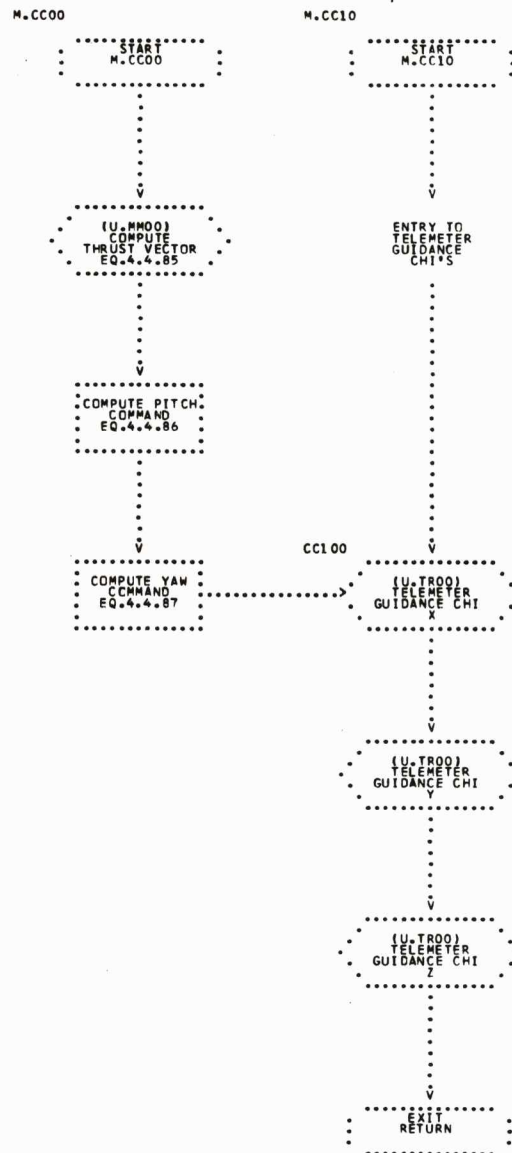
V.SQ3 {previous value of $(1-V.YSP^2)^{1/2}$ } and D.VTP2 (Sin Chi Z in 4-system) must be initialized prior to first pass through this module.

Notes

None.

CHI COMPUTATIONS (CC)

1



2.6 PHASE II CONTROL (CM)

Function

The Phase II Control module maintains the navigation and guidance timeline for the orbital phases of operation, Phase 2.

Entries

- M.CM00 Entered from Timer 2 every eight seconds during Phase 2 for navigation processing.
- M.CM01 Entered by the Navigation Update Implementation module when a DCS update has been processed for navigation processing. Control is returned to Timer 2 rather than the module.
- M.CM10 Entered from Timer 2 every eight seconds during Phase 2 for saving guidance chi's and total gimbals for midpoint navigation processing.
- M.CM20 Entered from Timer 2 every second between navigation (M.CM00) entries for guidance processing.

External

- M.DP00 Discrete Processor
- M.EB10 ETC/BTC
- M.EC00 Minor Loop Error Monitor
- M.GL00 Acquisition Gain/Loss
- M.MS00 Minor Loop Support
- M.NI00 Navigation Extrapolation
- M.NZ00 Orbit Navigation
- M.OG00 Orbit Guidance
- M.PA00 Phase Activator
- M.PC01 One second Data Compression
- M.OV00 Orbit Velocity Calculations
- M.AC00 APS Attitude Control Test

M.PG00 Simulated Platform Gimbal Angles
M.TU00 Phase II Time Update
U.TR00 Real Time Telemetry Delay Utility
E.QT Phase II Control (M.CM00, M.CM10, and M.CM20)

Input Data

D.FLT	Flight Mode indicator	
D.VNTM	Time in mission from GRR of navigation extrapolation	*
D.VER8	Accumulated eight second composite EMR word	*
D.VGCY	Guidance Chi Y (Desired pitch gimbal angle)	*
D.VGCZ	Guidance Chi Z (Desired yaw gimbal angle)	*
D.VTHY	Total pitch gimbal angle	*
D.VTHZ	Total yaw gimbal angle	*
G.PA00	Variable Hop constant to M.PA00 or C.M10	*
G.GL00	Variable Hop constant to M.GL00 or C.M20	
G.DP00	Variable Hop constant to M.DP00 or C.M25	
G.MSPG	Variable Hop constant to M.MS00 or M.PG00	
G.OV00	Variable Hop constant to M.OV00 or M.CM35	
G.AC00	Variable Hop constant to M.AC00 or M.CM40	

Output Data

D.VTAS	Total time elapsed in mission	*
V.NTMO	Time of previous navigation pass	*
D.VMTY	Midpoint pitch gimbal angle for drag integration	*
D.VMTZ	Midpoint yaw gimbal angle for drag integration	*
D.VT8S	Timer II schedule for M.CM00 entry	
D.VT4S	Timer II schedule for M.CM10 entry	
D.VT1S	Timer II schedule for M.CM20 entry	

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>	
D.VTAS	0000	Total time elapsed in mission	*
D.VER8	0535	Accumulated eight second composite EMR word	*

Initial Conditions

The scheduled times for the three entries, D.VT8S, D.VT4S, and D.VT1S, are set to phase start plus 8, 4, and 1 seconds respectively.

Notes

Although both entries M.CM00 and M.CM10 have an eight second rate, they are scheduled so that there is a four second off-set between entering M.CM00 and M.CM10.

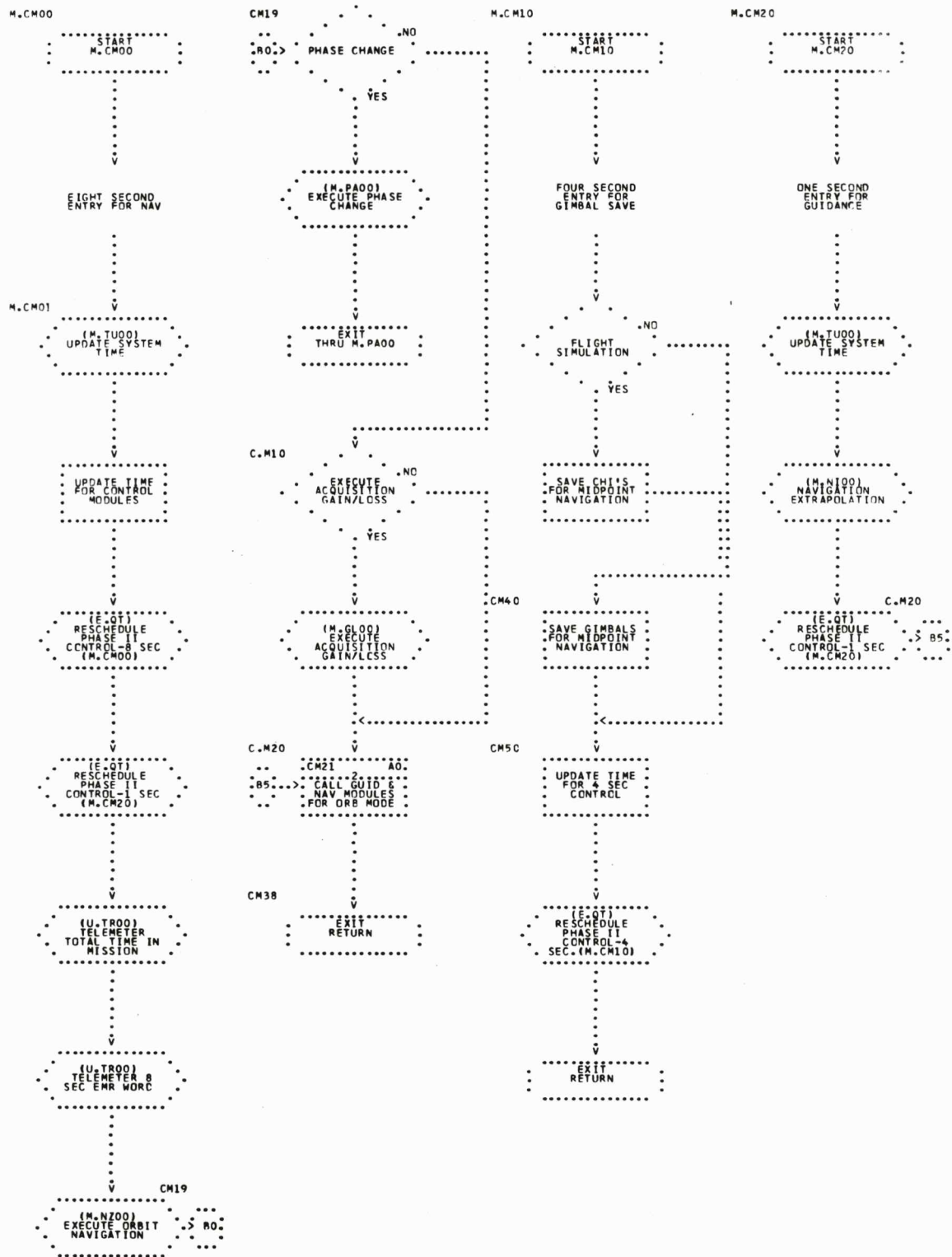
In Flight Simulation Mode, the guidance chi's are used for midpoint navigation gimbals. In Flight Mode, the actual gimbals are used.

In Flight Simulation Mode, the Simulated Platform Gimbal Angles module will be executed instead of the Minor Loop Support module. The Simulated Platform Gimbal Angles module is not executed in the Flight Mode.

The Phase Activator module is given control when the phase is to be terminated. Control is not returned to the Phase II Control module.

PHASE II CONTROL (CM)

1





2.7 CUTOFF LOGIC (CO)

Function

The function of the Cutoff Logic module is to provide the capability for accurately issuing S-IVB cutoff.

Entries

M.CO00 Entered via Minor Loop when S-IVB cutoff is imminent.

External

E.GP07	Timer 2 Halt and Reschedule	*
E.GP20	Control Program Time Update	*
H.S027	High Speed Loop S-IVB cutoff switch selector read issuance	
E.IHBT	Inhibit interrupts	
E.Q	S-IVB Cutoff Prediction (M.HS00)	
E.QT	Switch Selector Processing (M.SS60)	
E.QT	Events Processor (M.EP00)	

Input Data

D.VCOR	RTC read associated with D.VTMS	
D.VTMM	Elapsed time in mission from GRR to last time update	*
D.VTMS	Current time-to-go to S-IVB cutoff	*
D.VTRR	Elapsed total time in current time base	*
V.YAW	Actual yaw ladder command	
V.PIT	Actual pitch ladder command	
V.ROL	Actual roll ladder command	

Output Data

D.VMLT	Execution time of next Minor Loop	
D.VTBI	Indication of time base initiation	*

D.VTMS Current time-to-go to S-IVB cutoff
D.GMLC Variable HOP constant to M.C000 or M.L299

*

Status Words

D.FILE Flight Program Status
D.FIL3 Timer 1 interrupt in progress indicator
D.FIL2 External int level in progress indicator
D.VMC5 Mode Code 25

Telemetry

None.

Initial Conditions

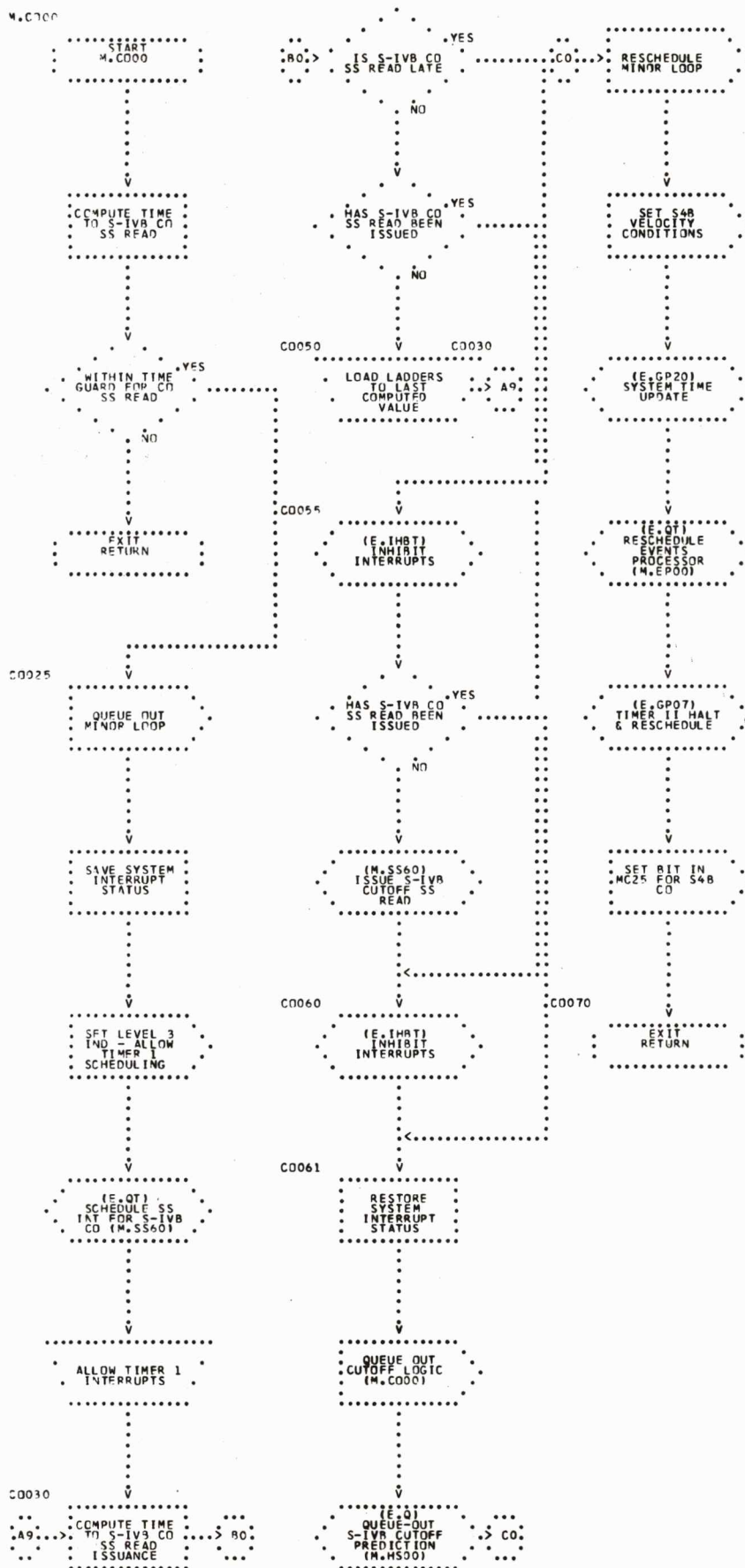
None.

Notes

To insure accurate issuance of S-IVB cutoff and to maintain the proper minor loop rate, the Cutoff Logic module will be called as a special subroutine of the Minor Loop module. To activate this module, the HOP Constant D.GMLC must be set to M.C000. This module will deactivate itself by resetting D.GMLC to M.L299 after the S-IVB cutoff switch selector read has been issued.

CUTOFF LOGIC (CO)

1



2.8 COMPRESSED DATA TELEMETRY (CT)

Function

This module serves as the active major loop after there has been a Compressed Data Dump DCS command. The module telemeters different types of data which have been stored from the previous dark period and/or which are sampled at known intervals. These data are telemetered from predefined data tables with each having its own associated PIO tag.

Entries

M.CT00 Entered from the Non-Interrupt Sequencer once per computation cycle during Phase 2 when queued-in by the Compressed Data Dump DCS command.

External

U.TR00 Real Time Telemetry Delay Utility
E.Q Compressed Data Telemetry (M.CT00)
E.IHBT Inhibit interrupts
E.ENBL Enable interrupts

*
*

Input Data

V.KSTR Compressed data storage flag
D.BPT1 Table 1 beginning pointer
D.BPT2 Table 2 beginning pointer
D.BPT3 Table 3 beginning pointer
D.BPT4 Table 4 beginning pointer
D.EPT1 Table 1 ending pointer
D.EPT2 Table 2 ending pointer
D.EPT3 Table 3 ending pointer
D.EPT4 Table 4 ending pointer
D.PPT1 Table 1 present pointer
D.PPT2 Table 2 present pointer

D.PPT3 Table 3 present pointer
D.PPT4 Table 4 present pointer
D.FDTL Memory dump/compressed data dump in progress flag *

Output Data

D.VER8 Accumulated eight second composite EMR word *
F.EMR8 Dark period EMR accumulation flag *

Status Words

None.

Telemetry

Compressed Data Table 1
Compressed Data Table 2
Compressed Data Table 3
Compressed Data Table 4

Initial Conditions

D.PPT1, ..., D.PPT4 must be set to D.EPT1, ..., D.EPT4, respectively.
V.ECT1, ..., V.ECT4 must be set to V.ECK1, ..., V.ECK4, respectively.
V.FPIO must be non-zero.

Notes

There are four telemetry tables. This module telemeters one entry from each table each computation cycle in such a manner that each table is completely telemetered once before beginning telemetry of any table a second time and each table is completely telemetered twice before beginning telemetry of any table a third time. Each table begins at corresponding locations in different data sectors so that the selection of a table is made by changing data sectors.

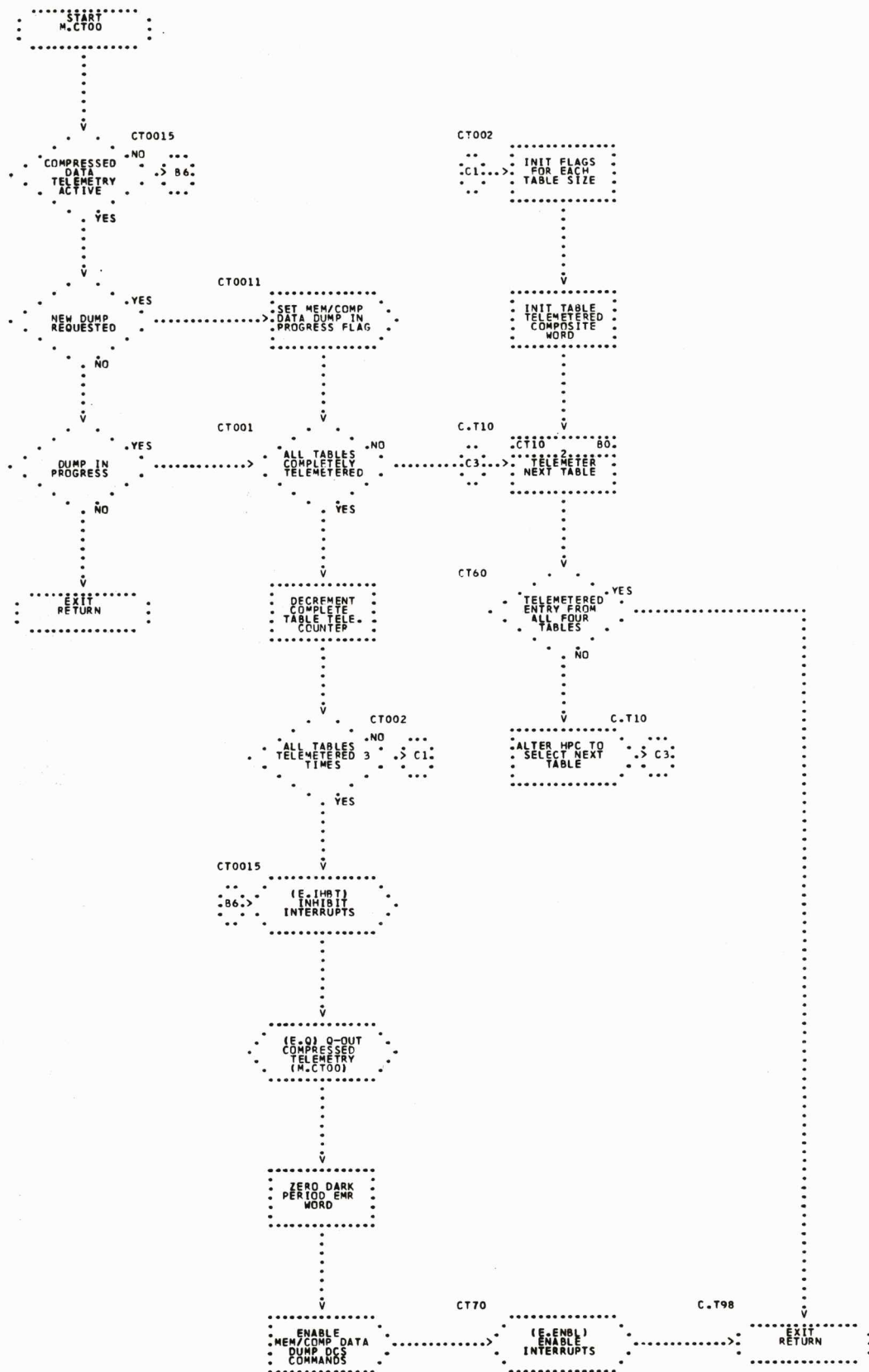
- o Telemetry Table 1: Contains 3 fine gimbal angles and time and 3 backup gimbal angles and time. Eight words per entry. PIO Tag 3404.
- o Telemetry Table 2: Contains the 60 second composite EMR word and time. Two words per entry. PIO Tag 3414.
- o Telemetry Table 3: Contains 3 accelerometers and time. Four words per entry. PIO Tag 3410.
- o Telemetry Table 4: Contains MC24 and time, TLC and time, switch selector and time, DIN's and time, etc. Two words per entry. PIO Tag 3041.

This module queues itself out when each compressed data table has been completely telemetered three times or is queued out by the DCS module (DS) upon receipt of a Terminate DCS command.

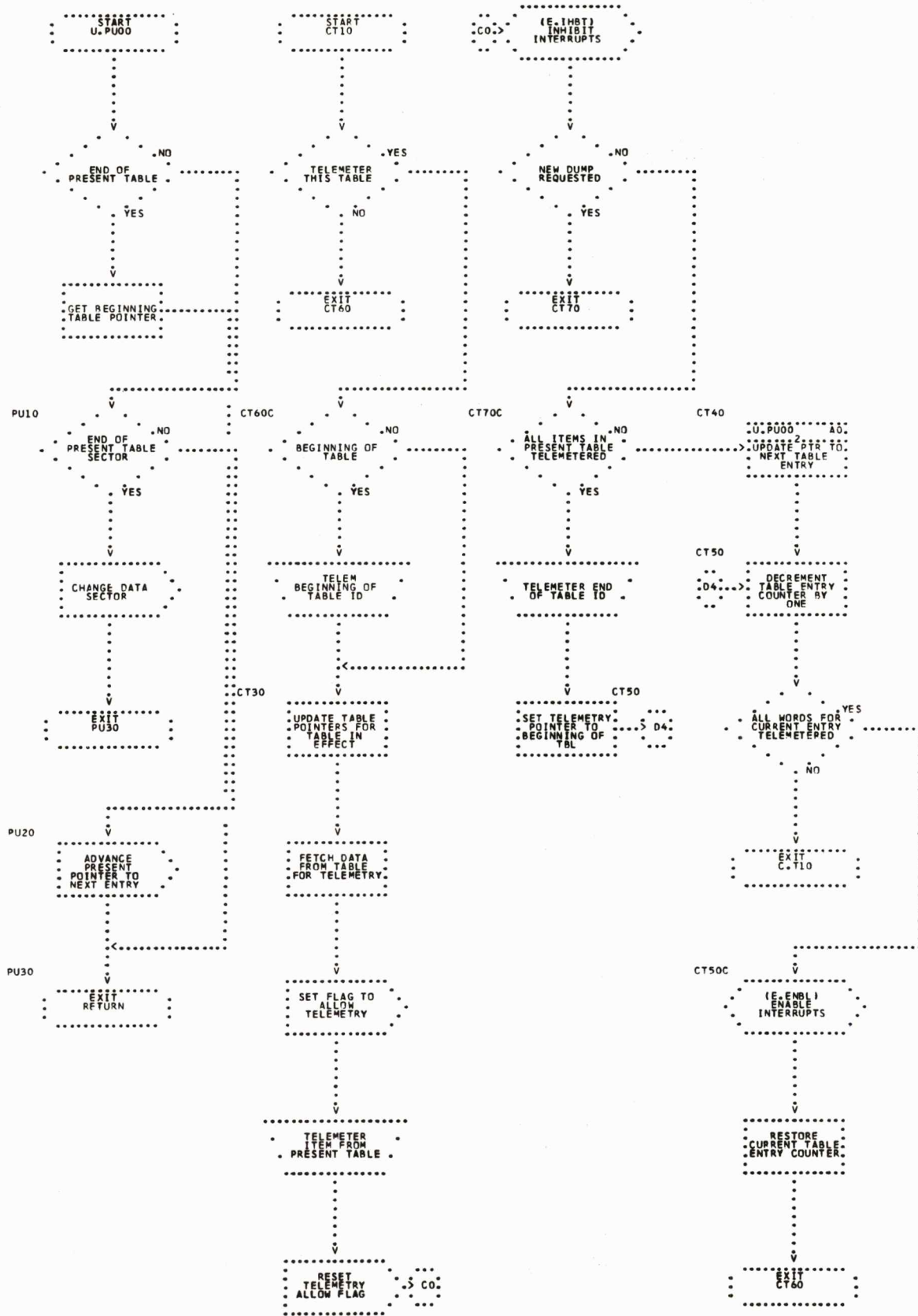
The flag V.KSTR must be set non-zero in the Flight Mode to indicate data compression is active. In the Flight Simulation Mode, the flag is set according to the Flight Simulator Indicator request for data compression.

COMPRESSED DATA TELEMETRY (CT)

M. CT00



COMPRESSED DATA TELEMETRY (CT)



STC
Graphs

(THIS PAGE HAS BEEN LEFT BLANK INTENTIONALLY)

2.9 DISAGREEMENT BIT PROCESSING (DG)

Function

The function of the Disagreement Bit Processing module is to determine the conditions under which LVDA hardware logic has set the disagreement bit and to select a good resolver for minor loop processing.

Entries

M.DG00 Entered from the Minor Loop module for all axes.

External

U.D000 Discrete Output Register Set/Reset Utility

U.TR00 Real Time Telemetry Delay Utility

Input Data

D.VDGW Raw gimbal read

D.VDGS Count of failures in disagreement bit circuitry

D.GMLT HOP constant to telemetry delay or telemetry bypass

D.VAFR Count of times A resolver determined bad

D.VBFR Count of times B resolver determined bad

D.VABF Count of times A & B resolvers determined bad

D.VHDB Count of times B multiplexer circuitry defective

Output Data

D.VDGS Count of failures in disagreement bit circuitry

D.VAFR Count of times A resolver determined bad

D.VBFR Count of times B resolver determined bad

D.VABF Count of times A & B resolvers determined bad

D.VHDB Count of times B multiplexer circuitry defective

D.FDBF Disagreement multiplexer failure flag

D.VGDF Flag for gimbal angle disagreement

*
*

Status Words

D.VMC4 Mode Code 24
D.VICR Internal Control Register

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.VDGW	0570	Error word

Initial Conditions

The following items are to be initialized at mission start:

D.VDGS Count of failures in disagreement bit circuitry
D.VAFR Count of times A resolver determined bad
D.VBFR Count of times B resolver determined bad
D.VABF Count of times A & B resolvers determined bad
D.VHDB Count of times B multiplexer circuitry defective

Notes

The Disagreement Bit Processing module has three unique exits to indicate the results of its processing. Control is returned to one of the three instruction locations following the HOP instruction to M.DG00 as follows:

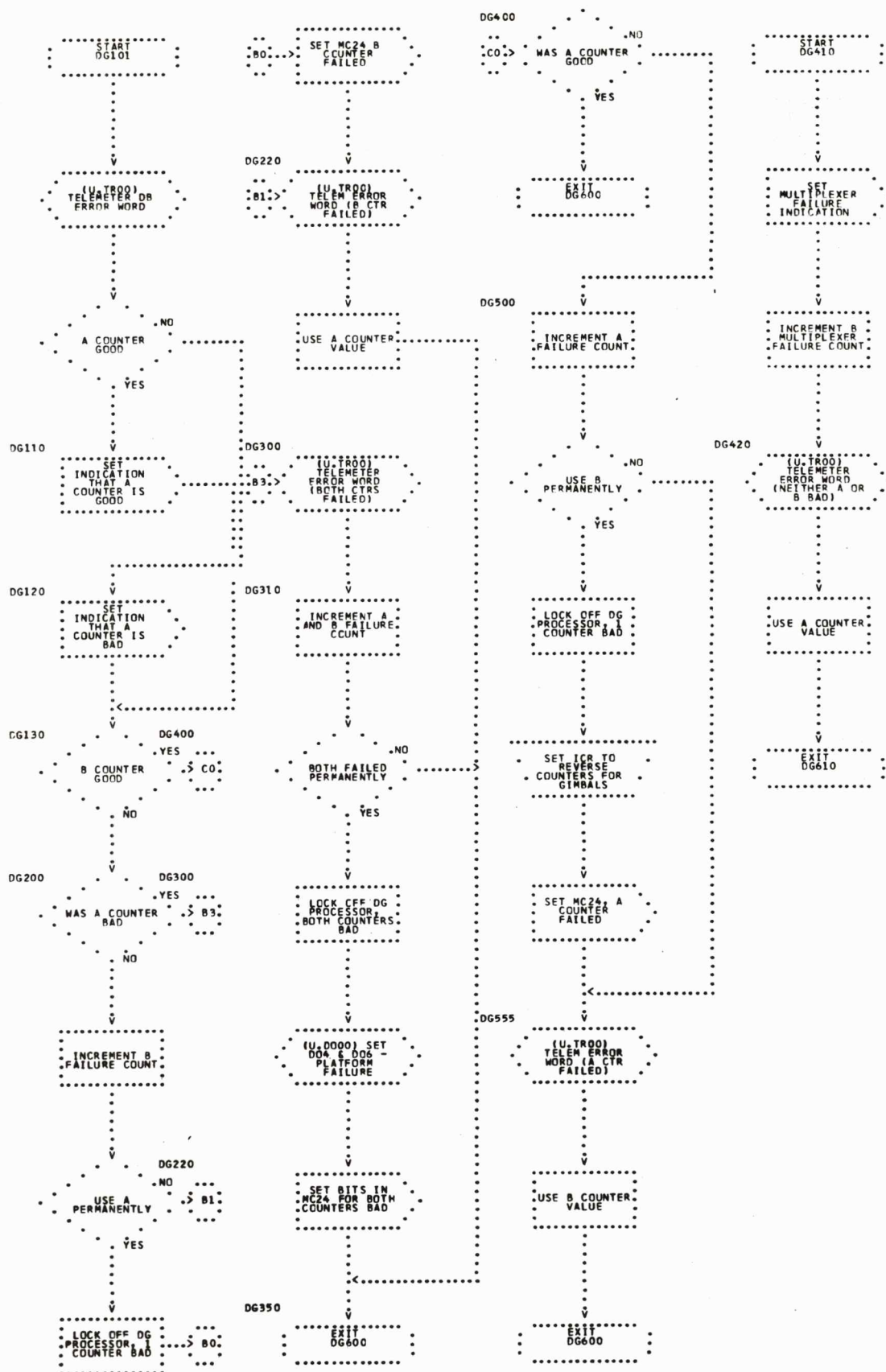
Word j	HOP* M.DG00
Word j+1	(Exit 1)
Word j+2	(Exit 2)
Word j+3	(Exit 3)

Exit 1: Either a cross over of the gimbals caused the DB to be set or a hardware DB circuitry set the DB invalidly. The gimbal is to be used for processing of the actual attitude.

Exit 2: No valid counter was found by using the test counters.

Exit 3: A valid counter was found when the test counters were checked.

The good counter is to be used for processing of the actual attitude.



2.10 DISCRETE OUTPUT REGISTER SET/RESET UTILITY (DO)

Function

This utility is entered whenever any bit is required to be set or reset in the LVDA Discrete Output Register (DOR).

Entries

U.D000 Entered by any module to set a specified bit or bits in the DOR.

U.D010 Entered by any module to reset a specified bit or bits in the DOR.

External

M.PC80 Data Compression Store Utility

U.TR00 Real Time Telemetry Delay Utility

E.ENBL Enable interrupts

E.IHBT Inhibit interrupts

Input Data

D.VDCT Time from GRR (in seconds positioned for data compression) *

D.FPHS Phase indicator flag: *

(-) = Boost

(+) = Orbit (any bits except sign bit) *

Output Data

V.DOFF Mask word which specifies which DIN's are to be processed
when they change from ON to OFF *

D.VDPM Mask word which specifies which DIN's are to be processed
when they change from OFF to ON

D.VM05 Rate limit for all ladders in Minor Loop *

D.GMLT HOP constant which inhibits Minor Loop Error Telemetry

F.IGCO Inhibit High Speed Loop flag *

Status Words

D.VDOR Discrete Output Register
D.VMC7 Mode Code 27

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.DORS	0425	DOR Telemetry word

Initial Conditions

The DOR Status Word, D.VDOR, must be initialized at mission start.

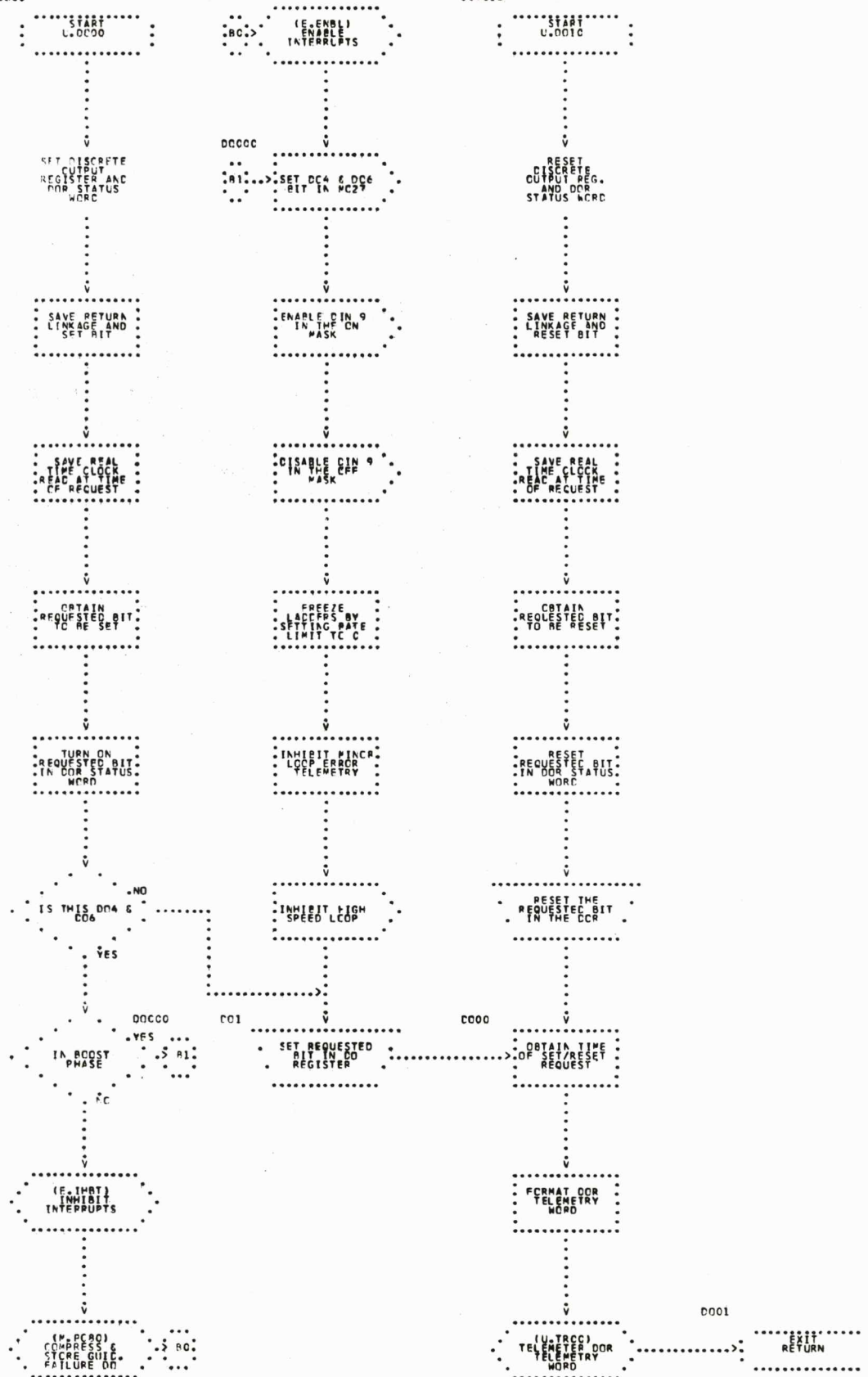
Notes

A 13 bit image, which indicates all bits to be set or reset, must be in the accumulator upon entry.

A test is made to determine if D04 and D06 are to be set. If these Discrete Output Register bits are requested, the time, tag, and DOR status word will be compressed and stored during Phase 2 in Data Compression Table 4. The designated bit in Mode Code 27 will be set when D04 and D06 are set.

U.DC00

U.DC10



2.11 DISCRETE PROCESSOR (DP)

Function

The Discrete Processor module accesses individual discrete and time base modules to perform all functions required when it detects change in the logic of any of the discretes specified for processing.

Entries

M.DP00 Entered from the Non-Interrupt Sequencer once per computation cycle during Phase 1 and once per second from the Phase II Control module during Phase 2.

External

M.DP09 Discrete Input 9

M.DP14 Discrete Input 14

M.DP15 Discrete Input 15

E.ENBL Enable interrupts

E.IHBT Inhibit interrupts

Input Data

D.VDPM Mask word which specifies which DIN's are to be processed when they change from OFF to ON

V.DOFF Mask word which specifies which DIN's are to be processed when they change from ON to OFF

D.VDPT Mask word which specifies which time base discretes are to be processed.

D.GTBD HOP constant to execute Time Base module

Output Data

D.VTRS Real Time Clock at discrete read time

V.DINT Current reading of the Discrete Input Register

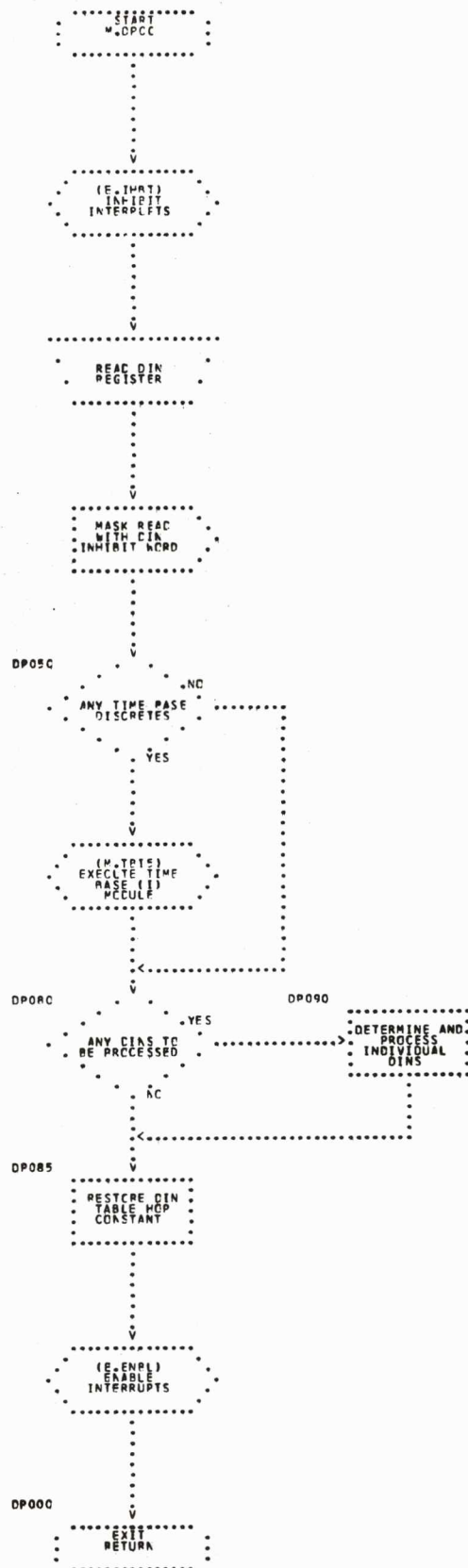
Status Word

None.

Telemetry

None.

M.CPCO



2.12 DISCRETE INPUTS 14 and 15 (DP1)

Function

This module will adjust the backup S-IB thrust (acceleration) and mass flow rate parameters and provide the Time Tilt module with an indicator for modification of the pitch tilt arrest time and pitch profile to compensate for an early S-IB Inboard or Outboard Engine Out condition (DI14 or 15).

Entries

- M.DP14 Entered from the Discrete Processor execution during a mission dependent interval and upon detection of the S-IB Outboard Engine Out discrete (DI14).
- M.DP15 Entered from the Discrete Processor execution during a mission dependent interval and upon detection of the S-IB Inboard Engine Out discrete (DI15).

External

- U.TR00 Real Time Telemetry Delay Utility

Input Data

- D.VDPM Mask word which specifies which DI's are to be processed when they change from OFF to ON
- D.VTAS Total time elapsed in mission *
- D.VTRS Real Time Clock at discrete read time
- D.KT1 Time Base 1 start time from GRR *
- D.KT2 Time Base 2 start time from GRR *
- D.KT3 Time Base 3 start time from GRR *

Output Data

D.VDPM	Mask word which specifies which DI's are to be processed when they change from OFF to ON	
D.VEOF	Engine out multiplication factor for backup F/M calculation	*
D.VEOT	Time of S-IB engine out	*
D.VSND	Constant used to compute V.ACZR in Accelerometer Processing	

Status Words

D.VMC5	Mode Code 25	
D.VMC4	Mode Code 24	
D.FILE	Flight Program Status	*

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.VDIT	0465	Engine Out Discrete Telemetry

Initial Conditions

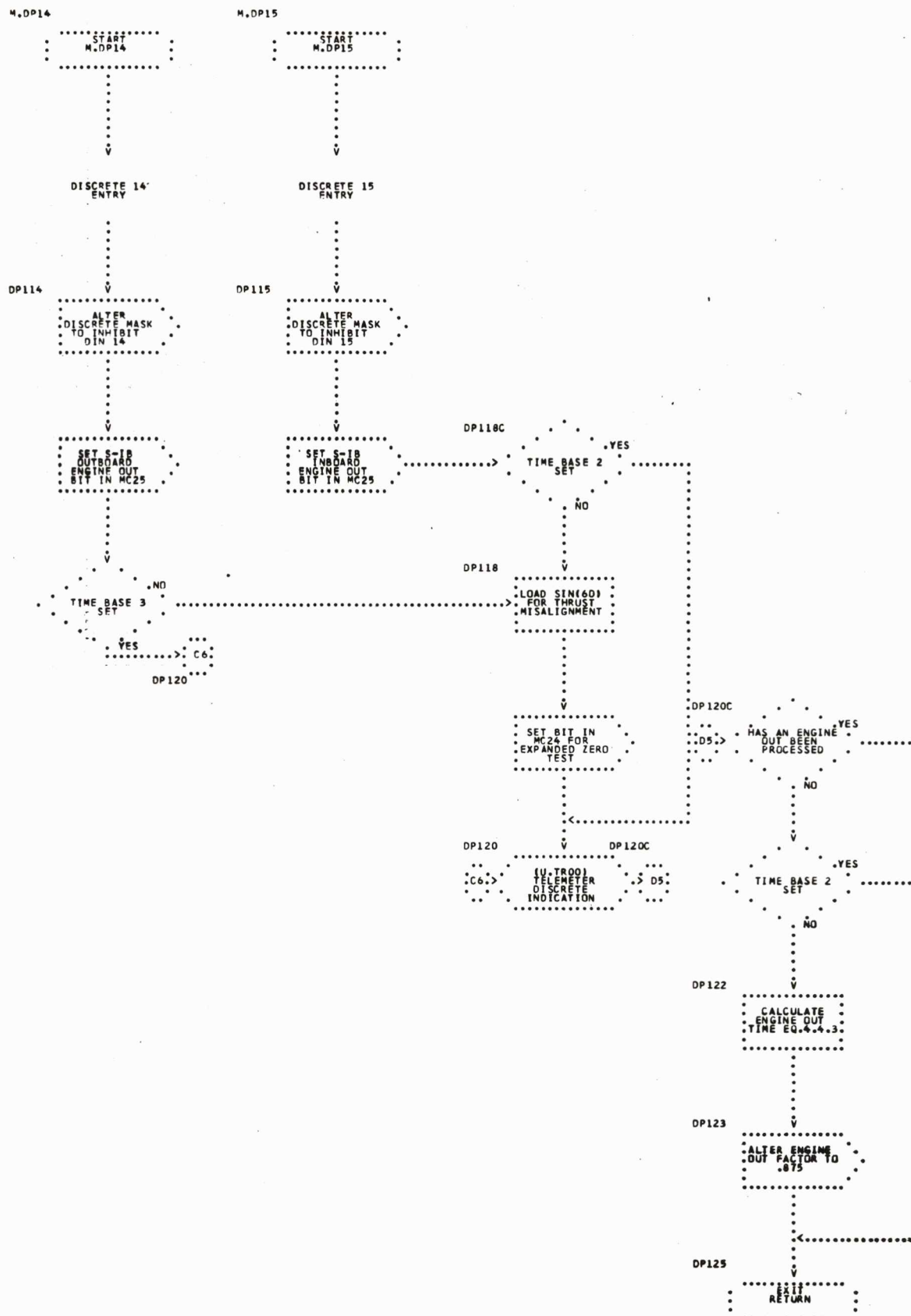
DI's 14 and 15 must be inhibited at start of mission; i.e., they must not be initially included in the Discrete Processor mask word D.VDPM.

Notes

DI's 14 and 15 must be enabled (included in mask word D.VDPM) at a mission dependent time in Time Base 1.

DI's 14 and 15 must be inhibited (removed from mask word D.VDPM) as follows:

- o DI's 14 and 15 when processed by this module
- o DI14 at start of Time Base 3
- o DI15 at start of Time Base 3



2.14 DISCRETE INPUT 9 (DP3)

Function

This module will set status and control indications when the spacecraft assumes or relinquishes attitude control (DI9).

Entries

M.DP09 Entered by the Discrete Processor on detection of a logic change of the SC Control discrete (DI9).

External

E.IHBT Inhibit interrupts

*

E.ENBL Enable interrupts

*

Input Data

V.DINT Current reading of the Discrete Input Register

D.VDPM Mask word which specifies which DI's are to be processed when they change from OFF to ON

Output Data

D.VDPM Mask word which specifies which DI's are to be processed when they change from OFF to ON

V.DOFF Mask word which specifies which DI's are to be processed when they change from ON to OFF

V.ROL Present and previous roll commands in sequential locations

V.YAW Present and previous yaw commands in sequential locations

V.PIT Present and previous pitch commands in sequential locations

Status Words

D.VMC7 Mode Code 27

D.FILE Flight Program status

*

Telemetry

None.

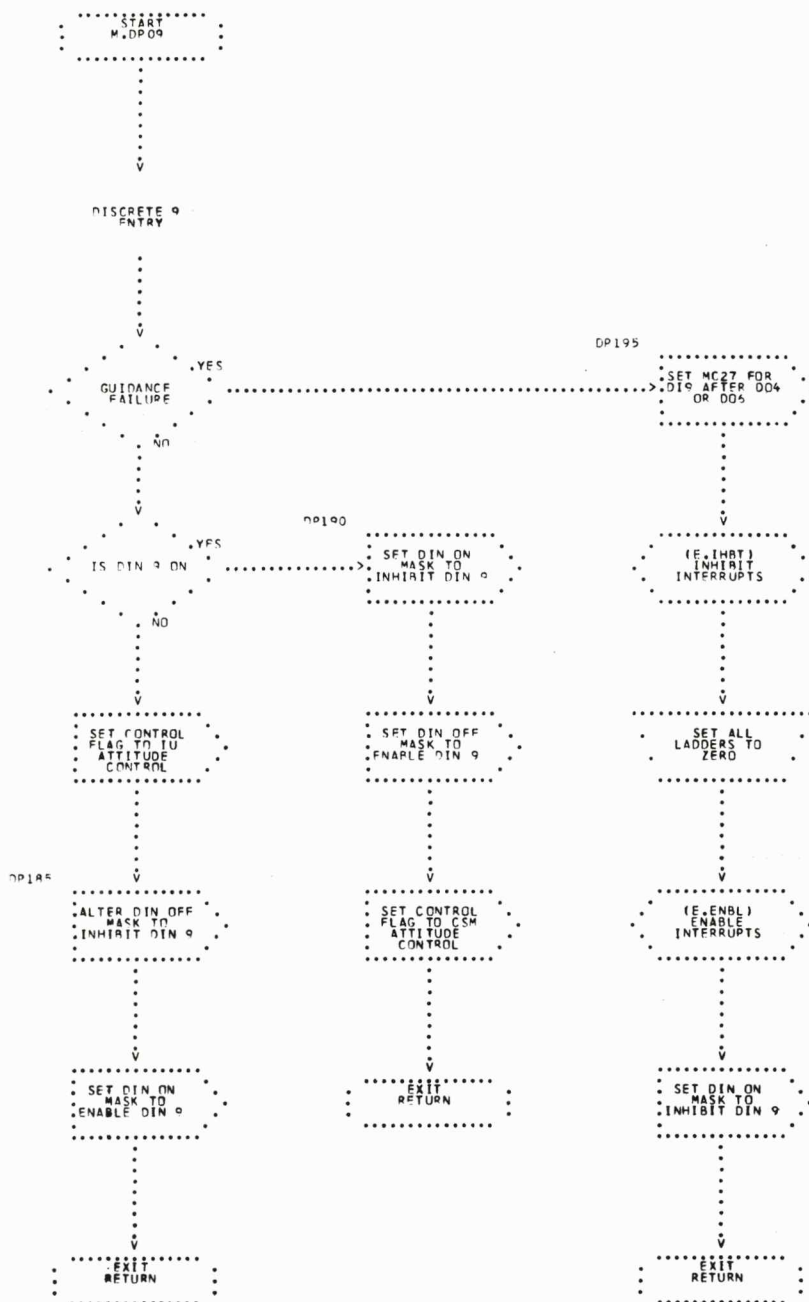
Initial Conditions

DI9 must be inhibited at the start of the mission, i.e., it must not be included in the Discrete Processor mask words D.VDPM and V.DOFF.

Notes

DI9 must be enabled at a mission dependent time in Time Base 4.

M.DP09



2.15 DIGITAL COMMAND SYSTEM (DCS) (DS)

Function

The function of the DCS module is to provide a real time method of controlling the LVDC Flight Program and obtaining program information. The DCS module will process all legal DCS commands and provide proper communication between the LVDC and the ground support equipment.

This module will also provide communications with the ground support equipment when initialized by the Preflight DCS Linkage module.

Entries

M.DS00 Entered via Interrupt Processor upon recognition of DCS interrupt.

External

M.SS00	Switch Selector Check	
U.TR00	Real Time Telemetry Delay Utility	
U.SSRQ	Generalized Switch Selector (M.SS00)	
E.Q	Water Coolant Valve Check (M.WM00)	
E.Q	Navigation Update Implementation (M.NU00)	
E.Q	Compressed Data Dump Telemetry (M.CT00)	
E.Q	Memory Dump Telemetry (M.DT00)	
E.Q	Maneuver Processor (M.MP55)	
E.Q	APS Attitude Control Test (M.AC00)	*
E.Q	Time Base 5 Start Logic (M.TB50)	
E.Q	Maneuver Processor (M.MP50)	*
E.QT	Navigation Update Implementation (M.NU00)	
E.QT	Maneuver Processor (M.MP55)	
E.QT	Time Base 5 Start Logic (M.TB50)	
E.QT	Maneuver Processor (M.MP50)	*
E.ENBL	Enable interrupts	
E.IHBT	Inhibit interrupts	

Input Data

F.DSEN	Data enable flag	*
F.SSAC	Switch Selector active flag	
D.FDTL	Memory Dump/compressed data dump in progress flag	*
V.GBIA	Time base bias to be implemented	
D.VTMM	Elapsed time in mission from GRR to last time update	*
D.VTRR	Elapsed total time in current time base	*
D.VRTC	Real time clock reading at last time update	*

Output Data

D.VUXS	X Navigation Update position
D.VUYS	Y Navigation Update position
D.VUZS	Z Navigation Update position
D.VUDX	X Navigation Update velocity
D.VUDY	Y Navigation Update velocity
D.VUDZ	Z Navigation Update velocity
D.VNUT	Navigation Update implementation time
D.FTUP	Time Update waiting flag
D.FDTL	Memory Dump/compressed data dump in progress flag
D.FPFT	Preflight Memory Dump Telemetry flag
D.VDLO	Memory Dump start address
D.VDHI	Memory Dump stop address
V.GBIA	Time Base bias to be implemented
V.GSSB	Formatted Generalized Switch Selector word
V.I	Inclination angle
V.LAMB	Descending node
V.IV	Insertion velocity
V.COR	Cutoff radius
V.THET	Path angle
V.LAMD	Rate of change of descending node
D.GRRT	Predicted time of GRR
V.TLWC	Time of Launch Window close
D.SMCK	Sum Check Factor for Target Load Parameters

F.DSEN	Data enable flag	
D.XREF	Inertial X-axis reference angle for commanded maneuver	
D.YREF	Inertial Y-axis reference angle for commanded maneuver	
D.ZREF	Inertial Z-axis reference angle for commanded maneuver	
F.MTYP	Commanded maneuver type flag	
F.TSOM	Time to implement commanded maneuver	
D.VTDS	Time to start Time Base 5	
D.VTLD	Duration of LOX dump included in De-orbit command	*
D.VTHD	Duration of LH2 dump included in De-orbit command	*
D.VMO6	Pitch and yaw ladder magnitude limit	
D.VM16	Roll ladder magnitude limit	
F.CLML	Ladder magnitude limit change word	
D.VER8	Accumulated eight second composite EMR word	
F.EMR8	Dark period EMR accumulation flag	
D.DLVR	Delta velocity to be attained for S-IVB/IU De-orbit	
D.ELVR	Delta velocity check allow flag	*
G.AC00	APS attitude control test HOP constant (M.AC00)	*
D.MPTB	Pointer for enabling the extraction maneuver	*
V.T4M6	Extraction enable start time	*
V.T4M7	Maneuver 7 start time	*
V.T4M8	Maneuver 8 start time	*

Status Words

D.VMC7	Mode Code 27
D.VLNK	DCS Status
D.VASW	Switch Selector Request word
D.VMC4	Mode Code 24

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.DSST	4030	DCS Mode Status Word (Telemetered 2 times)
V.DSDT	4574	DCS Data Status Word (Telemetered 2 times)
V.DSER	4055	DCS Error Word (Telemetered 2 times)

Initial Conditions

None.

Notes

The seven navigation update parameters must be transmitted in the order: ZDS, XDS, YDS, ZS, XS, YS, and NUPTIM. These parameters will be stored in memory module 4, sector 15, locations 371 through 377.

The nine target load parameters will be stored in memory module 0, sector 16, locations 360 through 370, and will be transmitted in the following order: V_T , R_T , θ_T , i , λ_O , $\dot{\lambda}$, T_{GRRO} , T_{LWC} , TLCK.

The four De-Orbit parameters will be stored in memory module 2, sector 14, location 374 through 377 and will be transmitted in the following order: T_{DSS} , T_{LDD} , T_{HDD} , DELVR.

The following programs are executed from this module (entry point in parentheses):

- (D.S260) Update Sequence Time Base
- (D.S330) Navigation Update
- (D.S380) Generalized Switch Selector
- (D.S430) Memory Dump
- (D.S510) Terminate
- (D.S770) Water Valve Control
- (D.S790) Target Load
- (D.S810) Generalized Maneuver Routine
- (D.S860) Return to Timed Maneuver Sequence
- (D.S900) Ladder Magnitude Limit
- (D.S925) Compressed Data Dump
- (D.S950) S-IVB/IU De-Orbit
- (D.S960) Extraction Maneuver Enable

*

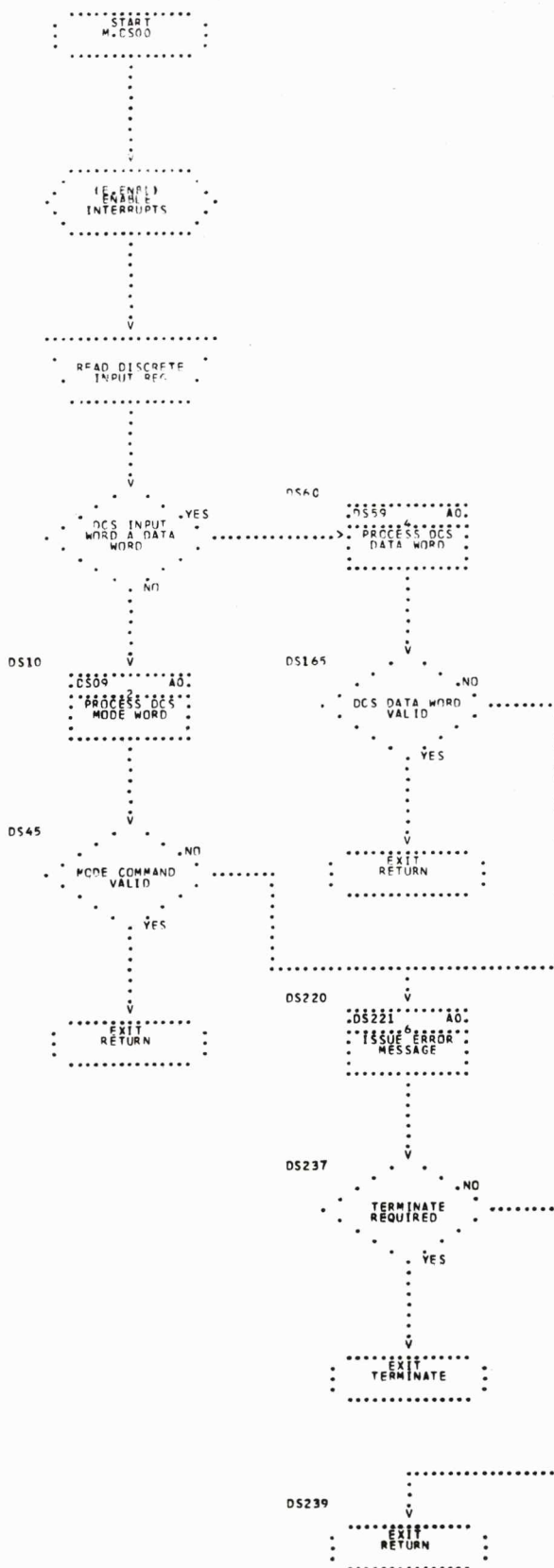
All DCS error messages are set through telemetry by the DCS Common Error Handler whose entry point is DS220.

The Computer Reset Pulse (CRP) is set and reset by the CRP Set/Reset Routine whose entry point is D.S200.

All DCS output data is sent through telemetry by the Output Data Routine whose entry point is DS250.

DIGITAL COMMAND SYSTEM (DCS) (DS)

M.DS00



```

graph TD
    subgraph DS15
        S15_1[START DS09] --> S15_2[READ DCS MODE WORD]
        S15_2 --> S15_3{IS THE MODE WORD FORMAT GOOD}
        S15_3 -- YES --> S15_4[SET ERROR CODE = 10]
        S15_3 -- NO --> S15_5[EXIT DS220]
        S15_4 --> S15_5
    end

    subgraph DS18
        S18_1[IS MODE COMMAND EXPECTED] -- YES --> S18_2[MEMORY DUMP COMMANDED]
        S18_2 -- NO --> S18_3[COMPRESSED DATA DUMP COMMANDED]
        S18_3 -- YES --> S18_4[MEMORY OR COMPRESSED DATA DUMP IN PROG]
        S18_4 -- NO --> S18_5[MODE COMMAND VALID T-T IS MISSION]
        S18_5 -- YES --> S18_6[PROCESS DCS MODE COMMAND]
        S18_6 --> S18_7[PROCESS COMMAND ROUTINE]
        S18_7 --> S18_8[RETURN]
    end

    subgraph DS20
        S20_1[MODE COMMAND VALID T-T IS MISSION] -- YES --> S20_2[PROCESS DCS MODE COMMAND]
        S20_2 --> S20_3[PROCESS COMMAND ROUTINE]
        S20_3 --> S20_4[RETURN]
    end

    subgraph DS21
        S21_1[PROCESS DCS MODE COMMAND] --> S21_2[PROCESS COMMAND ROUTINE]
        S21_2 --> S21_3[RETURN]
    end

    subgraph DS22
        S22_1[MODE COMMAND VALID T-T IS MISSION] -- YES --> S22_2[PROCESS DCS MODE COMMAND]
        S22_2 --> S22_3[PROCESS COMMAND ROUTINE]
        S22_3 --> S22_4[RETURN]
    end

    subgraph DS24
        S24_1[SEQUENCE BIT CORRECT IN MODE INPUT] -- YES --> S24_2[SET ERROR CODE = 24]
        S24_2 --> S24_3[EXIT DS220]
        S24_1 -- NO --> S24_3
    end

    subgraph DS30
        S30_1[START DS30] --> S30_2[SET ERROR CODE = 14]
        S30_2 --> S30_3[EXIT DS220]
    end

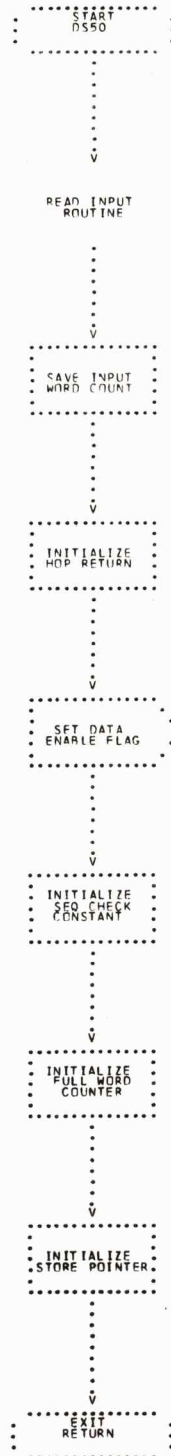
    subgraph DS35
        S35_1[SET ERROR CODE = 64] --> S35_2[EXIT DS220]
    end

    subgraph DS36
        S36_1[SET ERROR CODE = 20] --> S36_2[EXIT DS220]
    end

    subgraph DS40
        S40_1[START DS40] --> S40_2[SET ERROR CODE = 74]
        S40_2 --> S40_3[EXIT DS220]
    end

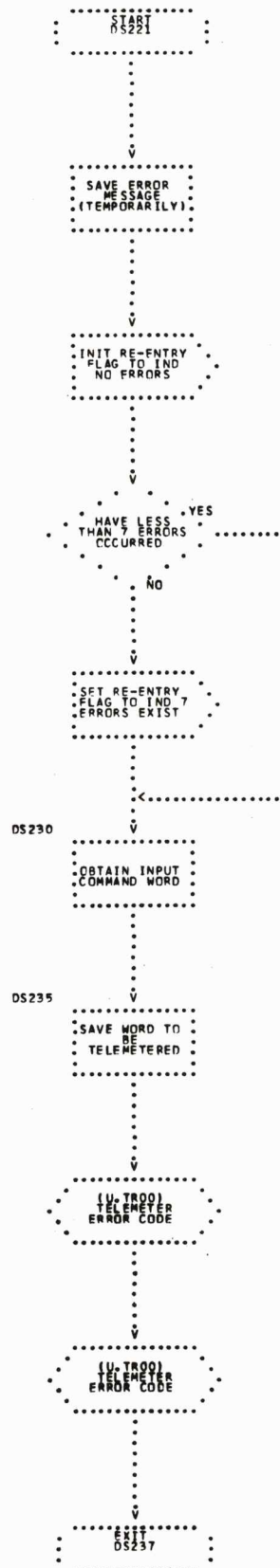
    S15_3 -- NO --> S15_5
    S15_4 --> S15_5
    S18_1 -- NO --> S18_3
    S18_3 -- YES --> S18_4
    S18_4 -- NO --> S18_5
    S18_5 -- YES --> S18_6
    S18_6 --> S18_7
    S18_7 --> S18_8
    S20_1 -- YES --> S20_2
    S20_2 --> S20_3
    S20_3 --> S20_4
    S21_1 --> S21_2
    S21_2 --> S21_3
    S22_1 -- YES --> S22_2
    S22_2 --> S22_3
    S22_3 --> S22_4
    S24_1 -- YES --> S24_2
    S24_2 --> S24_3
    S24_1 -- NO --> S24_3
    S30_1 --> S30_2
    S30_2 --> S30_3
    S35_1 --> S35_2
    S36_1 --> S36_2
    S40_1 --> S40_2
    S40_2 --> S40_3

```

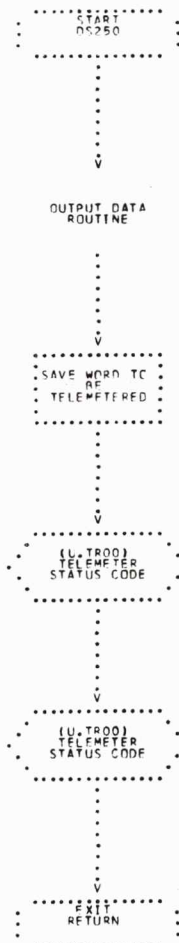




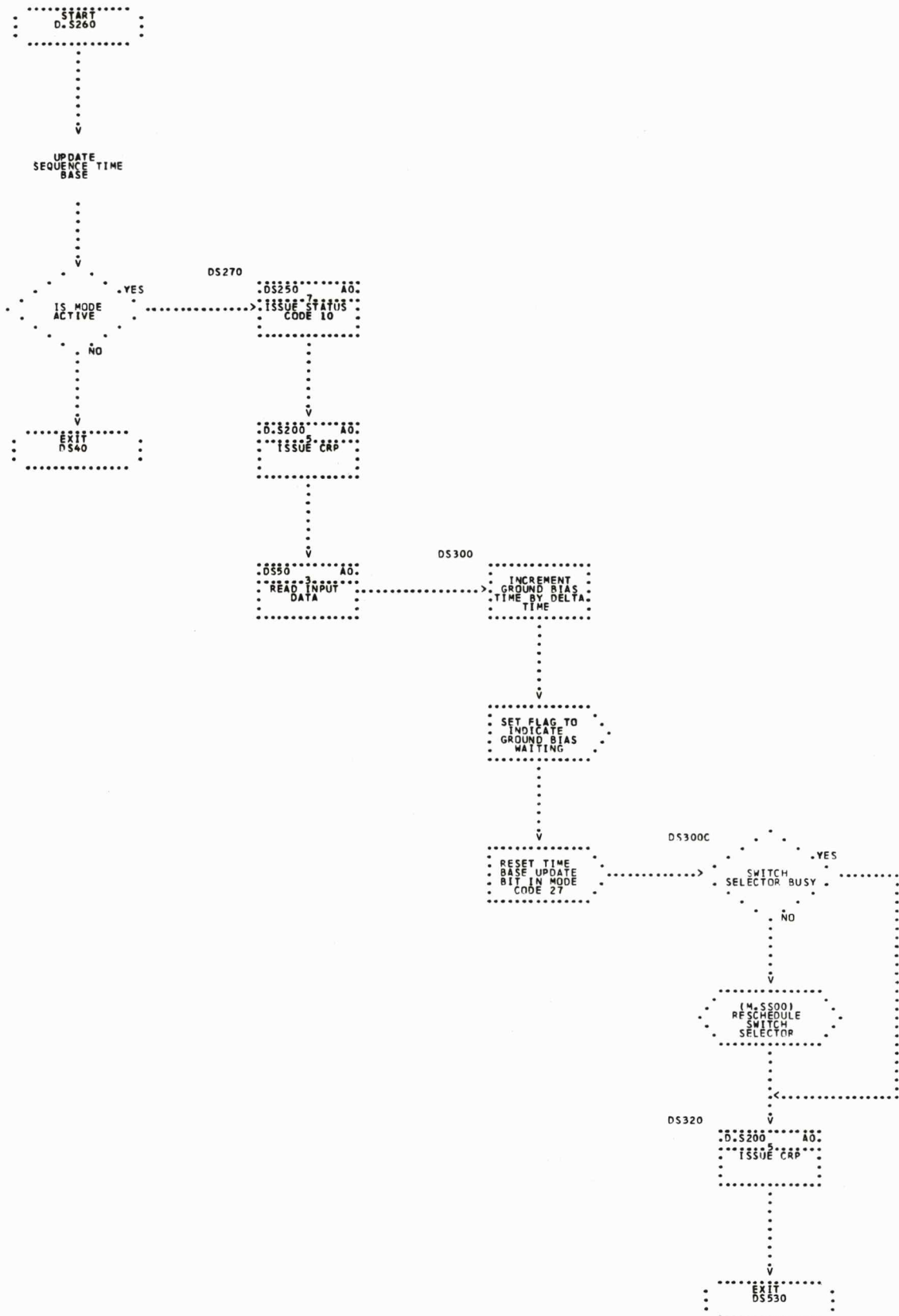




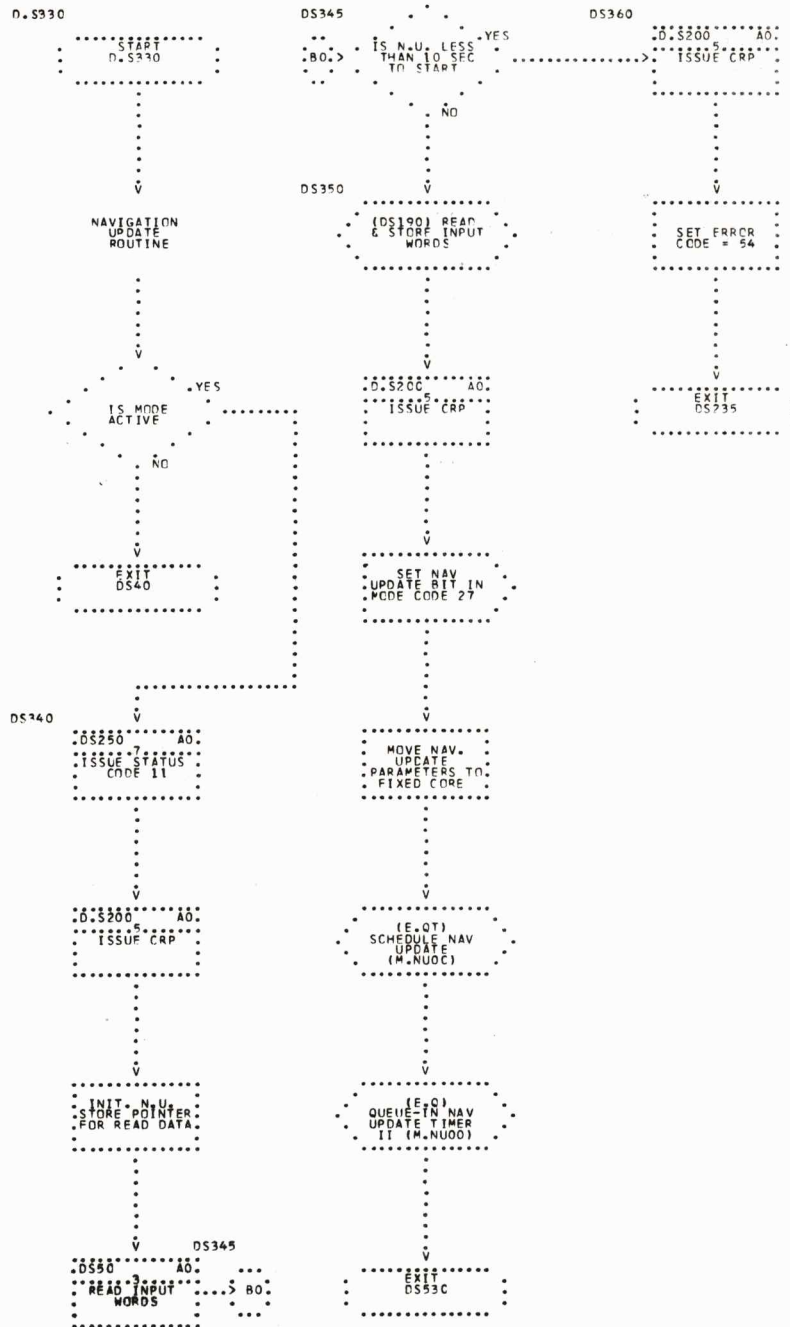
DIGITAL COMMAND SYSTEM (DCS) (DS)



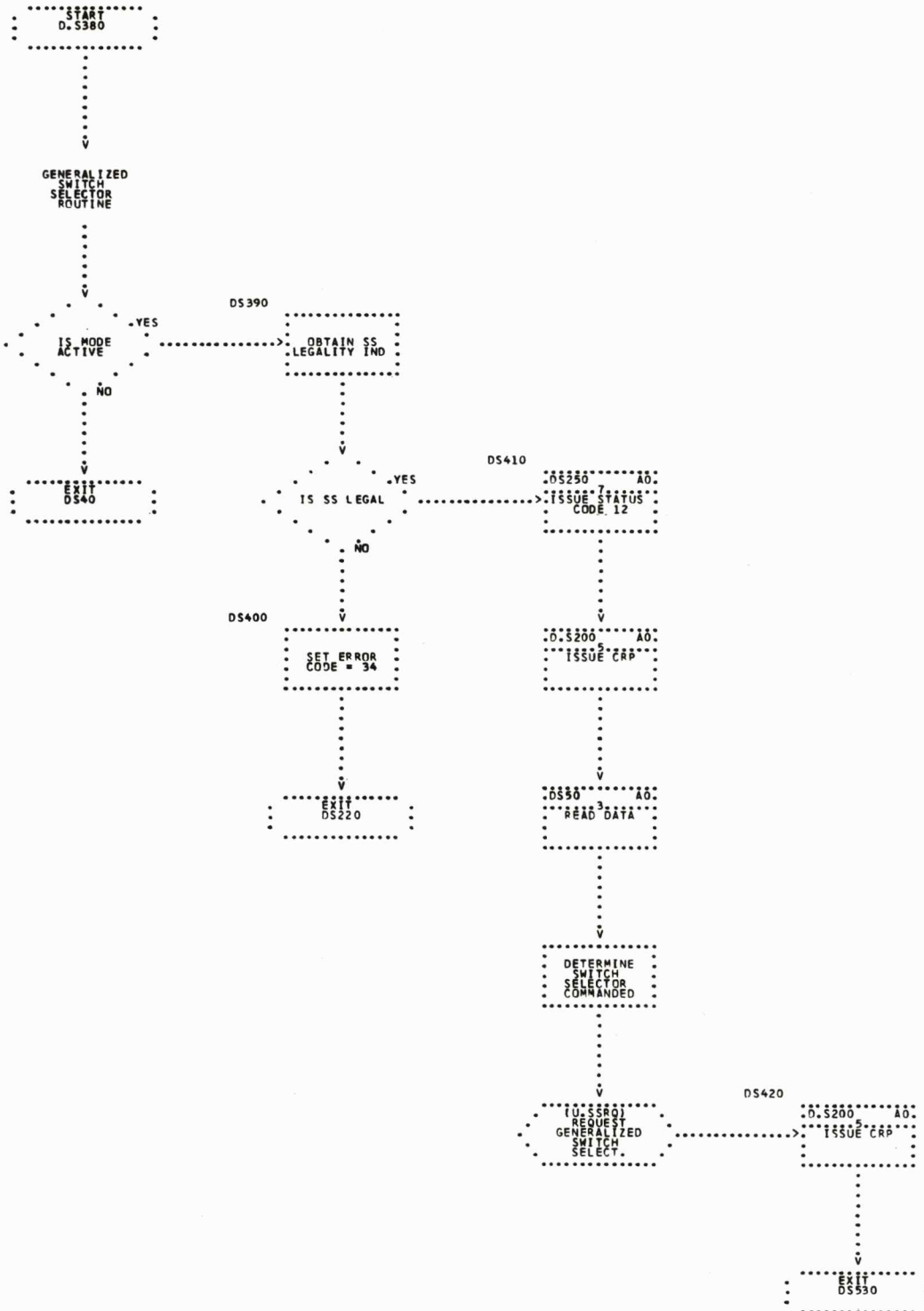
D.S260



DIGITAL COMMAND SYSTEM (DCS) (DS)

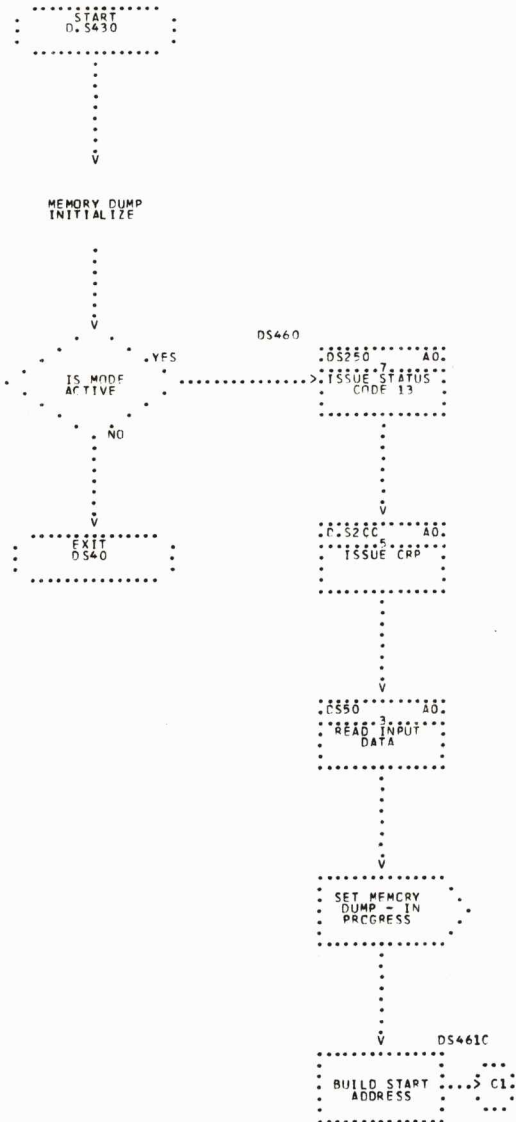


D.S380

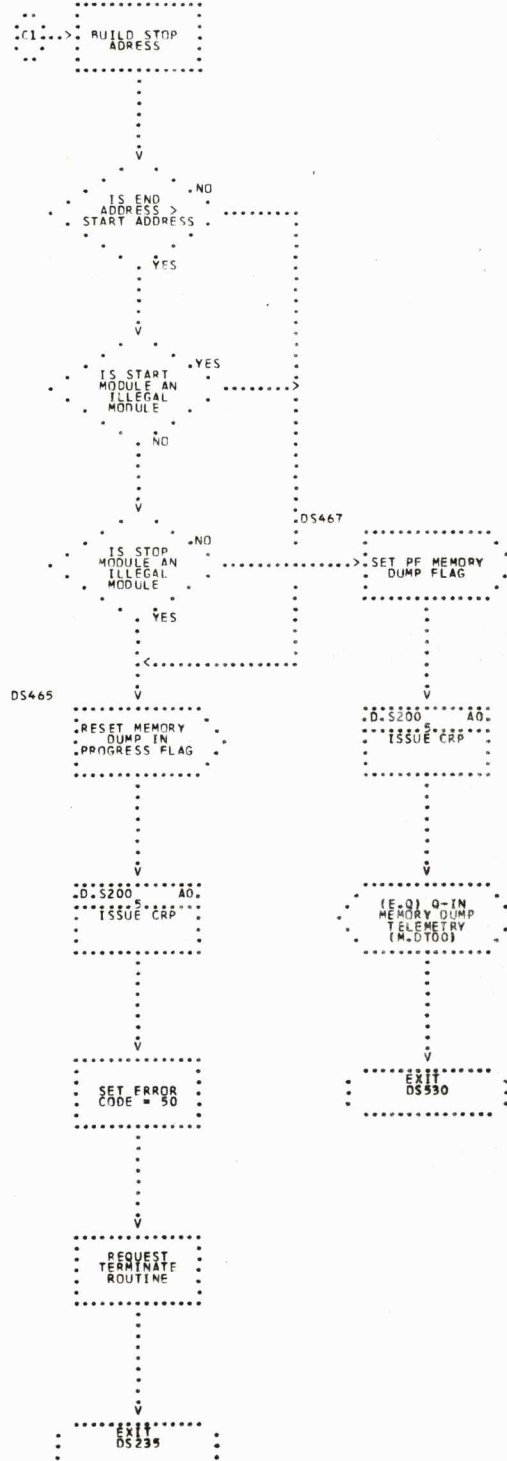


DIGITAL COMMAND SYSTEM (DCS) (DS)

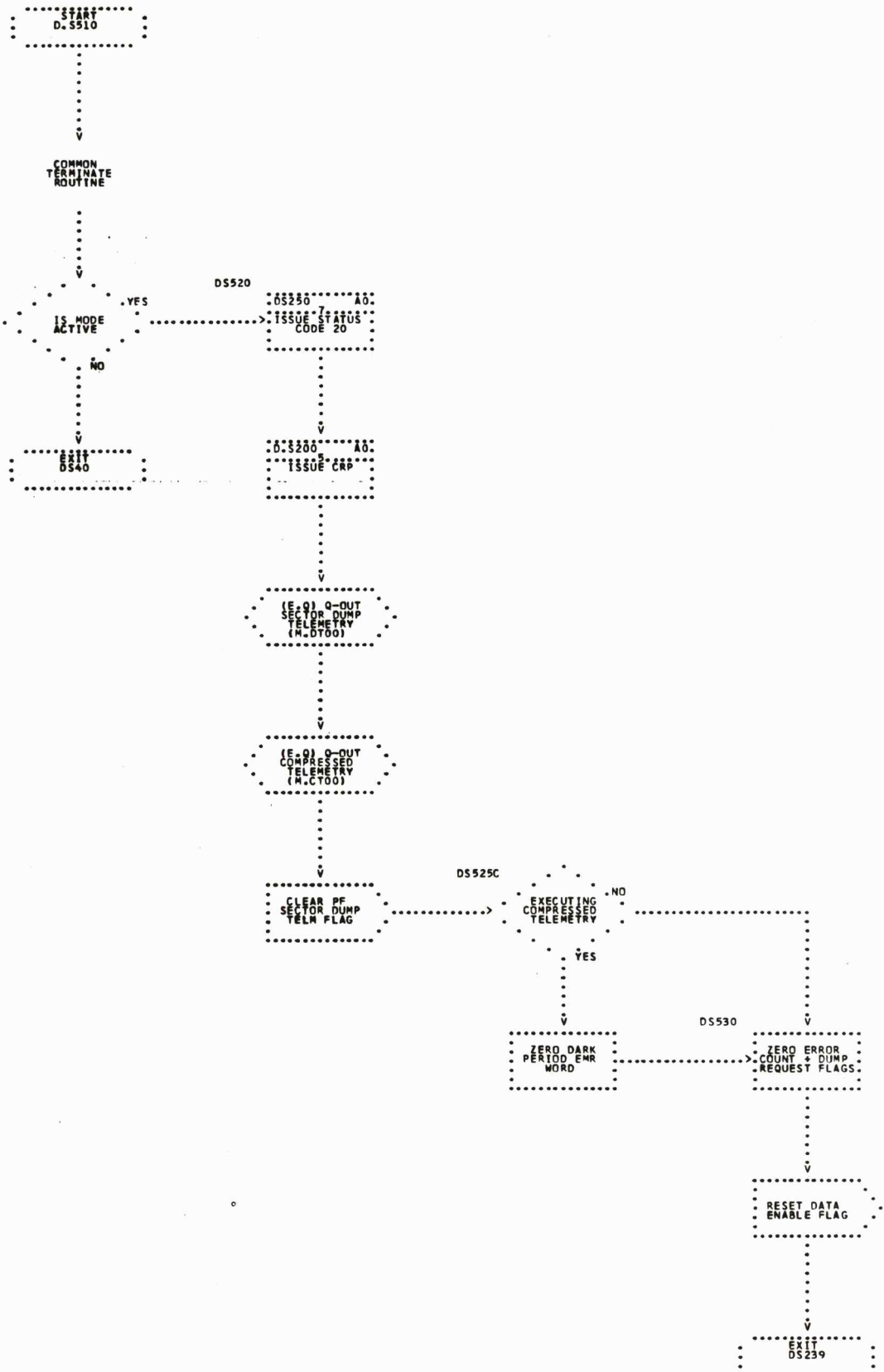
D.5430



DS461C

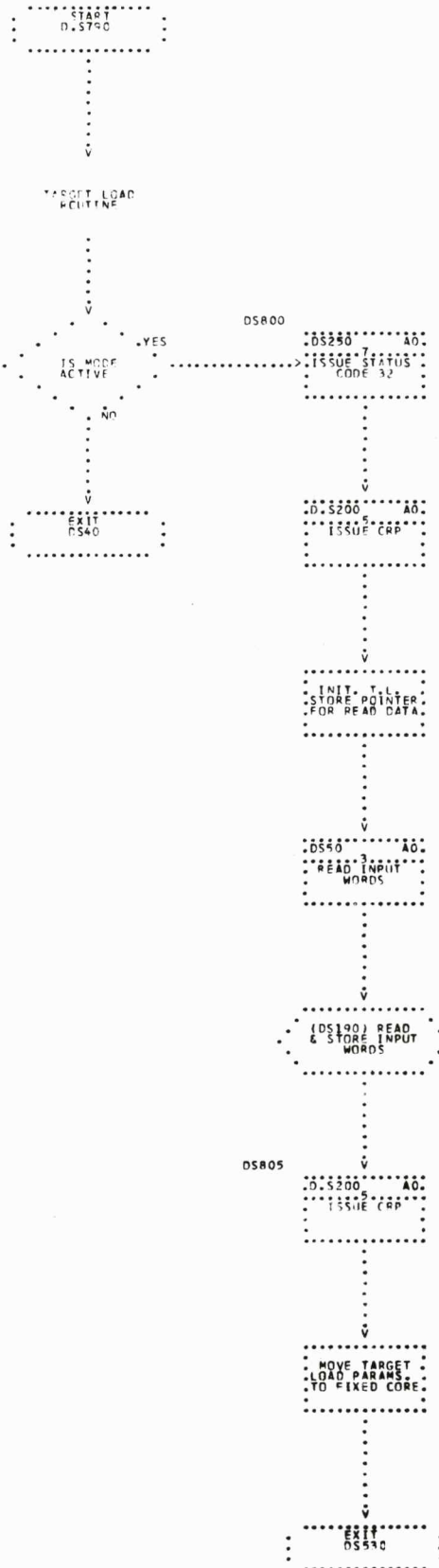


D.S510



DIGITAL COMMAND SYSTEM (DCS) (DS)

D.S700



DIGITAL COMMAND SYSTEM (DCS) (DS)

D.S770

START
D.S770

V

WATER VALVE
CONTROL

V

IS MODE
ACTIVE

YES

DS780

DS250 A0.
ISSUE STATUS
CODE 45

NO

EXIT
DS40

DS5200 A0.
ISSUE ERP

RESET INHIBIT
WATER CONTROL
VALVE IN MC27

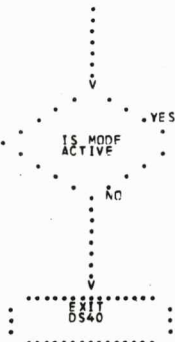
(E.O.)
QUEUE-OUT
WATER CONTROL
VALVE LOGIC
(P.WM00)

EXIT
DS530

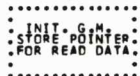
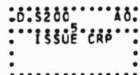
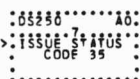
D.SR10



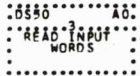
GENERALIZED
MANUEVER
ROUTINE



DSR20



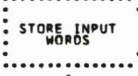
DSR22C



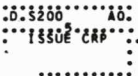
DSR22C



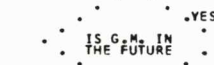
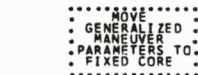
DSR24



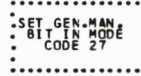
DSR24C



DSR24C



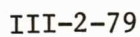
DSR25



DSR26



D.S860



```

      .....
      START
      D-S900
      .....

```

LADDER
MAGNITUDE
LIMIT

IS MODE
ACTIVE

EXIT
DS40

```

:CS250*****AO:
>:ISSUE STATUS:
:  CODE 41

```

.D.S200 A0:
..ISSUE⁵ CRP..
.:

```

DS50 ..... AO.
READ INPUT
DATA

```

```

: : DETERMINE : :
: : COMMANDED : :
: : LADDER : :
: : MAGNITUDE : :
: : LIMIT : :

```

COMMANDED
LADDER LIMIT
 > 15.3
DEGREES

```
.....  
:SET LIMITS TO:  
:  COMMANDED  
:    LIMITS  
:
```

```

> SET LADDER
> MAGNITUDE
> LIMIT TO 15.3
> DEGREES

```

```

.....
: SET FLAG TO
>: INDICATE
: COMMANDED LML
: IN USE

```

```

      V
.....
*  RESET APS
*  CONTROL LOSS

```

062

```

      SET LML DCS
.D6...>ACCEPTED BIT
      IN MC27

```

(E.Q)
 QUEUE-OUT APS
 ATTITUDE
 CONTROL TEST
 (M.ACOO)

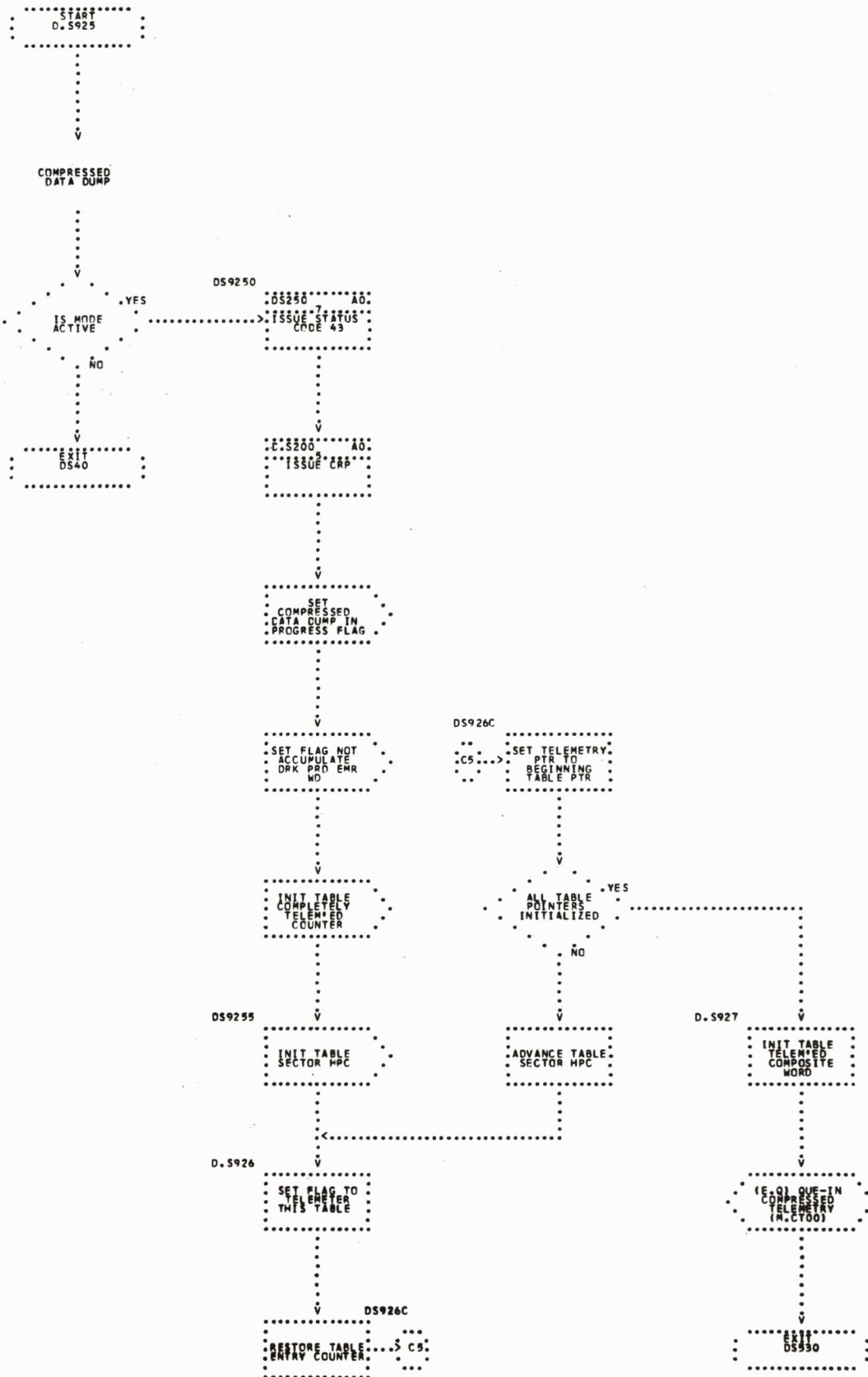
V

0-5200 AOJ
ISSUE^S CRP

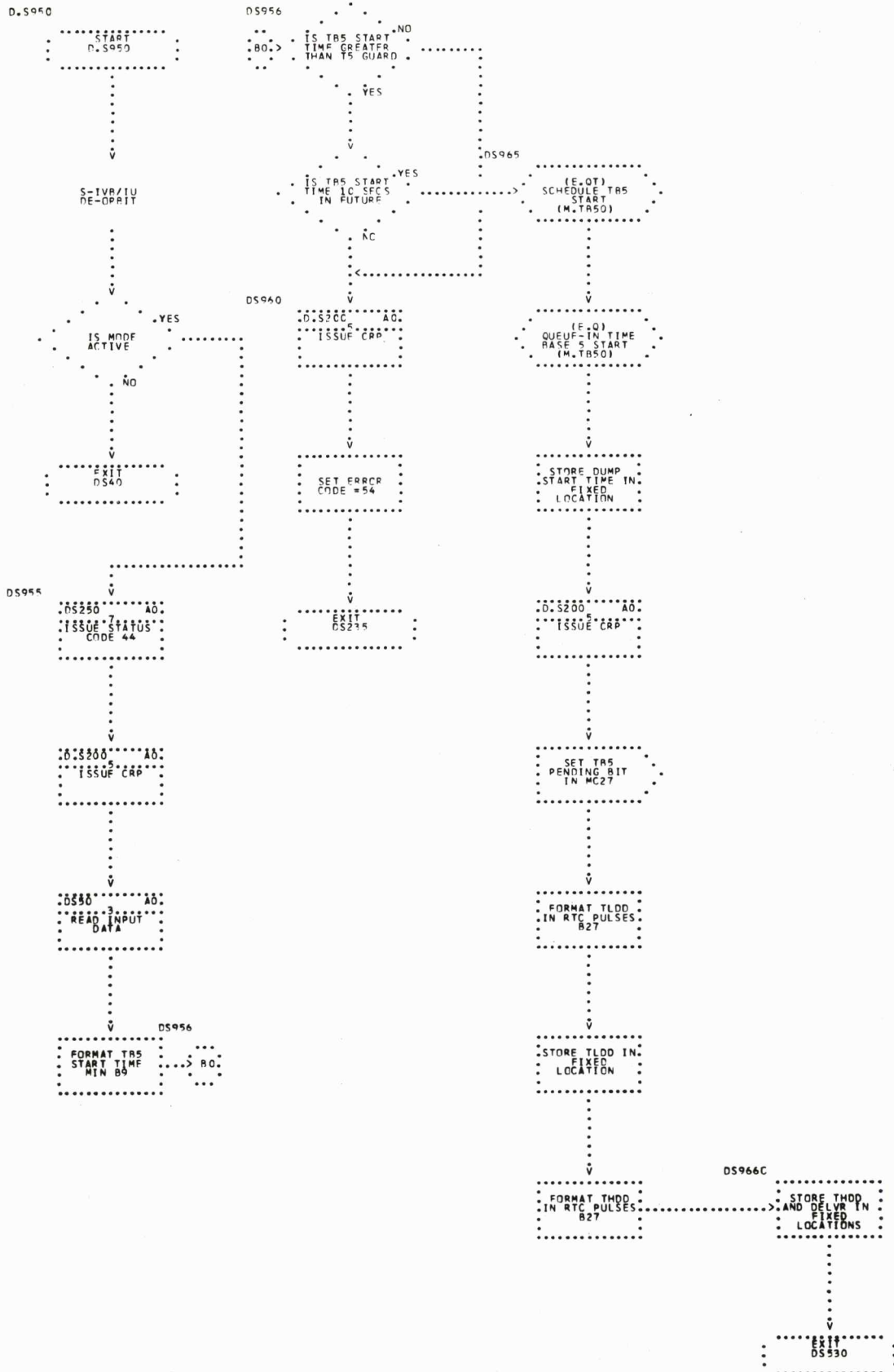
EXIT
DS530

DIGITAL COMMAND SYSTEM (DCS) (DS)

D.S925

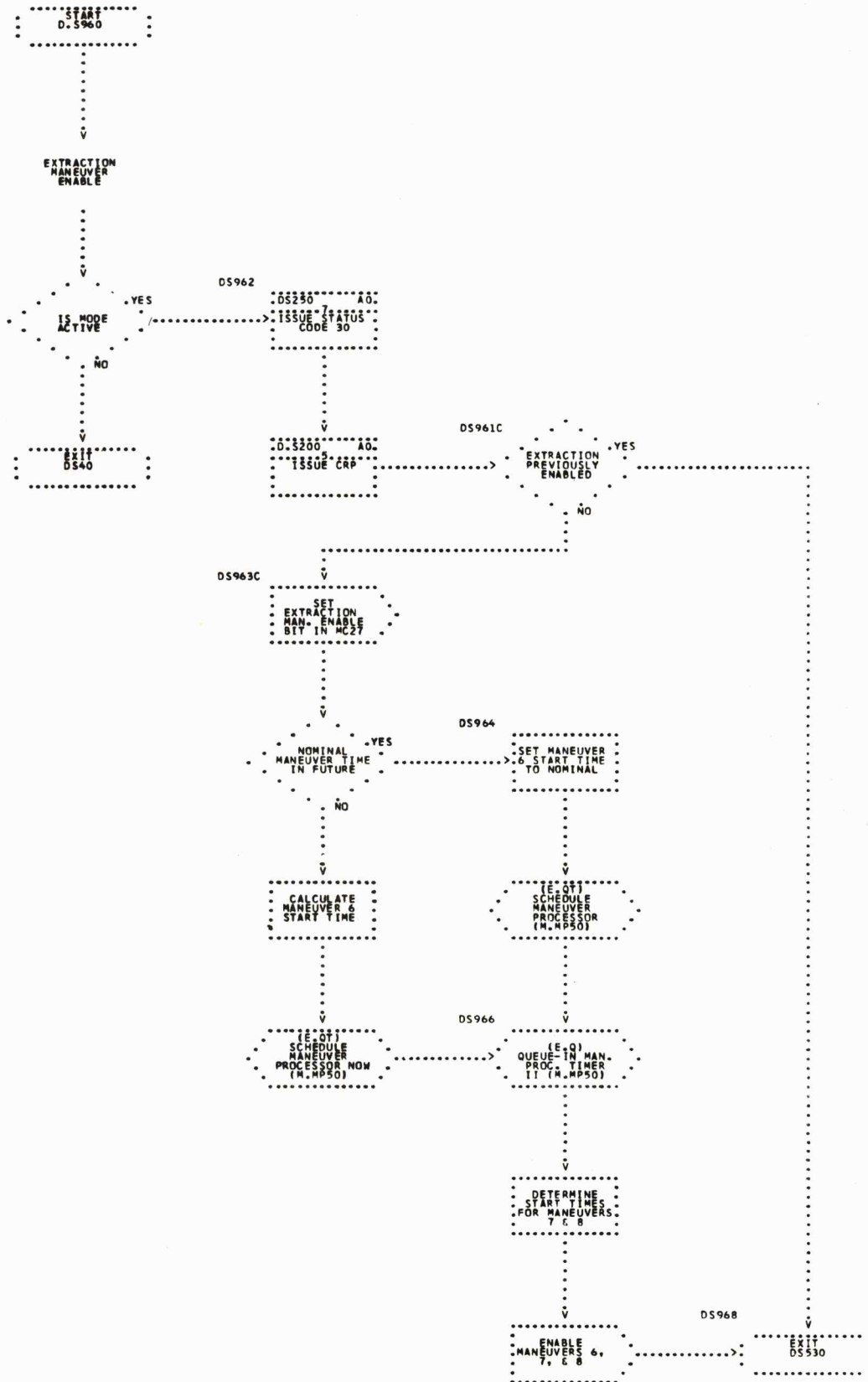


DIGITAL COMMAND SYSTEM (DCS) (DS)



DIGITAL COMMAND SYSTEM (DCS) (DS)

D.S960





2.16 MEMORY DUMP TELEMETRY (DT)

Function

The Memory Dump Telemetry module telemeters the contents of the requested LVDC memory location(s) when a DCS memory dump is requested.

Entries

M.DT00 Entered from the Non-Interrupt Sequencer once a computation cycle when queued-in by the DCS Processor.

External

U.TR00 Real Time Telemetry Delay Utility

E.ENBL Enable interrupts

E.IHBT Inhibit interrupts

E.Q Memory Dump Telemetry (M.DT00)

Input Data

D.VDLO Memory Dump start address

*

D.VDHI Memory Dump stop address

*

D.FDTL Memory dump/compressed data dump in progress flag

*

Output Data

D.FDTL Memory dump/compressed data dump in progress flag

*

D.FPFT Preflight DCS dump in progress flag

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.DDTL	4470	Header Word of Memory Dump Block
	4474	Word 1 of Memory Dump Block
	4500	Word 2 of Memory Dump Block
	4504	Word 3 of Memory Dump Block
	4510	Word 4 of Memory Dump Block
	4514	Word 5 of Memory Dump Block
	4520	Word 6 of Memory Dump Block
	4524	Word 7 of Memory Dump Block
	4530	Word 8 of Memory Dump Block
	4534	Word 9 of Memory Dump Block
	4540	Word 10 of Memory Dump Block
	4544	Word 11 of Memory Dump Block
	4550	Word 12 of Memory Dump Block
	4554	Word 13 of Memory Dump Block
	4560	Word 14 of Memory Dump Block
	4564	Word 15 of Memory Dump Block
	4570	Word 16 of Memory Dump Block

Initial Conditions

The memory dump in progress flag must be initialized to zero at start of mission.

Notes

On entry to this module, the starting and ending addresses (including sector and even module) must be given in D.VDLO and D.VDHI respectively, and the memory dump in progress flag, D.FDTL, must be set to minus.

If a dump is requested when the in progress flag, D.FDTL, is already set, the new request will override the current dump in progress. It is

*

therefore the responsibility of the requesting module to test the in progress flag D.FDTL.

If a dump is requested in which the starting or ending address is from an odd module, the data from the duplexed even module will be telemetered.

To terminate a dump in progress, a module must queue-out the Memory Dump Telemetry (M.DT00) and clear the memory dump in progress flag, D.FDTL, to zero.

2.17 F/M CALCULATIONS (DV)

Function

The function of the F/M Calculations module is to compute the actual acceleration from the accelerometer readings.

Entries

M.DV00 Entered from the Non-Interrupt Sequencer once per computation cycle when the module is queued in.

External

U.FD00 Fixed Point Divide Utility
U.SR00 Square Root Utility
U.TR00 Real Time Telemetry Delay Utility
E.Q F/M Calculations (M.DV00)

Input Data

D.VXDM X component of measured velocity
D.VYDM Y component of measured velocity
D.VZDM Z component of measured velocity
D.VDT Elapsed time between current and previous computation cycle
accelerometer reads in seconds *

D.VOXM Previous X component of measured velocity
D.VOYM Previous Y component of measured velocity
D.VOZM Previous Z component of measured velocity
F.FM Flag to indicate that F/M should be set to a constant value
D.GMFS Variable HOP constant to compute M/F smoothed or to avoid
computing M/F smoothed
D.VFOM Total vehicle acceleration determined from accelerometer readings *

Output Data

D.VFOM Total vehicle acceleration determined from accelerometer readings *

D.VXDD X component of platform measured acceleration

D.VYDD Y component of platform measured acceleration

D.VZDD Z component of platform measured acceleration

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.VXDD	1134	X component of platform measured acceleration
D.VYDD	1140	Y component of platform measured acceleration
D.VZDD	1130	Z component of platform measured acceleration
D.VFOM	1144	Total vehicle acceleration determined from accelerometer * readings

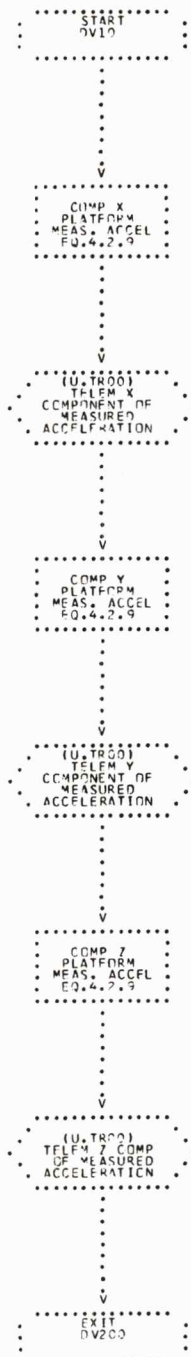
Initial Conditions

None.

Notes

During periods of flight in which the F/M computations may be unusually noisy, preset constants are used in place of computed values of D.VFOM. The F/M Calculations module will not be executed after setting D.VFOM to the constant value, as this module will then queue itself out. F.FM will contain the preset constant if it is a non-zero positive value. If the constant is zero, F.FM will be set negative.

This module will call the M/F Smoothing module as required in the mission timeline (controlled by the Events Processor).



2.18 ETC/BTC (EB)

Function

At the end of each computation cycle, the module will mark the end of the current telemetry cycle and the beginning of the next telemetry cycle with an end telemetry cycle (ETC) word followed by a begin telemetry cycle word (BTC). This module will also serve as a telemetry catch-up module; i.e., all telemetry items not telemetered from other modules will be telemetered by the ETC/BTC module. The ETC/BTC module will also reset system status words if necessary.

Entries

- M.EB00 Entered once per computation cycle from the Non-Interrupt Sequencer during Phase 1.
- M.EB10 Entered once per second from the Phase II Control module during Phase 2.

External

- M.OD00 Digital Output Multiplexer (DOM) Telemetry
- U.TR00 Real Time Telemetry Delay Utility

Input Data

- G.OD00 HPC to Digital Output Multiplexer Telemetry routine (M.OD00)

Output Data

None.

Status Words

D.VMC4 Mode Code 24
D.VMC5 Mode Code 25
D.VMC7 Mode Code 27
D.VDOR Discrete Output Register
D.VICR Internal Control Register
D.VEMR Error Monitor Register
D.FILE Flight Program Status

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.BTC	0575	Begin telemetry cycle word
K.ETC	0571	End telemetry cycle word
V.LOTW	0415	Mode Code 24
D.VMC5	0421	Mode Code 25
D.VMC7	0420	Mode Code 27
D.VDOR	0425	Discrete Output Register
V.DINT	0445	Discrete Input Register
D.VICR	0431	Internal Control Register
V.EMR	0435	Error Monitor Register

Initial Conditions

The begin telemetry cycle word (V.BTC) must be set to zero at mission start.

Notes

The ETC/BTC module must be the last module in the Phase I Non-Interrupt Sequential Control Table and the last module called by the Phase II Control module.

The Hop constant, G.OD00, is set to M.OD00 in Phase 1 Initialize and at start of TB4 to activate the Digital Output Multiplexer Telemetry module.

The Iterative Guidance Mode module resets the Hop constant to M.EB20 to deactivate M.OD00 at time for S-IVB cutoff.

M.EB00

```

      .----- .
      | START |
      | M.EB00 |
      |----- .
  
```

V

```

      .----- .
      | ENTRY IN |
      | BOOST MODE |
      | VIA |
      | NON-INTERRUPT |
      | SEQUENCER |
      |----- .
  
```

V

```

      .----- .
      | RESET FMR |
      | STATUS WORD |
      | FOR NEXT COMP. |
      | CYCLE |
      |----- .
  
```

V

```

      .----- .
      | (U.T000) |
      | TELEMETER FMR |
      | STATUS WORD |
      |----- .
  
```

V

```

      .----- .
      | RESET SYSTEM |
      | FLAG TO |
      | INDICATE MC24 |
      | TELEMETRY |
      |----- .
  
```

V

```

      .----- .
      | (U.T000) |
      | TELEMETER |
      | MODF CODE |
      | WORD 24 |
      |----- .
  
```

ER10

M.EB10

```

      .----- .
      | START |
      | M.EB10 |
      |----- .
  
```

V

```

      .----- .
      | ENTRY IN |
      | ORBIT MODE |
      | VIA PHASE 2 |
      | CONTROL |
      | MODULE |
      |----- .
  
```

V

ER10

```

      .----- .
      | EB12 2 AO. |
      | TEL. ETC/RTC |
      | ITEMS FOR |
      | BOOST/ORBIT |
      |----- .
  
```

V

ER20

```

      .----- .
      | UPDATE |
      | BEGINNING |
      | COMP CYCLE |
      | WORD |
      |----- .
  
```

V

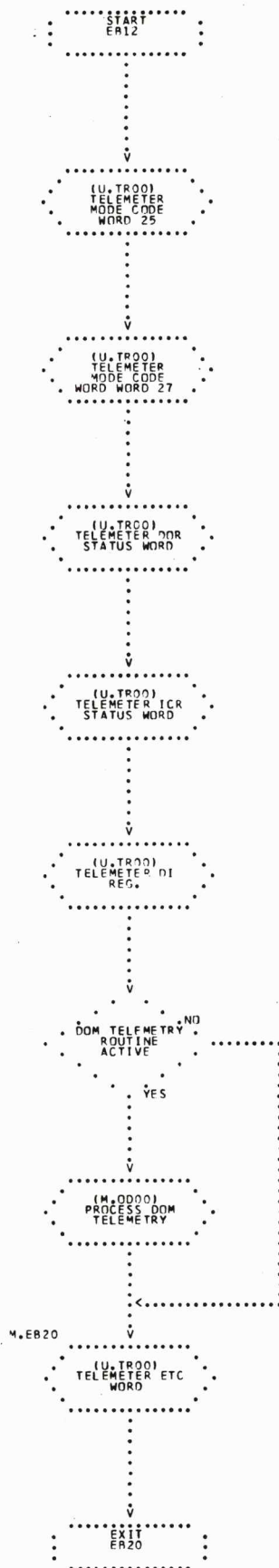
```

      .----- .
      | (U.T000) |
      | TELF4 |
      | BEGINNING |
      | COMP CYCLE |
      | WORD |
      |----- .
  
```

V

```

      .----- .
      | EXIT |
      | RETURN |
      |----- .
  
```



2.19 MINOR LOOP ERROR MONITOR (EC)

Function

The Minor Loop Error Monitor module has three basic functions:

- (1) To monitor the counters of resolver unreasonableness failures to select the resolver to be used by the Minor Loop module.
- (2) To monitor the disagreement bit failure counters to control counter selection by the Disagreement Bit Processing module.
- (3) To determine if ladder converter A is to be selected by the Minor Loop module.

Entries

M.EC00 Entered from Timer 2 at a 0.8 second rate during Phase 1. Entered from the Phase II Control module once per second during Phase 2.

External

E.IHBT Inhibit interrupts
E.ENBL Enable interrupts

Input Data

D.VXRE	X resolver reasonableness failure count	
D.VYRE	Y resolver reasonableness failure count	
D.VZRE	Z resolver reasonableness failure count	
D.VLDB	Count of times switched to ladder converter B	*
D.VDGS	Count of failures in disagreement bit circuitry	
D.VHDA	Count of A multiplexer failures	*
D.VHDB	Count of B multiplexer failures	*
V.IRE	Resolver failure limit 2	
D.VGDG	Flag for gimbal angle disagreement	*

Output Data

D.VXRE	X resolver reasonableness failure count	
D.VYRE	Y resolver reasonableness failure count	
D.VZRE	Z resolver reasonableness failure count	
D.VLDB	Count of times switched to ladder converter B	*
D.VDGS	Count of failures in disagreement bit circuitry	
D.VAFR	Count of times A resolver determined bad	*
D.VBFR	Count of times B resolver determined bad	*
D.VABF	Count of times A and B resolvers determined bad	*
D.VHDA	Count of A multiplexer failures	*
D.VHDB	Count of B multiplexer failures	*

Status Words

D.VICR Internal Control Register
D.VMC4 Mode Code 24
D.FILE Flight Program Status

Telemetry

None.

Initial Conditions

None.

Notes

$$V.IRE = (R_E * R_M * 3/5 + 1)$$

where

R_E = Rate Minor Loop Error Monitor cycled

R_M = Minor Loop Rate

MINOR LOOP ERROR MONITOR (EC)

4. EC00

START
M, EC00

V

(IF, INTR)
INITIALIZE
INTERRUPTS

V

EC010

EC400 A0.
TEST X
RESOLVER
FAILURES

V

EC400 A
TEST Y
RESOLVER
FAILURES

V

EC400 A0.
TEST Z
RESOLVER
FAILURES

V

SET FL
PROC. STATUS
WD TO IND
ERROR MONITOR
PASS

V

EC200

EC201 A0.
INSTR
DISAGREEMENT
BIT COUNTERS

V

EC300

EC301 A0.
DETERMINE
PROPER LANDER
CONVERTED

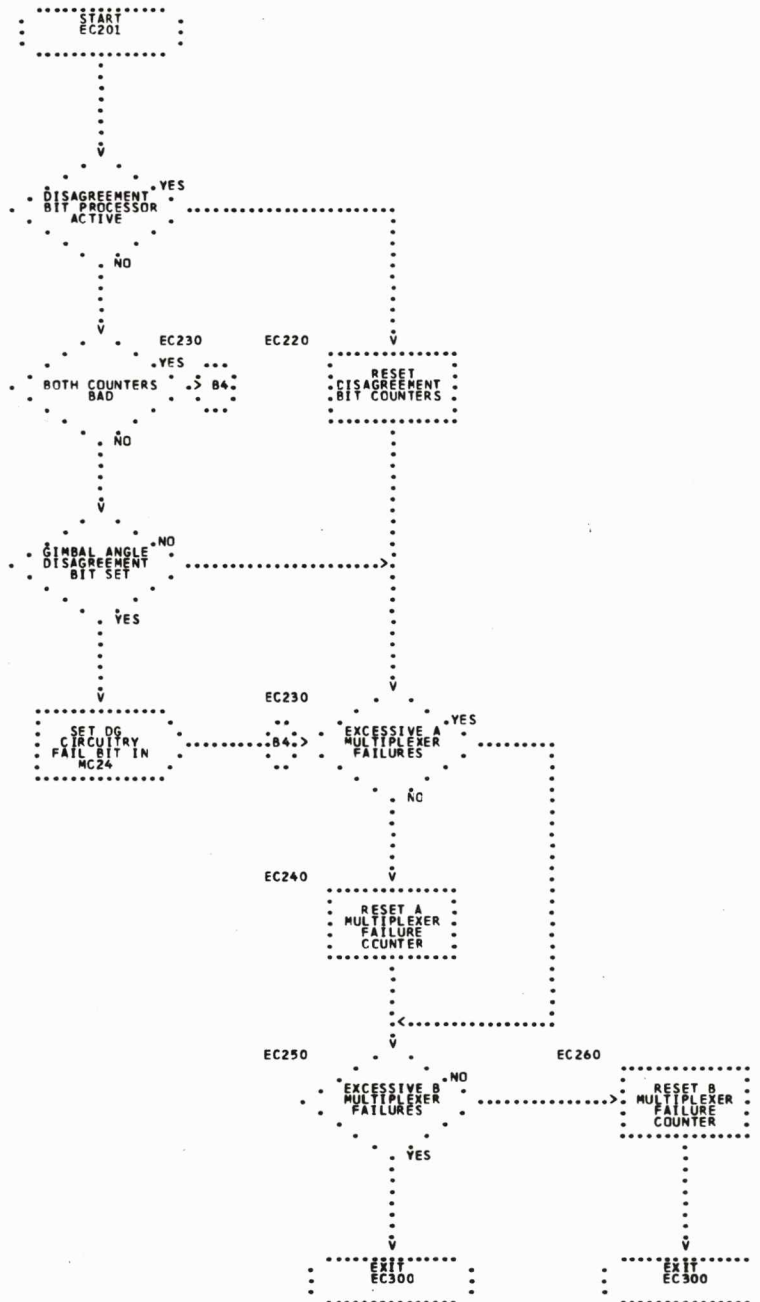
V

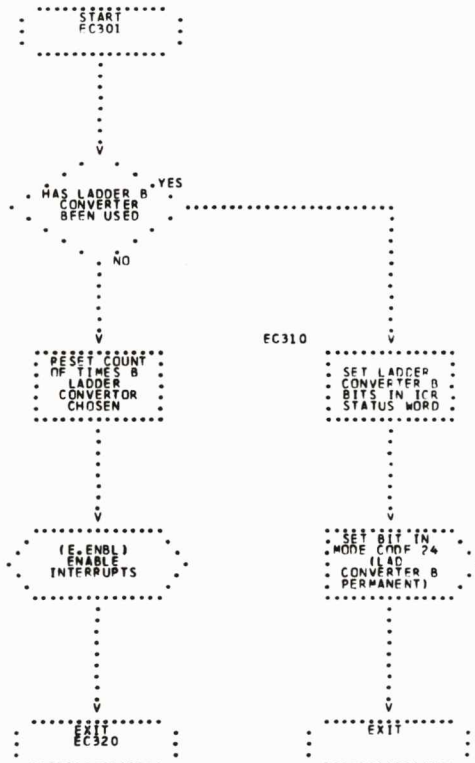
(IF, ENRI)
ENABLE
INTERRUPTS

V

EC320

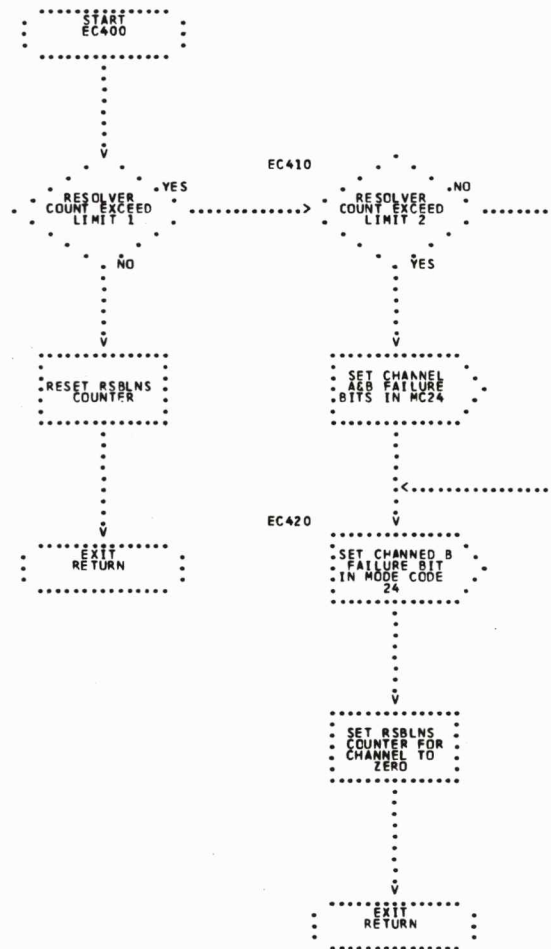
EXIT
RETURN





MINOR LOOP ERROR MONITOR (EC)

4



2.20 EVENTS PROCESSOR (EP)

Function

This module will schedule the time of execution of various time base referenced functions specified in a mission dependent table. When the time for execution of a scheduled event occurs, the Events Processor will give control to the appropriate event submodule which will perform the actual function.

Entries

M.EP00 Entered from Timer 2 when the time for a scheduled function occurs.

M.EP05 Entered from the Time Base Set module for proper initialization of the events table at each time base change.

External

L.Exxx Mission dependent event submodules

E.ENBL Enable interrupts

E.IHBT Inhibit interrupts

E.Q Events Processor (M.EP00)

E.QT Events Processor (M.EP00)

Input Data

D.FTBC Time base change flag

Output Data

None.

Status Words

None.

Telemetry

None.

Initial Conditions

The table advance pointers L.ER00 and L.TBRO must be initialized to the first event table entry. Variable HOP constants for the entries in the Timed Events Table which are affected by the mode of operation (Flight or Flight Simulation) must be initialized according to the mode of operation selected. For example, the entries for the timed start of time bases are active only for the Repeatable Flight Simulation Mode.

*

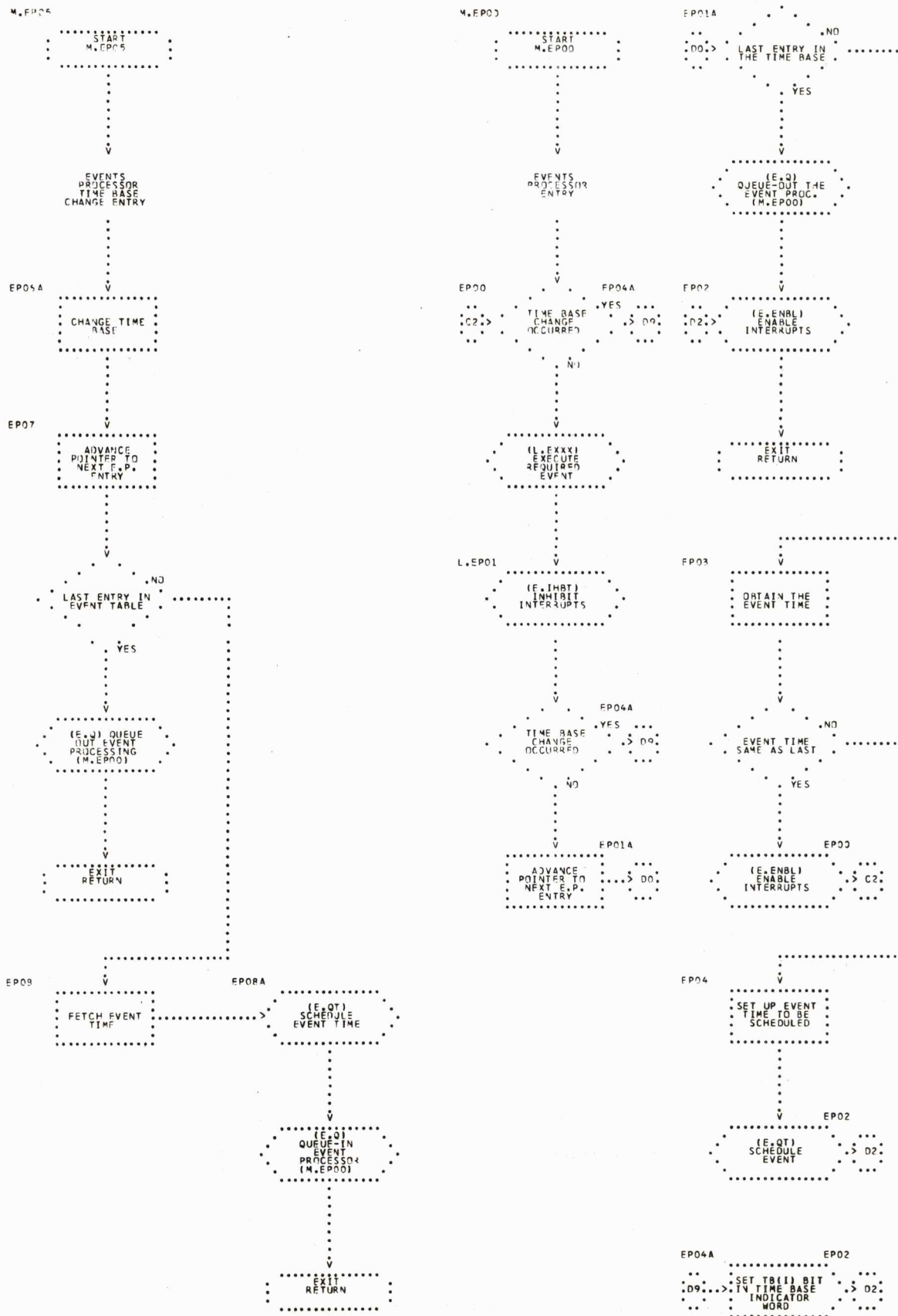
Notes

There is an event submodule, L.Exxx, for each entry in the Timed Events Tables. These modules perform time referenced mission dependent functions such as enabling and inhibiting discretes, queuing of modules, setting reasonableness test constants, etc.

The Timed Events Table is sub-divided according to time bases. Events which are referenced to a specific time base will only be processed in that time base. Any events which are scheduled but not processed are bypassed at the time base end.

Each Timed Events Table entry is composed of two words. The first word is a HPC to the event subroutine. The second word contains the time of execution referenced from time base start (RTCP scaled B27).

A detailed description of the mission dependent events required for this mission are given in the Events Processor Timed Events Table and the flowcharts of the Events Processor Submodules in Part IV.



2.21 FIXED POINT DIVIDE UTILITY (FD)

Function

The function of the Fixed Point Divide Utility is to perform a divide operation with specified inputs and to delay the proper time for the DIV instruction to operate.

Entries

U.FD00 Entered by any module.

External

None.

Input Data

D.VUTB Divide denominator
See notes

*

Output Data

See notes

*

Status Words

None

Telemetry

None

Initial Conditions

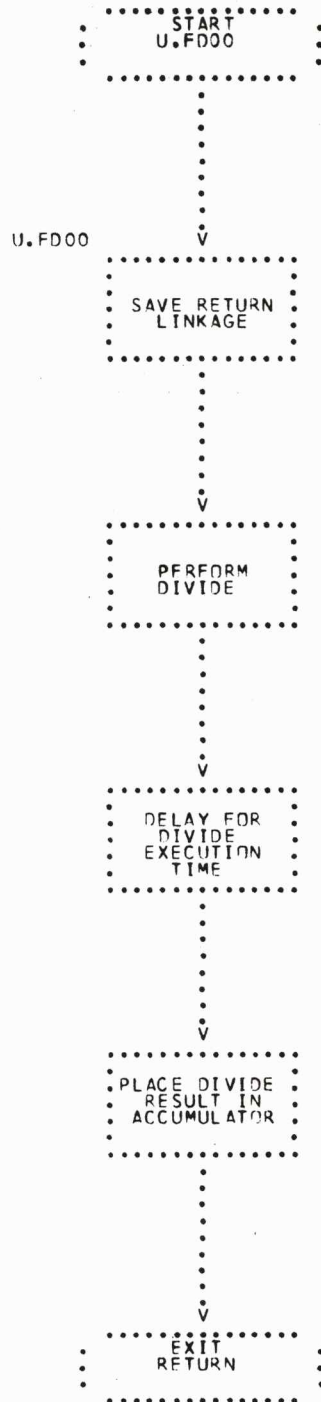
None.

Notes

Upon entry, the numerator is to be in accumulator. Upon exit, the quotient will be in PQ and accumulator.

FIXED POINT DIVIDE UTILITY (FD)

1



2.22 ABSOLUTE/COMMAND FREEZE UTILITY (FR)

Function

The function of this utility module is to perform either an absolute freeze or a command freeze of the vehicle.

Entries

U.FR00 Entered by the Orbit Guidance module to perform an absolute freeze *

U.FR10 Entered from Non-Interrupt Sequencer modules to perform a command freeze

External

E.ENBL Enable interrupts

E.IHBT Inhibit interrupts

Input Data

D.FLT	Flight Mode indicator	
D.VTHX	Total roll gimbal angle	*
D.VTHY	Total Pitch gimbal angle	*
D.VTHZ	Total Yaw gimbal angle	*
D.VCCX	Commanded Chi X (Rate limited roll gimbal angle)	*
D.VCCY	Commanded Chi Y (Rate limited pitch gimbal angle)	*
D.VCCZ	Commanded Chi Z (Rate limited yaw gimbal angle)	*

Output Data

D.VCCX	Commanded Chi X (Rate limited roll gimbal angle)	*
D.VCCY	Commanded Chi Y (Rate limited pitch gimbal angle)	*
D.VCCZ	Commanded Chi Z (Rate limited yaw gimbal angle)	*

D.VGCX Guidance Chi X (Desired roll gimbal angle)
D.VGCY Guidance Chi Y (Desired pitch gimbal angle)
D.VGCZ Guidance Chi Z (Desired yaw gimbal angle)
D.VDCX Delta Chi X (Roll commanded chi increment)
D.VDCY Delta Chi Y (Pitch commanded chi increment)
D.VDCZ Delta Chi Z (Yaw commanded chi increment)
D.FDCO Bypass setting delta chi flag

*
*
*
*
*
*

Status Words

None.

Telemetry

None.

Initial Conditions

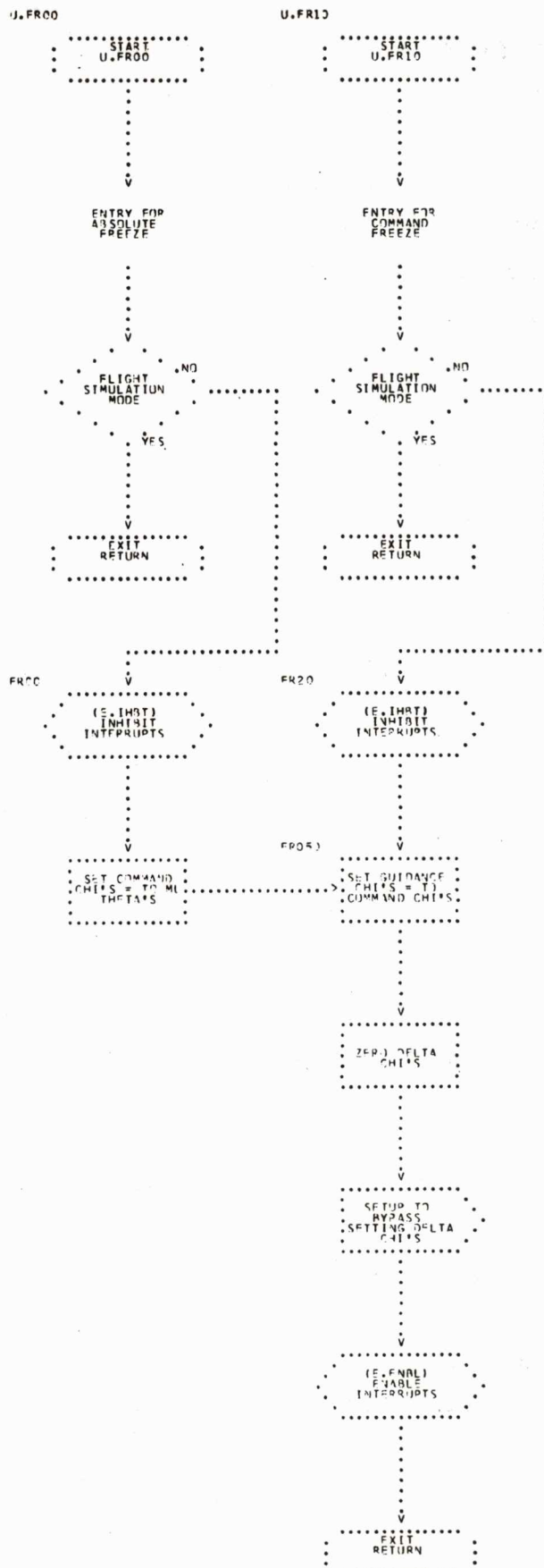
None.

Notes

For an absolute freeze in the Flight Mode, the command chi's and guidance chi's are set to the current gimbal angles. The delta chi's are set to zero. When in the Flight Simulation Mode, these quantities are not set.

For a command freeze in the Flight Mode, the guidance chi's are set to the command chi's and the delta chi's are set to zero. When in the Flight Simulation Mode, these quantities are not set.

When setting the guidance chi's, command chi's, and delta chi's, the program must be under interrupt protection.



2.23 PREFLIGHT G & C STEERING LINKAGE (GC)

Function

The function of this module is to provide the necessary linkage between the LVDC Preflight Program and the LVDC Flight Program Minor Loop module when operating the Prelaunch LVDC Steering Test. The Preflight G & C Steering Linkage module, the Minor Loop module, and the LVDC Preflight Program's Preflight G & C Steering test routine establish the necessary LVDC software to operate in the Preflight G & C Steering mode.

Entries

- M.GC00 Entered by the LVDC Preflight Program (G&C Steering Test routine) upon command from the ground computer for initialization.
- M.GC10 Entered from the LVDC Preflight Program (G&C Steering Test routine) upon occurrence of the Timer 1 (Minor Loop) interrupt.

External

- M.ML20 Minor Loop
- E.GP15 Control Program Timer 1 Scheduler
- I.N02 Phase I Initialize

Input Data

None.

Output Data

- | | | |
|--------|---|--|
| D.VA1 | } | Coefficients used to convert the inertial platform readings to body frame attitude corrections |
| D.VA5 | | |
| D.VSST | | Timer 1 scheduling word |
| D.VRTC | | Real time clock reading at last time update |

*

D.VCCX	Commanded Chi X (Rate limited roll gimbal angle)	*
D.VTHX	Total roll gimbal angle	*
D.VMLD	Minor loop rate of execution	*
D.VMLT	Execution time for next minor loop	
D.VTMM	Elapsed time in mission from GRR to last time update	*
F.GNC	G & C active indicator	
D.FMDI	Flight mode indicator	
V.XCG	X high order gimbal, backup or coarse resolution	
V.XOLD	X old platform gimbal	

Status Words

D.FIL3 Timer 1 interrupt in progress indicator

Telemetry

None.

Initial Conditions

None.

Notes

The Preflight Program Interrupt Processor is used to process all interrupts, although the Control Program Interrupt Processor is used to schedule the first Timer 1 interrupt. The Preflight Program maintains control of the interrupt inhibit status via its interrupt inhibit image word INTH.

In order to maintain consistent interface with the LVDC Preflight Program, absolute linkages are defined and maintained. The linkages required for the proper operation of this module are as follows:

Preflight Program

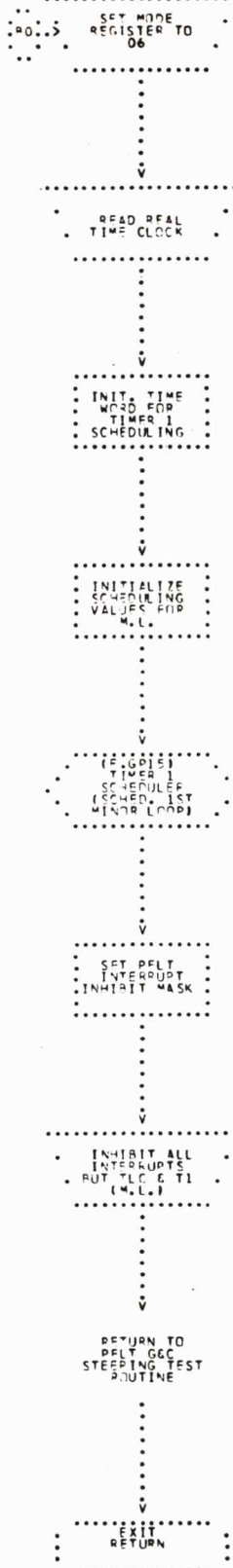
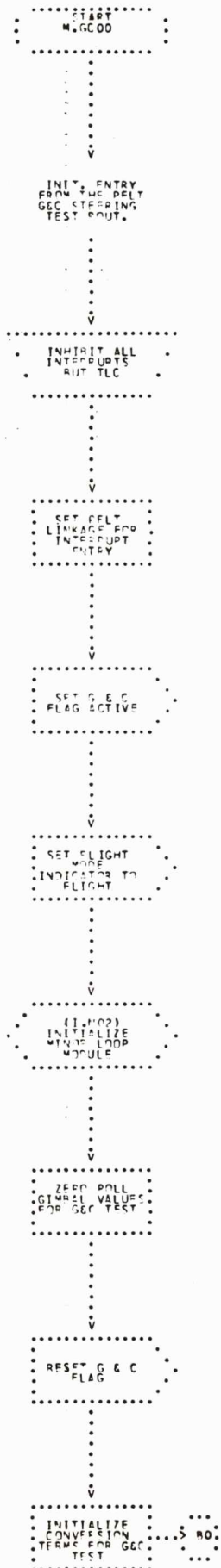
<u>Location</u> <u>Mod/Sect/Addr/S-D</u>	<u>Program</u> <u>Label</u>	<u>Description</u>
0, 07, 006 (S)	GCFIHC	Linkage from Preflight G&C Steering Test routine to the Preflight G&C Steering Linkage module initialization entry point, M.GC00. (HPC 0, 11, 0, 002, 2, 17 (D))
0, 07, 007 (S)	FIRHC	Linkage from Preflight G&C Steering Test routine to the Preflight G&C Steering Linkage module minor loop interrupt entry point, M.GC10. [Linkage HPC is provided by the G&C Steering Linkage module when entered for initialization at M.GC00.]
0, 17, 216 (S)	INTH	Preflight Interrupt Processor interrupt inhibit image. Modified by the Preflight G&C Steering Linkage module as needed.

Flight Program

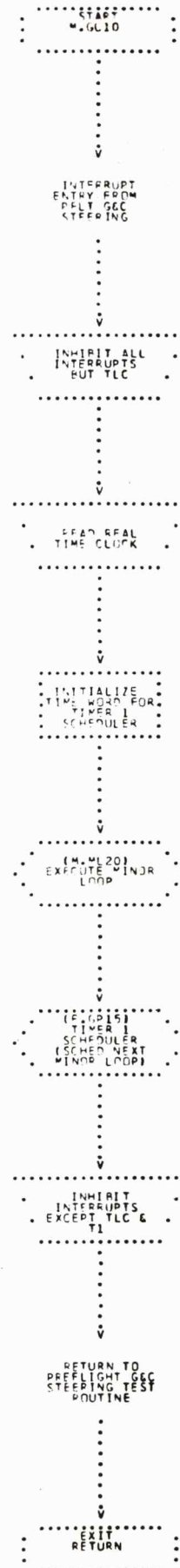
<u>Location</u> <u>Mod/Sect/Syl/Addr/S-D</u>	<u>Program</u> <u>Label</u>	<u>Description</u>
0, 11, 0, 002 (D)	M.GC00	Initialization entry point to Preflight G&C Steering Linkage module. Used by Preflight G&C Steering Test routine.

This module is never executed in the Flight Mode.

M.GC00



M.GC10



2.24 GIMBAL ANGLE RTC TEST UTILITY (GF)

Function

The function of the Gimbal Angle RTC Test Utility Routine is to test if there have been gimbal angle failures and to set up reasonableness constants.

Entries

U.GF00 Entered by any module.

External

None.

Input Data

F.BUGX	Backup X gimbal active, alter RTC	*
F.BUGY	Backup Y gimbal active, alter RTC	*
F.BUGZ	Backup Z gimbal active, alter RTC	*
V.ML12	X low order gimbal resolution	
V.ML22	Y low order gimbal resolution	
V.ML32	Z low order gimbal resolution	
V.CG10	Gimbal reasonableness rate limit constant (Backup 1st pass)	*
V.CG11	Gimbal reasonableness rate limit constant (Backup 1st pass)	*
	(crossover)	*
V.CG20	Yaw & Pitch gimbal reasonableness rate limit constant	*
	(Backup 2nd pass)	*
V.CG21	Yaw & Pitch gimbal reasonableness rate limit constant	*
	(Backup 2nd pass) (crossover)	*
V.CG30	Roll gimbal reasonableness rate limit constant	*
	(Backup 2nd pass)	*
V.CG31	Roll gimbal reasonableness rate limit constant	*
	(Backup 2nd pass) (crossover)	*

Output Data

V.ML10 X gimbal reasonableness test limits
V.ML11 X gimbal reasonableness test limits (crossover case)
V.ML20 Y gimbal reasonableness test limits
V.ML21 Y gimbal reasonableness test limits (crossover case)
V.ML30 Z gimbal reasonableness test limits
V.ML31 Z gimbal reasonableness test limits (crossover case)

Status Words

D.VMC4 Mode Code 24

Telemetry

None.

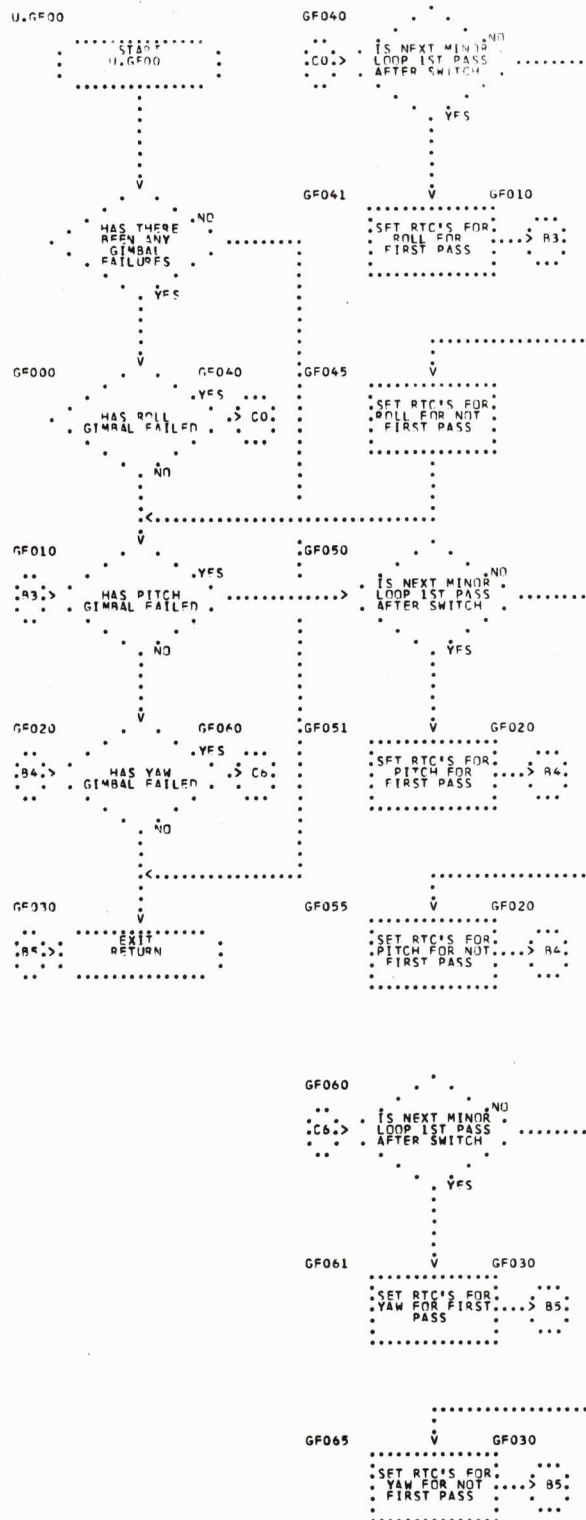
Initial Conditions

None.

Notes

This module must be under interrupt protection when entered.

GIMBAL ANGLE RTC TEST UTILITY (GF)



2.25 ACQUISITION GAIN/LOSS (GL)

Function

The function of the Acquisition Gain/Loss module is to determine whether the vehicle is within range of a telemetry station.

Entries

M.GL00 Entered via Phase 2 Timer 2 Control module once per 8 seconds.

External

U.TR00	Real Time Telemetry Delay Utility	
U.VD00	Vectory Dot Product Utility	*
U.SC00	Sine-Cosine Utility	*
E.GP20	Control Program Time Update	*
U.SSRQ	Request Switch Selector Sequence for Acquisition Gain (M.SS00)	

Input Data

D.VNTM	Time in mission from GRR of navigation extrapolation	*
D.VXG	X coordinate of position vector, G-system	*
D.VYG	Y coordinate of position vector, G-system	*
D.VZG	Z coordinate of position vector, G-system	*
D.VTRR	Elapsed total time in current time base	*
D.VTGB	Accumulated ground bias time	

Output Data

None.

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.TBA	0074	Time in time base of Station Acquisition Gain
V.TBL	0105	Time in time base of Station Acquisition Loss

Initial Conditions

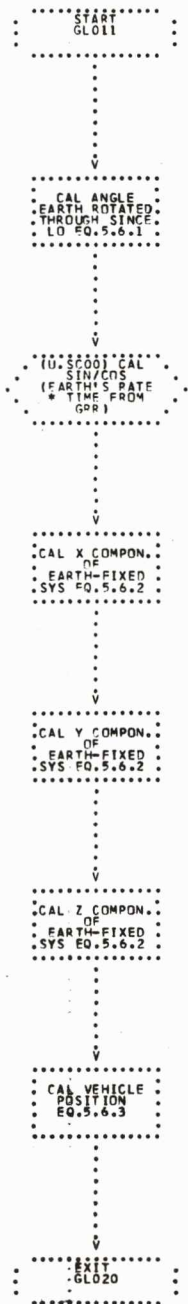
None.

Notes

A table of station unit vectors is used for projecting the distance the vehicle is above or below the horizon of each station. Each station is tested to determine if the station is within range of the vehicle.



ACQUISITION GAIN/LOSS (GL)



2.26 GRAVITATION ACCELERATION (GR)

Function

The function of the Gravitation Acceleration module is to calculate the gravity acceleration of the vehicle as a function of its position vector.

Entries

M.GR00 Entered by the Boost Navigation module during Phase 1.
Entered by the Orbit Navigation module during Phase 2.

External

U.VD00 Vector Dot Product Utility

Input Data

D.VXS	X component of radius vector, in S-system	*
D.VYS	Y component of radius vector, in S-system	*
D.VZS	Z component of radius vector, in S-system	*
V.MA4+1	Product of the cosine of the geodetic latitude of the launch site and the sine of the actual launch azimuth	
V.MA4	Sine of the geodetic latitude of the launch site	
V.MA4+2	Product of the cosine of the geodetic latitude of the launch site and the cosine of the actual launch azimuth	
D.FPHS	Phase indicator flag:	
	(-) = Boost	
	(+) = Orbit (any bits except sign bit)	

Output Data

D.VRR2	Radius squared	
D.VI0R	Inverse of radius	*

D.VGX X component of gravity acceleration
D.VGY Y component of gravity acceleration
D.VGZ Z component of gravity acceleration
D.V1R2 Inverse of square of radius

*
*
*

Status Words

None.

Telemetry

None.

Initial Conditions

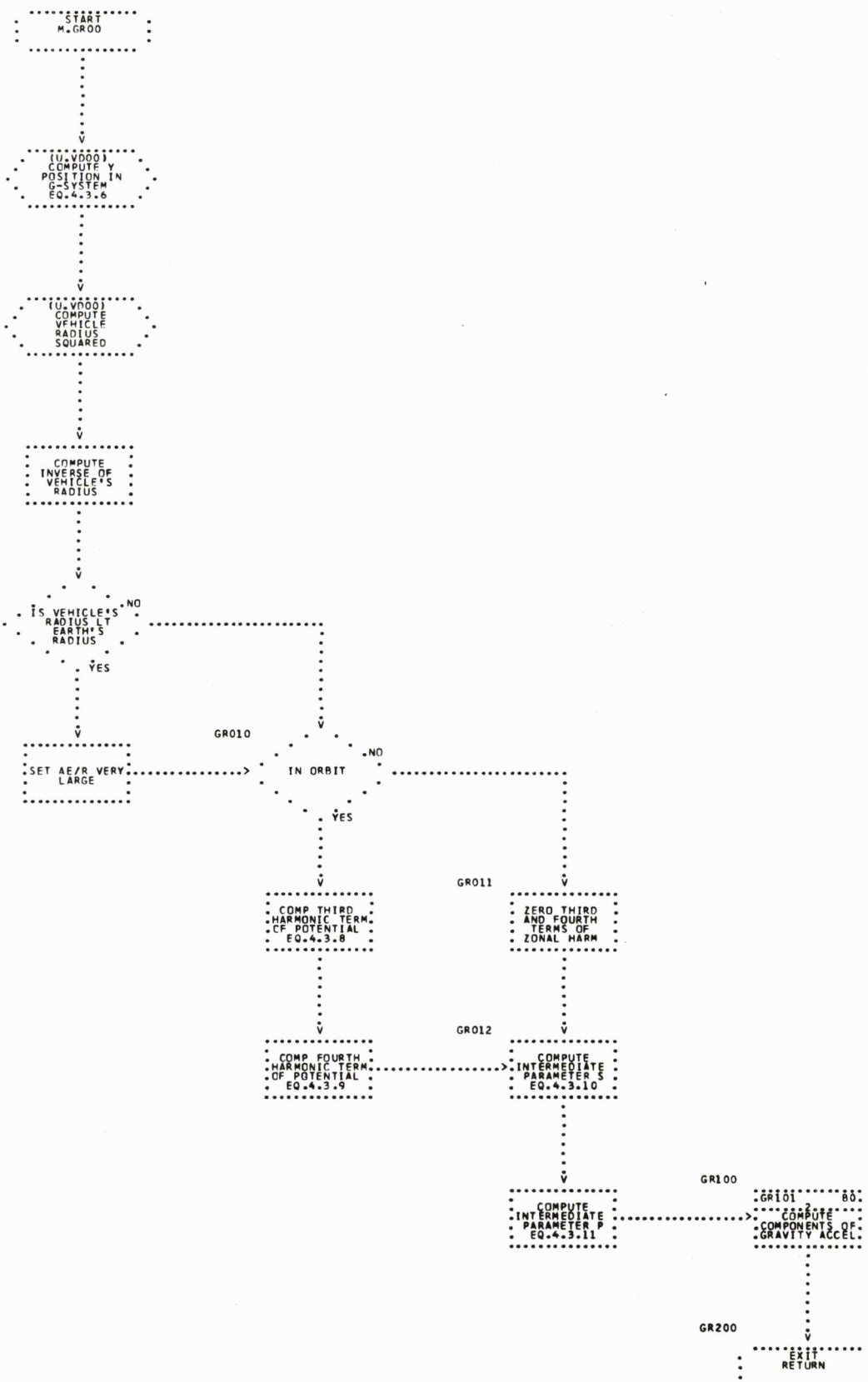
The phase indicator flag, D.FPHS, must be initialized at the start of each phase.

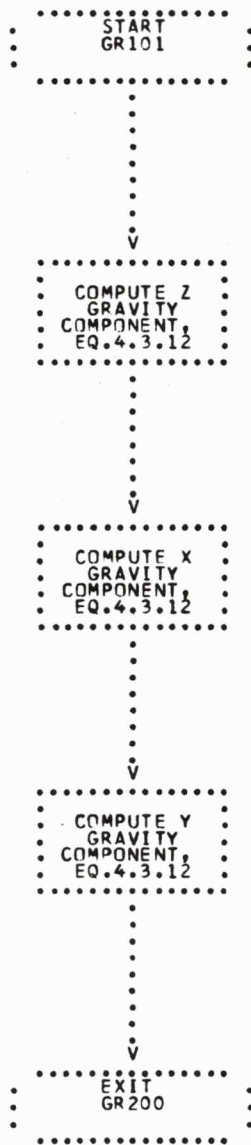
D.V1R2 must be set to a predefined value at mission start. The second row of "A" matrix, V.MA4, V.MA4+1, V.MA4+2, must be calculated at mission start.

Notes

None.

M.GR00





2.27 PREFLIGHT DCS LINKAGE (HD)

Function

The function of this module is to provide proper linkage between the LVDC Preflight Program and LVDC Flight Program DCS module when operating the Preflight DCS Interface Test. This linkage allows prelaunch testing of the ability to receive and acknowledge DCS commands.

This module provides the capability to execute an automatic memory dump of the target load parameters when commanded by the RCA 110-A via the LVDC Preflight Program. This module also provides the proper linkage for the housekeeping processing performed when the LVDC Preflight Program is commanded to calculate memory sum checks.

*
*

Entries

- M.HD00 Entered by the LVDC Preflight Command Receiver Linkage routine upon command from the ground computer for initialization. Entered by the LVDC Preflight Program upon command for automatic memory dump.
- M.HD10 Entered from the LVDC Preflight Interrupt Processor upon occurrence of the Command Receiver (CR) Interrupt.
- M.HD20 Entered from the LVDC Preflight Sum Check routine upon command from the ground computer for memory sum checks.

External

- D.S200 CRP Set/Reset
- M.DS00 Digital Command System
- M.DT00 Memory Dump Telemetry
- M.ZR05 Memory Clear Routine
- E.Q Memory Dump Telemetry (M.DT00)

*

Input Data

D.FPFT Preflight Memory Dump Telemetry flag

Output Data

D.VIH Interrupt inhibit image

D.FDTL Memory dump/compressed data dump in progress flag

*

F.DSEN Data enable flag

V.DSRC DCS transmission error counter

D.HTR4 HOP constant to special telemetry delay

F.SSAC Switch Selector Active flag

D.FPFT Preflight Memory Dump Telemetry flag

D.VDLO Memory Dump start address

D.VDHI Memory Dump stop address

Status Words

D.FIL3 Timer 1 interrupt in progress indicator

D.VLNK DCS Status

Initial Conditions

None.

Notes

In order to maintain consistent interface with the LVDC Preflight Program, absolute linkages are defined and maintained. The linkages required for the proper operation of this module are as follows:

Preflight Program

<u>Location</u> <u>Mod/Sect/Addr/S-D</u>	<u>Program</u> <u>Label</u>	<u>Description</u>
0, 07, 211 (S)	OCRLHC	Link from Preflight Command Receiver Linkage routine to the Preflight DCS Linkage module initialization entry point, M.HD00. [HPC 4, 17, 0, 274, 2, 17 (D)] *
0, 17, 154 (S)	CRAIHC	Link from Preflight Interrupt Processor to the Preflight DCS Linkage module CR interrupt entry point, M.HD10. [Linkage HPC is provided by DCS Linkage module when entered for initialization at M. HD00.]
0, 17, 210 (S)	INACC	Preflight Interrupt Processor save location for accumulator contents. Used by Preflight DCS Linkage module to restore accumulator after CR interrupt processing.
0, 17, 212 (S)	INHC	Preflight Interrupt Processor save location for interrupted program location. Used by Preflight DCS Linkage module to return to interrupted location after CR interrupt processing.
0, 17, 214 (S)	INPQ	Preflight Interrupt Processor save location of PQ. Used by Preflight DCS Linkage module to restore PQ after CR interrupt processing.

0, 17, 216 (S)

INTH

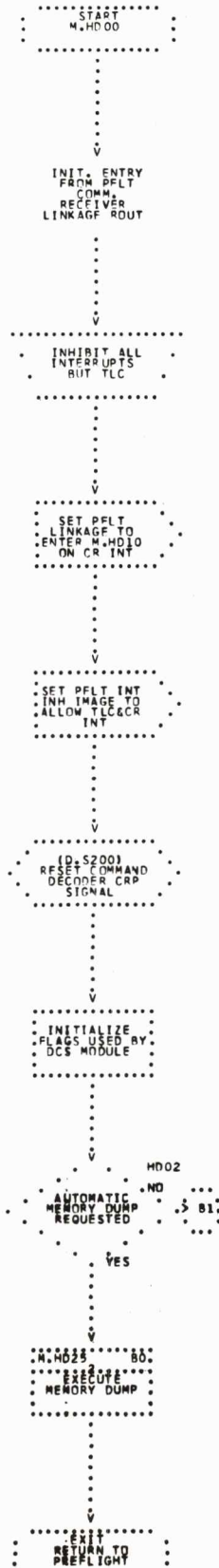
Preflight Interrupt Processor interrupt inhibit image. Modified by the Preflight DCS Linkage module as needed.

Flight Program

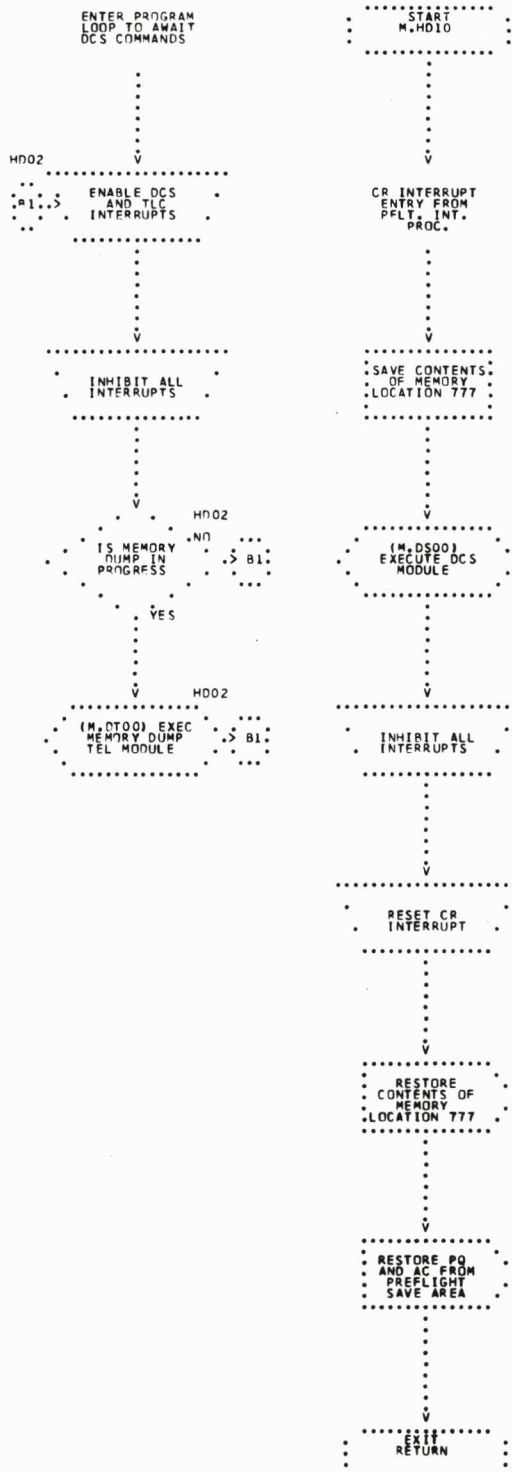
<u>Location</u> <u>Mod/Sect/Syl/Addr/S-D</u>	<u>Program</u> <u>Label</u>	<u>Description</u>
4, 17, 0, 274 (D)	M.HD00	Initialization entry point to the Preflight DCS Linkage module. Used by Preflight Command Receiver Linkage routine.
0, 11, 0, 005 (D)	M.PFSC	Entry point for the Preflight Sum Check Routine

This module is never used in the Flight Mode.

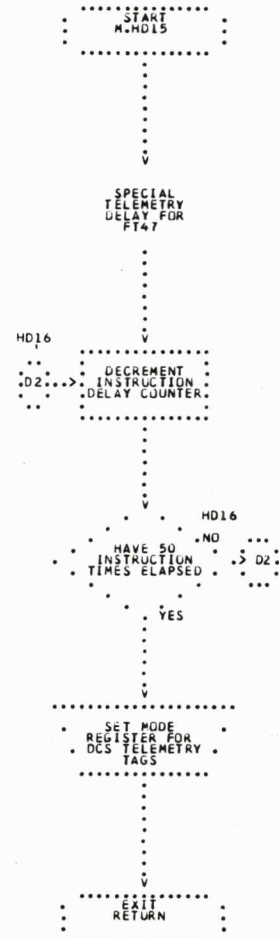
M.HD00



M.HD10



M.HD15



M.HD20

START
M.HD20

V

PROCESSING
FOR SUM CHECK
PREPARATION

V

SAVE RETURN
TO PREFLIGHT

V

CLEAR
PREDEFINED
DCS STORAGE
AREA

V

CLEAR
ROLL-OVER
INSTRUCTIONS

V

INITIALIZE
776 HOP SAVE
FOR SUM CHECK

V

IS-81
QUEUE-OUT
MEMORY
DUMPELN
(M.DT00)

V

EXIT
RETURN TO
PREFLIGHT

START
M.HD25

V

PROCESSING
FOR AUTO
MEMORY DUMP

V

SET START
ADDRESS TO
0,16,360

V

SET STOP
ADDRESS TO
0,16,370

V

ENABLE MEMORY
DUMP FLAGS

V

M.DT00
EXECUTE
MEMORY DUMP

V

EXIT
RETURN TO
PREFLIGHT

(This Page Left Blank Intentionally)

2.28 S-IVB CUTOFF PREDICTION (HS)

Function

The function of the S-IVB Cutoff Prediction module is to insure the required velocity is achieved by precisely determining when to issue the S-IVB Cutoff Switch Selector.

Entries

- M.HS00 Entered from the Non-Interrupt Sequencer in Phase 1 when the predicted cutoff time is less than a mission dependent preset time.
- H.S027 Entered from the Cutoff Module (CO) if the S-IVB Cutoff Switch Selector read issuance is late.

External

- M.SS60 Switch Selector Processing
- U.FD00 Fixed Point Divide Utility
- U.FR10 Absolute/Command Freeze Utility
- U.TR00 Real Time Telemetry Delay Utility
- E.ENBL Enable interrupts
- E.IHBT Inhibit interrupts
- E.Q S-IVB Cutoff Prediction (M.HS00)
- E.QT Switch Selector Processing (M.SS60)

Input Data

- D.VGVT Desired terminal velocity
- D.VTBI Indication of time base initiation *
- D.VT3I Second phase IGM time to go *
- D.VDT Elapsed time between current and previous computation cycle
accelerometer reads in seconds *
- D.VIT1 Length of 2nd previous computation cycle

D.VIT2	Length of previous computation cycle	
D.VV0	Second previous total velocity	
D.VV1	Previous total velocity	
D.VV2	Current total velocity	
D.VACT	Real time clock reading associated with D.VTAS	*
D.VRTC	Real time clock reading at last time update	*
D.VTMM	Elapsed time in mission from GRR to last time update	*
V.GTVB	Time bias for thrust decay for high cutoff accuracy	*
V.GDVB	Thrust decay velocity bias	*
F.HSPF	High Speed Loop flag	

Output Data

F.HSPF	High Speed Loop flag	
D.VSST	Time for S-IVB cutoff switch selector read to be issued	
D.VT3I	Second phase IGM time-to-go	*
D.VCOR	RTC read associated with D.VTMS	*
D.VTMS	Current time-to-go to S-IVB cutoff	
D.GMLC	Variable HOP constant to M.C000 or M.L299	
V.SSCO	High Speed S-IVB cutoff switch selector time	
D.VTBI	Indication of time base initiation	*

Status Words

D.FILE Flight program status

Telemetry

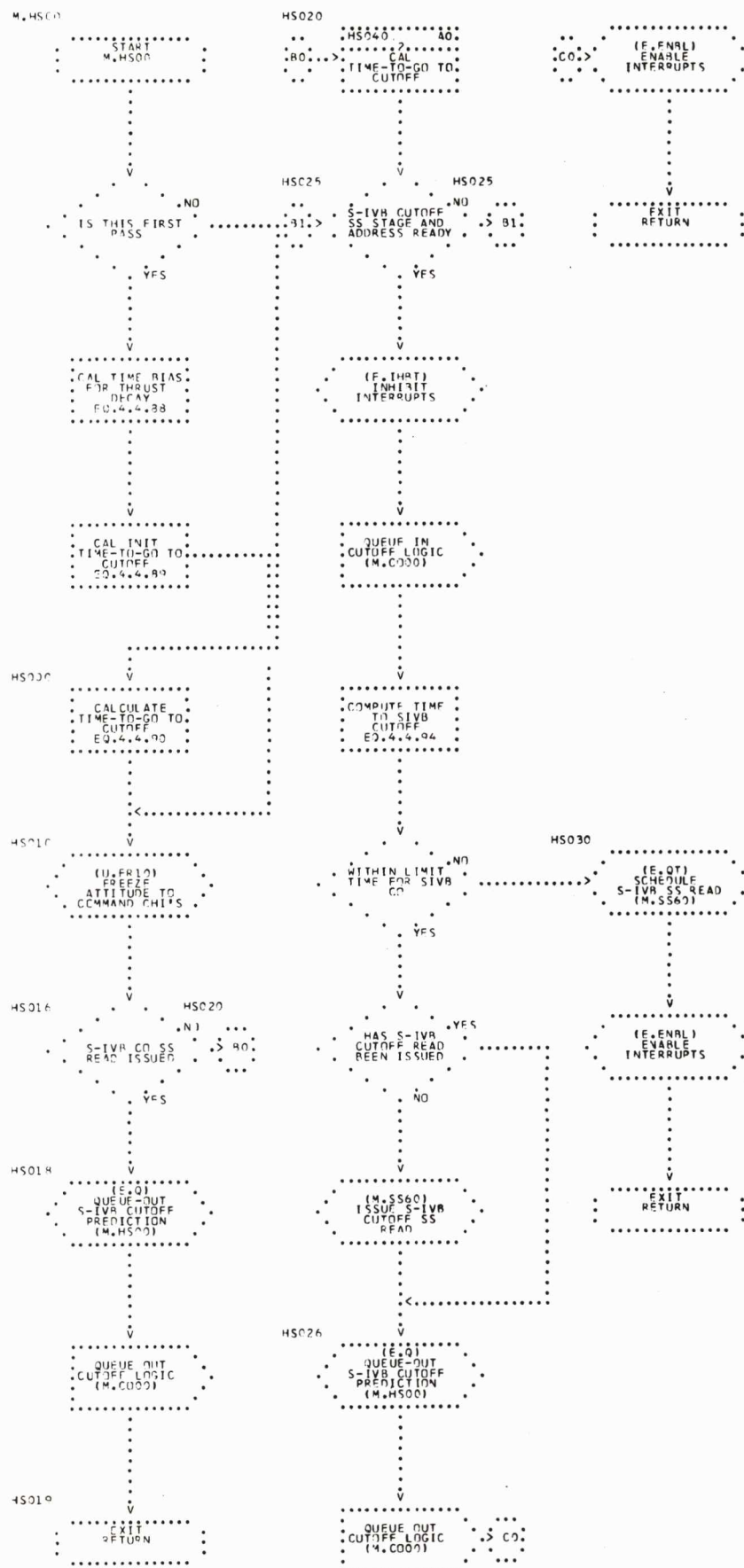
<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
T.T3I	0544	Time remaining in 3rd phase IGM

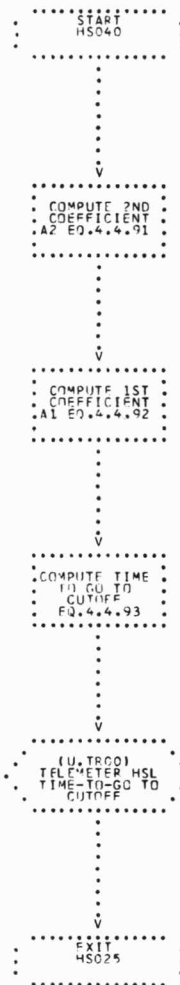
Initial conditions

The High Speed Loop flag, F.HSPF, must be set to zero and V.GDVB must be initialized to the proper value at mission start.

Notes

This module modifies the variable HOP constant D.GMLC used by the Minor Loop module to access or to bypass the Cutoff Logic module. D.GMLC is set to M.C000 to activate the Cutoff Logic module and to M.L299 to bypass it.





2.29 ITERATIVE GUIDANCE MODE (IG)

Function

The function of the Iterative Guidance Mode module is to guide the Saturn IB vehicle to desired end conditions for orbital insertion in a near optimum fashion. This module computes steering commands during the S-IVB burn using a path-adaptive scheme.

Entries

M.IG00 Entered from the Non-Interrupt Sequencer once per computation cycle.

External

M.CC00 Chi Computations module
M.SM00 Steering Misalignment Correction module
U.AT00 Arctangent Utility
U.FD00 Fixed Point Divide Utility
U.LN00 Natural Log Utility
U.MM00 Matrix Multiply Utility
U.SC00 Sine-Cosine Utility
U.SD00 Scaled Divide Utility
U.SR00 Square Root Utility
U.TR00 Real Time Telemetry Delay Utility
U.VD00 Vector Dot Product Utility
U.SSRQ Request S-IVB Cutoff SS (M.SS00)
E.IHBT Inhibit interrupts
E.ENBL Enable interrupts
E.Q Chi Computations Telemetry (M.CC10)
E.Q. S-IVB Cutoff Prediction (M.HS00)
E.Q. Iterative Guidance Mode (M.IG00)
E.Q. Minor Loop Support (M.MS00)

*

*

Input Data

D.VDT	Elapsed time between current and previous computation cycle accelerometer reads in seconds	*
D.VMFS	Filtered value of current M/F	*
D.VXS	X component of radius vector, in S-system	*
D.VYS	Y component of radius vector, in S-system	*
D.VZS	Z component of radius vector, in S-system	*
D.VXDS	X component of velocity vector, in S-system	*
D.VYDS	Y component of velocity vector, in S-system	*
D.VZDS	Z component of velocity vector, in S-system	*
D.VVS	Total space-fixed velocity	*
D.VGX	X component of gravity acceleration	*
D.VGY	Y component of gravity acceleration	*
D.VGZ	Z component of gravity acceleration	*
D.FLT	Flight mode indicator	
D.VGVT	Terminal velocity	
V.GTVB	Time bias for thrust decay for high cutoff accuracy	
D.GSMC	Variable HOP constant to Steering Misalignment Calculations	
D.VAP1 thru D.VAP9	(M.SM00) Elements of the [MS4] Transformation Matrix	*
F.IGCO	Inhibit High Speed Loop flag	*
D.KT4	Time Base 4 start time from GRR	*

Output Data

D.VT3I	Second phase IGM time-to-go	*
D.VlMR	Reciporcal of the number of minor loops per pass through Minor Loop Support	
D.VMLR	The number of minor loops per computation cycle	
D.VTP1	Cosine Chi Y in 4-system	*
D.VTP2	Sine Chi Z in 4-system	*

D.VTP3	Sine Chi Y in 4-system	*
D.VTP4	Cosine Chi Z in 4-system	*
G.OD00	Variable HOP constant to DOM module (M.OD00)	
D.GMFS	Variable HOP constant to M/F smoothing module (M.ST00)	*

Status Words

D.VMC5 Mode Code 25

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>	
D.VT1I	0400	First phase IGM time-to-go	
V.TLZ4	0434	Z component of radius vector, in 4-system	*
V.TLX4	0444	X component of radius vector, in 4-system	*
V.TLY4	0450	Y component of radius vector, in 4-system	*
V.GT20	2414	Terminal range angle	*
V.TU3	2410	Second phase IGM ideal burn time	*
V.GT22	2424	X component of radius vector, in V-system	*
V.GT19	2434	Z component of velocity vector, in V-system	*
V.GT24	2444	X component of velocity vector, in V-system	*
V.GT17	0460	Total time to go to S-IVB cutoff	
V.GT30	2474	Y component of velocity-to-be-gained vector, in V-system	*
V.GT31	2464	Z component of velocity-to-be-gained vector, in V-system	*
V.GT29	2470	X component of velocity-to-be-gained vector, in V-system	*
V.GT7	2500	Velocity correction in yaw guidance angle	
V.GT8	2504	Velocity correction in pitch guidance angle	
V.DYV	2510	Y component of distance-to-be gained vector, in V-system	*

V.GT12	2534	Ideal IGM yaw rate, position constrained	
V.K3	2530	Altitude correction in IGM yaw guidance angle	
V.GT23	2524	X component of distance-to-be gained vector, in V-system	*
			*
V.GT10	2514	Altitude correction in IGM pitch guidance angle	
V.GT11	2520	Ideal IGM pitch rate, position constrained	
D.VT3I	0464	Second phase IGM time-to-go	*
V.TU1	2404	First phase IGM ideal burn time	
V.GT2	2550	Yaw guidance command in the target system	*
V.GT1	2554	Pitch guidance command in the target system	*
V.GT5	2450	Y component of velocity vector, in V-system	*
V.GT21	2454	Second phase IGM time-to-go correction	*

Initial Conditions

The following quantities must be initialized before IGM initiation:

D.VT1I	First phase IGM time-to-go	
D.VT3I	Second phase IGM time-to-go	*
V.TU3	Second phase IGM ideal burn time	*
V.IGPC	Time since nominal EMR shift	
V.TAUN	Nominal TAU3 for S-IVB burn	
V.GSQ1	First guess at SQRT (DZDV**2) + (DXDV**2)	
V.GXVT	Terminal radius for S-IVB burn	
F.IGCO	Flag to enable/inhibit high speed loop	
F.THLC	Thrust level change flag	
F.PASS	Chi bar steering first pass flag	
D.GSMC	Variable HPC to Steering Misalignment Correction (M.SM00)	*
D.VAP1 thru D.VAP9	Elements of the [MS4] Transformation Matrix	*
F.INIG	Indicator for IGM first pass	
D.FS4T	Artificial TAU flag	

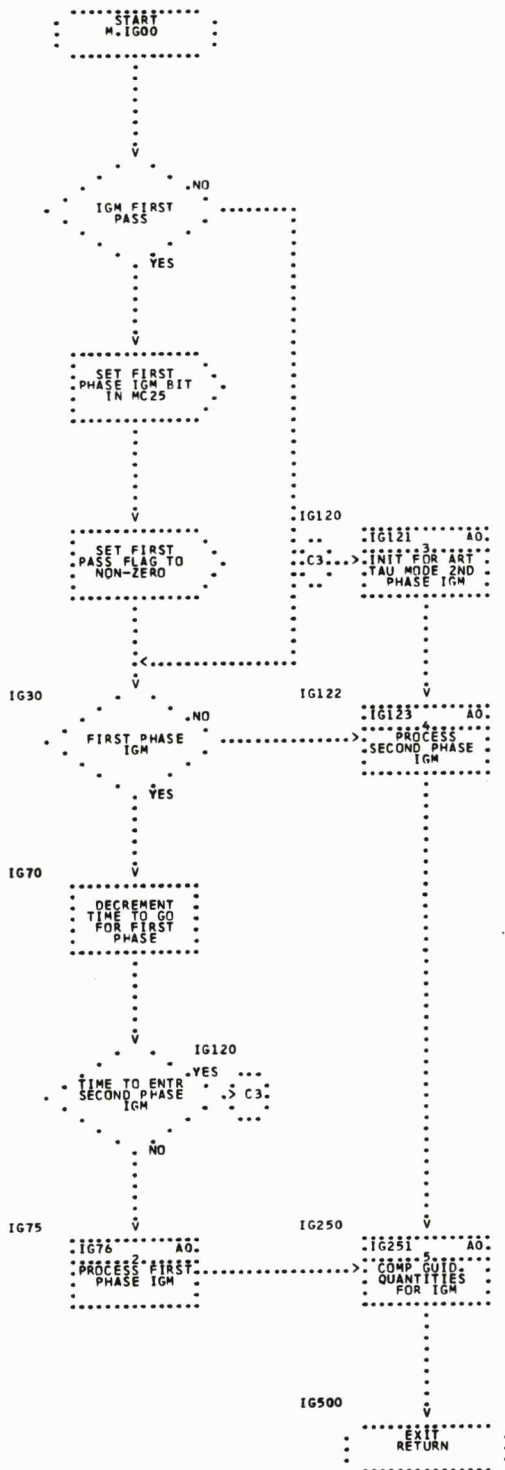
Notes

When the calculated time-to-go to S-IVB cutoff equals or passes a predefined time, IGM will queue itself out, and queue in the S-IVB cutoff prediction module.

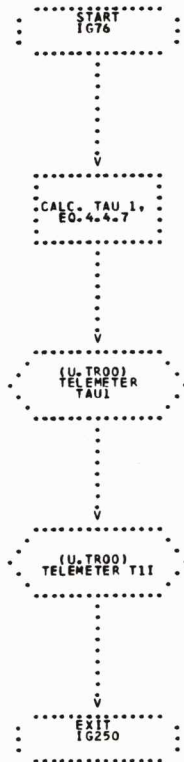
(THIS PAGE INTENTIONALLY LEFT BLANK)

ITERATIVE GUIDANCE MODE (IG)

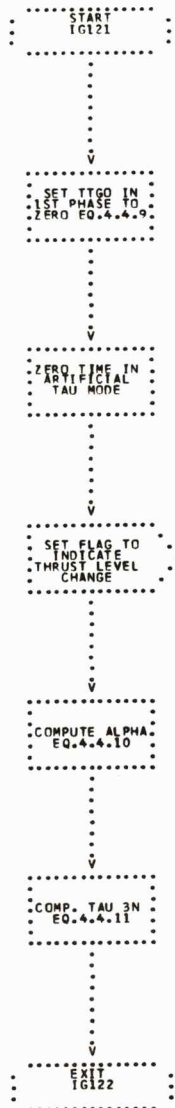
M. IG00



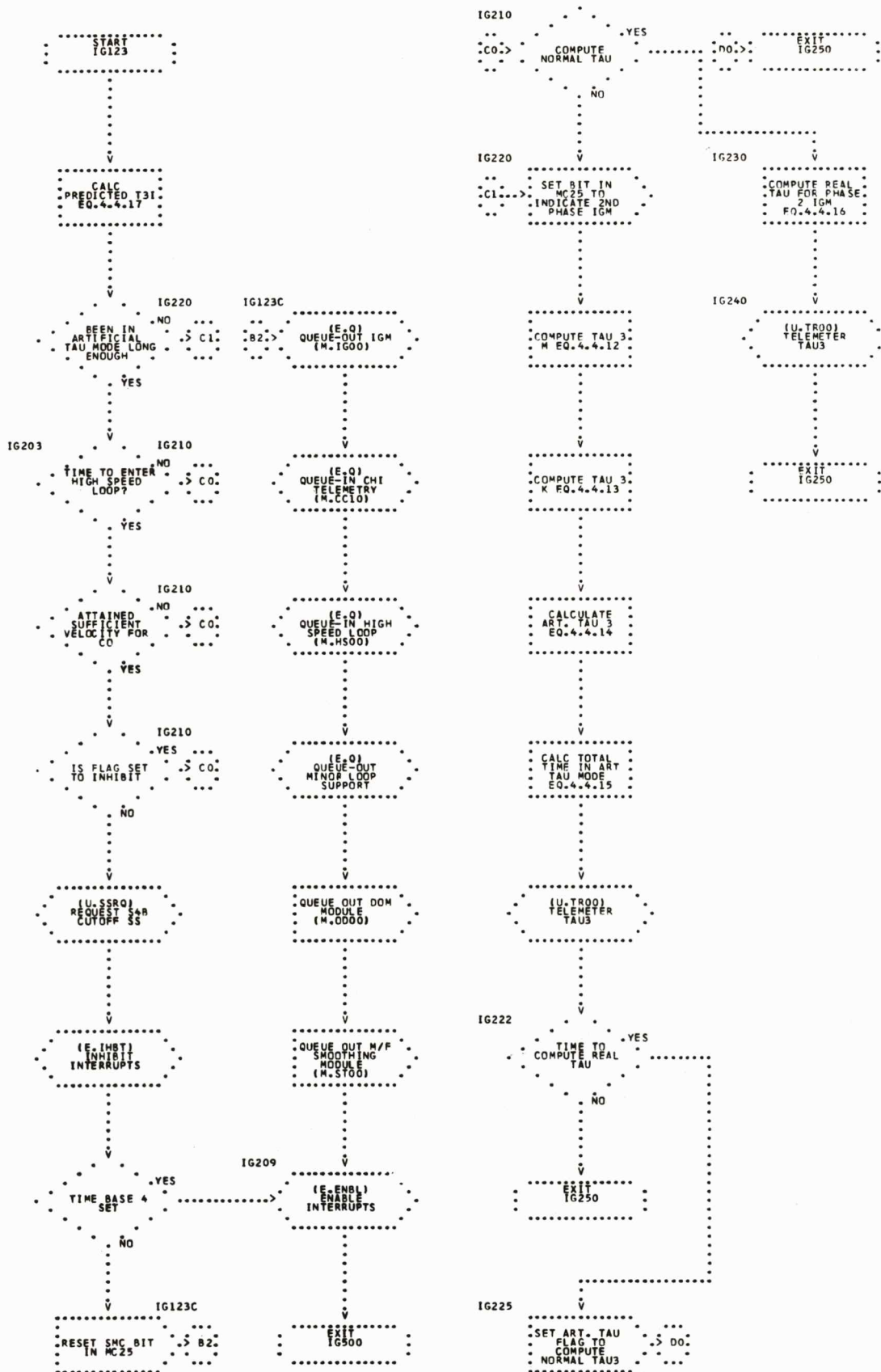
ITERATIVE GUIDANCE MODE (IG)



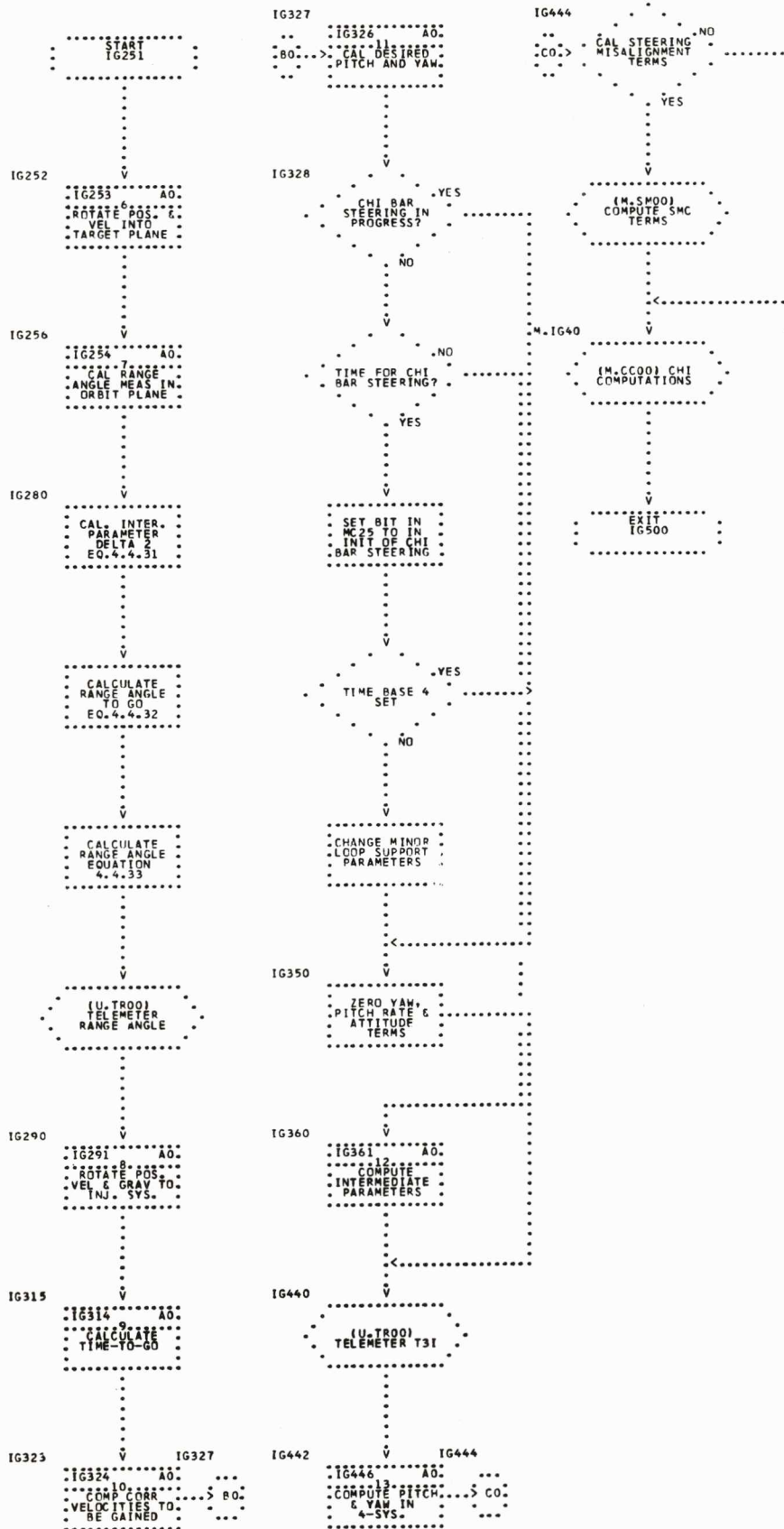
ITERATIVE GUIDANCE MODE (IG)

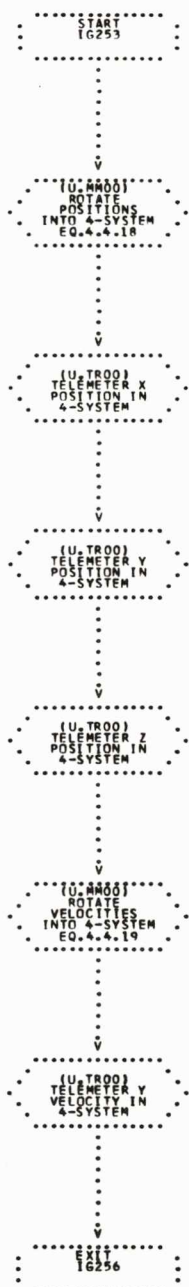


ITERATIVE GUIDANCE MODE (IG)

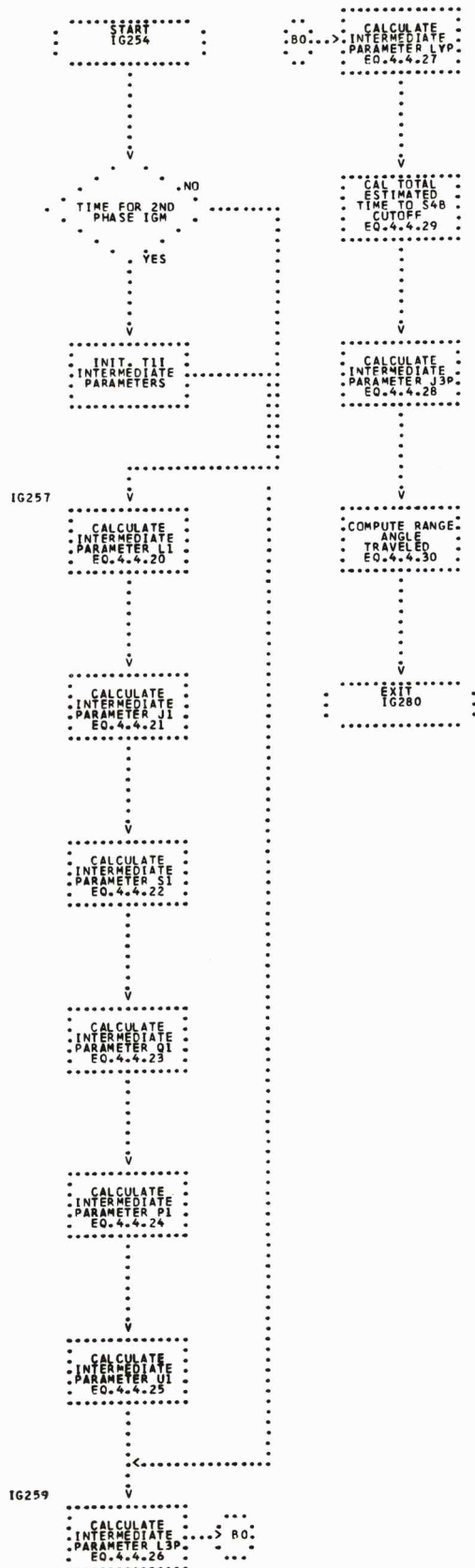


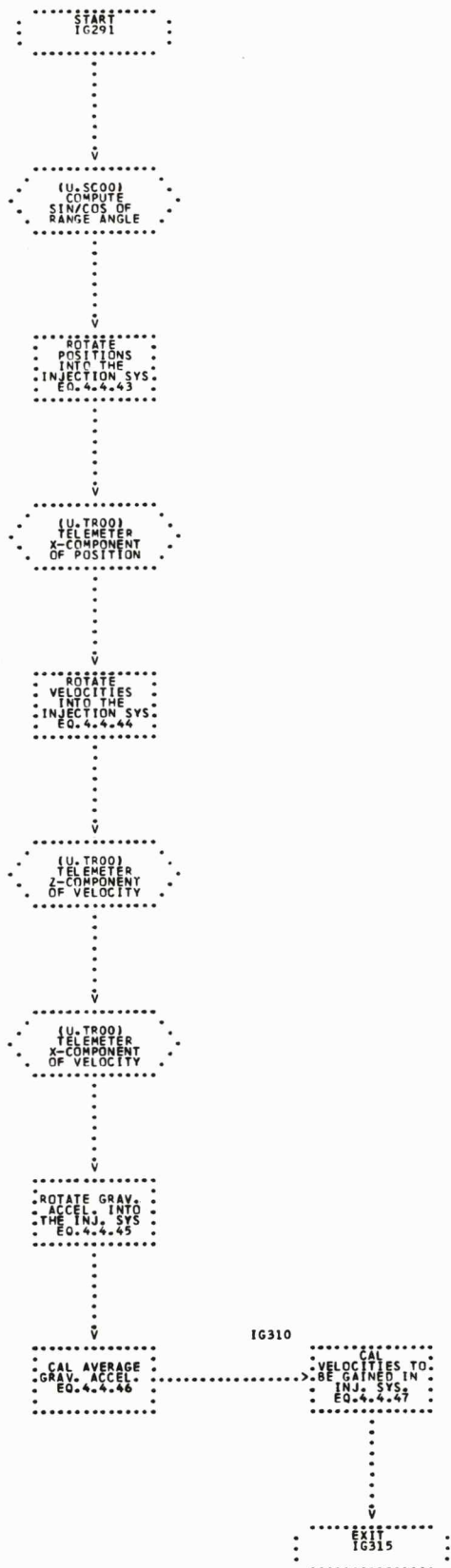
ITERATIVE GUIDANCE MODE (IG)

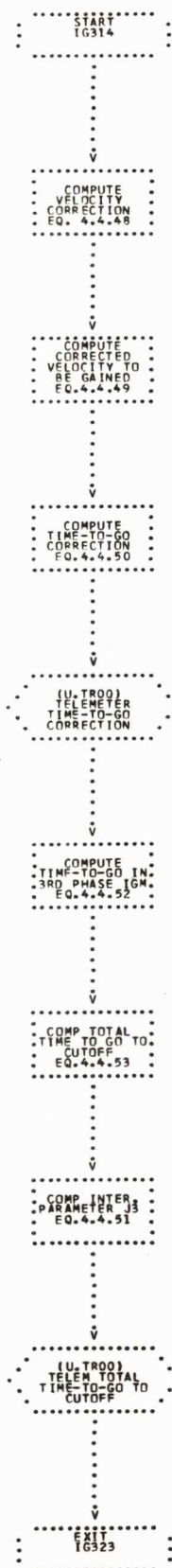


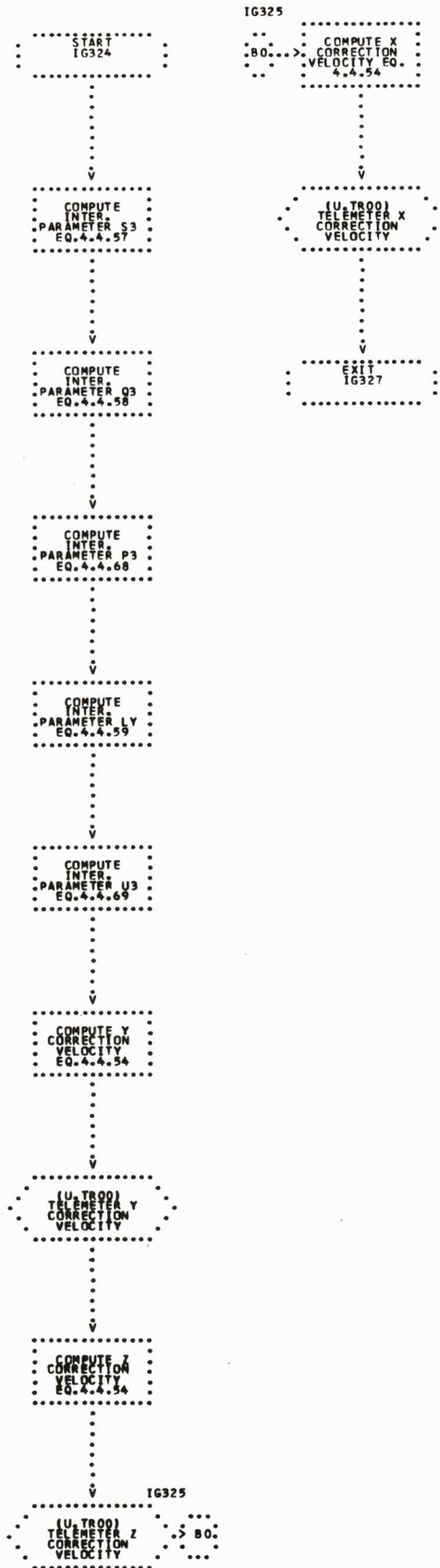


ITERATIVE GUIDANCE MODE (IG)

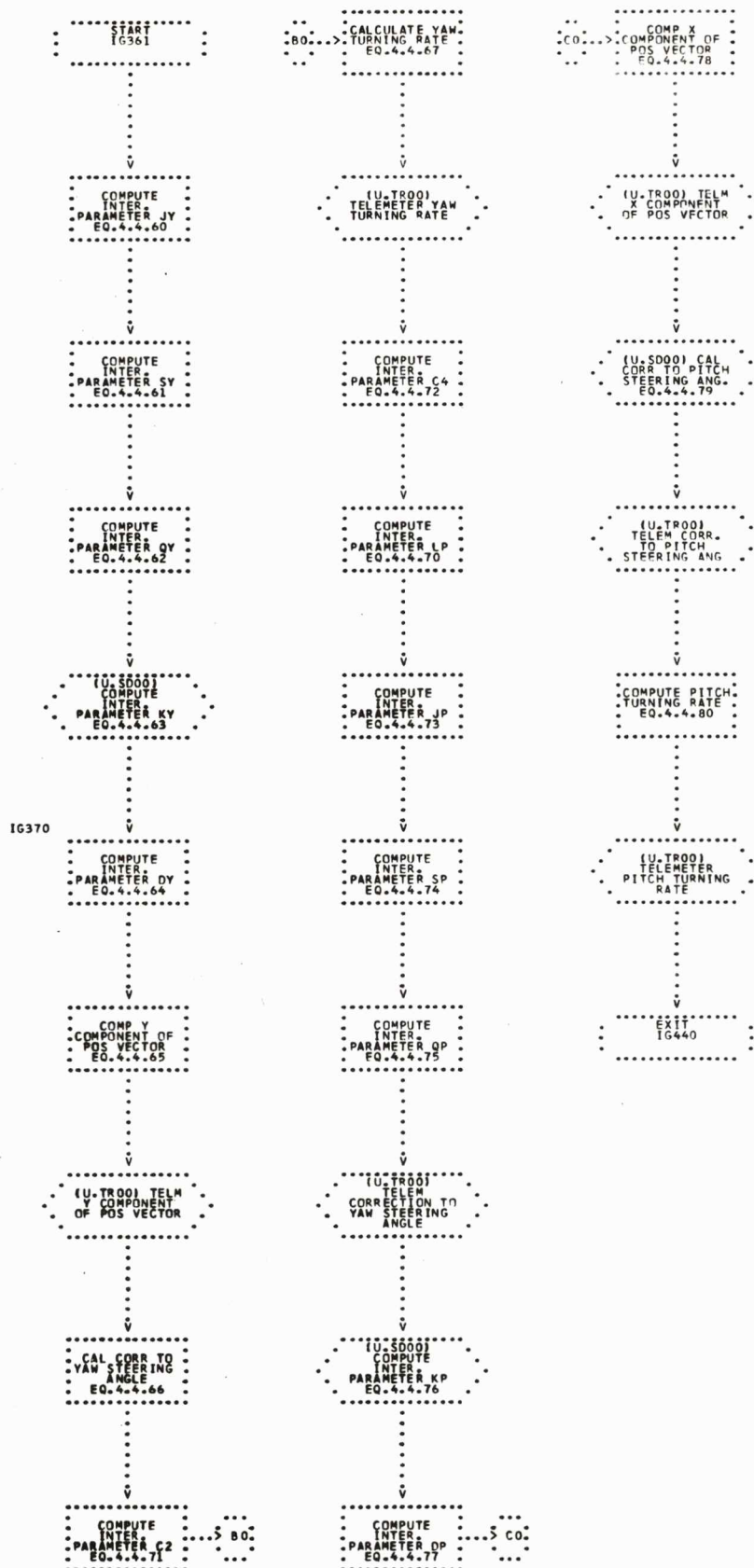


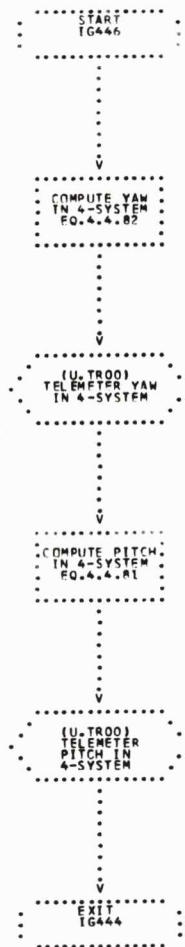












(THIS PAGE INTENTIONALLY LEFT BLANK)

2.30 PHASE INITIALIZATION (IN)

Function

This module will initialize all variable data and activate or deactivate any module which differs from the current mission definition for each phase according to the various test modes of operation.

Entries

M.IN00 Entered from the Control Program before starting Phase 1.
M.IN10 Entered from the Control Program before starting Phase 2.

External

M.EP05 Events Processor
M.GC02 Preflight G&C Steering Test
M.GR00 Gravitation Acceleration
M.LR35 Ladder Ramp Processor
M.ML20 Minor Loop
M.ZR00 Memory Zero Routine
U.DO00 Discrete Output Register Set/Reset Utility
U.MM00 Matrix Multiply Utility
U.MT00 Matrix Transpose Utility
U.SC00 Sine-Cosine Utility
U.TR00 Real Time Telemetry Delay Utility
U.GF00 Gimbal Angle RTC Test Utility
U.FD00 Fixed Point Divide Utility
E.ENBL Enable interrupts
E.GP08 Control Program Timer 1
E.GP20 Control Program Time Update
E.IHBT Inhibit interrupts
E.Q Phase 1 Time Update (M.UM00)
E.Q Discrete Processor on Non-Interrupt Sequencer (M.DP00)
E.Q Simulated Platform Gimbal Angles (M.PG00)
E.Q Simulated Accelerometers (M.SA00)

*

E.Q Minor Loop Support (M.MS00)
 E.Q Phase 2 Control (M.CM00, M.CM10, M.CM20)
 E.QT Phase 1 Time Update (M.UM00)
 E.QT Phase 2 Control (M.CM00, M.CM10, M.CM20)

*

Input Data

The input data for this module is mission dependent and too large to enumerate. For more detailed information, see the program listing.

Output Data

The output data for this module is mission dependent and too large to enumerate. For more detailed information, see the program listing.

Status Words

D.VICR Internal Control Register Status Word
 D.VMC5 Mode Code 25
 D.VMC7 Mode Code 27
 D.VLNK DCS Status Word

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.OACX	0010	Initial X, Y, Z accelerometer readings at Phase 1 start
V.OACY	0014	
V.OACZ	0004	
D.VACT	0174	RTC at time of accelerometer read
V.BTC	0575	Begin telemetry cycle word
T.FAZ	0470	Platform azimuth corrected up to liftoff - 3 minutes
D.VGCX	0005	Initial Guidance Chi X
D.VTBI	0455	Indication of Time Base 0 start
D.VINT	0441	Indication of GRR interrupt
V.TD	0035	Delta between actual time or GRR and targeted time or GRR

*

D.VIA6 0024 Descending node angle

D.VIA7 0041 Inclination angle

Initial Conditions

None.

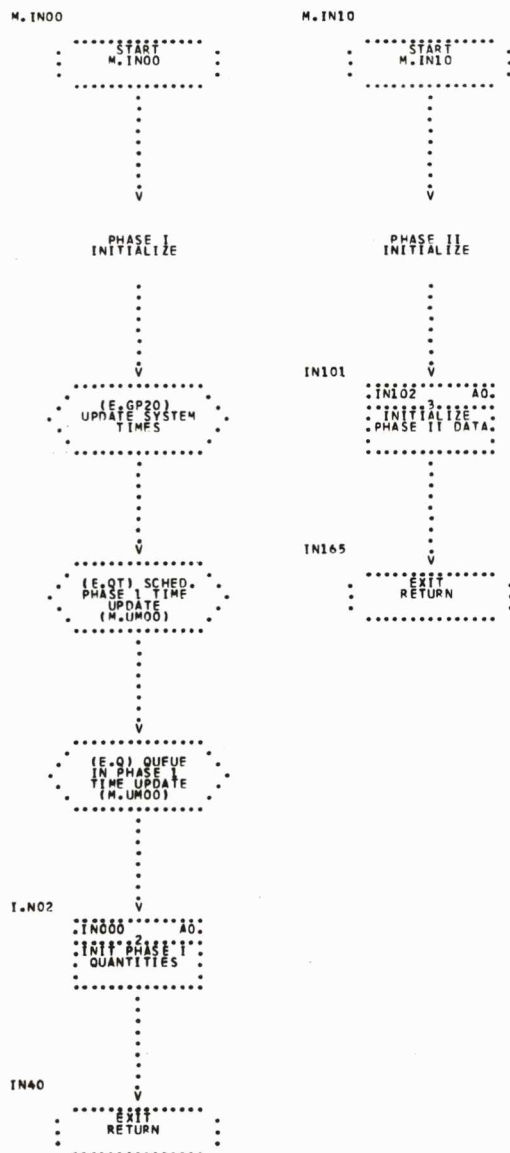
*

Notes

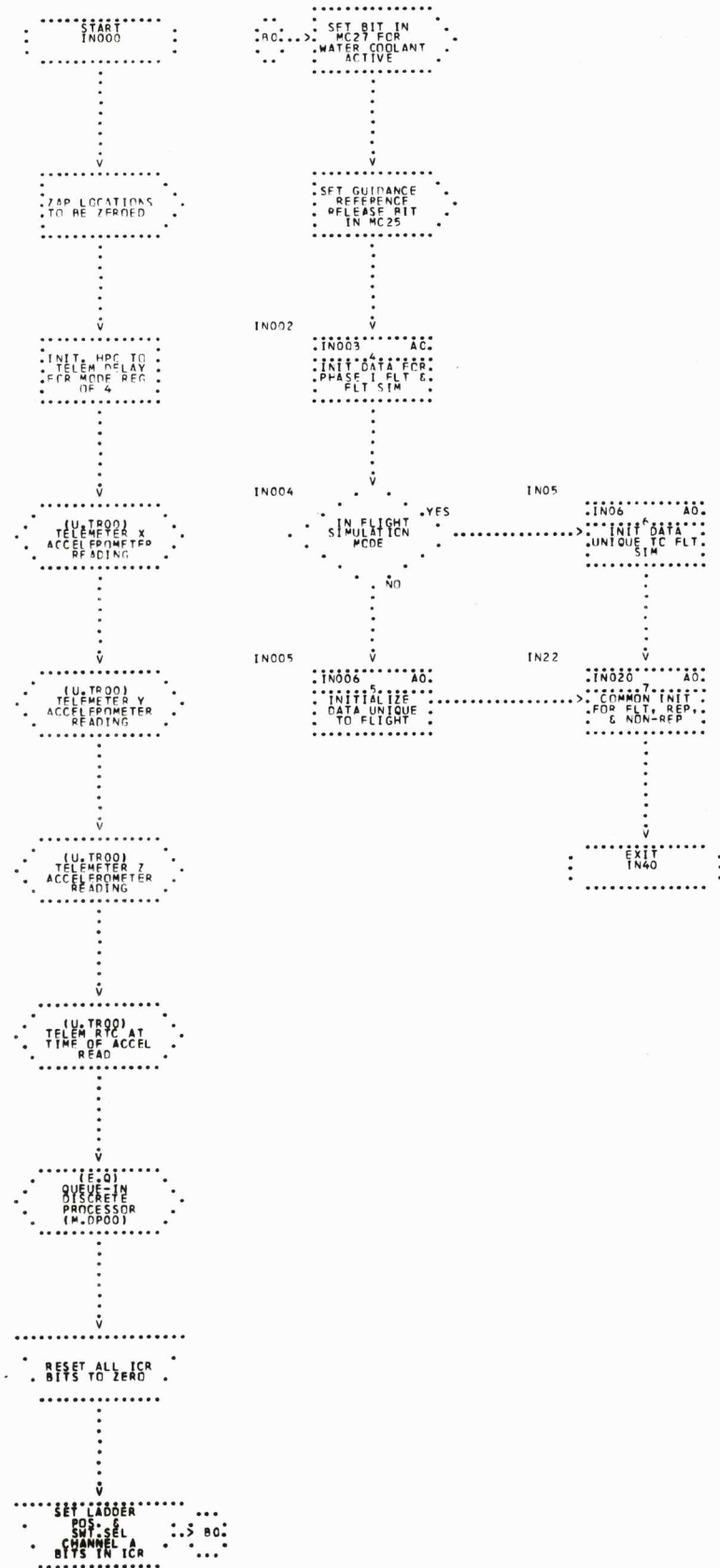
None.

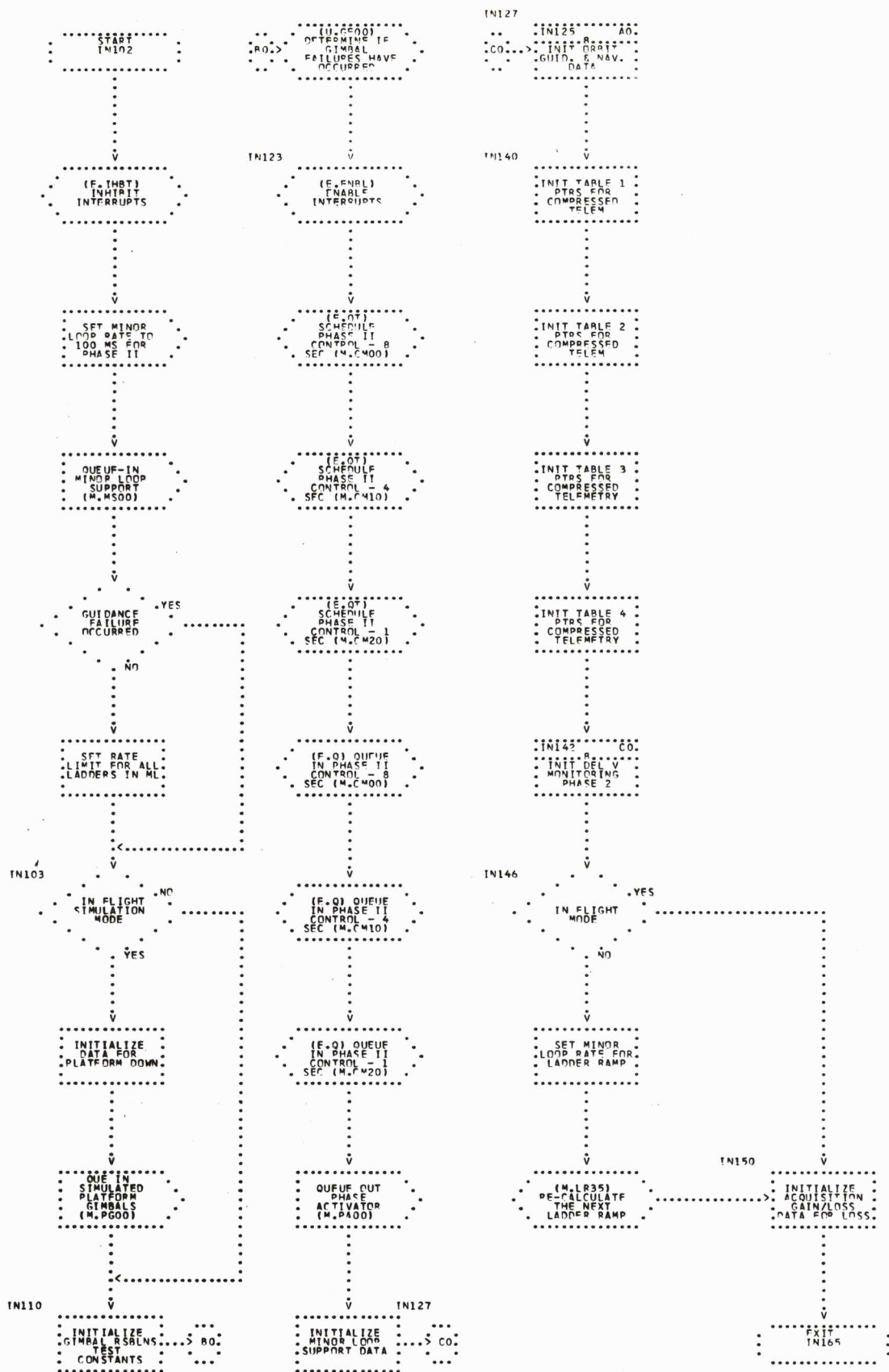
*

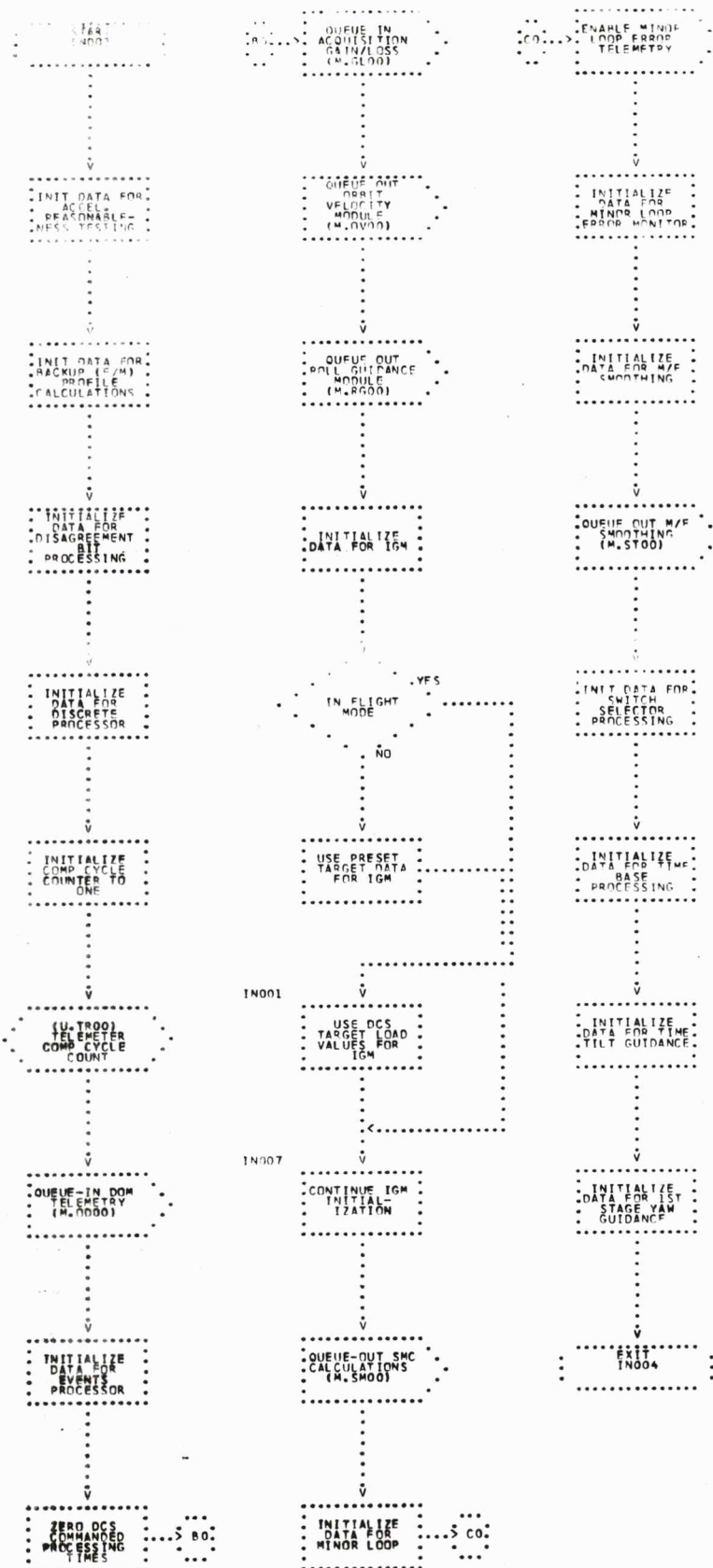
PHASE INITIALIZATION (IN)



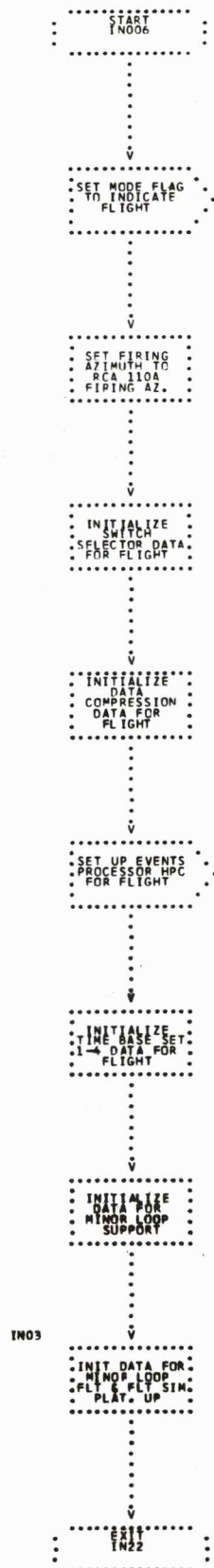
PHASE INITIALIZATION (INI)



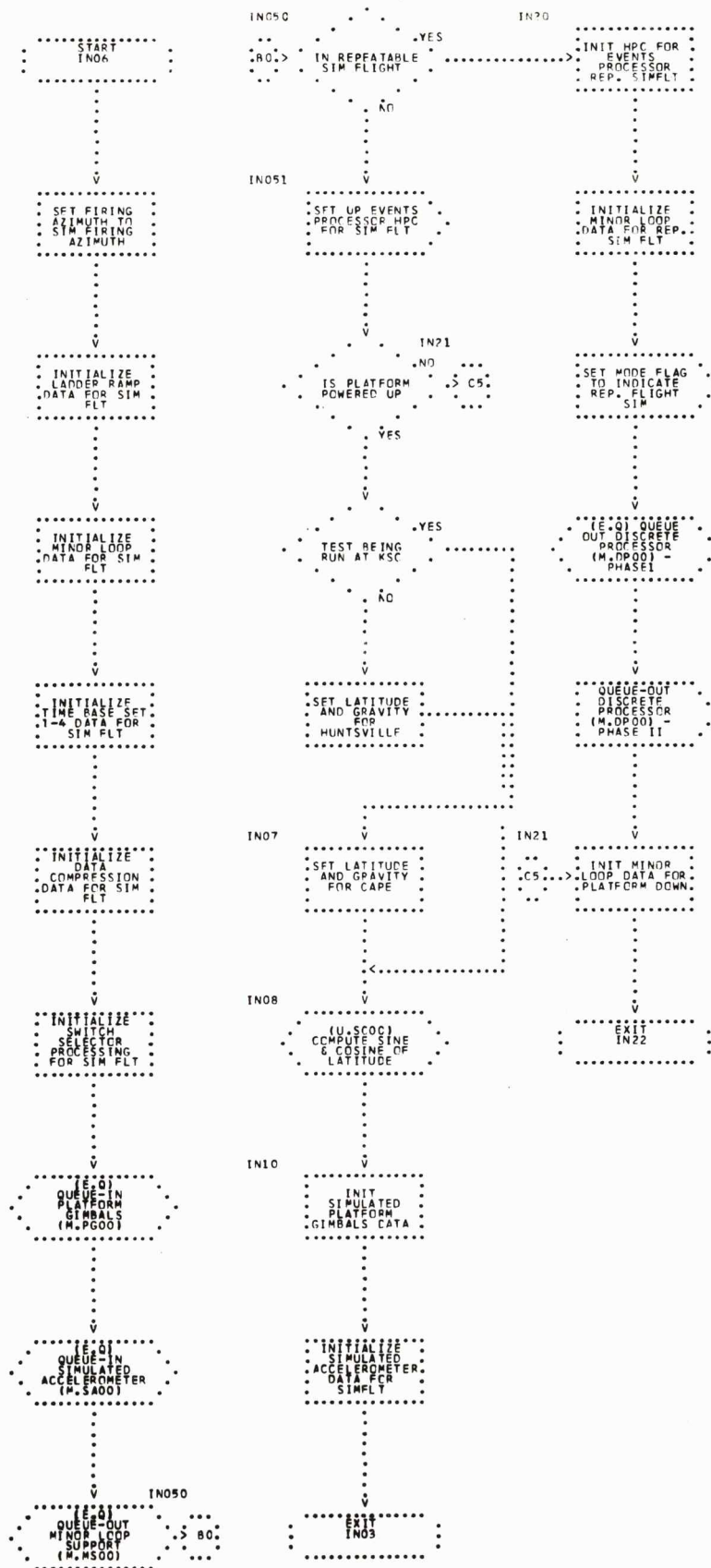


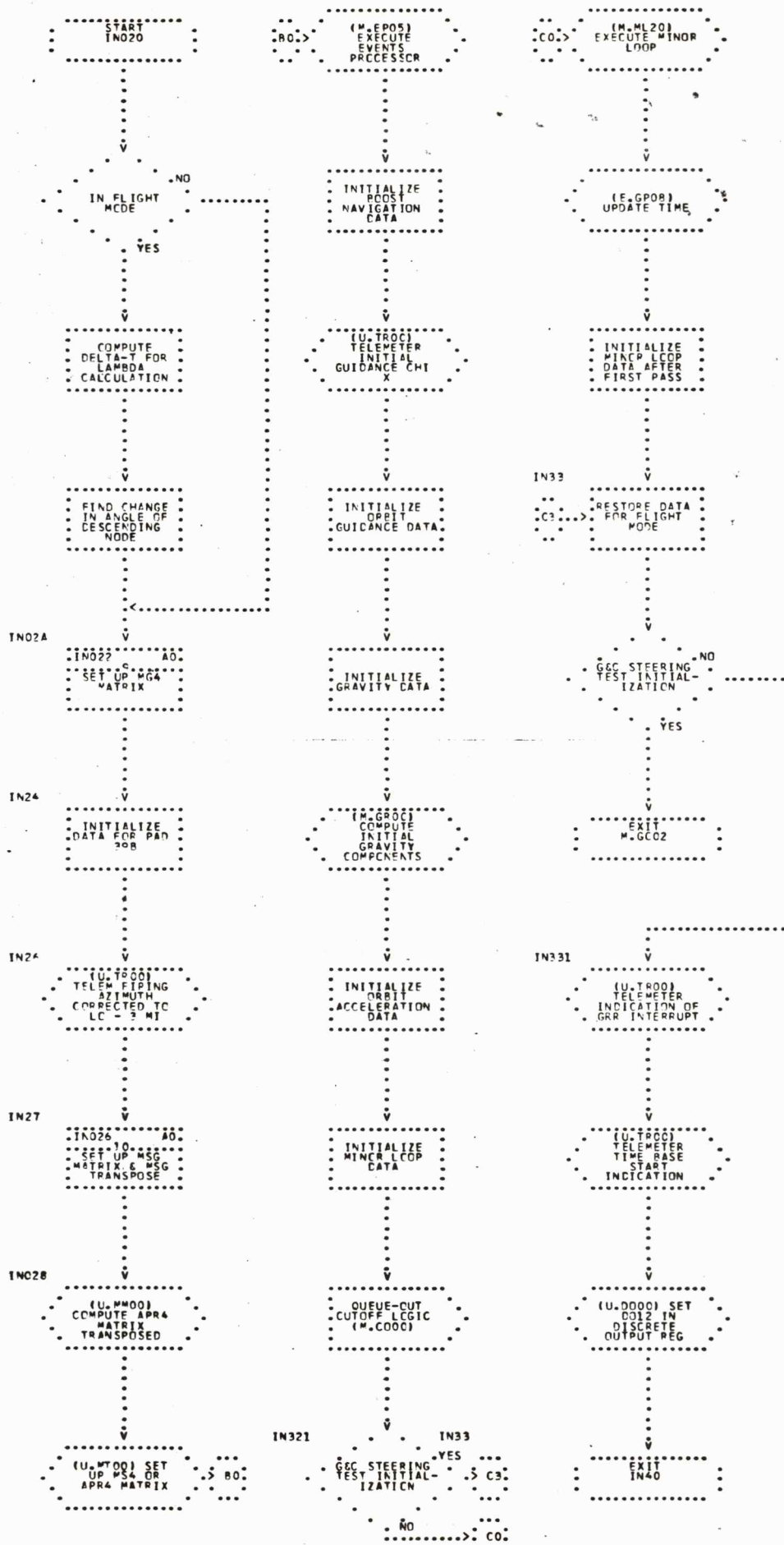


PHASE INITIALIZATION (IN)

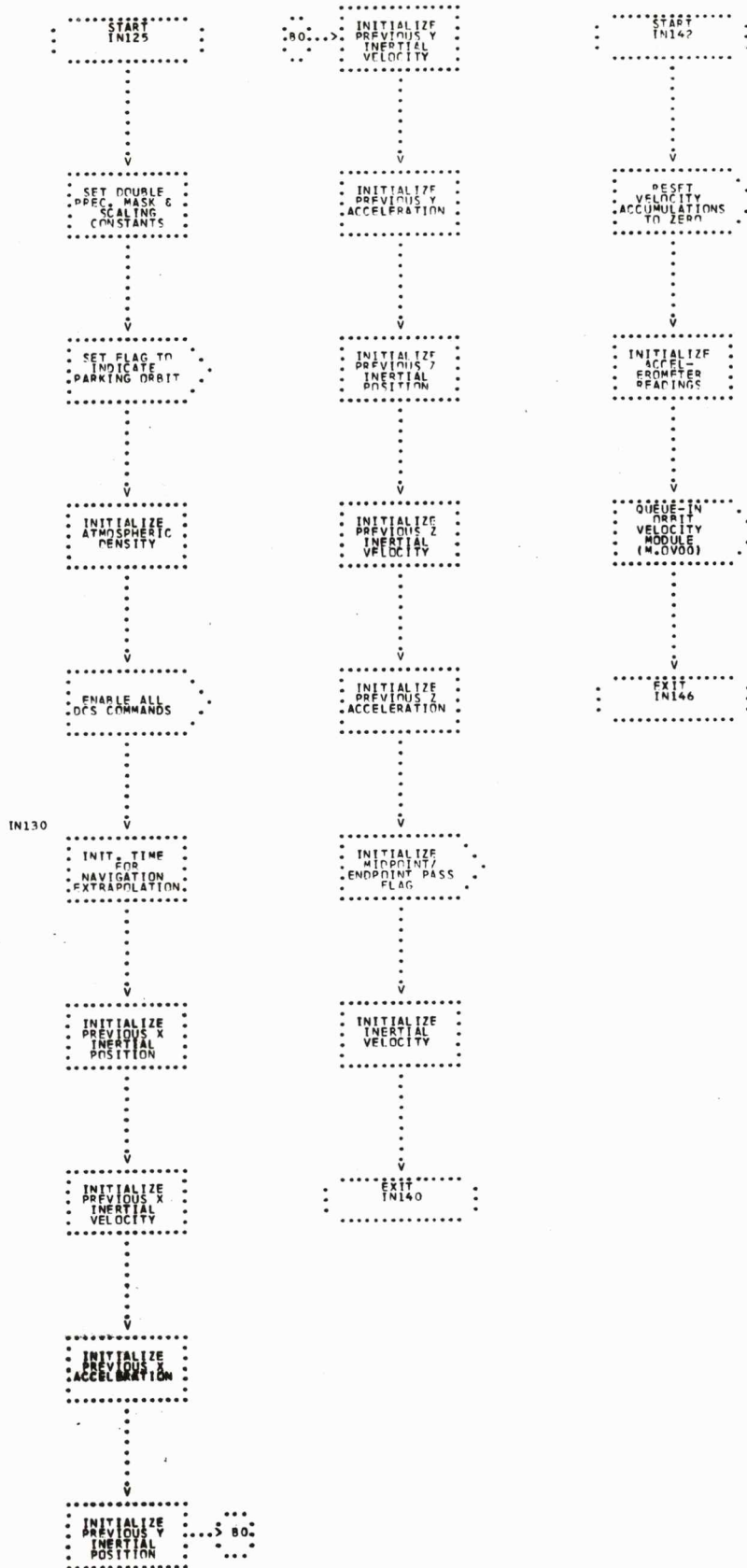


PHASE INITIALIZATION (IN)

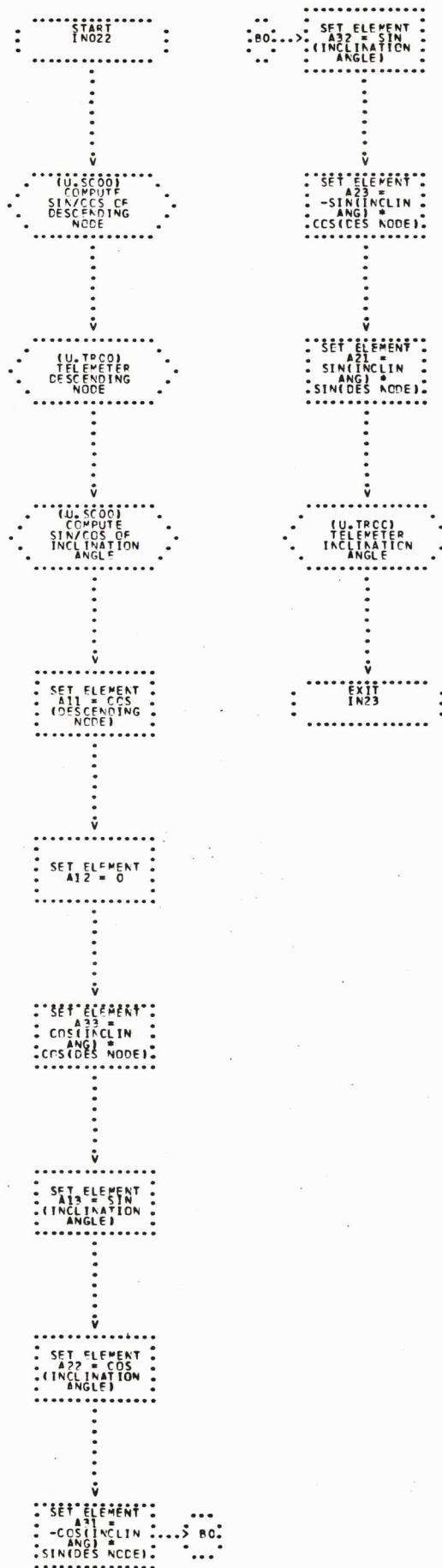




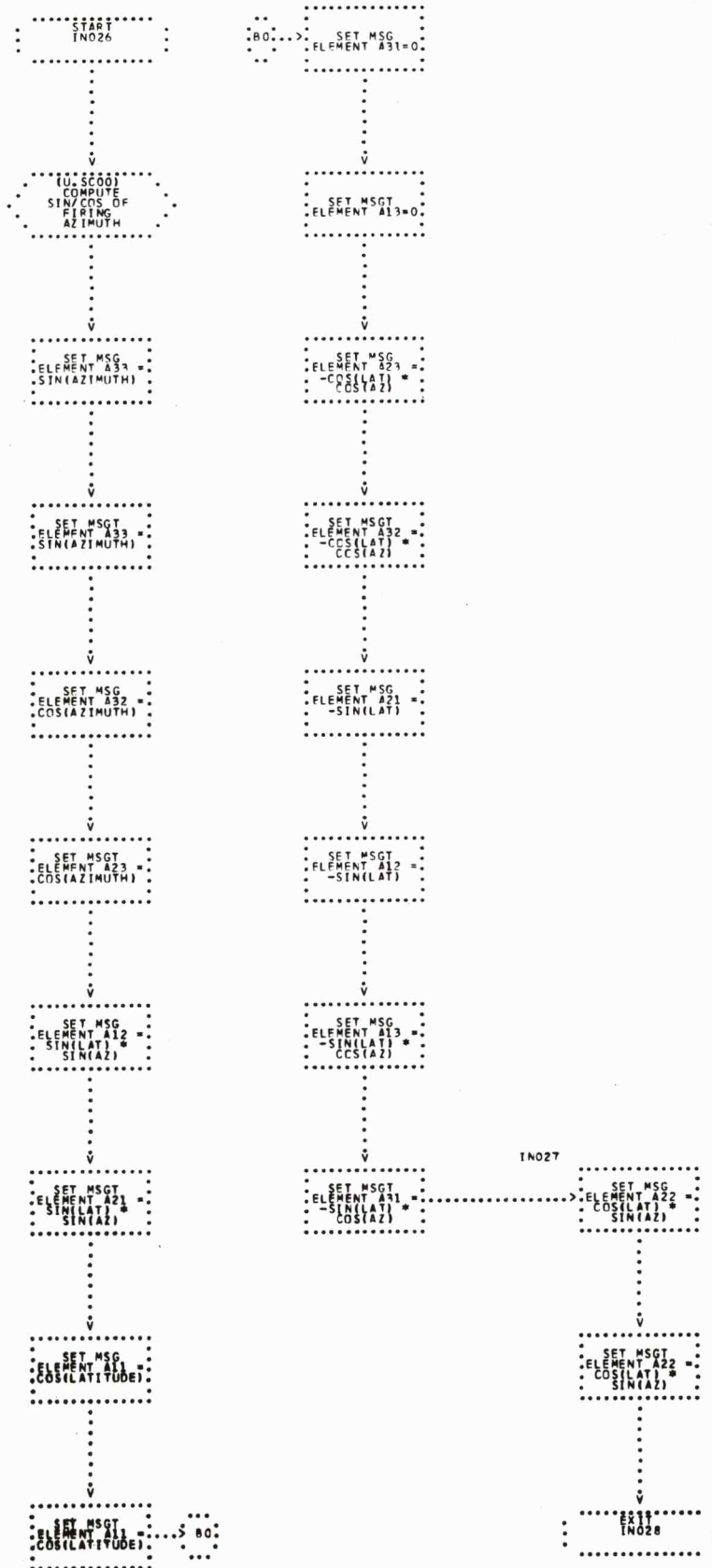
PHASE INITIALIZATION (IN)



PHASE INITIALIZATION (IN)



PHASE INITIALIZATION (IN)



2.31 NATURAL LOGARITHM UTILITY (LN)

Function

The function of the Natural Logarithm Utility is to compute the natural logarithm of the quotient of two specified numbers by means of an algorithm which utilizes a polynomial approximation to the natural logarithm function.

Entries

U.LN00 Entered by any module.

External

None.

Input Data

See Notes.

Output Data

See Notes.

Status Words

None.

Telemetry

None.

Initial Conditions

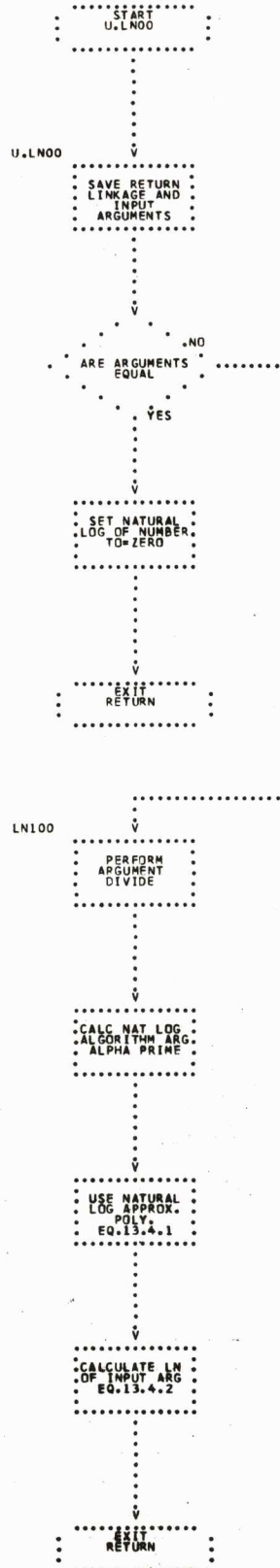
None.

Notes

Upon entry, two input arguments of identical scaling must be specified. The accumulator must contain the numerator value (ARG1), and the PQ must contain the denominator value (ARG2)

Upon exit, the desired natural logarithm value, $\ln (\text{ARG1}/\text{ARG2})$, will be in the accumulator scaled B1.

NATURAL LOG UTILITY (LN)



2.32 LIFTOFF SEARCH (LO)

Function

The function of the Liftoff Search module is to monitor the discrete input register for indication of vehicle liftoff (DI 24 or DI 7) and on detecting either discrete, access the Time Base 1 module to initiate Time Base 1.

Entries

M.L000 Entered from Timer 1 every 40 milliseconds from GRR + 17.4 seconds until Time Base 1 is initiated.

External

M.ML20 Minor Loop
M.TB15 Time Base 1
U.TR00 Real Time Telemetry Delay Utility

Input Data

None.

Output Data

D.VTRS RTC reading to be used by the Control Program Time Update routine at a new time base start.
D.VTBI Indication of time base initiation *

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.LOWD	0465	Special liftoff detection word

Initial Conditions

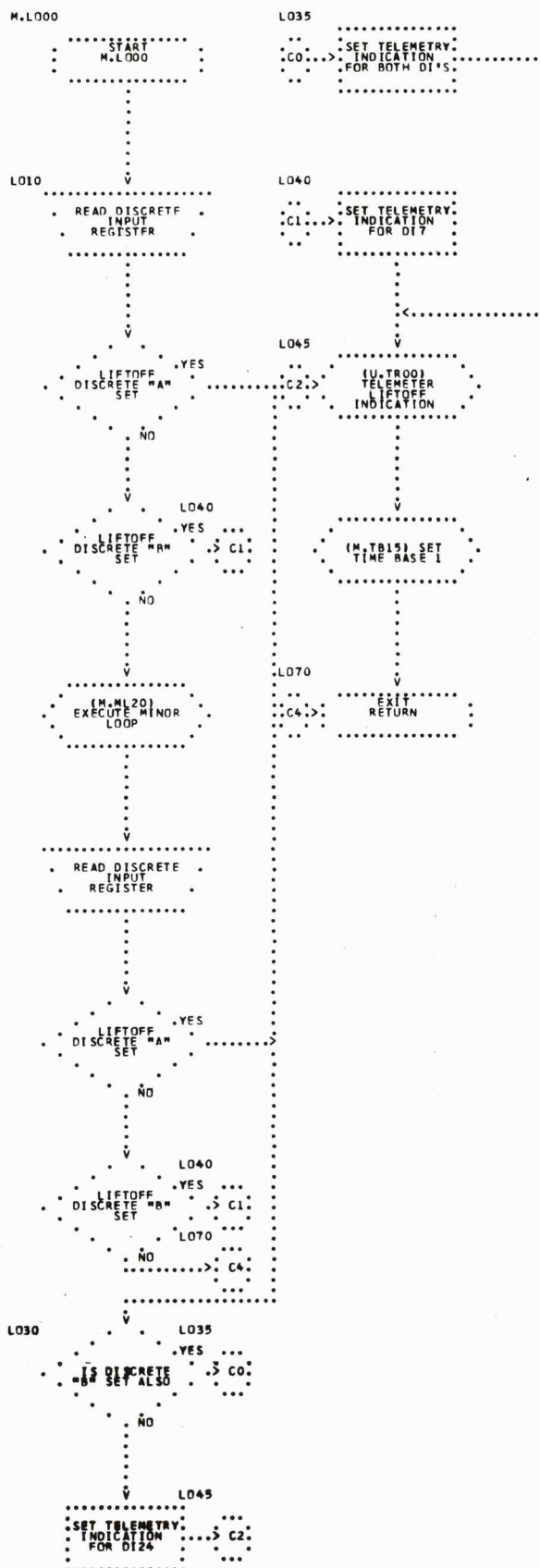
None.

Notes

The Liftoff Search module is activated by the Events Processor at 17.4 seconds after GRR by altering the Timer 1 HPC value, D.GML, to M.L000 for the Liftoff Search module instead of the Minor Loop module. Liftoff Search accesses Minor Loop on each entry so that Minor Loop continues to execute at a 40 millisecond rate. The Time Base 1 module deactivates Liftoff Search by resetting the Timer 1 HPC to M.ML00 for Minor Loop.

On entry, the Liftoff Search module checks for liftoff. If liftoff has not occurred, it accesses Minor Loop and then repeats the check for liftoff. Since the Minor Loop operation requires approximately 20 milliseconds, the effective rate of checking for liftoff is every 20 milliseconds.

1



2.33 LADDER RAMP PROCESSOR (LR)

Function

This module will provide the Minor Loop module with all data needed to output a predefined ladder ramp profile when operating in the Flight Simulation Mode.

Entries

- M.LR10 Entered from Timer 2 when time to issue the next ramp.
- M.LR20 Entered from the Time Base Set module for initialization at each time base change.
- M.LR35 Entered from the Phase Initialization module at the beginning of Phase 2.

*

External

- E.ENBL Enable interrupts
- E.IHBT Inhibit interrupts
- E.Q Ladder Ramp Processor (M.LR10)
- E.QT Ladder Ramp Processor (M.LR10)

Input Data

- D.FTBC Time base change flag

Output Data

- D.VDCX X delta chi (ladder increment in Pirads)
- D.VDCY Y delta chi (ladder increment in Pirads)
- D.VDCZ Z delta chi (ladder increment in Pirads)

D.VLRC Ladder ramp command chi update counter

*

Status Word

None.

Telemetry

None.

Initial Conditions

L.LRR1 The master table point must be initialized.

V.MLR1 The rate of Minor Loop execution must be set whenever changed.

Notes

This module is executed only in the Flight Simulation Mode of program operation. It requires as input the Ladder Ramp Table (see attached format).

Both the ladder rate and magnitude are limited by the Minor Loop module according to the maximum limits specified for control of the ladder outputs in a mission dependent table.

The Ladder Ramp Table is sub-divided according to time bases. If a requested ramp cannot be completed within the current time base, the processing of the ramp is cancelled and the ladders are zeroed. The ramps referenced to the new time base are then started.

If two ramps are requested too close together, the second ramp will be superimposed on the first.

*

Ladder Ramp Table Format

The Ladder Ramp Table is composed of two words per entry. The format for each entry is as follows:

Word 1	Ladder ramp
Word 2	Ramp occurrence time

Word 1

S	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
A	B	B	B	S	C	C	C	C	C	S	S	S	S	S	S	S	S	S	S	S	D	D	D	D	D

Where A = Specifies ramp direction.

0 = down ramp; 1 = up ramp.

B = Specifies pitch, roll, yaw or combined ladder ramp.

001=Yaw, 010=Roll, 100=Pitch, 011=Yaw & Roll,

111=Pitch, Yaw & Roll, etc.

C = Specifies the ladder rate in degrees per second.

LSB=0.5°/sec, MSB=8°/sec, MAX=15.5°/sec.

D = Specifies the ladder magnitude in degrees.

LSB=0.5°, MSB=8°, MAX=15.5°.

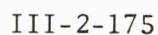
S = Spares, i.e., unused bits.

Word 2

S	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F

Where F = The time for the ramp to occur in real time clock pulses scaled B27, referenced from time base.





2.34 MINOR LOOP (ML)

Function

The Minor Loop module has four functions:

- (1) To read platform gimbal angles, to compute yaw, roll and pitch attitude corrections needed to drive the gimbals to the desired position, and to output these commands to the LVDA.
- (2) To perform various tests on the platform gimbal to verify the hardware in the LVDA and the ST-124M platform.
- (3) To insure that the gimbal reads and attitude error commands will be telemetered by the DOM.
- (4) To access the Cutoff Logic module when requested.

Entries

M.ML00 Entered from Timer 1 only for Flight Simulation Mode.
M.ML20 Entered from Timer 1 or from the Liftoff Search module.

External

M.C000 Cutoff Logic
M.DG00 Disagreement Bit Processing
U.TR00 Real Time Telemetry Delay Utility
U.D000 Discrete Output Register Set/Reset Utility

Input Data

D.VLDB	Count of times switched to ladder converter B	*
D.VDCX	Delta Chi X (Roll commanded chi increment)	*
D.VDCY	Delta Chi Y (Pitch commanded chi increment)	*

D.VDCZ	Delta Chi Z (Yaw commanded chi increment)	*
D.VLRC	Ladder ramp command chi update counter	
V.CG10	Gimbal reasonableness rate limit constant (Backup 1st pass)	
V.CG11	Gimbal reasonableness rate limit constant (Backup 1st pass) (crossover)	
V.CG20	Yaw and Pitch gimbal reasonableness rate limit constant (Backup 2nd pass)	
V.CG21	Yaw and Pitch gimbal reasonableness rate limit constant (Backup 2nd pass) (crossover)	
V.CG30	Roll gimbal reasonableness rate limit constant (Backup 2nd pass)	
V.CG31	Roll gimbal reasonableness rate limit constant (Backup 2nd pass) (crossover)	
D.GMLC	Variable HOP constant to M.C000 or M.L299	
D.VA1	Coefficients used to convert the inertial platform readings to body frame attitude corrections	
D.VA2		
D.VA3		
D.VA4		
D.VA5		
F.BUGX	Backup X gimbal active, alter RTC	
F.BUGY	Backup Y gimbal active, alter RTC	
F.BUGZ	Backup Z gimbal active, alter RTC	
D.VTMM	Elapsed time in mission from GRR to last time update	*
D.VMLT	Execution time of minor loop	
D.VDGS	Count of failures in disagreement bit circuitry	*
D.FDBF	Disagreement Multiplexer failure flag	
D.VXRE	X resolver reasonableness failure count	*
D.VYRE	Y resolver reasonableness failure count	*
D.VZRE	Z resolver reasonableness failure count	*
D.GMLT	HOP constant to telemetry delay or to telemetry bypass	
V.IRE	Resolver failure limit 2	
D.VHDB	Count of B multiplexer failures	
D.VHDA	Count of A multiplexer failures	
D.VM05	Rate limit for all ladders in minor loop	
V.OCK	Zero gimbal test constant	*

D.VM06	Pitch and yaw ladder magnitude limit	*
D.VM16	Roll ladder magnitude limit	*
D.VRTC	Real time clock reading at last time update	*
D.VMLD	Minor loop rate of execution	
V.ML10	X gimbal reasonableness test limits	
V.ML11	X gimbal reasonableness test limits (crossover)	
V.ML20	Y gimbal reasonableness test limits	
V.ML21	Y gimbal reasonableness test limits (crossover)	
V.ML30	Z gimbal reasonableness test limits	
V.ML31	Z gimbal reasonableness test limits (crossover)	

Output Data

D.VTHX	Total roll gimbal angle	*
D.VTHY	Total pitch gimbal angle	*
D.VTHZ	Total yaw gimbal angle	*
D.VCCX	Commanded Chi X (Rate limited roll gimbal angle)	*
D.VCCY	Commanded Chi Y (Rate limited pitch gimbal angle)	*
D.VCCZ	Commanded Chi Z (Rate limited yaw gimbal angle)	*
D.FDBF	Disagreement multiplexer failure flag	
D.FSMC	Steering Misalignment Correction inhibit flag	*
V.XGR	Fine X gimbal angle reading	
V.YGR	Fine Y gimbal angle reading	
V.ZGR	Fine Z gimbal angle reading	
V.ROL	Roll commands: present, previous and actual	
V.PIT	Pitch commands: present, previous and actual	
V.YAW	Yaw commands: present, previous and actual	
D.VLDB	Count of times switched to ladder converter B	*
D.VDGW	Data storage for Disagreement Bit Processing	
D.VDGS	Count of failures in disagreement bit circuitry	*
D.VHDB	Count of B multiplexer failures	
D.VHDA	Count of A multiplexer failures	
D.VM05	Rate limit for all ladders in Minor Loop	
V.ML12	X low order gimbal resolution	
V.ML22	Y low order gimbal resolution	

V.ML32	Z low order gimbal resolution	
D.VXRE	X resolver reasonableness failure count	*
D.VYRE	Y resolver reasonableness failure count	*
D.VZRE	Z resolver reasonableness failure count	*
D.VMLT	Execution time of next minor loop	

Status Words

D.VMC4	Mode Code 24	
D.FIL3	Timer 1 interrupt in progress indicator	
D.VEMR	Error Monitor Register	
D.VICR	Internal Control Register	
D.VMC7	Mode Code 27	*

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.MLET	0570	Error telemetry word

Initial Conditions

Minor Loop magnitude and rate limits and test constants are initialized at mission start, phase start and as required by flight sequencing.

Notes

The ICR must be set to select ladder converter A and to start the yaw fine resolver on minor loop interrupt. The ICR must be set so that the resolver will be read in B-A order.

The HOP constant, D.GMLC, is set to M.C000 by the S-IVB Cutoff Prediction module to activate the Cutoff Logic module. The Cutoff module resets the HOP constant to M.L299 to deactivate itself.

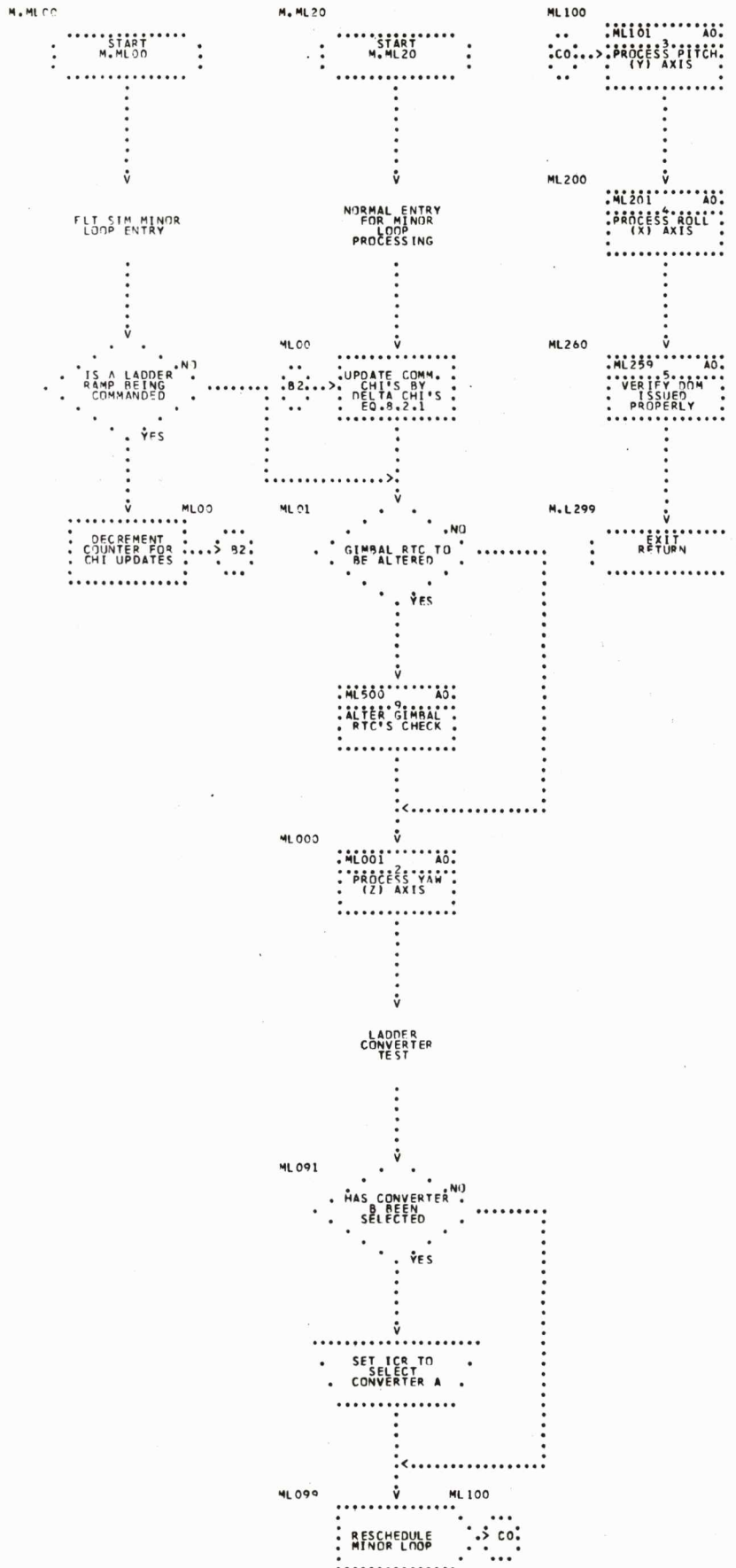
The resolver failure constant for setting D04 and D06 (V.IRE) must be set according to the phase which the program is in:

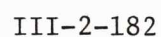
$$V.IRE = (R_E * R_M * 3/5 + 1)$$

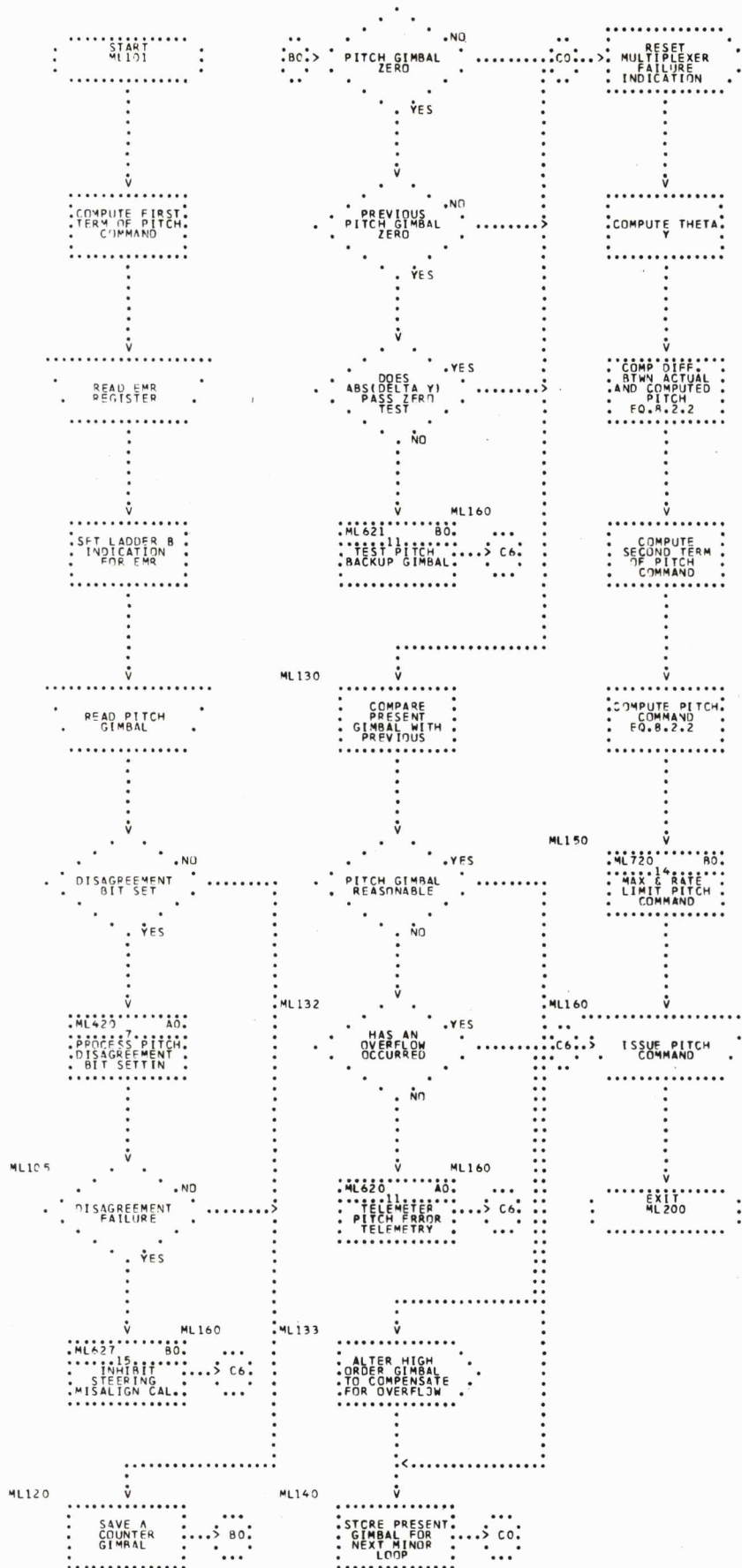
where

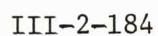
R_E = Rate Minor Loop Error Monitor cycled

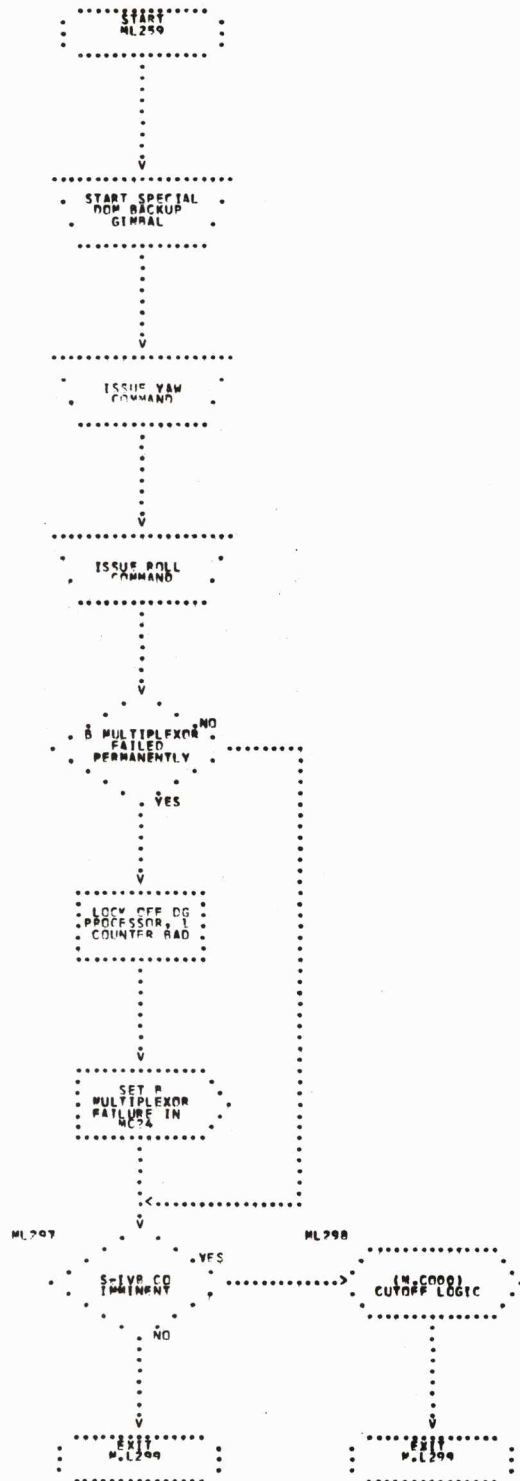
R_M = Minor Loop Rate



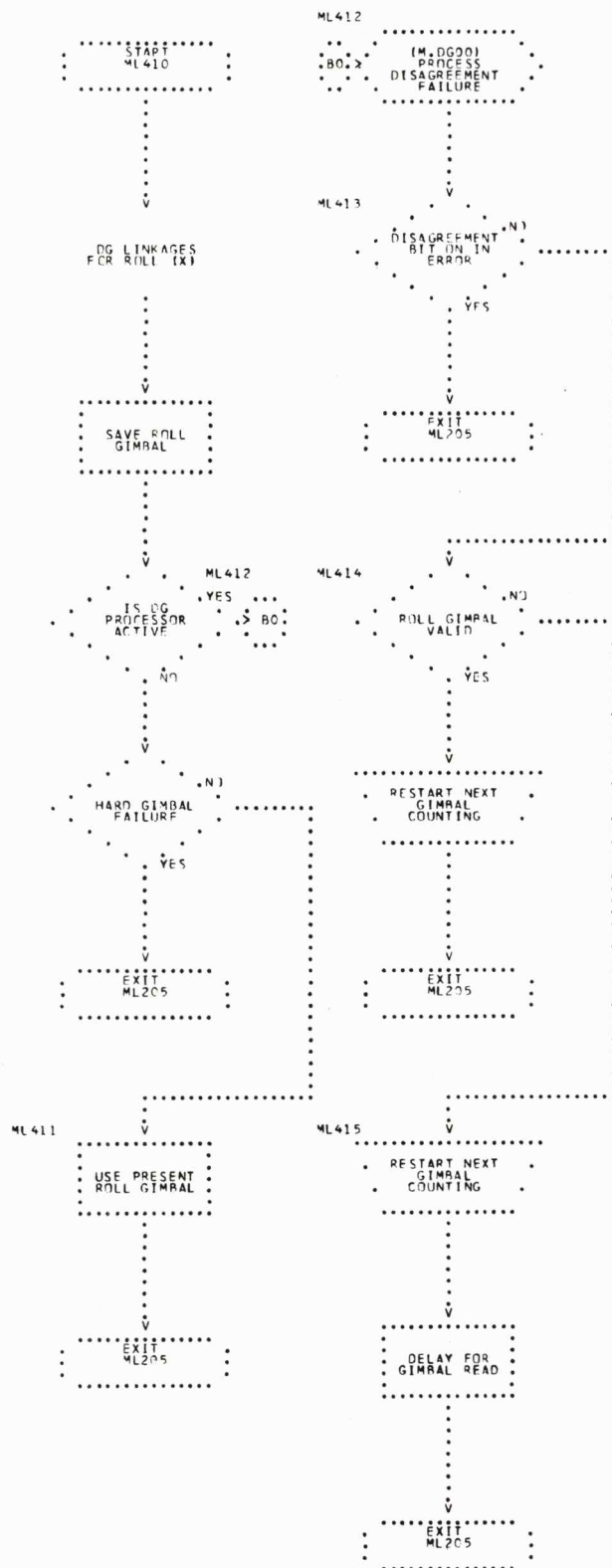






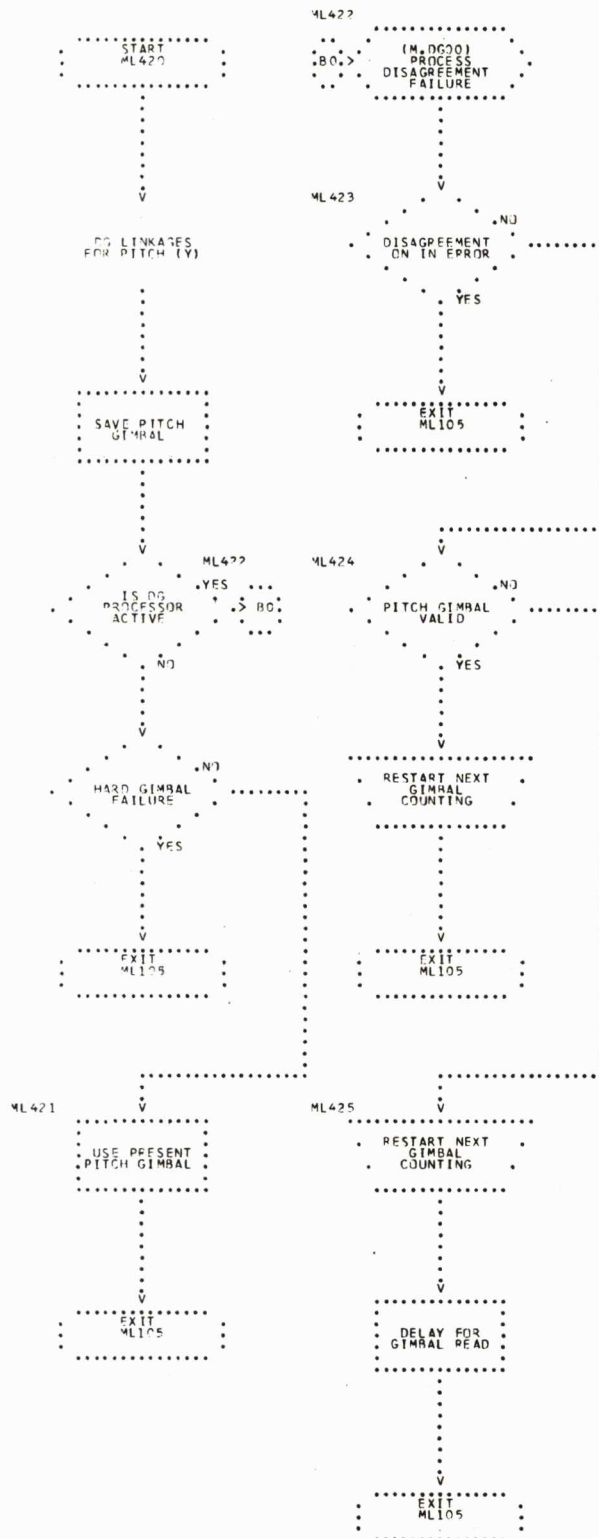


MINOR LOOP (ML)

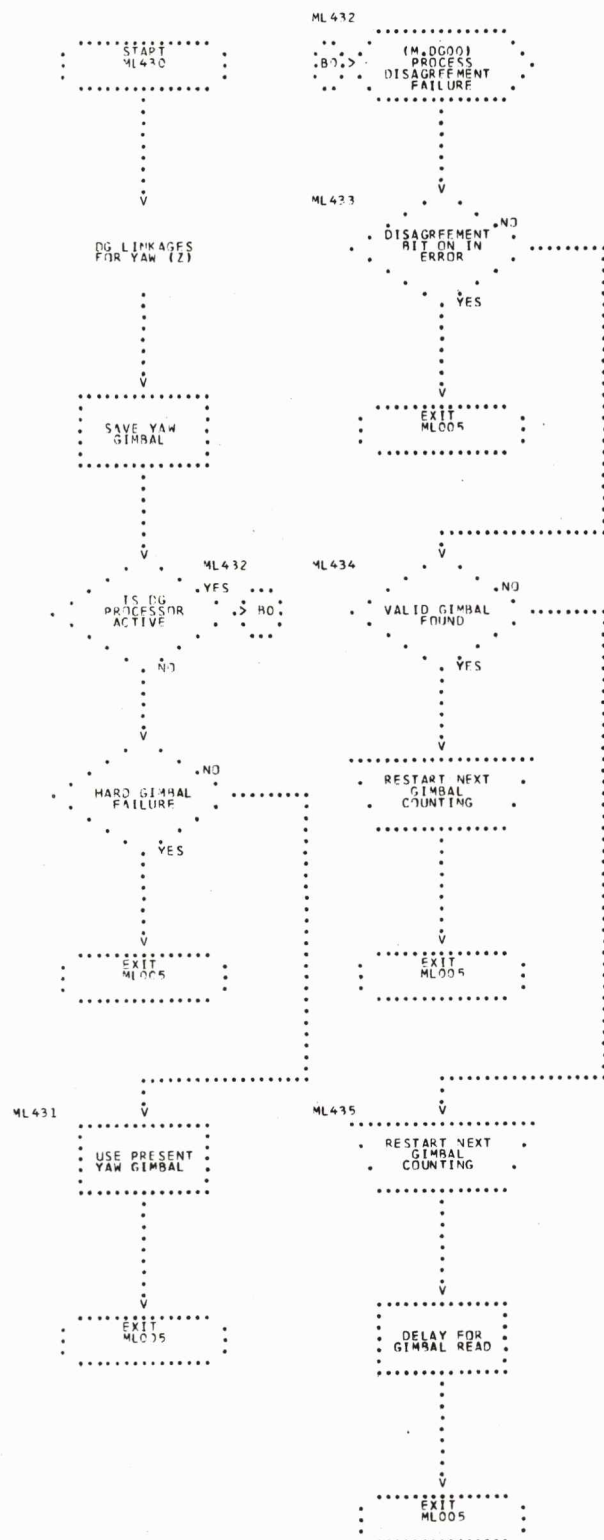


MINOR LOOP (ML)

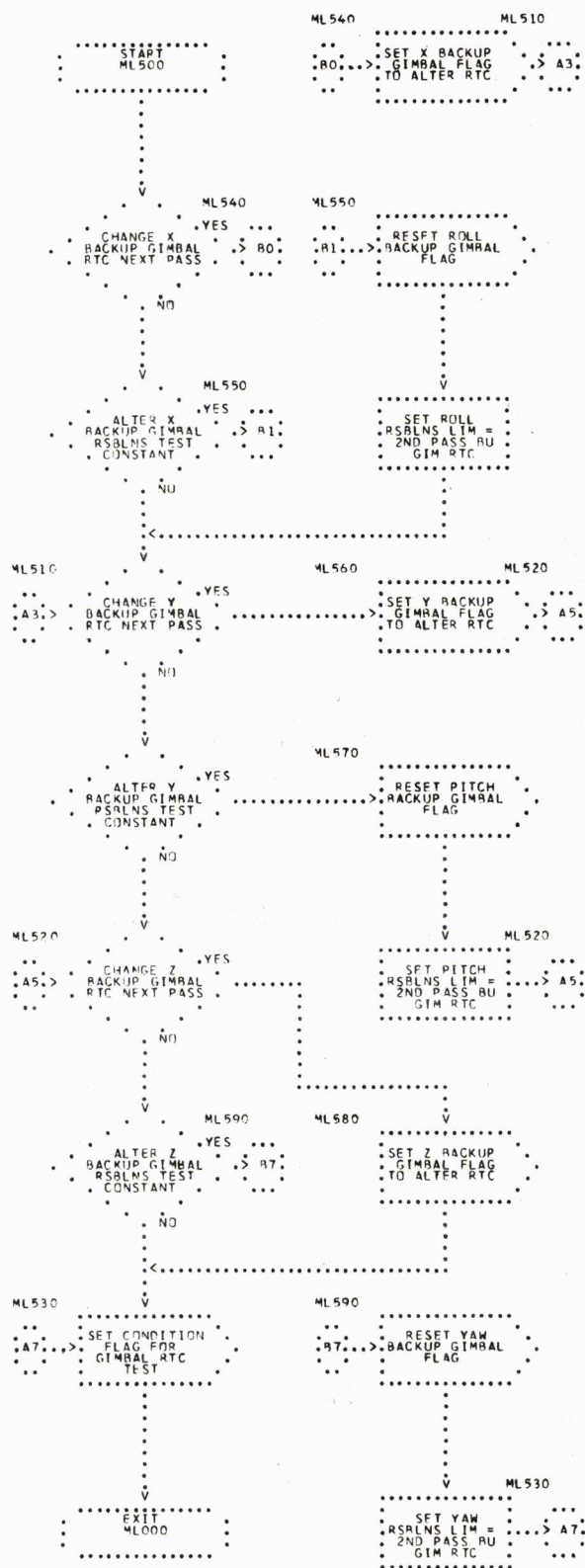
7



MINOR LOOP (ML)

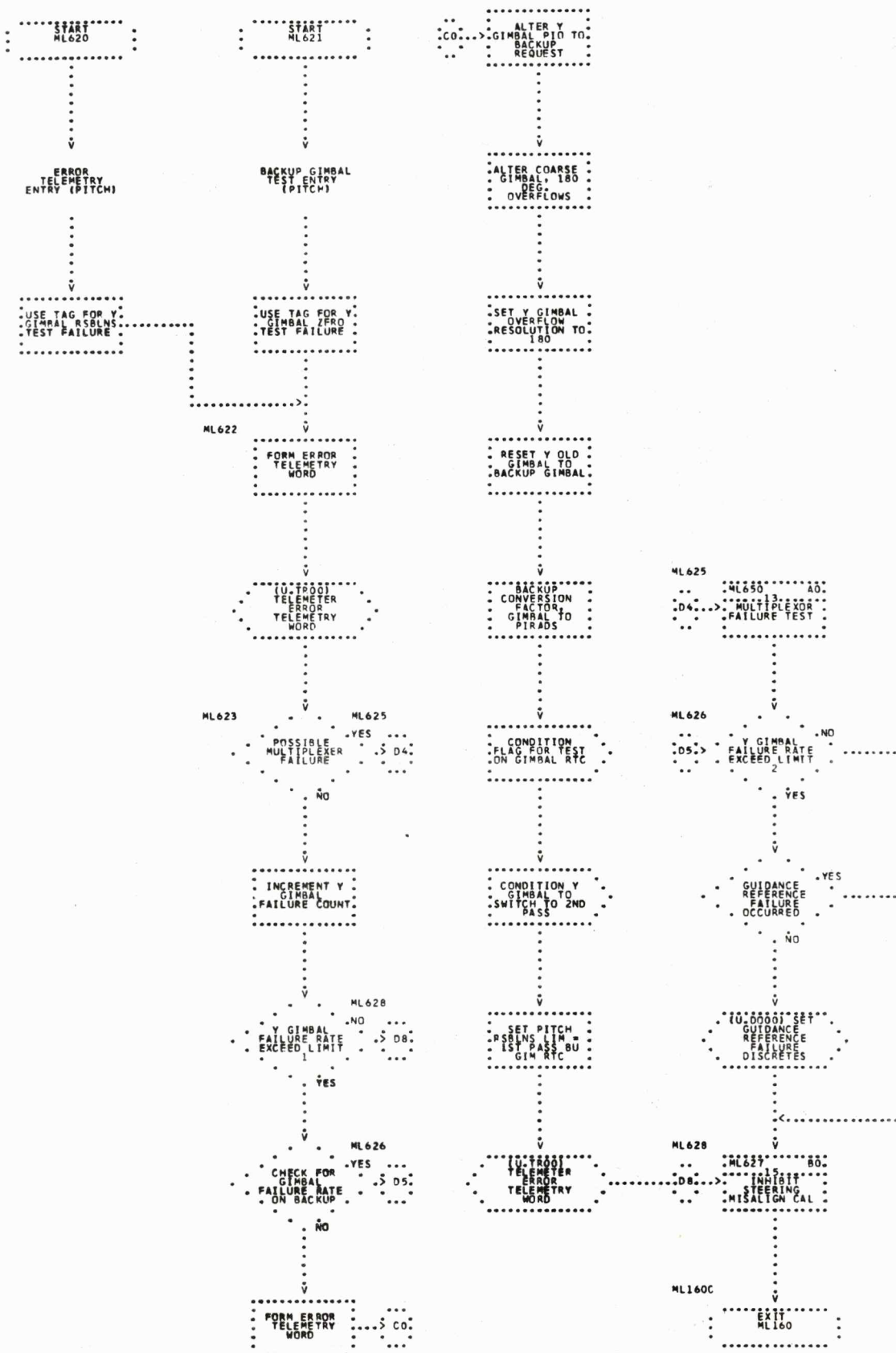


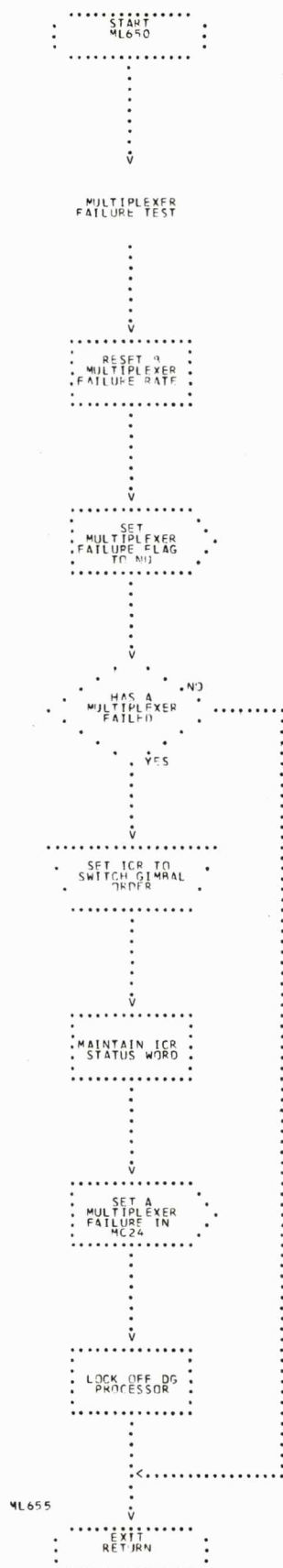
MINOR LOOP (ML)



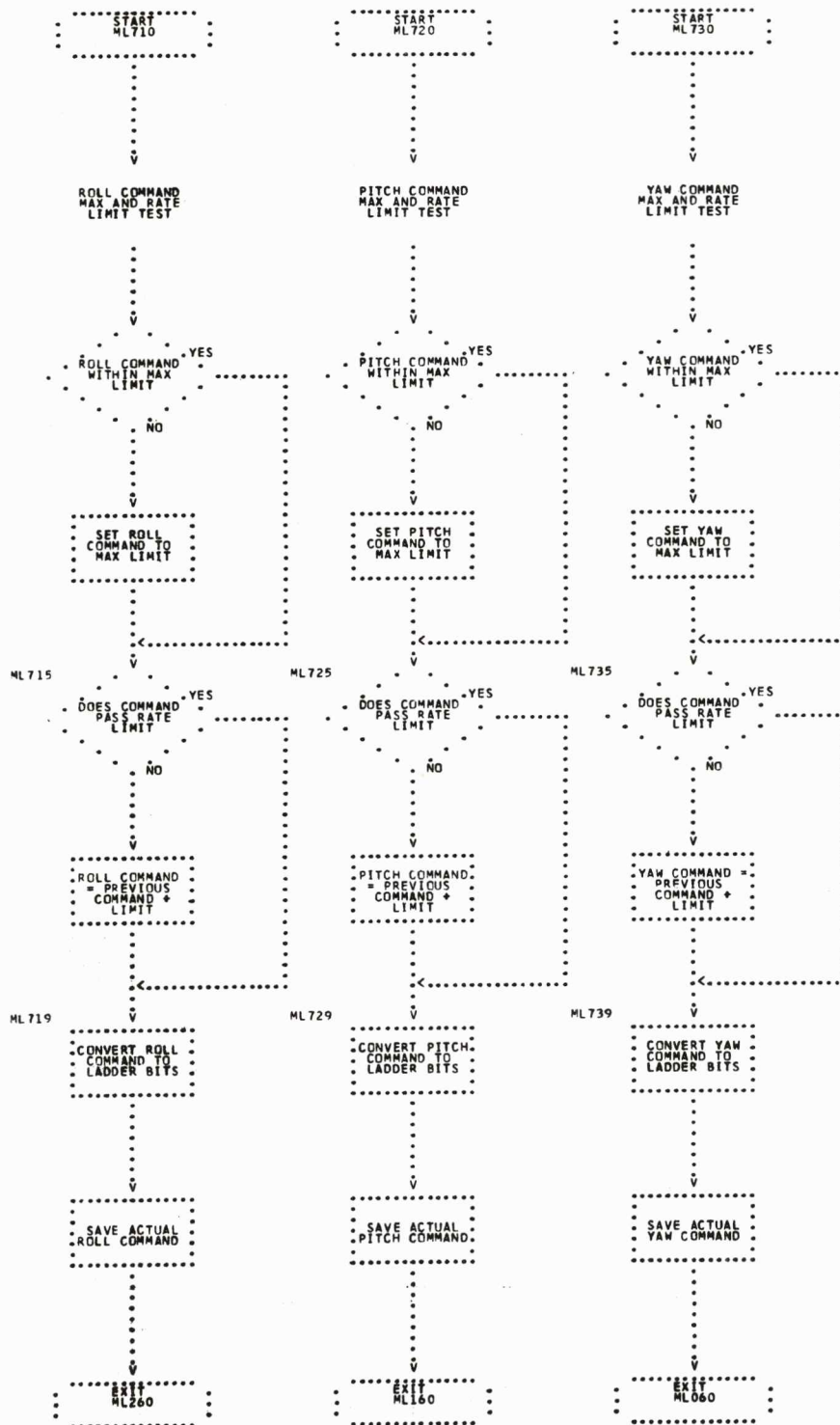


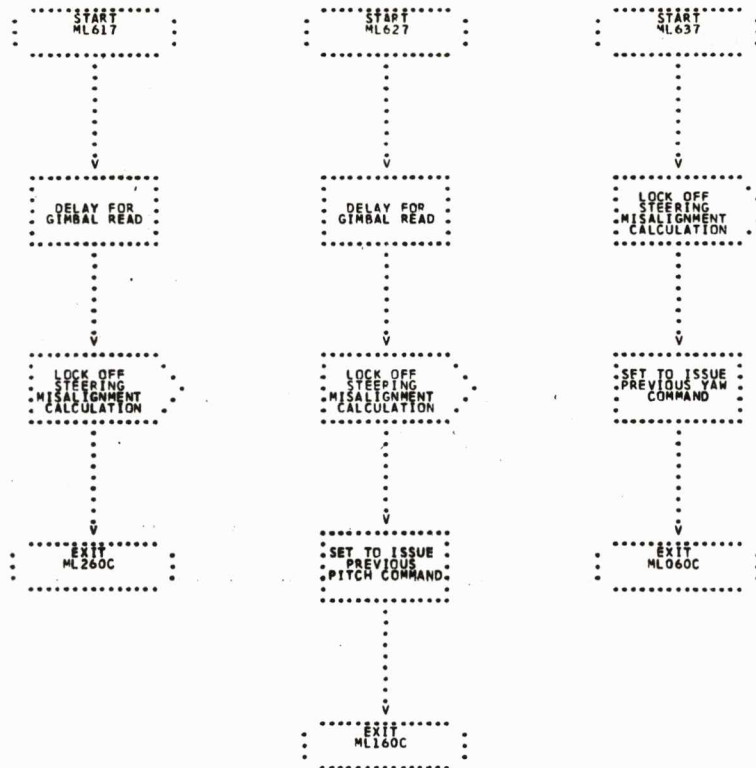
MINOR LOOP (ML)





MINOR LOOP (ML)





2.35 MATRIX MULTIPLY UTILITY (MM)

Function

The function of the Matrix Multiply Utility module is to compute the product of two specified matrices.

Entries

U.MM00 Entered by any module.

External

U.VD01 Vector Dot Product Utility

Input Data

See Notes.

Output Data

See Notes.

Status Words

None.

Telemetry

None.

Initial Conditions

None.

Notes

Two arguments must be supplied upon entry:

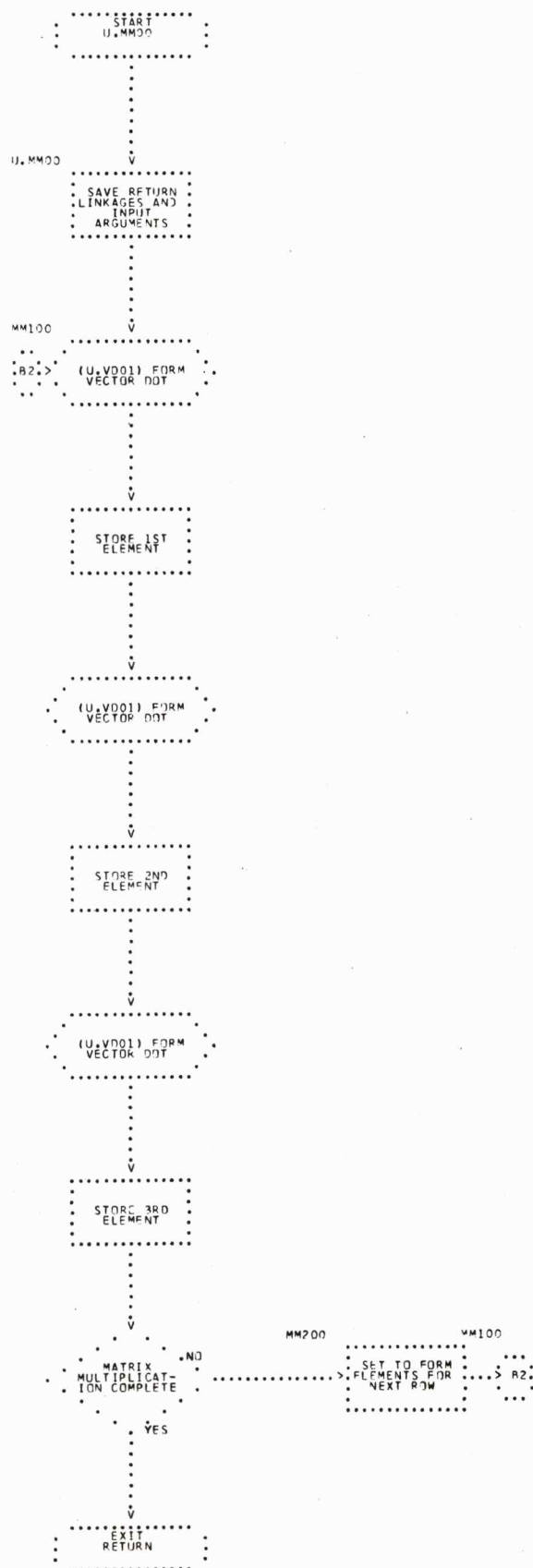
- (1) The accumulator must contain a computer word of the form CLA A; MPY B, where A and B are the initial locations of the matrices to be multiplied.
- (2) The PQ must contain a computer word of the form STO C in syllable 1 and a counter (scaled B25) in syllable 0. C must be the initial location of the matrix in which the product matrix is to be stored.

Upon exit, the resultant product matrix will be stored in the matrix specified by the computer word in the PQ.

All matrices must be grouped in vector form. Each column or row is to be stored as a vector using the VEC assembler pseudo operation. The three vectors required to represent a 3x3 matrix must be stored using the MAT assembler pseudo operation.

The A matrix must be stored by rows and the B matrix by columns. The resultant C matrix will be stored by columns.

Since utility modules may not normally access each other, this module saves its return linkage and sets up the necessary linkage to use the Vector Dot Product Utility.



2.36 ORBITAL GUIDANCE MANEUVER PROCESSOR (MP)

Function

The function of the Maneuver Processor is to initialize the parameters for proper execution of the orbital guidance maneuvers. This module is executed only one time for each maneuver.

Entries

- M.MP00 Entry made via variable HOP constant from the orbital guidance module.
- M.MP50 Entry made from Timer 2 when it is time for a new maneuver.
- M.MP55 Entry made from Timer 2 when it is time for a generalized maneuver.

External

- E.QT Next Maneuver Processor Entry (M.MP50)
- E.Q Maneuver Processor (M.MP50)
- E.Q Maneuver Processor (M.MP55)
- U.SC00 Sine-Cosine Utility

Input Data

- | | | |
|--------|--|---|
| F.MTYP | Commanded maneuver type flag | * |
| D.VACT | Real Time Clock reading associated with D.VTAS or D.VTNM | * |
| D.VTB | Time elapsed in current time base | * |
| D.KT5 | Time base 5 start time from GRR | * |

Output Data

V.RAM	Inertial roll angle in pirads	
V.SPM	Inertial pitch angle in pirads or sine of pitch delta from local horizontal	*
V.SYW	Inertial yaw angle in pirads or sine of yaw delta from local horizontal	*
V.CPM	Cosine of pitch delta from local horizontal	*
V.CYW	Cosine of yaw delta from local horizontal	*
F.SCAH	Flag to indicate S/C returned control	
F.SLR	Special local reference maneuver first pass flag	
F.SCIC	First pass flag for Spacecraft Control return	
G.MPCK	Variable HOP to either the Maneuver Processor or next Orbital Guidance instruction	
F.MANT	Maneuver type flag	

Status Word

D.VMC7 Mode Code 27

Telemetry

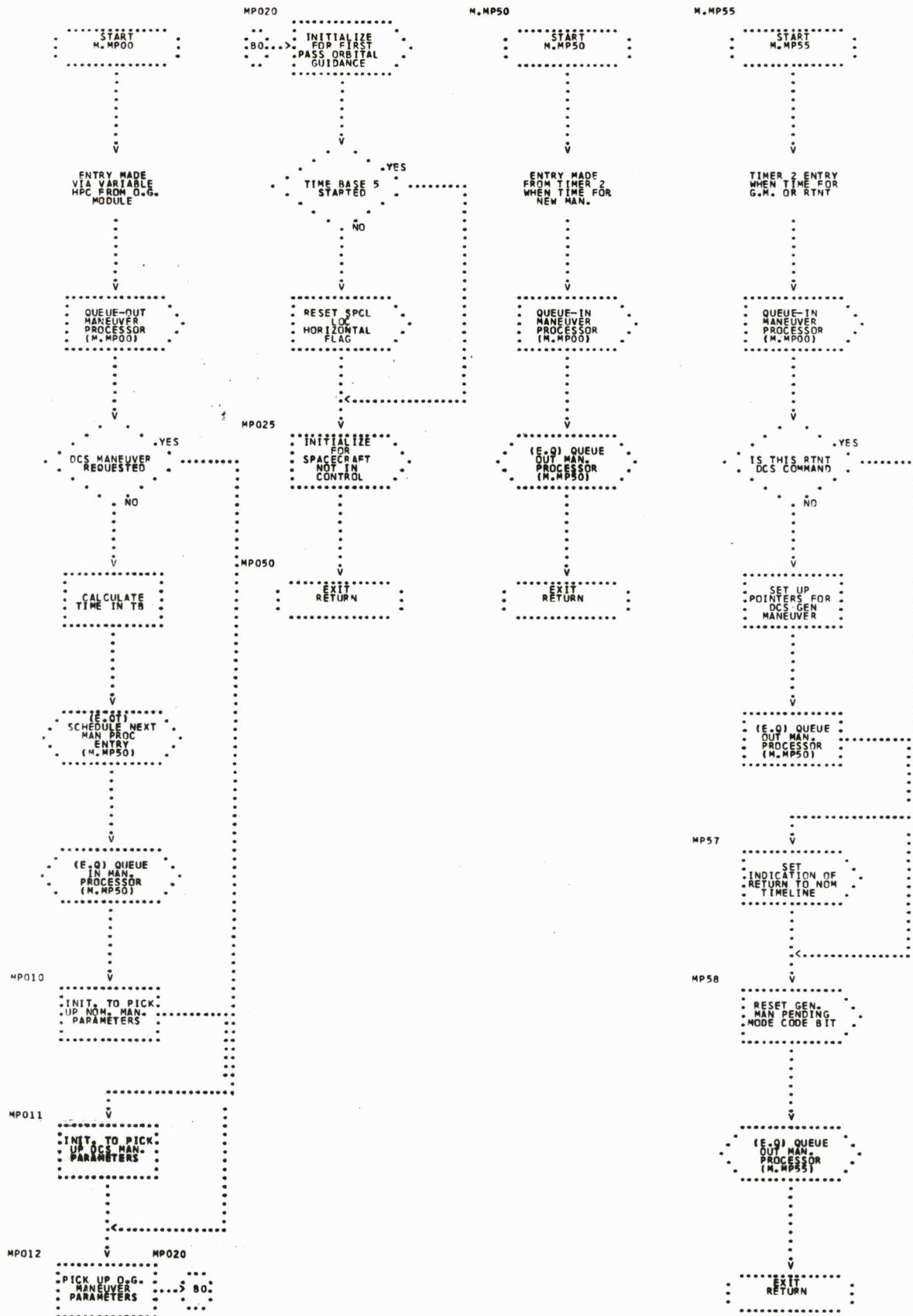
None.

Initial Conditions

None.

Notes

None.



2.37 MINOR LOOP SUPPORT (MS)

Function

The Minor Loop Support module calculates the attitude increments (delta chi's) used by the Minor Loop module and calculates the terms of the platform to body coordinate transformation used by the Minor Loop module.

Entries

M.MS00 Entered from the Non-Interrupt Sequencer once per computation cycle during Phase 1. Entered from the Phase II Control module once per second during Phase 2.

External

U.SC00 Sine-Cosine Utility
U.TR00 Real Time Telemetry Delay Utility
E.ENBL Enable interrupts
E.IHBT Inhibit interrupts

*

Input Data

D.VLM1 Maximum permitted rate of change of roll command chi per Minor Loop
D.VLM2 Maximum permitted rate of change of the yaw and pitch command chi's per Minor Loop
D.VGCX Guidance Chi X (Desired roll gimbal angle)
D.VGCY Guidance Chi Y (Desired pitch gimbal angle)
D.VGCZ Guidance Chi Z (Desired yaw gimbal angle)
D.VCCX Commanded Chi X (Rate limited roll gimbal angle)
D.VCCY Commanded Chi Y (Rate limited pitch gimbal angle)
D.VCCZ Commanded Chi Z (Rate limited yaw gimbal angle)
D.VMLR The number of minor loops per computation cycle

*
*
*
*
*
*

D.VTHX	Total roll gimbal angle	*
D.VTHY	Total pitch gimbal angle	*
D.VTHZ	Total yaw gimbal angle	*
D.V1MR	Reciprocal of the number of minor loops per pass through Minor Loop Support	
D.FDC0	Bypass setting delta chi's flag	

Output Data

D.VDCX	Delta Chi X (Roll commanded chi increment)	*
D.VDCY	Delta Chi Y (Pitch commanded chi increment)	*
D.VDCZ	Delta Chi Z (Yaw commanded chi increment)	*
D.VA1	Coefficients used to convert the inertial platform readings to body frame attitude corrections	
D.VA2		
D.VA3		
D.VA4		
D.VA5		
D.FSMC	Steering Misalignment Correction inhibit flag	*
D.VGCZ	Guidance Chi Z (Desired yaw gimbal angle)	*
D.FDC0	Bypass setting delta chi's flag	

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>	
D.VDCX	1451	Delta Chi X (Roll commanded chi increment)	*
D.VDCY	1455	Delta Chi Y (Pitch commanded chi increment)	*
D.VDCZ	1445	Delta Chi Z (Yaw commanded chi increment)	*
V.THZT	0015	Yaw gimbal angle at time of delta chi computations	*
V.THXT	0021	Roll gimbal angle at time of delta chi computations	*
V.THYT	0025	Pitch gimbal angle at time of delta chi computations	*

V.CCZT	0501	Command Chi Z at time of delta chi computations
V.CCXT	0505	Command Chi X at time of delta chi computations
V.CCYT	0511	Command Chi Y at time of delta chi computations
V.DCHI	2024	Delta Chi RTC reference

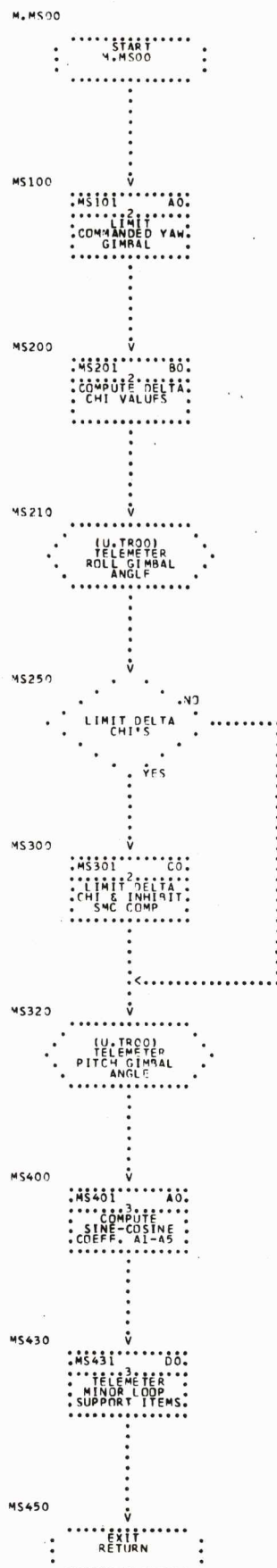
Initial Conditions

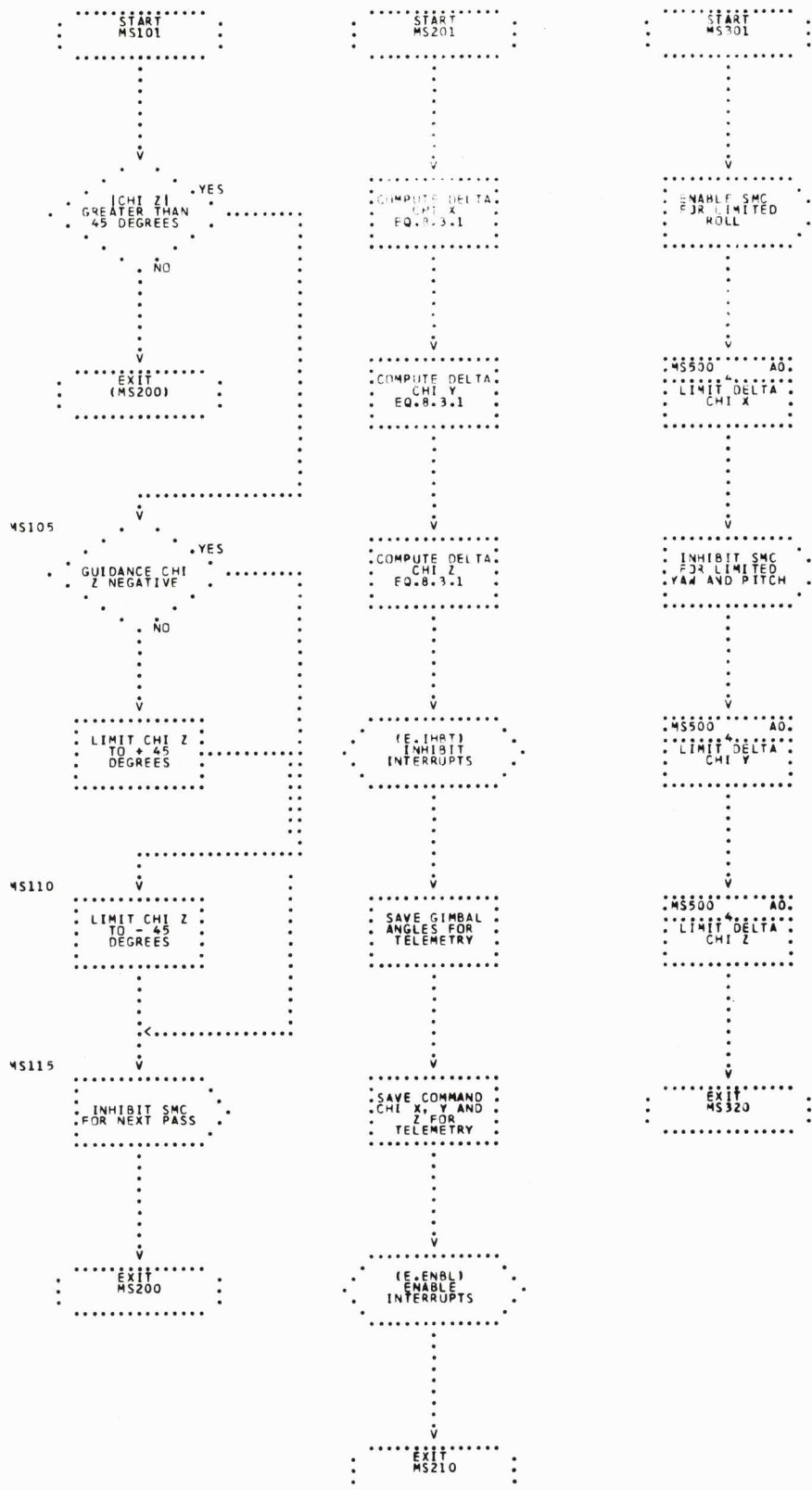
D.VMLR must be initialized to number of minor loops executed per computation cycle. D.VlMR must be initialized to reciprocal of number of minor loops executed per computation cycle. D.VLM1 and D.VLM2, rate change limits, are to be initialized to mission specifications.

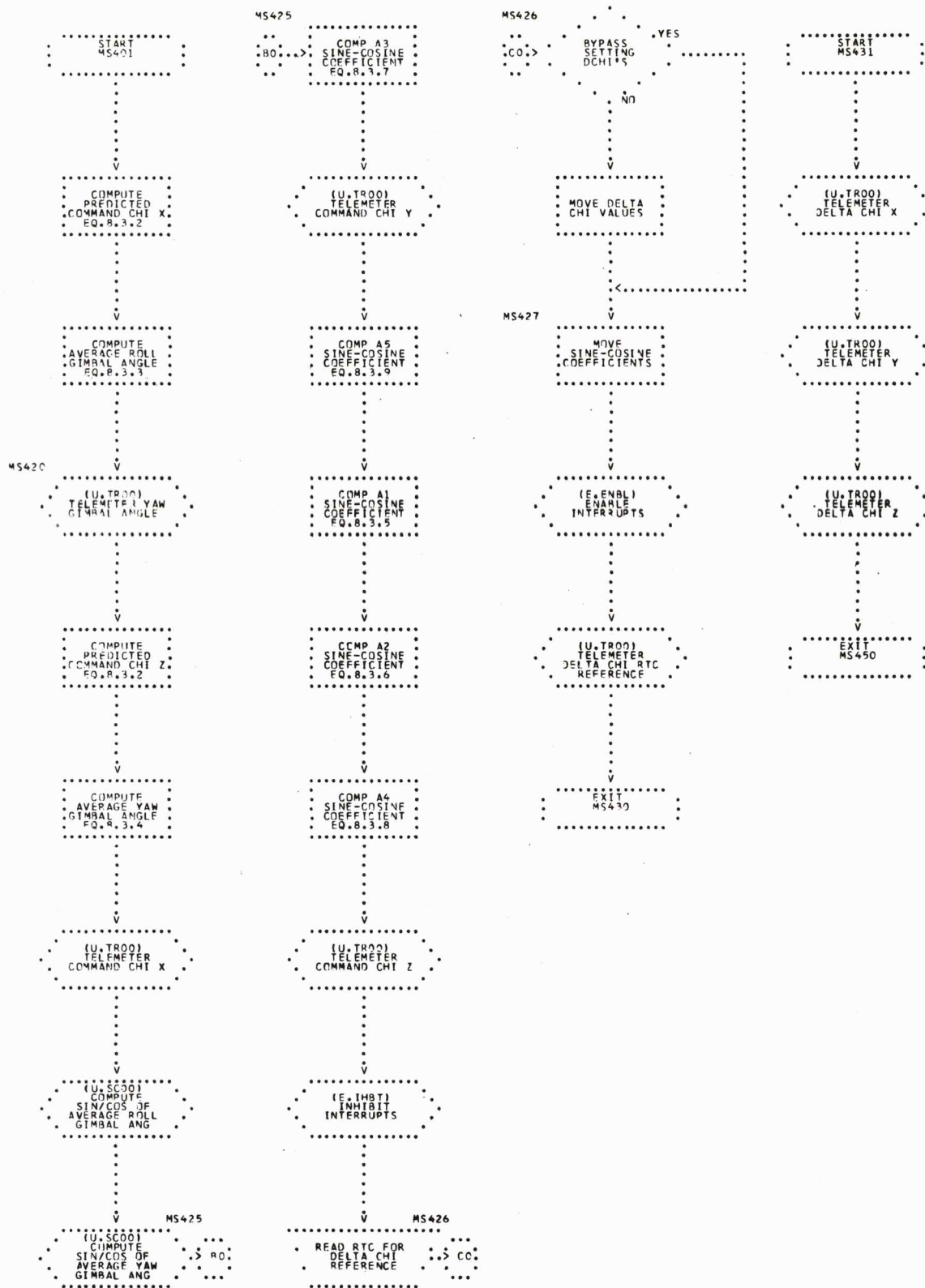
Notes

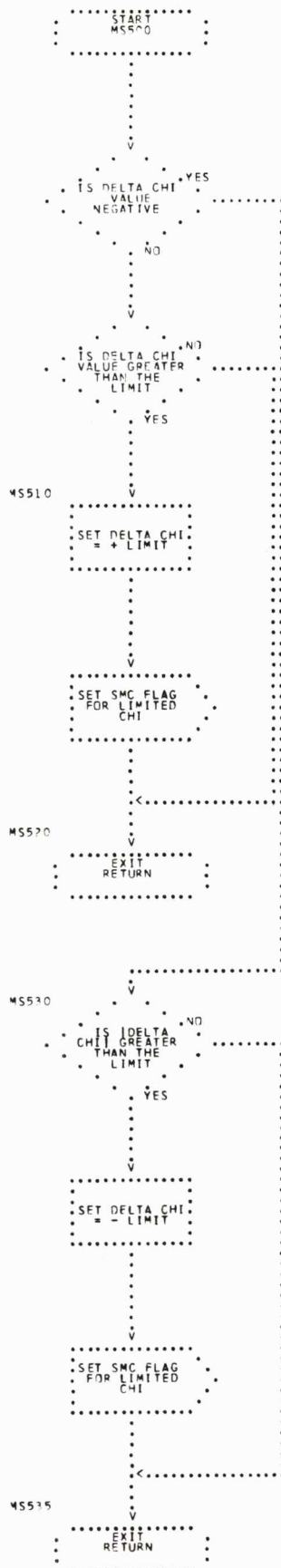
The quantities D.VMLR and D.VlMR must be altered by the program whenever the rate of the minor loop is altered or the computation cycle length is changed considerably.

The guidance chi Z will be magnitude limited to 45 degrees.









2.38 MATRIX TRANSPOSE UTILITY (MT)

Function

The function of the Matrix Transpose Utility module is to form and store the transpose of a specified matrix.

Entries

U.MT00 Entered by any Non-Interrupt Sequencer and Periodic Processor module.

External

None.

Input Data

See Notes.

Output Data

See Notes.

Status Words

None.

Telemetry

None.

Initial Conditions

None.

Notes

Upon entry, a computer word of the form STO B; CLA A must be in the accumulator, where A is the initial location of the matrix to be transposed and B is the initial location of the matrix in which the transpose is to be stored.

Both matrices are to be grouped in vector form using the VEC assembler pseudo-operation. The three vectors required to represent a 3x3 matrix must be stored using the MAT assembler pseudo-operation.

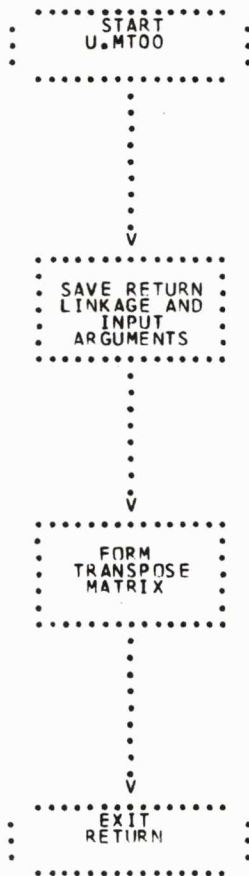
The matrices A and B may be the same. In that case, the A matrix is transposed and the transpose is then stored over A. Matrices may be grouped by either column or row vectors.

Upon exit, the transposed matrix will be stored in the matrix specified by the word on entry.

MATRIX TRANSPOSE UTILITY (MT)

1

U.MTCO



2.39 BOOST NAVIGATION (NE)

Function

The function of the Boost Navigation module is to determine the vehicle position and velocity in an earth centered inertial coordinate frame.

Entries

M.NE00 Entered from the Non-Interrupt Sequencer once per computation cycle during Phase 1.

External

M.GR00 Gravitation Acceleration
U.SR00 Square Root Utility
U.FD00 Fixed Point Divide Utility
U.TR00 Real Time Telemetry Delay Utility

Input Data

D.VDT Elapsed time between current and previous computation cycle
accelerometer reads in seconds *

D.VZDM Z component of measured velocity (m/sec)

D.VOZM Previous Z component of measured velocity

D.VXDM X component of measured velocity (m/sec)

D.VOXM Previous X component of measured velocity

D.VYDM Y component of measured velocity (m/sec)

D.VOYM Previous Y component of measured velocity

D.VGX X component of gravity acceleration

D.VGY Y component of gravity acceleration

D.VGZ Z component of gravity acceleration

D.VT3I Second phase IGM time-to-go *

F.HSPF High Speed Loop flag

V.GDVB Thrust decay velocity bias

Output Data

D.VXS	X component of radius vector, in S-system	*
D.VYS	Y component of radius vector, in S-system	*
D.VZS	Z component of radius vector, in S-system	*
D.VXDS	X component of velocity vector, in S-system	*
D.VYDS	Y component of velocity vector, in S-system	*
D.VZDS	Z component of velocity vector, in S-system	*
D.VOXM	Previous X component of measured velocity	
D.VOYM	Previous Y component of measured velocity	
D.VOZM	Previous Z component of measured velocity	
D.VVS	Total space-fixed velocity	
D.VV0	Second previous total velocity	*
D.VV1	Previous total velocity	*
D.VV2	Current total velocity	*
D.VIT1	Length of second previous computation cycle	*
D.VIT2	Length of previous computation cycle	*
V.GTVB	Time bias for thrust decay for high cutoff accuracy	

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>	
D.VZS	0034	Z component of radius vector, in S-system	*
D.VGX	0064	X component of gravity acceleration	
D.VXS	0044	X component of radius vector in S-system	*
D.VGY	0070	Y component of gravity acceleration	
D.VYS	0050	Y component of radius vector, in S-system	*
D.VGZ	0060	Z component of gravity acceleration	
D.V1OR	0054	Inverse of radius	*

D.VXDS	0114	X component of velocity vector, in S-system	*
D.VYDS	0120	Y component of velocity vector, in S-system	*
D.VZDS	0110	Z component of velocity vector, in S-system	*
D.VVS	0124	Total space-fixed velocity	
V.IGZ	1074	Z component of gravity velocity	
V.IGX	1100	X component of gravity velocity	
V.IGY	1104	Y component of gravity velocity	

Initial Conditions

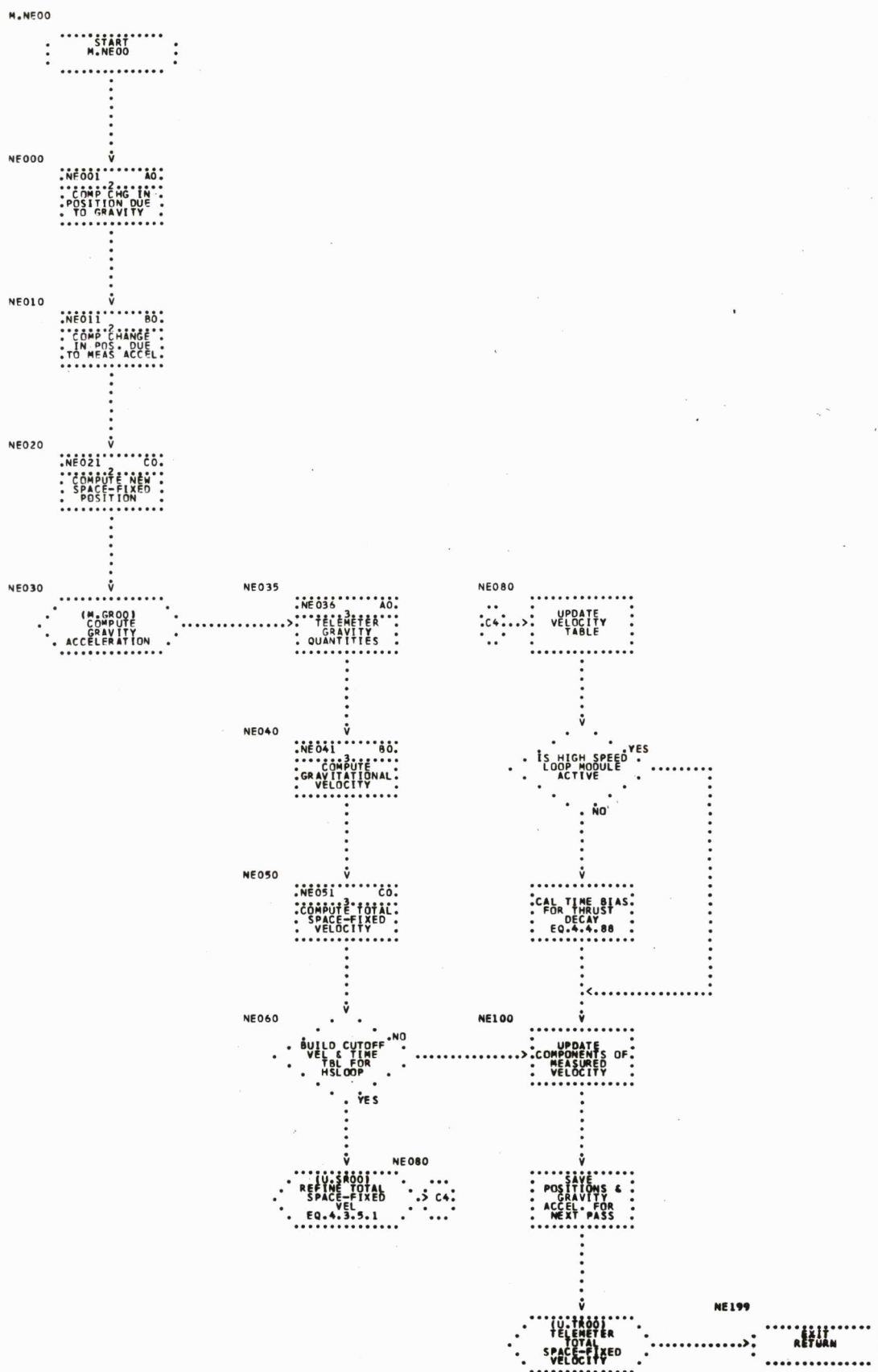
Previous components of measured velocity (D.VOZM, D.VOXM, and D.VOYM) must be set to zero at start of mission as well as the components of gravity velocity (V.IGX, V.IGY, and V.IGZ).

The components of earth rate (V.XDO, V.YDO, and V.ZDO) must be computed at mission start.

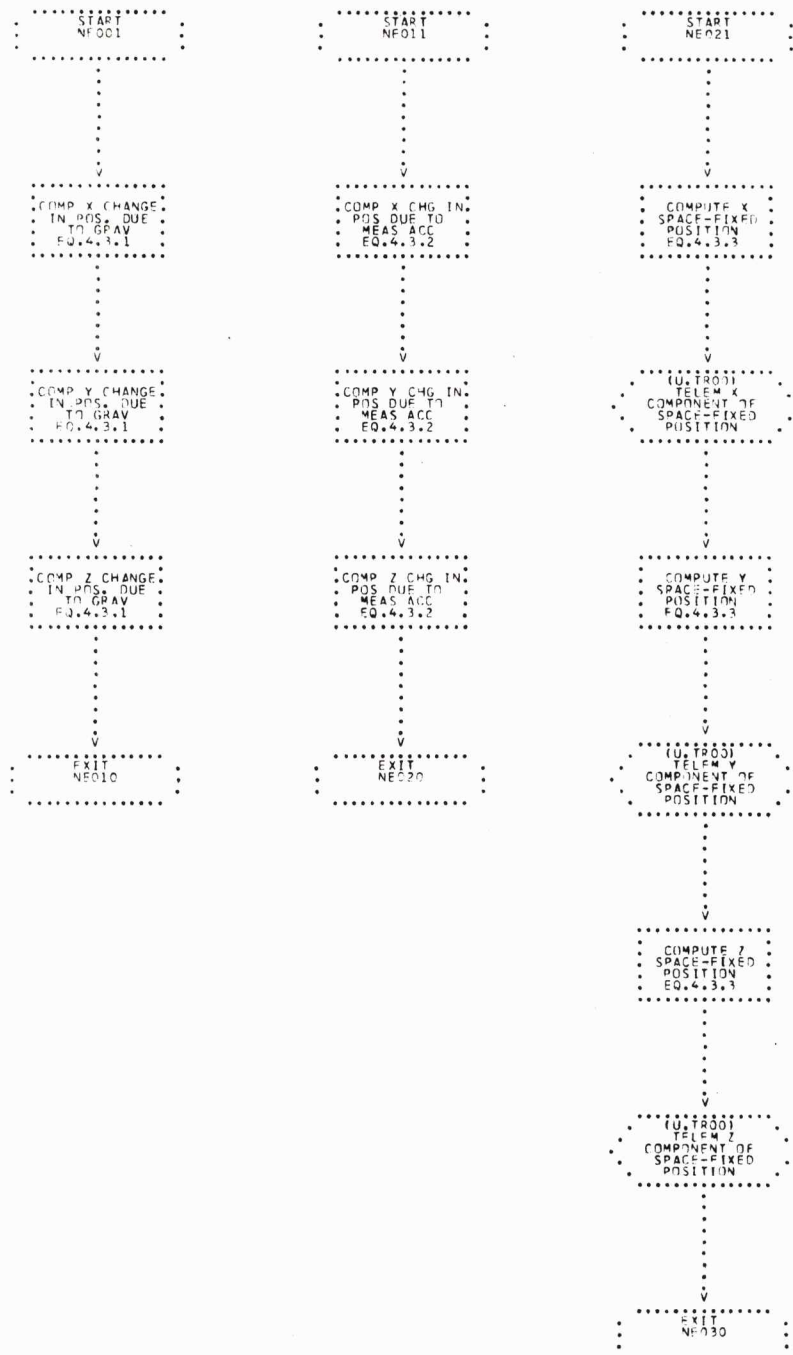
Previous components of gravity acceleration (V.OGX, V.OGY, V.OGZ) must be set to initial gravity values at start of mission. Previous components of position (V.OXS, V.OYS, and V.OZS) must be computed at start of mission as well as total velocity (D.VVS).

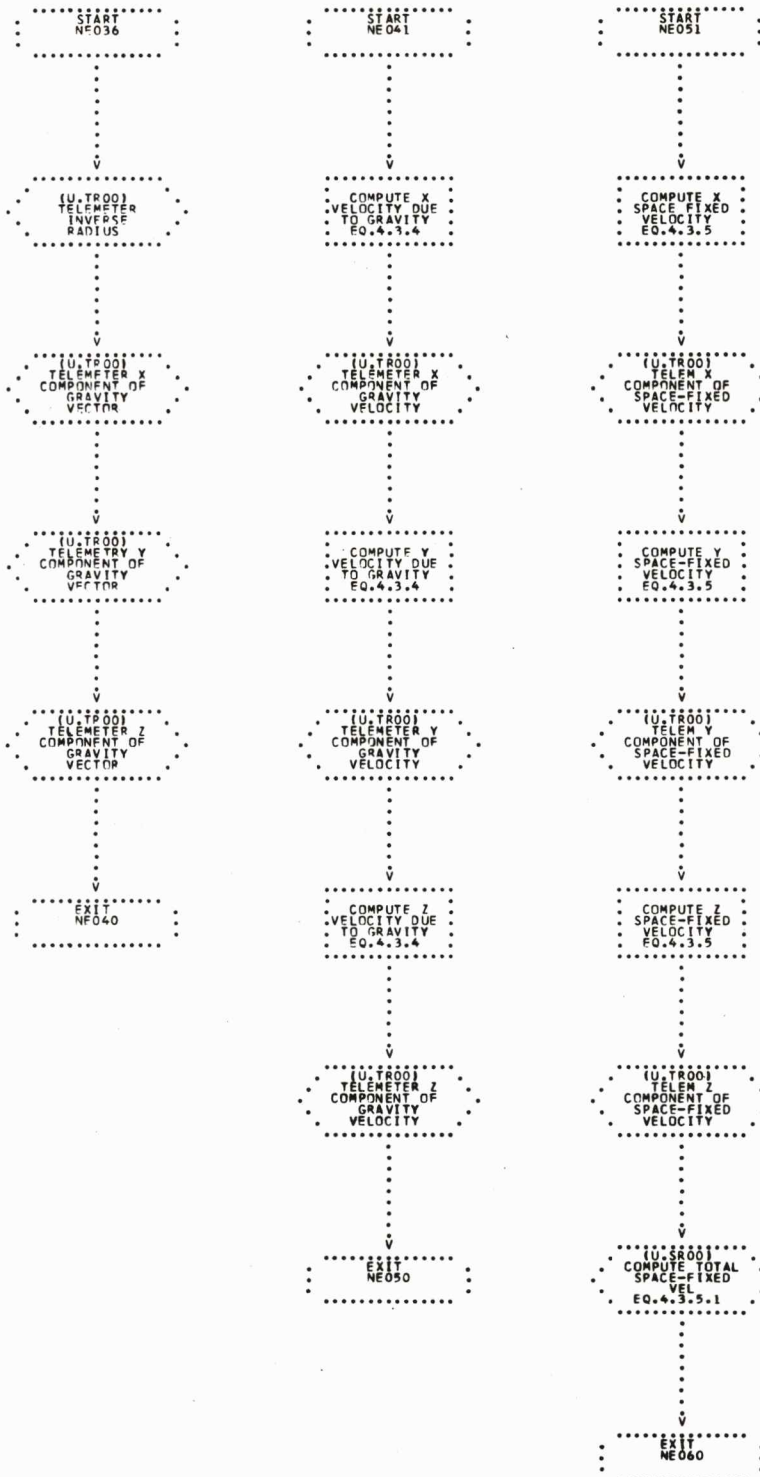
Notes

At a fixed time prior to S-IVB cutoff, a second iteration is made to the square root algorithm to obtain a more accurate total space-fixed velocity. Also at this time, velocity and time values are stored for second order extrapolation in the S-IVB Cutoff Prediction module.



BOOST NAVIGATION (ONE)





2.40 NAVIGATION EXTRAPOLATION (NI)

Function

This module will compute the extrapolated positions between the navigation computations made by the Orbit Navigation module.

Entries

M.NI00 Entered by the Phase II Control module once per second except on the eight second boundaries when the Orbit Navigation module is executed.

External

U.TR00 Real Time Telemetry Delay Utility

Input Data

D.VNTM	Time in mission from GRR of navigation extrapolation	*
D.VXDS	X component of velocity vector, in S-system	*
D.VYDS	Y component of velocity vector, in S-system	*
D.VZDS	Z component of velocity vector, in S-system	*
D.VXS	X component of radius vector, in S-system	*
D.VYS	Y component of radius vector, in S-system	*
D.VZS	Z component of radius vector, in S-system	*
V.NTMO	Time of previous navigation pass	

Output Data

D.VXS	X component of radius vector, in S-system	*
D.VYS	Y component of radius vector, in S-system	*
D.VZS	Z component of radius vector, in S-system	*
V.NTMO	Time of previous navigation pass	

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>	
V.XS	0144	X component of radius vector, in S-system	*
V.YS	0150	Y component of radius vector, in S-system	*
V.ZS	0134	Z component of radius vector, in S-system	*

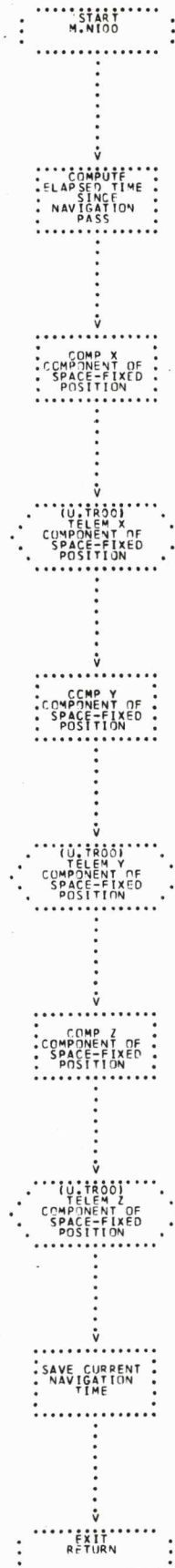
Initial Conditions

None.

Notes

None.

M.N100



2.41 NAVIGATION UPDATE IMPLEMENTATION (NU)

Function

The function of this module is to insert navigation update state vectors and reschedule the orbital timeline.

Entries

M.NU00 Entered from Timer 2 when scheduled by the DCS Processor module to process a navigation update command.

External

M.CM01 Phase II Control

E.IHBT Inhibit interrupts

E.Q Navigation Update Implementation (M.NU00)

E.QT Phase II Control (M.CM10)

E.ENBL Enable interrupts

Input Data

D.VNUT Navigation update implementation time

D.VUXS }
D.VUYS } Navigation update position vector
D.VUZS }

D.VUDX }
D.VUDY } Navigation update velocity vector
D.VUDZ }

Output Data

V.XDPR }
V.YDPR } Position extra precision words
V.ZDPR }

V.XDSP }
 V.YDSP } Velocity extra precision words
 V.ZDSP }
 D.FON Orbit navigation pass flag
 V.ONXN }
 V.ONYN } Previous positions, S-system
 V.ONZN }
 D.VXS }
 D.VYS } Positions, S-system
 D.VZS }
 V.XDN }
 V.YDN } Previous velocities, S-system
 V.ZDN }
 D.VXDS }
 D.VYDS } Velocities, S-system
 D.VZDS }
 D.VT8S Phase II Control module schedule time for navigation processing
 D.VT4S Phase II Control module schedule time for midpoint processing

Status Words

D.VMC7 Mode Code 27
 D.FILE Flight Program Status

Telemetry

None.

Initial Conditions

None.

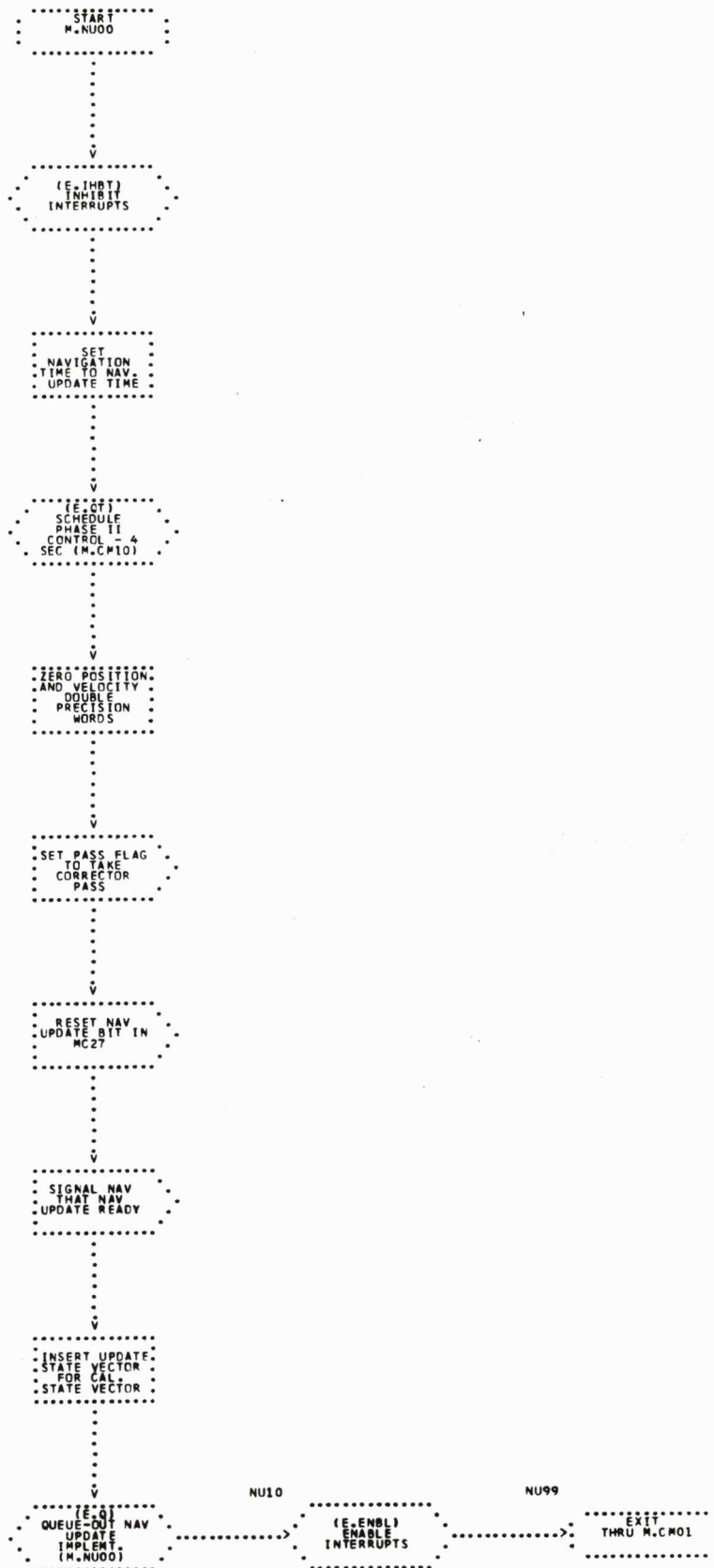
Notes

When a DCS Navigation Update is received and verified, the Navigation Update Implementation module is scheduled for the implementation time and queued-in. It queues itself out before exiting.

The Orbit Navigation module flag is set to cycle the corrector pass.

This module exits to the Phase II Control module, which will return control to Timer 2 rather than returning to this module.

M.NU00



2.42 ORBIT NAVIGATION (NZ)

Function

The function of the Orbit Navigation module is to compute position, velocity, and total acceleration relative to the S-coordinate system during the orbital phases.

Entries

M.NZ00 Entered by the Phase II Control module once every eight seconds during Phase 2.

External

M.GR00 Gravitation Acceleration
M.OA00 Orbit Accelerations
U.SR00 Square Root Utility
U.TR00 Real Time Telemetry Delay Utility

Input Data

D.FON Midpoint/Endpoint pass flag
D.FLT Flight Mode indicator
D.VDMX }
D.VDMY } Drag accelerations, S-system
D.VDMZ }
D.VGX }
D.VGY } Gravity accelerations, S-system
D.VGZ }
V.ONXN }
V.ONYN } Previous Positions Updated by the Navigation Update
V.ONZN } Implementation module
D.VGCY Guidance Chi Y (Desired pitch gimbal angle) *
D.VGCZ Guidance Chi Z (Desired yaw gimbal angle) *

D.VTHY	Total pitch gimbal angle	*
D.VTHZ	Total yaw gimbal angle	*
V.XDN	} Previous Velocities Updated by the Navigation Update Implementation module	
V.YDN		
V.ZDN		

Output Data

D.VXS	} Present navigation positions in inertial frame; to be extrapolated	
D.VYS		
D.VZS		
D.VETY	Endpoint pitch gimbal angle	
D.VETZ	Endpoint yaw gimbal angle	
D.VXDS	} Current inertial velocities	
D.VYDS		
D.VZDS		
D.VDXS	} Rescaled positions for drag	
D.VDYS		
D.VDZS		
D.VVS	Total space-fixed velocity	
D.FON	Midpoint/Endpoint pass flag	

Status Words

D.FILE Flight Program Status

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.VZS	0034	} Components of space-fixed position
D.VXS	0044	
D.VYS	0050	

D.VZDS 0110	}	Components of space-fixed velocity
D.VXDS 0114		
D.VYDS 0120		
D.VVS 0124		Total space-fixed velocity
D.V10R 0054		Inverse of radius
D.VGX 0064	}	Components of gravity vector
D.VGY 0070		
D.VGZ 0060		
D.VDMX 0524	}	Components of drag vector
D.VDMY 0530		
D.VDMZ 0520		

*

Initial Conditions

The following items are to be initialized at the start of orbit phases:

V.ONXN	}	Previous navigation positions in inertial frame (Set to previous boost inertial positions)
V.ONYN		
V.ONZN		
V.XDN	}	Previous inertial velocities (Set to previous boost inertial velocities)
V.YDN		
V.ZDN		
V.XDDN	}	Previous accelerations (Set to previous gravity accelerations)
V.YDDN		
V.ZDDN		
V.XDPR	}	Least significant part of extra precision of inertial positions (Set to zero)
V.YDPR		
V.ZDPR		
V.XDSP	}	Least significant part of extra precision of inertial velocities (Set to zero)
V.YDSP		
V.ZDSP		
D.FON Midpoint/Endpoint pass flag set to -2		

Notes

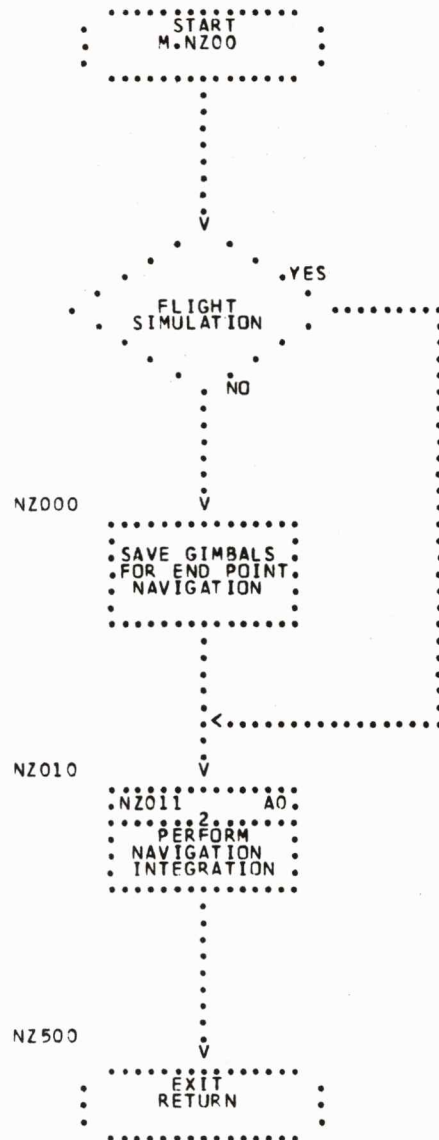
When running in the Flight Simulation Mode, the guidance chi's are used for the midpoint and endpoint navigation gimbals. In the Flight Mode, the actual gimbals are used.

If a navigation update is received through the Digital Command System, the update will be implementated through the Navigation Update Implementation module and time for the next orbit navigation will be rescheduled.

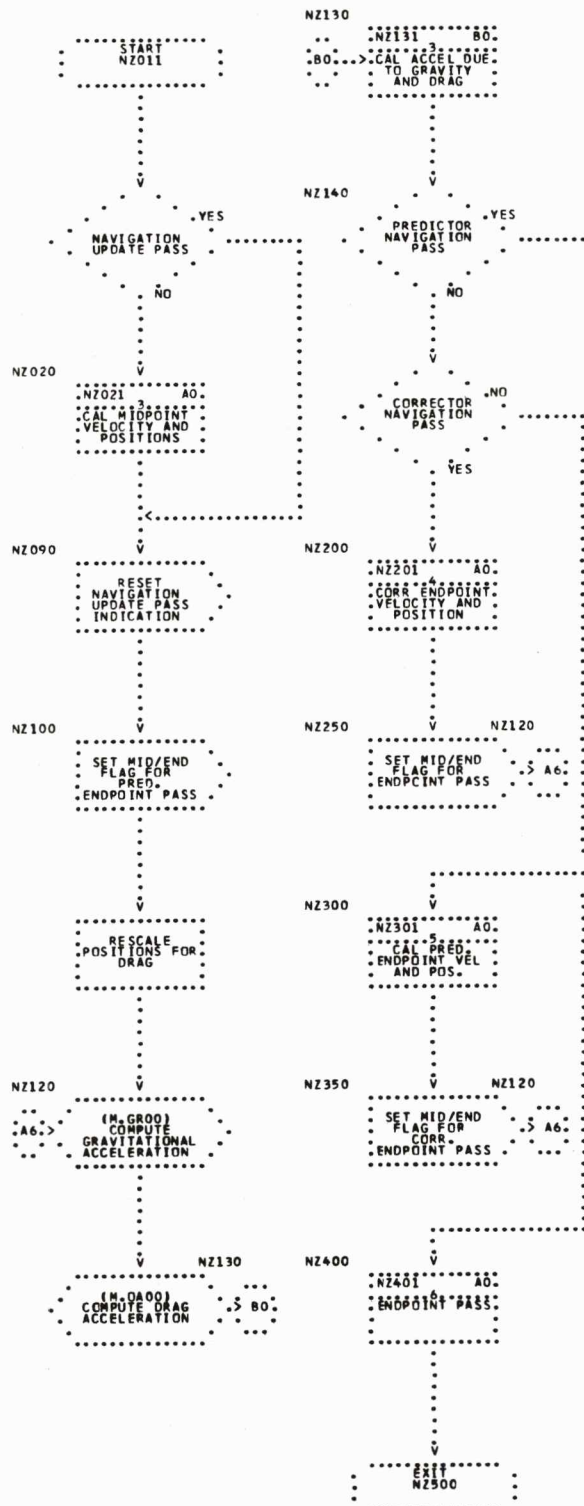
ORBIT NAVIGATION (NZ)

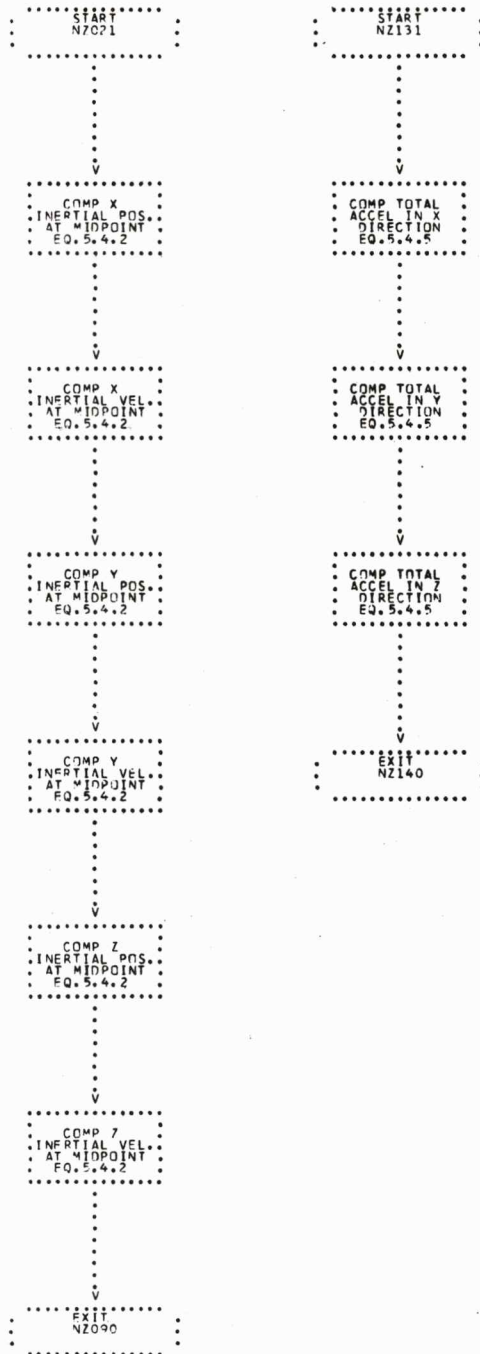
1

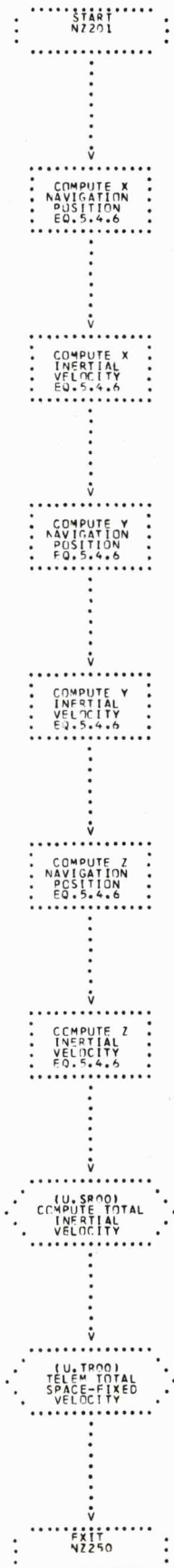
M.NZ00

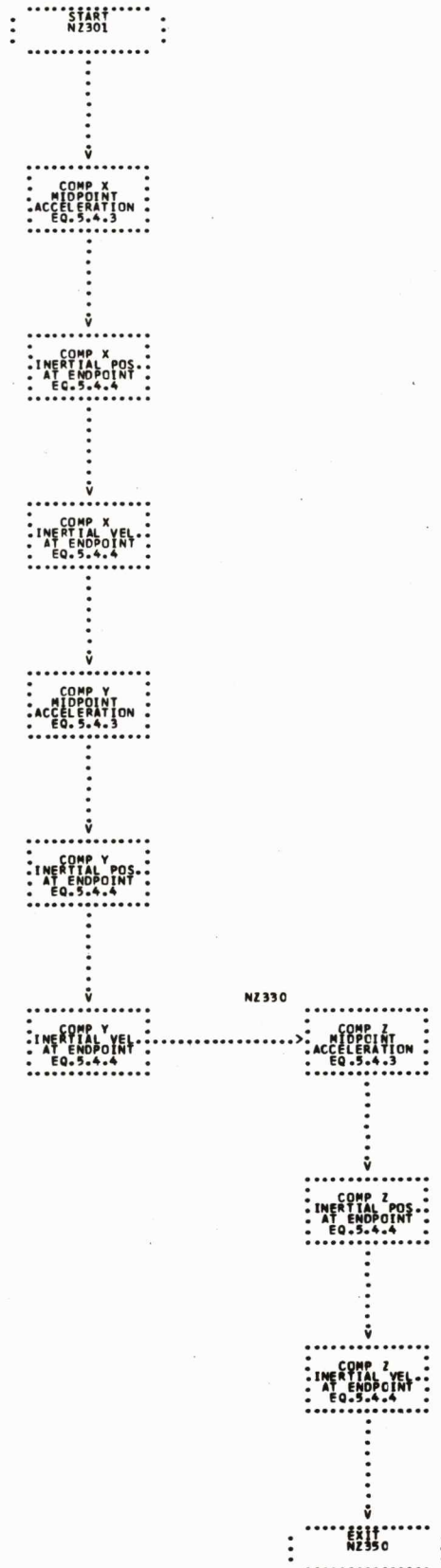


ORBIT NAVIGATION (NZ)

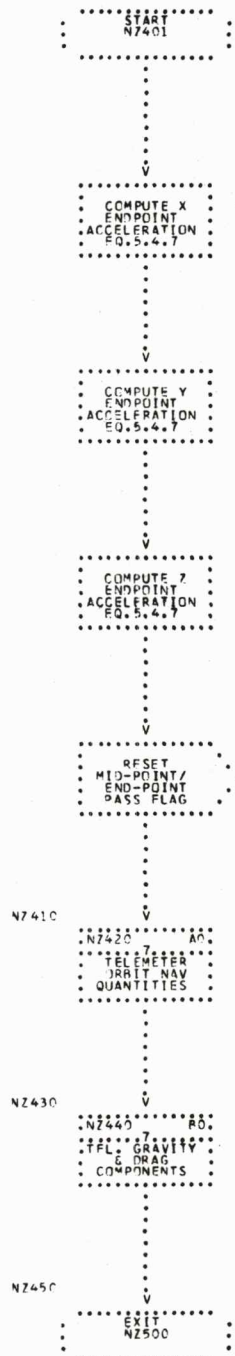


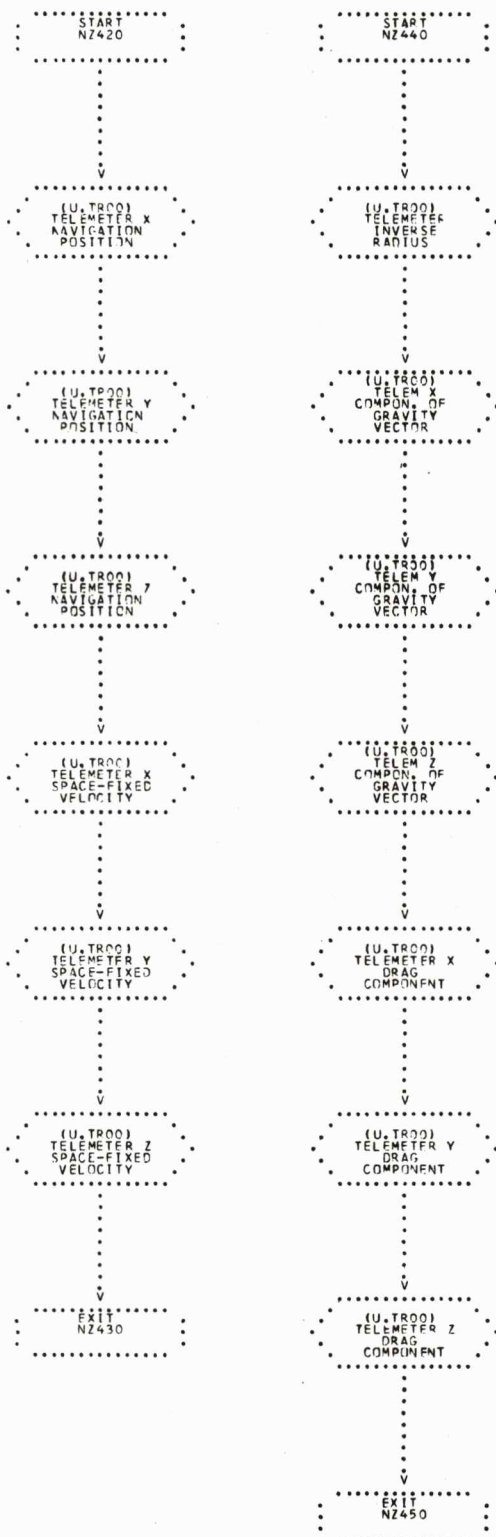






ORBIT NAVIGATION (NZ)





2.43 ORBIT ACCELERATIONS (OA)

Function

The function of the Orbit Accelerations module is to calculate the total drag acceleration of the vehicle due to atmospheric conditions.

Entries

M.OA00 Entered by Orbit Navigation during Phase 2.

External

U.FD00 Fixed Point Divide Utility
U.SC00 Sine-Cosine Utility
U.SR00 Square Root Utility
U.TR00 Real Time Telemetry Delay Utility
U.VC00 Vector Cross Product Utility
U.MM00 Matrix Multiply Utility

Input Data

D.VDXS	X coordinate of position vector, S-system	
D.VDYS	Y coordinate of position vector, S-system	
D.VDZS	Z coordinate of position vector, S-system	
D.VRR2	Square of radius	
D.V1OR	Inverse of radius	
D.VXDS	X component of velocity vector, in S-system	*
D.VYDS	Y component of velocity vector, in S-system	*
D.VZDS	Z component of velocity vector, in S-system	*
D.VMTY	Midpoint pitch gimbal angle for drag integration	*
D.VMTZ	Midpoint yaw gimbal angle for drag integration	*
D.VETY	Endpoint pitch gimbal for drag integration	
D.VETZ	Endpoint yaw gimbal for drag integration	
D.FON	Midpoint/Endpoint pass flag	

Output Data

D.VDMX X component of drag acceleration, S-system
D.VDMY Y component of drag acceleration, S-system
D.VDMZ Z component of drag acceleration, S-system
D.VXG X coordinate of position vector, G-system
D.VYG Y coordinate of position vector, G-system
D.VZG Z coordinate of position vector, G-system

Status Words

None

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.ALT	0534	Vehicle altitude

Initial Conditions

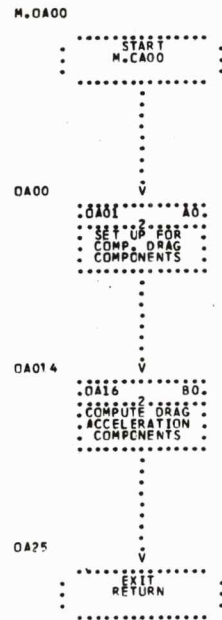
The components of earth's rotation rate (V.OMX, V.OMY, V.OMZ) about the inertial X, Y, Z Axes must be initialized at mission start.

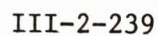
Notes

The Midpoint/Endpoint flag (D.FON) is set to proper condition by the Orbit Navigation module.

Components of the time varying matrix [MBS] which transforms body coordinates into space-fixed coordinates are stored by rows in X, Y, Z order.

ORBIT ACCELERATIONS (OA)







2.44 DIGITAL OUTPUT MULTIPLEXER (DOM) TELEMETRY (OD)

Function

The function of the DOM Telemetry module is to insure certain LVDA registers will be placed on the DOM.

Entries

M.OD00 Entered by the ETC/BTC module.

External

E.ENBL Enable interrupts

E.IHBT Inhibit interrupts

U.TR00 Real Time Telemetry Delay Utility

Input Data

D.VLDB Count of times switched to ladder converter B

Output Data

D.VLDB Count of times switched to ladder converter B

Status Words

D.VEMR Error Monitor Register

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.VAX1	1505	X accelerometer
D.VAY1	1511	Y accelerometer

D.VAZ1	1501	Z accelerometer
D.VRT1	1515	Real Time Clock

Initial Conditions

None.

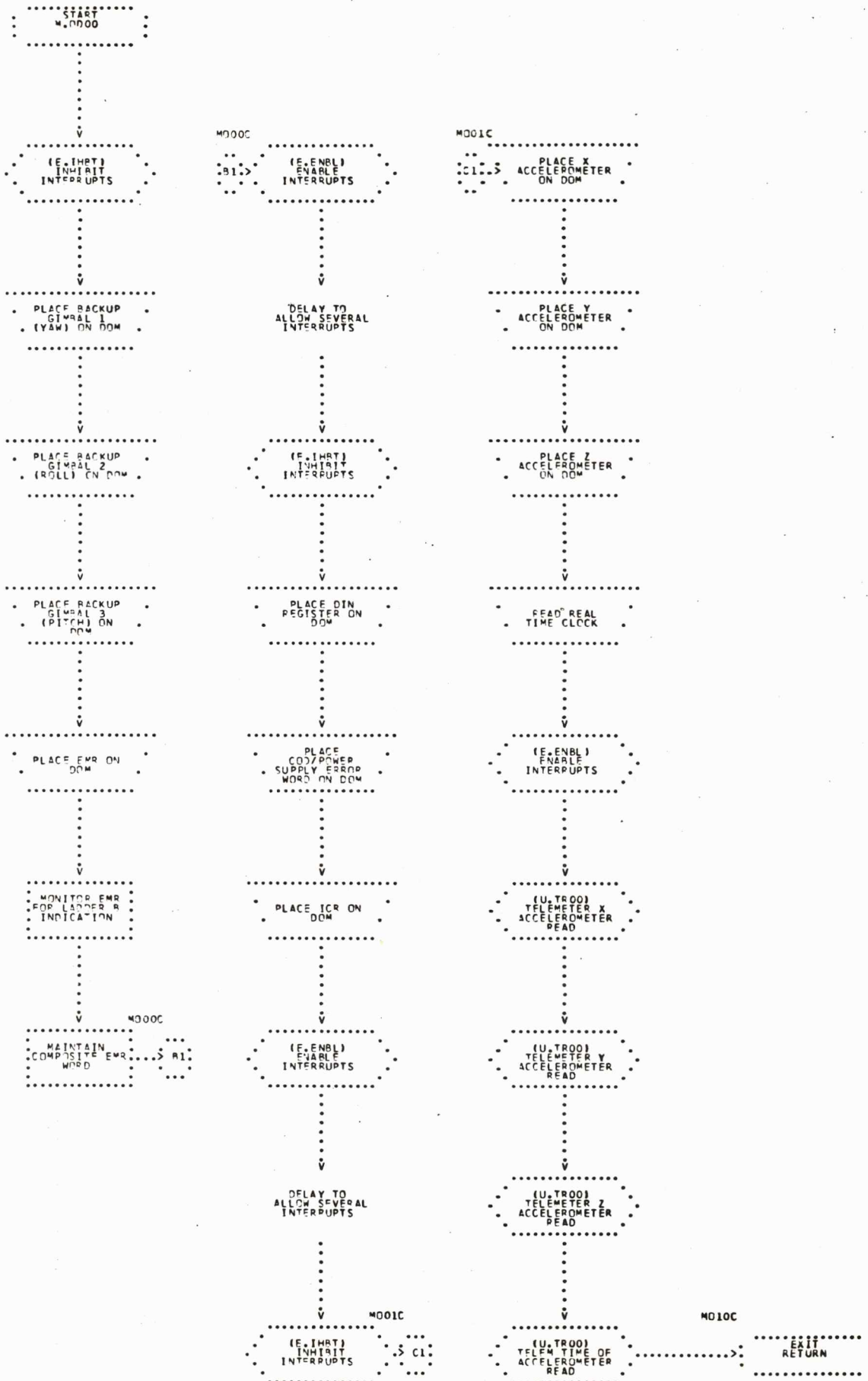
Notes

The following items are guaranteed a place on the DOM:

- X accelerometer
- Y accelerometer
- Z accelerometer
- X Backup Gimbal (Roll)
- Y Backup Gimbal (Pitch)
- Z Backup Gimbal (Yaw)
- Discrete Input Register (DI)
- Internal Control Register (ICR)
- Error Monitor Register
- COD Power Supply Word

The composite EMR word is maintained and the EMR bit indicating that ladder converter B was selected is monitored.

M.0000



2.45 ORBITAL GUIDANCE (OG)

Function

The function of the Orbital Guidance routine is to compute the desired vehicle attitude during periods when no propulsion thrust is available and attitude control is provided by the APS system.

Entries

M.OG00 Phase 1 via Non-Interrupt Sequencer once per computation cycle *
(when queued-in). Entered once per second during Phase 2 via *
once per second Timer 2 control module.

External

M.MP00 Orbital Guidance Maneuver Processor
U.FD00 Fixed Point Divide Utility
U.MM00 Matrix Multiply Utility
U.SR00 Square Root Utility
M.CC00 Chi Computations
M.CC10 Chi Telemetry
U.TR00 Real Time Telemetry Delay Utility
U.FR00 Attitude Freeze Utility
U.AT00 Arctangent Utility
U.SC00 Sine-Cosine Utility

Input Data

F.SCIC First pass flag for Spacecraft Control return *
G.MPCK Variable HOP to process or bypass the Maneuver Processor
D.VXS X component of radius vector, in S-system *
D.VYS Y component of radius vector, in S-system *
D.VZS Z component of radius vector, in S-system *

D.VXDS	X component of velocity vector, in S-system	*
D.VYDS	Y component of velocity vector, in S-system	*
D.VZDS	Z component of velocity vector, in S-system	*
F.SCAH	Flag to indicate S/C returned control	
F.MANT	Maneuver type flag	*
V.RAM	Inertial Roll angle in pirads	*
V.SPM	Inertial Pitch angle in pirads or sine of pitch delta from local horizontal	
V.SYW	Inertial Yaw angle in pirads or sine of yaw delta from local horizontal	
V.CPM	Cosine of pitch delta from local horizontal	
V.CYW	Cosine of yaw delta from local horizontal	
D.VGCX	Guidance Chi X (Desired roll gimbal angle)	*
D.VGCY	Guidance Chi Y (Desired pitch gimbal angle)	*
D.VGCZ	Guidance Chi Z (Desired yaw gimbal angle)	*
D.VTHX	Total roll gimbal angle	*
D.VTHY	Total pitch gimbal angle	*
D.VTHZ	Total yaw gimbal angle	*
D.VAP1 thru D.VAP9	Elements of the matrix rotating from S-system to 4-system	
D.FLT		
F.SLR		

Output Data

D.VGCX	Guidance Chi X (Desired roll gimbal angle)	*
D.VGCY	Guidance Chi Y (Desired pitch gimbal angle)	*
D.VGCZ	Guidance Chi Z (Desired yaw gimbal angle)	*
D.VTP1	Cosine Chi Y in 4-system	*
D.VTP2	Sine Chi Z in 4-system	*
D.VTP3	Sine Chi Y in 4-system	*

D.VTP4	Cosine Chi Z in 4-system	*
D.VOGR	Radius in the Target Plane	
V.XSP	X component of desired unit thrust vector in S-system	*
V.YSP	Y component of desired unit thrust vector in S-system	*
V.ZSP	Z component of desired unit thrust vector in S-system	*

Status Words

D.FILE Flight Program Status
D.VMC7 Mode Code 27

Telemetry

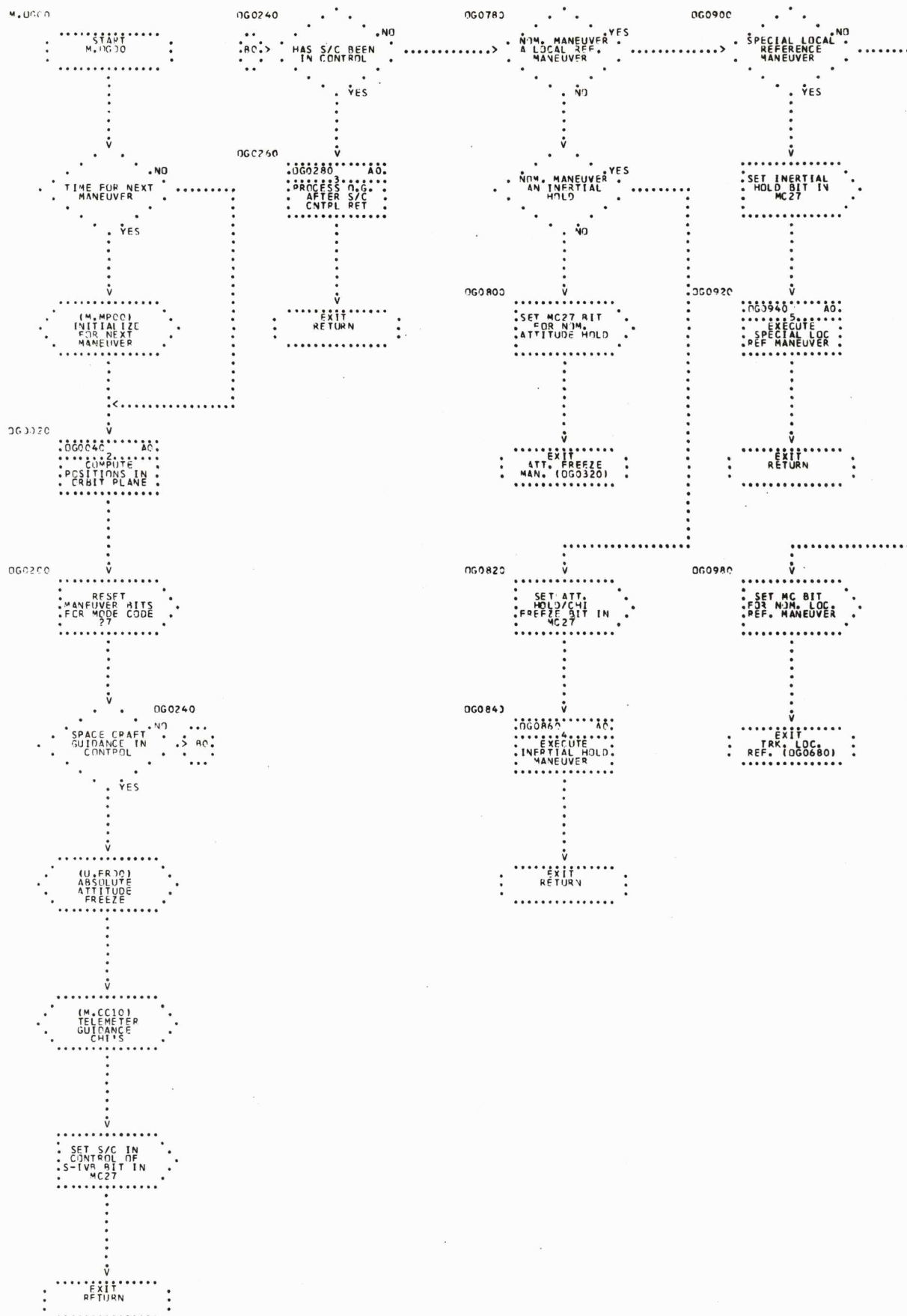
<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.TLZ4	0434	Z position in target plane
V.TLX4	0444	X position in target plane

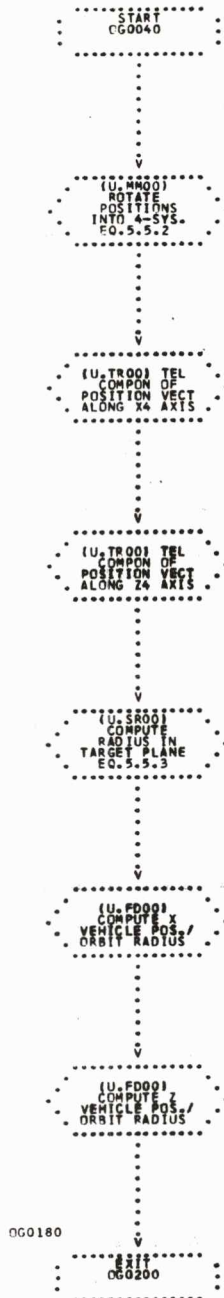
Initial Conditions

G.MPCK Must be set to go to Maneuver Processor

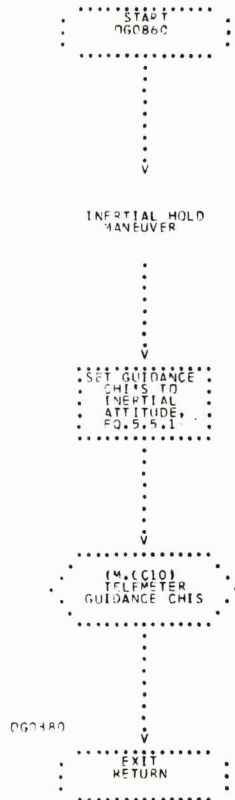
Notes

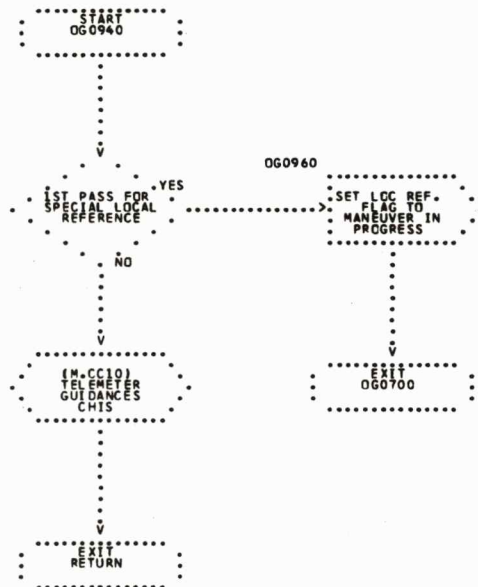
To initiate special maneuvers, the DCS Processor module modifies the variable HOP constant G.MPCK to initialize for execution of the Special Maneuver.

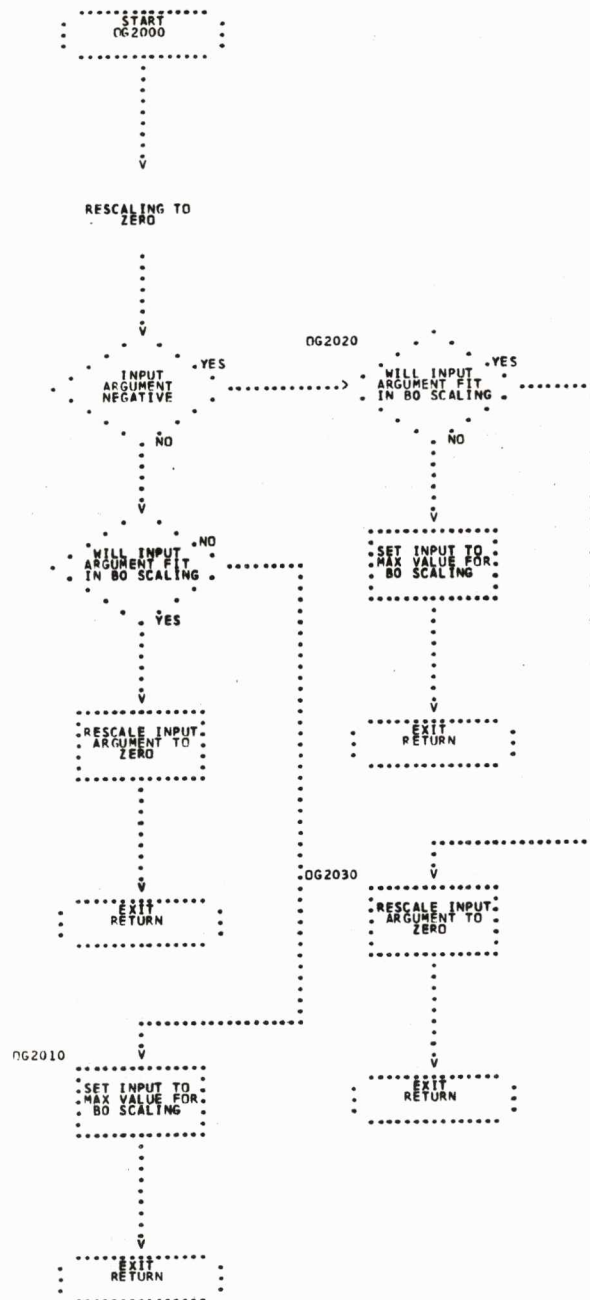




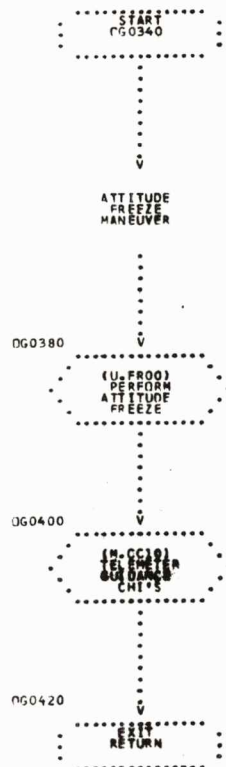
ORBIT GUIDANCE (OG)

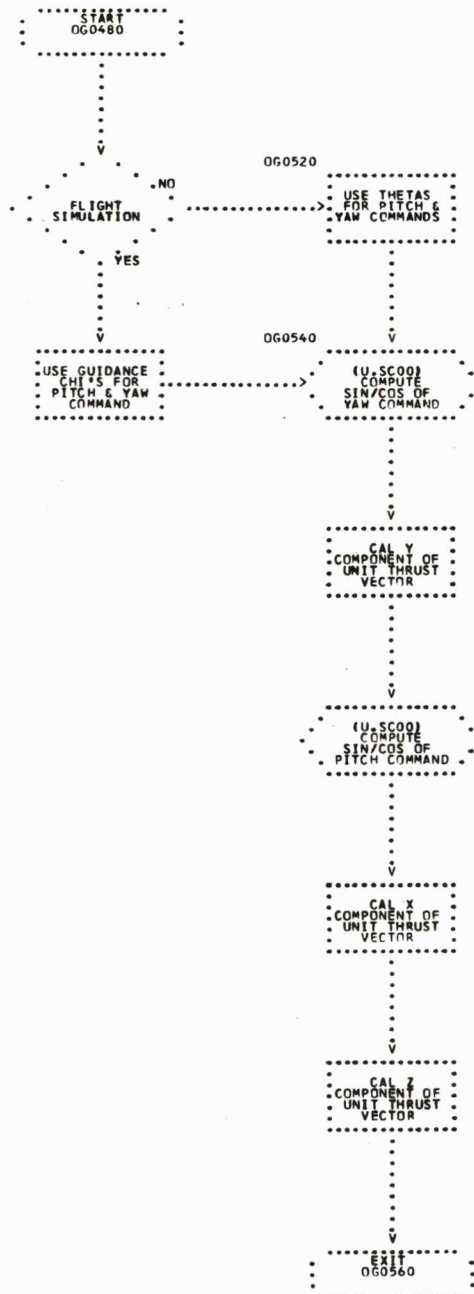




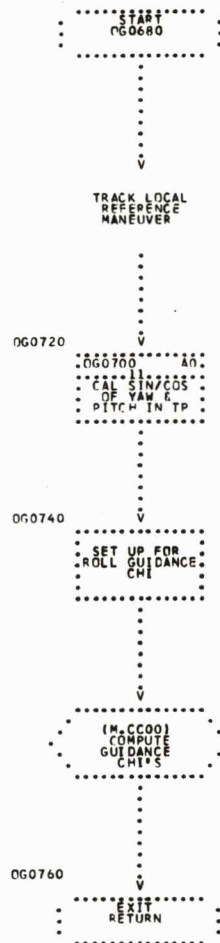


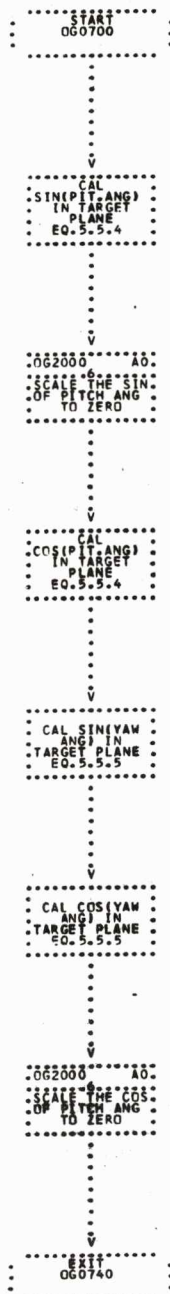
(THIS PAGE HAS BEEN LEFT BLANK INTENTIONALLY)











2.45.1 ORBIT VELOCITY CALCULATIONS (OV)

Function

The Orbit Velocity Calculations Module reads the platform accelerometers and accumulates the velocity measured by these readings.

Entries

M.OV00 Entered from the Phase II Control Module once per second.

External

E.IHBT Inhibit Interrupts

E.ENBL Enable Interrupts

U.TR00 Real Time Telemetry Delay Utility

U.SR00 Square Root Utility

E.Q Alternate Sequence (M.AS00)

U.SSRQ Request Safing Switch Selector Sequence (M.SS00)

Input Data

D.VACX X accelerometer reading

D.VACY Y accelerometer reading

D.VACZ Z accelerometer reading

D.VXDM X component of measured velocity

*

D.VYDM Y component of measured velocity

*

D.VZDM Z component of measured velocity

*

D.ELVR Delta velocity check allow flag

Output Data

D.VXDM X component of measured velocity
D.VYDM Y component of measured velocity
D.VZDM Z component of measured velocity
D.VACX X accelerometer reading
D.VACY Y accelerometer reading
D.VACZ Z accelerometer reading
D.ELVR Delta velocity check allow flag

*
*
*

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.VXDM	1024	X component of measured velocity
D.VYDM	1030	Y component of measured velocity
D.VZDM	1020	Z component of measured velocity

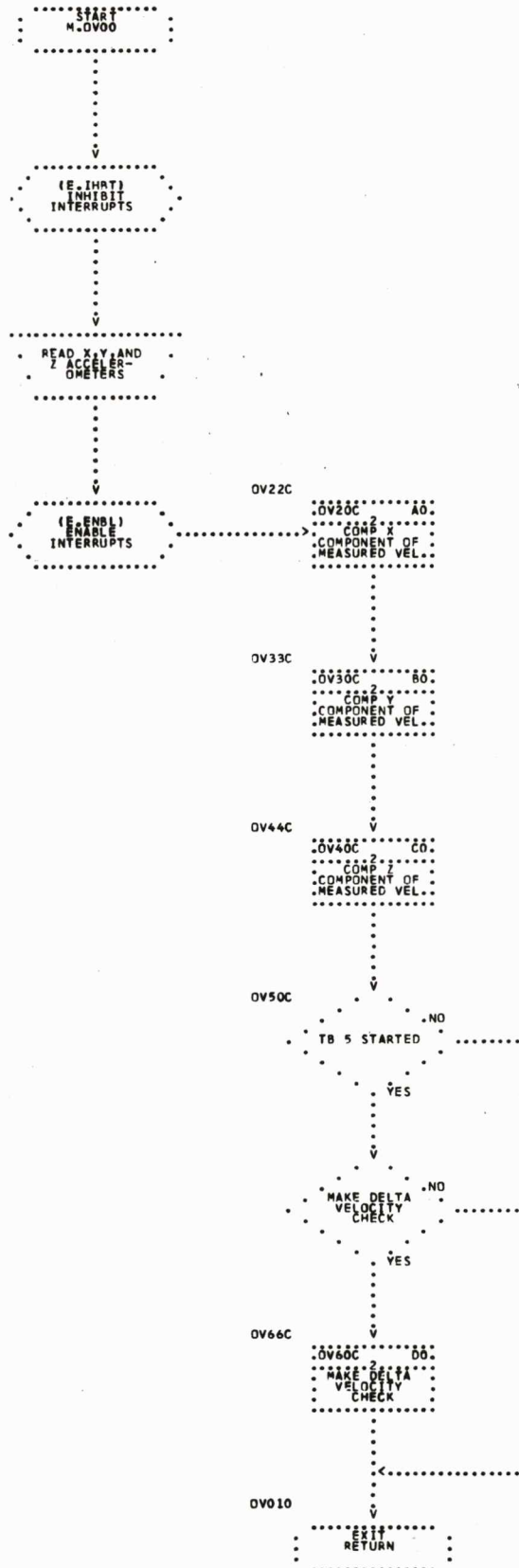
Initial Conditions

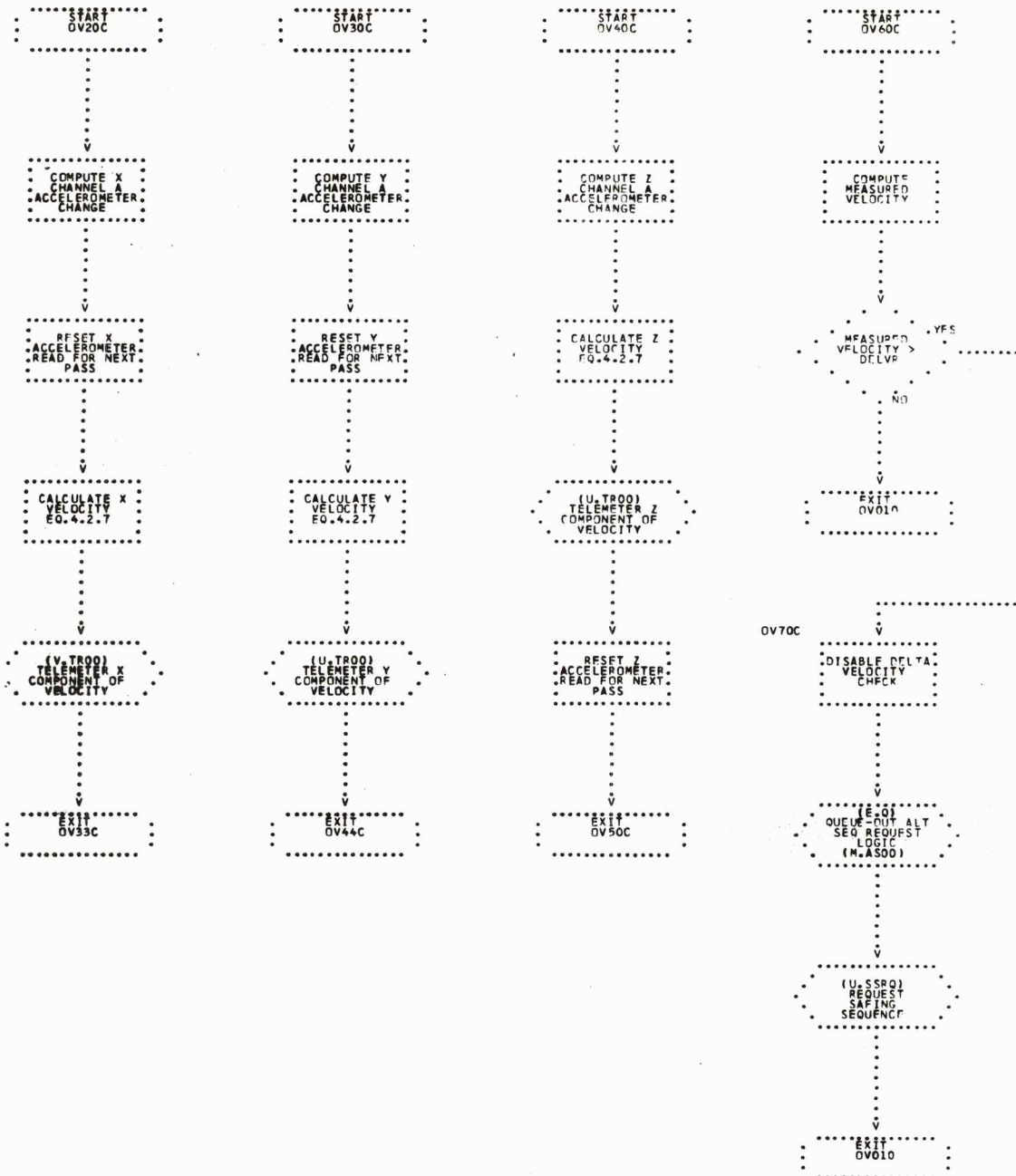
None.

Notes

The three components of the velocity vector (D.VXDM, D.VYDM and D.VZDM) are accumulated for the "A" counter only.

M.OV00





2.46 PHASE ACTIVATOR (PA)

Function

The function of this module is to terminate the present phase and start the specified phase.

Entries

M.PA00 Entered from the Non-Interrupt Sequencer when queued-in during Phase 1. Entered from the Phase II Control module in Phase 2.
Entered from the Simultaneous Memory Error Recovery module if a TLC occurred during phase change.

External

E.GP03 Control Program Phase Resequence

Input Data

D.VP Phase to be started

Output Data

D.FPHC Phase change in progress flag

Status Words

None.

Telemetry

None.

Initial Conditions

None.

Notes

This module does not return control but exits to the Control Program.

PHASE ACTIVATOR (PA)

1

M. PA00



2.47 ONE SECOND DATA COMPRESSION (PC01)

Function

This module compresses and stores Mode Code 24 if a change has occurred, accumulates a sixty second EMR word, accumulates an eight second EMR word, and telemeters a one second composite EMR word and Mode Code 24.

Entries

M.PC01 Entered by the Phase II Control module once per second.

External

M.PC80 Data Compression Store Utility
U.TR00 Real Time Telemetry Delay Utility
E.IHBT Inhibit interrupts
E.ENBL Enable interrupts

Input Data

D.VDCT Time from GRR (in seconds positioned for data compression)
F.EMR8 Dark period EMR accumulation flag

*

Output Data

D.VER0 Accumulated sixty second EMR word
D.VER8 Accumulated eight second composite EMR word
Data Compression Table 4 entries

Status Words

D.VMC4 Mode Code 24
D.VEMR Error Monitor Register

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
T.MC24 0415		Mode Code 24
D.VTEA 0435		One second EMR composite

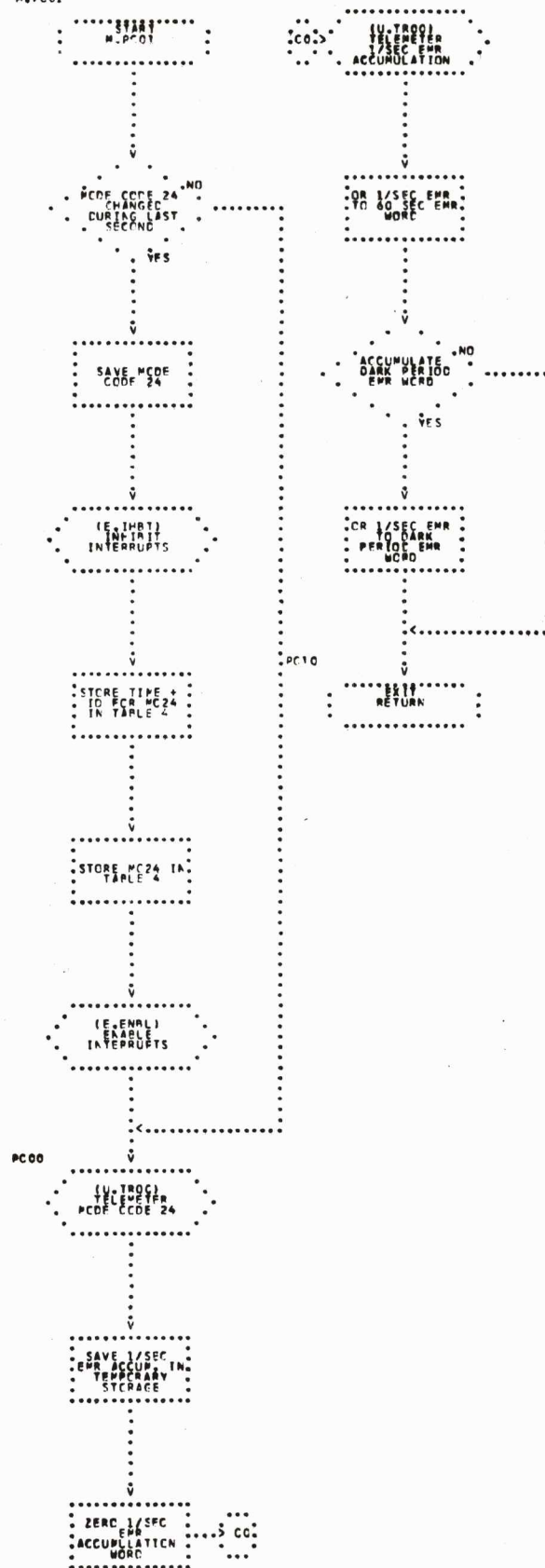
Initial Conditions

The eight second composite EMR word, D.VER8, must be initially set to zero.

Notes

Mode Code 24, time, and proper identification will be stored in Data Compression Table 4.

M.PC01



2.48 FIFTY SECOND DATA COMPRESSION (PC50)

Function

This module compresses and stores accelerometer readings every 50 seconds; also, the Discrete Input Register is sampled once every 50 seconds and stored if a change has occurred.

*
*
*

Entries

M.PC50 Entered from the Periodic Processor every fifty seconds.

External

M.PC80 Data Compression Store Utility
M.PC71 Data Compression Position Utility
E.IHBT Inhibit interrupts
E.ENBL Enable interrupts

Input Data

D.VDCT Time from GRR (in seconds positioned for data compression)
D.FLT Flight mode indicator

Output Data

Data Compression Table 3 entries
Data Compression Table 4 entries

Status Words

None.

Telemetry

None.

Initial Conditions

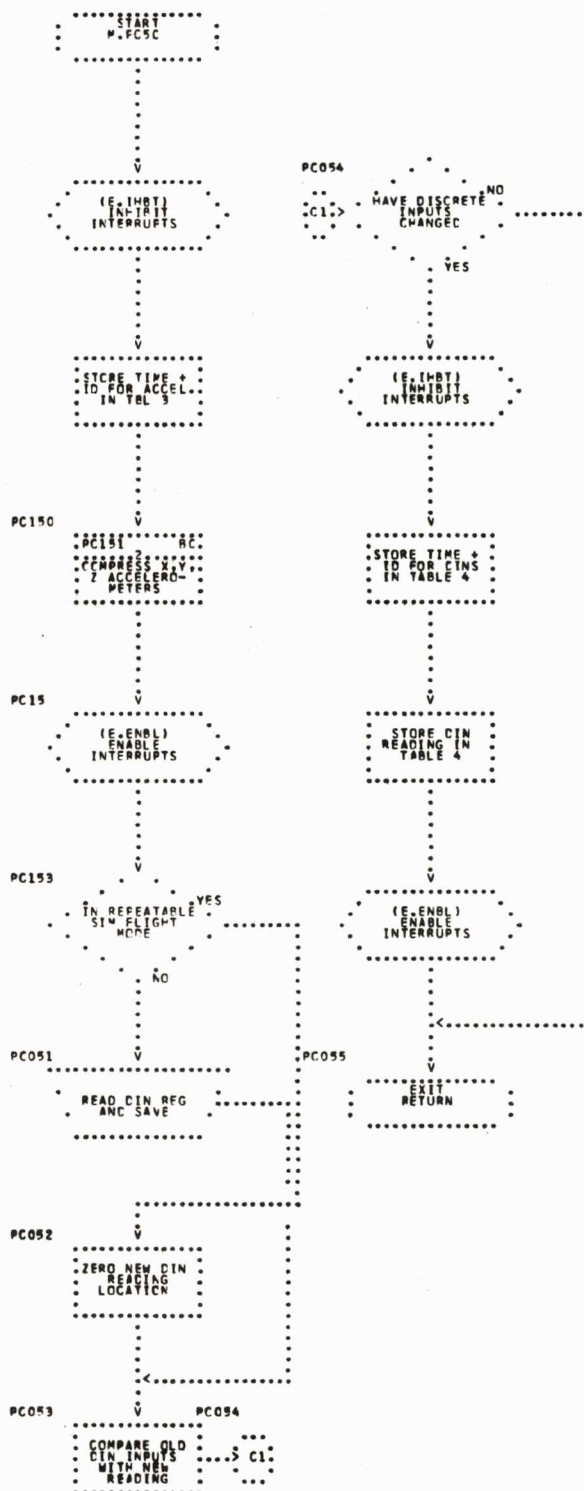
D.VDIO must be initially set to zero.

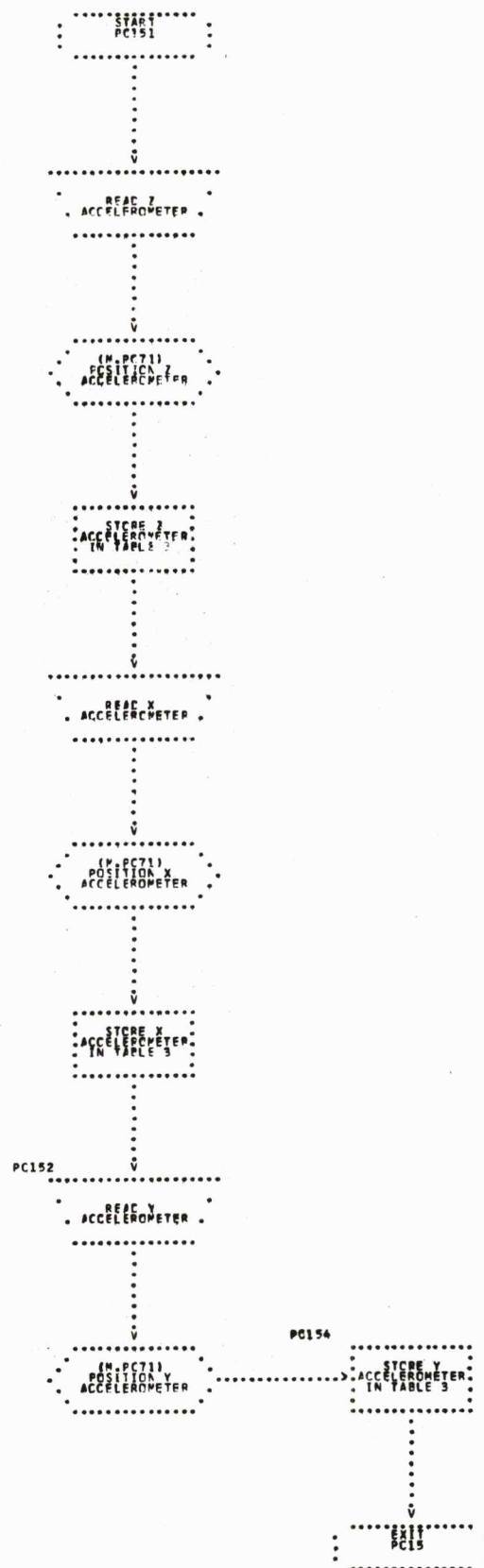
Notes

Accelerometer readings, time, and proper tags will be stored in Data Compression Table 3. Discrete Inputs, time, and proper tags will be stored in Data Compression Table 4.

When running in the Flight Mode or in the non-repeatable Flight Simulation Mode, the Discrete Input Register will be read from the LVDA. In the repeatable Flight Simulation Mode, they will be assumed to be zero.

M-PC50





2.49 SIXTY SECOND DATA COMPRESSION (PC60)

Function

This module compresses and stores the sixty second composite EMR word if a change occurred during the sixty second interval.

Entries

M.PC60 Entered from the Periodic Processor every sixty seconds.

External

M.PC80 Data Compression Store Utility

E.IHBT Inhibit Interrupts

E.ENBL Enable Interrupts

Input Data

D.VDCT Time from GRR (in seconds positioned for data compression)

D.VERO Accumulated sixty second EMR word

Output Data

Data Compression Table 2 entries

Status Words

None.

Telemetry

None.

Initial Conditions

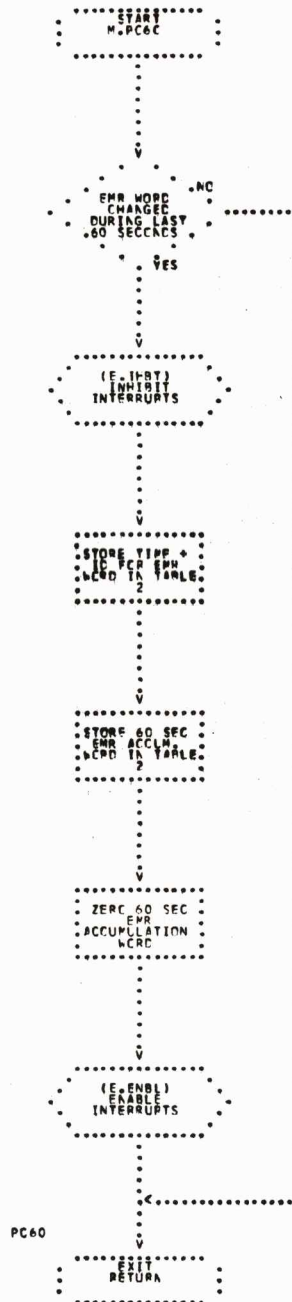
The accumulated sixty second EMR word, D.VERO, must be initially set to zero.

Notes

The accumulated sixty second EMR word will be reset to zero upon completion of the data store.

Accumulated sixty second EMR word, time, and proper identification will be stored in Data Compression Table 2.

M.PC60



2.50 DATA COMPRESSION POSITION UTILITY (PC70)

Function

This module positions gimbal angle and accelerometer readings for compression.

Entries

M.PC70 Entered by the One Hundred Second Data Compression module to position gimbal angles

M.PC71 Entered by the Fifty Second Data Compression module to position accelerometers

External

None.

Input Data

See Notes.

Output Data

See Notes.

Status Words

None.

Telemetry

None.

Initial Conditions

None.

Notes

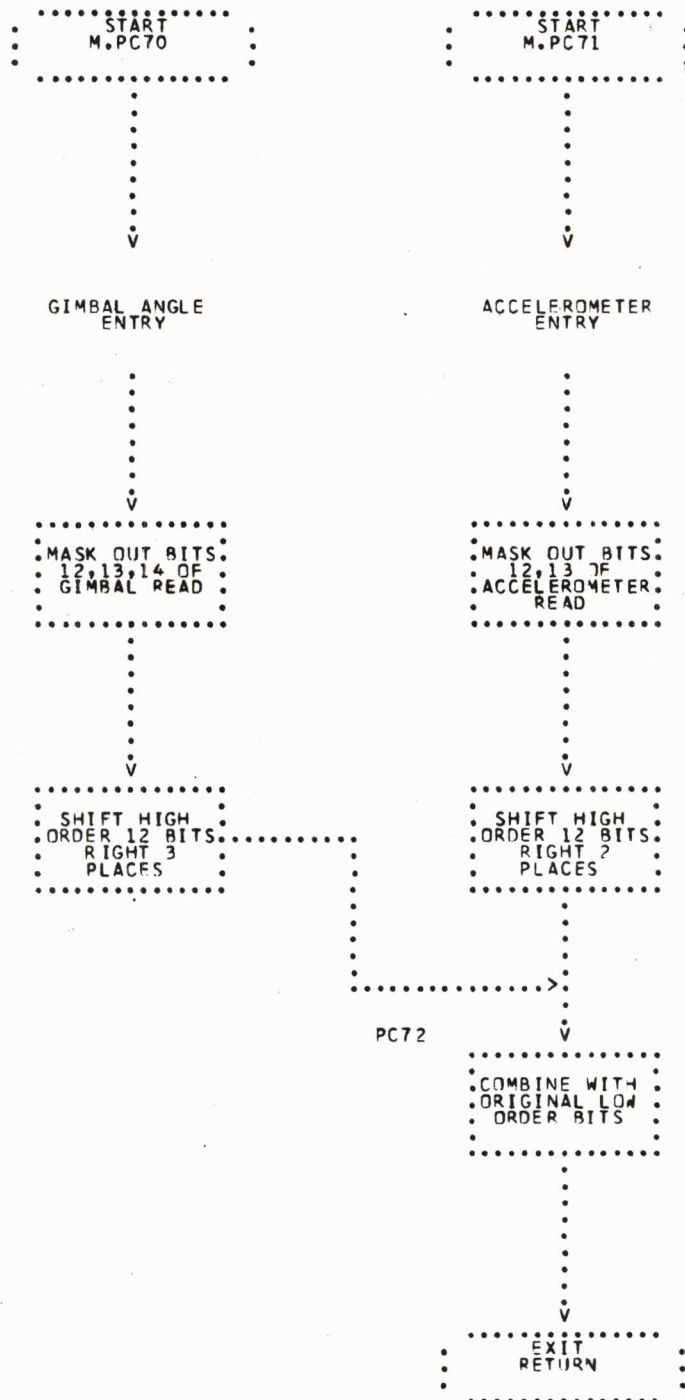
Argument must be in the accumulator upon entry, and the positioned data will be in accumulator upon exit.

DATA COMPRESSION POSITION UTILITY (PC70)

1

M. PC70

M. PC71



2.51 DATA COMPRESSION STORE UTILITY (PC80)

Function

This module stores compressed data into the table whose data sector is in effect.

Entries

M.PC80 Entered by the data compression modules for data compression.

External

None.

Input Data

V.KSTR	Compressed data storage flag
D.PPT1	Table 1 present pointer
D.PPT2	Table 2 present pointer
D.PPT3	Table 3 present pointer
D.PPT4	Table 4 present pointer
D.BPT1	Table 1 beginning pointer
D.BPT2	Table 2 beginning pointer
D.BPT3	Table 3 beginning pointer
D.BPT4	Table 4 beginning pointer
D.EPT1	Table 1 ending pointer
D.EPT2	Table 2 ending pointer
D.EPT3	Table 3 ending pointer
D.EPT4	Table 4 ending pointer

Output Data

D.PPT1 Table 1 present pointer

D.PPT2 Table 2 present pointer

D.PPT3 Table 3 present pointer

D.PPT4 Table 4 present pointer

Data Compression Table 1

Data Compression Table 2

Data Compression Table 3

Data Compression Table 4

Status Words

None.

Telemetry

None.

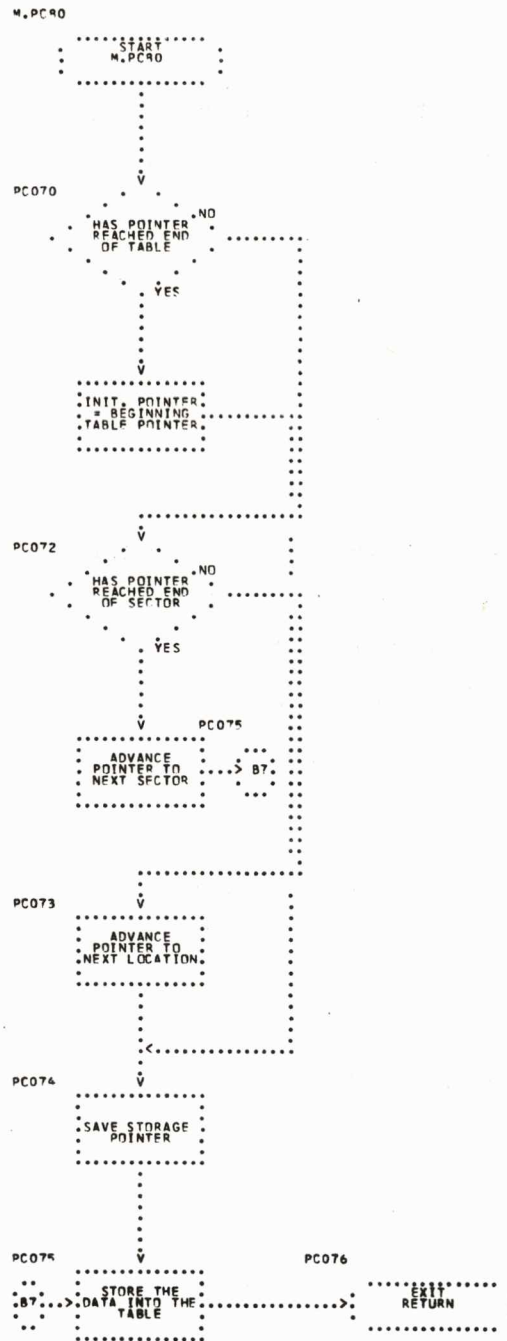
Initial Conditions

None.

Notes

The data to be stored must be in the accumulator on entry.

DATA COMPRESSION STORE UTILITY (PC80)



2.52 ONE HUNDRED SECOND DATA COMPRESSION (PC99)

Function

This module compresses and stores fine gimbal angles and backup gimbal angles every one hundred seconds.

*

Entries

M.PC99 Entered from the Periodic Processor every one hundred seconds.

External

M.PC80 Data Compression Store Utility
M.PC70 Data Compression Position Utility
E.ENBL Enable interrupts
E.IHBT Inhibit interrupts

Input Data

D.VDCT Time from GRR (in seconds positioned for data compression)

Output Data

Data Compression Table 1 entries

Status Words

None.

Telemetry

None.

Initial Conditions

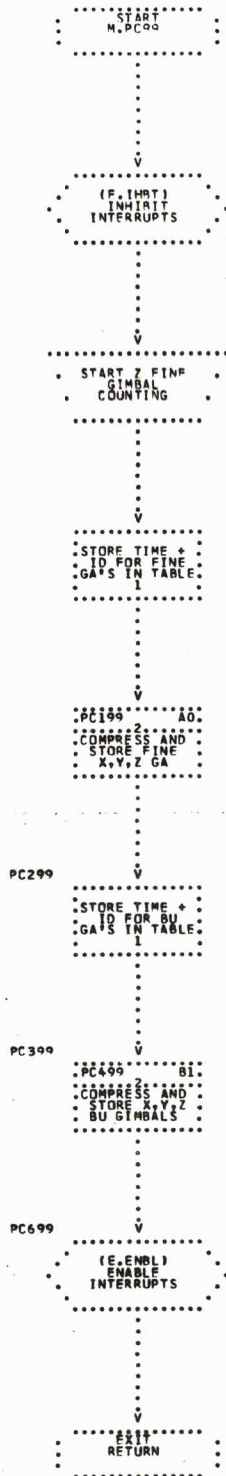
None.

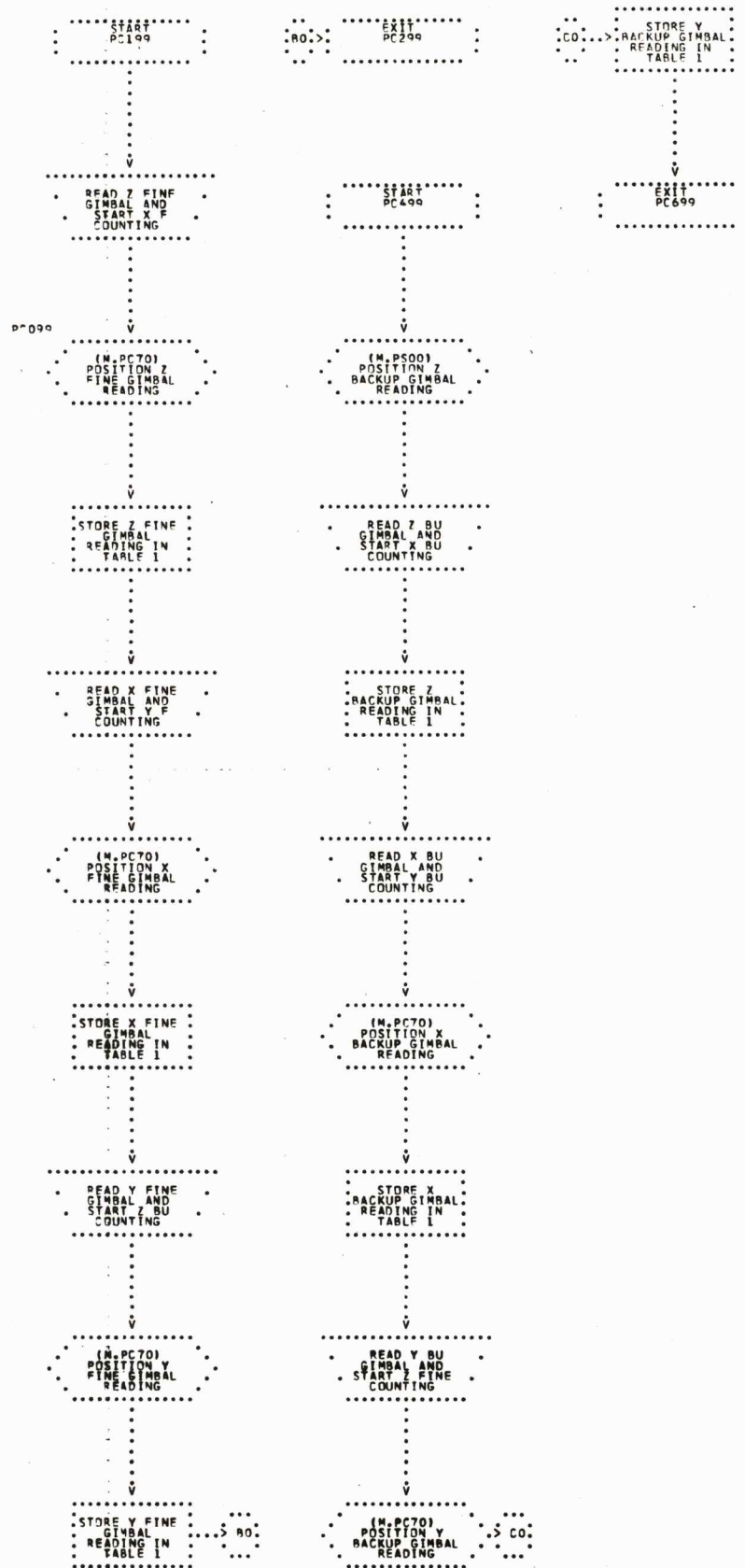
Notes

The gimbals, time, and proper tags will be stored in Data Compression Table 1.

ONE HUNDRED SECOND DATA COMPRESSION (PC99)

M.PC00





2.53 SIMULATED PLATFORM GIMBAL ANGLES (PG)

Function

The function of the Simulated Platform Gimbal Angles module is to compute the expected gimbal angles due to the earth's rotational rate in order to compensate for the actual platform gimbals read during Flight Simulation Mode of program operation with the platform powered up and connected.

Entries

M.PG00 Entered once per computation cycle from the Non-Interrupt Sequencer only when operating in the Flight Simulation Mode.

External

U.FD00 Fixed Point Divide Utility
U.SC00 Sine-Cosine Utility
U.TR00 Real Time Telemetry Delay Utility
E.ENBL Enable interrupts
E.IHBT Inhibit interrupts

Input Data

D.VDT	Elapsed time between current and previous computation cycle accelerometer reads in seconds	
D.FMDI	Flight mode indicator	
D.VDCX	X Delta Chi (ladder increments in pirads)	*
D.VDCY	Y Delta Chi (ladder increments in pirads)	*
D.VDCZ	Z Delta Chi (ladder increments in pirads)	*
D.VTHX	Total roll gimbal for platform-up, guidance chi X in platform- down mode	*
D.VTHY	Total pitch gimbal for platform-up, guidance chi Y in platform- down mode	*
D.VTHZ	Total yaw gimbal for platform-up, guidance chi Z in platform- down mode	*

Output Data

D.VCCX } Roll, pitch, and yaw command chi's modified by expected changes
D.VCCY } in gimbal angles due to earth's rate
D.VCCZ }

Status Words

None.

Telemetry

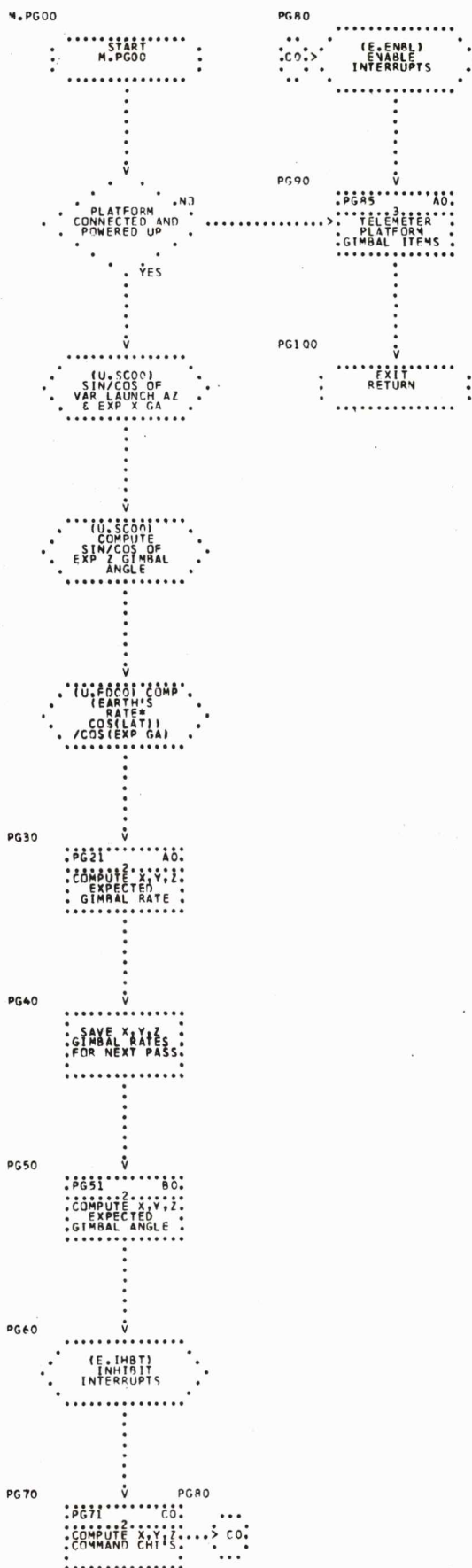
<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.DCXT	1451	Delta chi X
V.DCYT	1455	Delta chi Y
V.DCZT	1445	Delta chi Z
V.CCXT	0505	Command chi X
V.CCYT	0511	Command chi Y
V.CCZT	0501	Command chi Z
V.THXT	0021	Roll gimbal angle
V.THYT	0025	Yaw gimbal angle
V.THZT	0015	Pitch gimbal angle

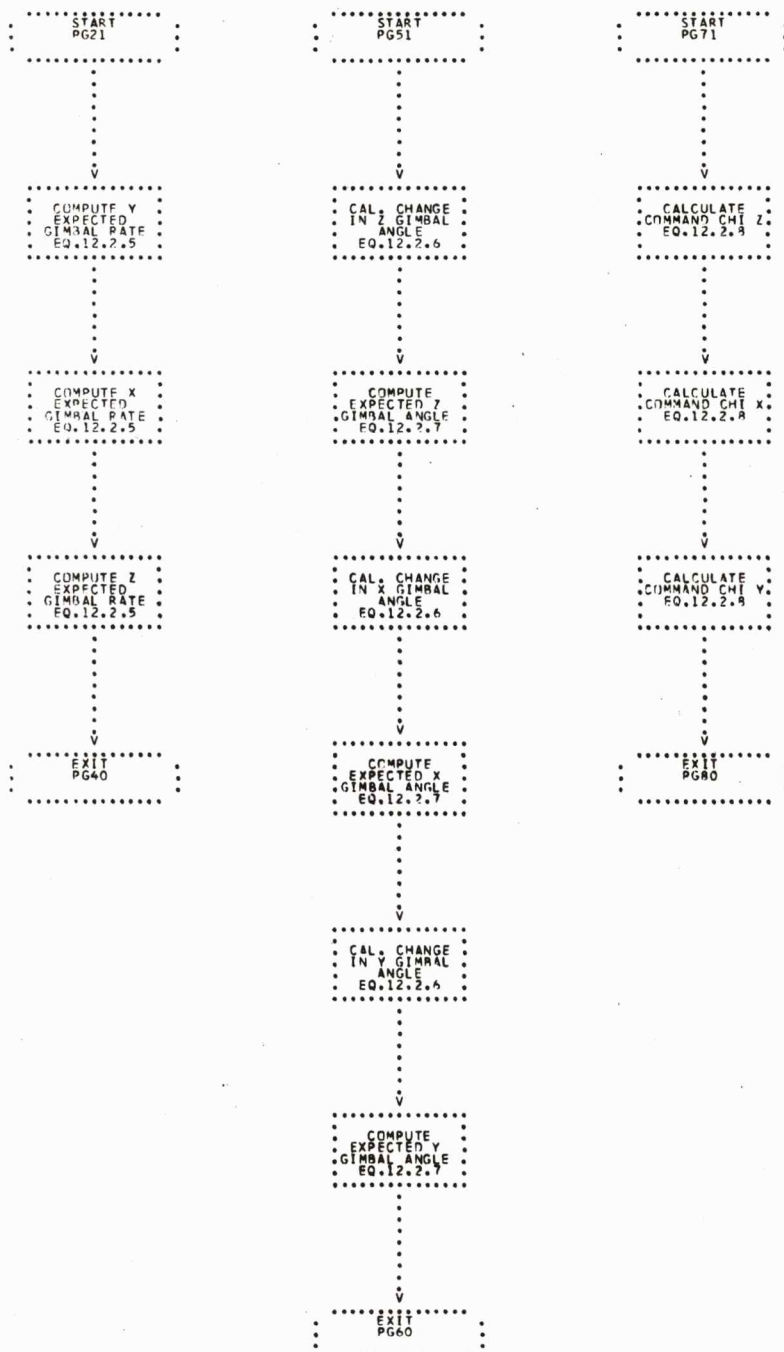
Initial Conditions

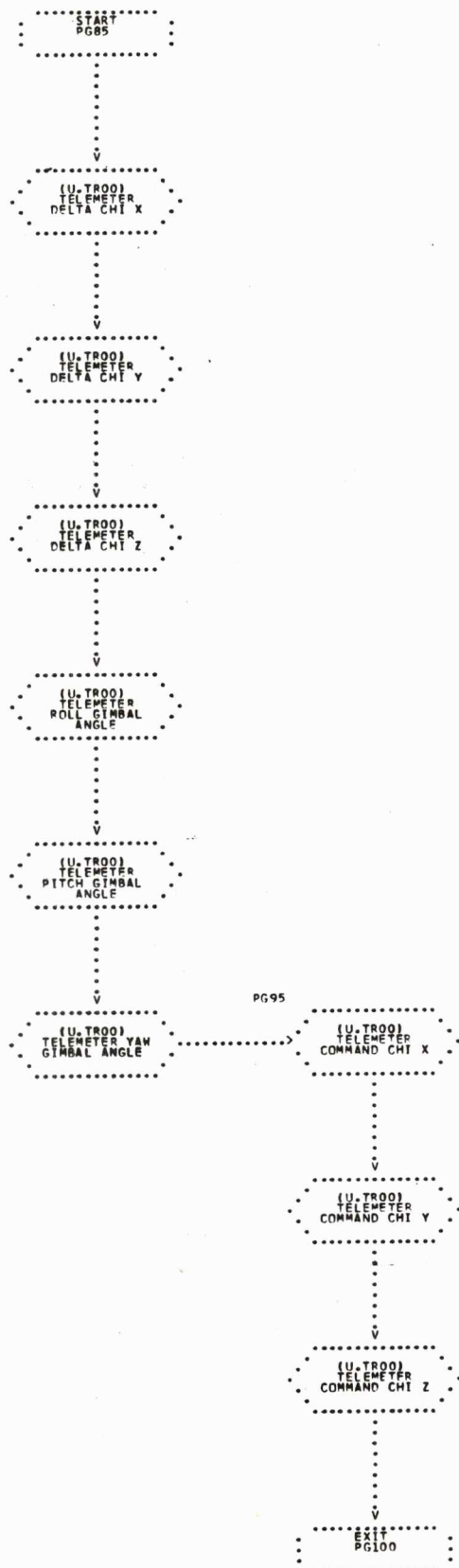
D.VSF1, D.VESL, V.STXE and D.VECL must be initialized to proper values at the start of the mission. V.STZE must be zeroed at mission start.

Notes

During the modification of the command chi's, interrupts must be inhibited to prevent a change in the chi's due to a Timer 1 interrupt which links to the Minor Loop module where the command chi's are updated by the delta chi's. This module is executed only in the Flight Simulation Mode.







2.54 ROLL GUIDANCE (RG)

Function

This module checks to see if roll gimbal is within the completion tolerance. When it is within the completion tolerance, this module is queued-out.

Entries

M.RG00 Entered from Time Tilt Module (M.TT00)

External

None.

Input Data

D.VTHX Total Roll gimbal angle

*

Output Data

D.VGCX Guidance Chi X (Desired roll gimbal angle)

*

H.RG00 Variable HOP to queue-out Roll Guidance (M.RG00)

*

Status Words

D.VMC5 Mode Code 25

Telemetry

None.

Initial Conditions

None.

Notes

None.

ROLL GUIDANCE (RG)

M.RG00

```

START
M.RG00
    
```

V

```

ZERO GUIDANCE
CHI X
    
```

V

RG101

```

ROLL GIMBAL
WITHIN
COMPLETION
TOLERANCE
    
```

NO

YES

V

```

SET ROLL
COMPLETE BIT
IN MC25
    
```

V

```

(E,Q)
QUEUE-OUT
ROLL GUIDANCE
(M.RG00)
    
```

V

RG150

```

EXIT
RETURN
    
```

2.55 SIMULATED ACCELEROMETERS (SA)

Function

The function of the Simulated Accelerometers module is to compute simulated velocity increments for a predetermined thrust model during the Flight Simulation Mode of program operation. The method of computation is dependent upon which of the following platform options is in effect:

1. No Platform Inputs Accepted

The Simulated Accelerometers module computes the velocity increments required to simulate platform outputs for the thrust model. These values are used in lieu of actual accelerometer readings throughout the program.

2. Platform Connected and Power Up

The Simulated Accelerometers module computes the simulated velocity increments by adjusting actual accelerometer readings for accelerations due to gravitation and the thrust model.

Entries

M.SA00 Entered from the Non-Interrupt Sequencer once per computation cycle only when operating in the Flight Simulation Mode.

External

U.SC00 Sine-Cosine Utility

Input Data

D.VGCY Guidance Chi Y (Desired pitch gimbal angle)

*

D.VGCZ Guidance Chi Z (Desired yaw gimbal angle)

*

D.VFMC	Generated acceleration (Backup)	
D.VDTP	Intermediate parameter (20*D.VDT)	*
D.VTAS	Total time elapsed in mission	*
D.VDAX	Optisyn A change in X velocity	*
D.VDBX	Optisyn B change in X velocity	*
D.VDAY	Optisyn A change in Y velocity	*
D.VDBY	Optisyn B change in Y velocity	*
D.VDAZ	Optisyn A change in Z velocity	*
D.VDBZ	Optisyn B change in Z velocity	*
D.FMDI	Flight mode indicator	

Output Data

D.VDAX	Optisyn A change in X velocity	*
D.VDBX	Optisyn B change in X velocity	*
D.VDAY	Optisyn A change in Y velocity	*
D.VDBY	Optisyn B change in Y velocity	*
D.VDAZ	Optisyn A change in Z velocity	*
D.VDBZ	Optisyn B change in Z velocity	*
D.VFX	Expected X platform velocity change	
D.VFY	Expected Y platform velocity change	
D.VFZ	Expected Z platform velocity change	
D.VDX	Intermediate X velocity change parameter	
D.VDY	Intermediate Y velocity change parameter	
D.VDZ	Intermediate Z velocity change parameter	

Status Words

None.

Telemetry

None.

Initial Conditions

V.GCL, V.SAZ, V.CAZ, V.CS, V.SS, VSL2, V.CL2, and V.GRV must be initialized to proper values prior to liftoff.

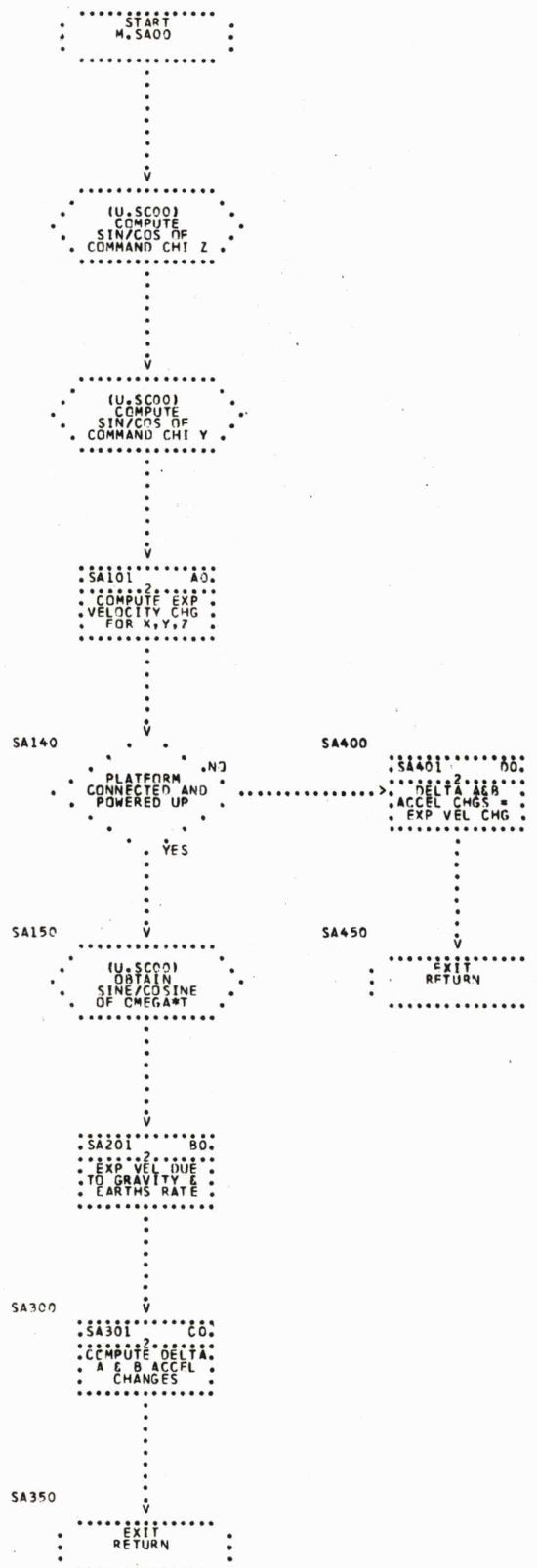
Notes

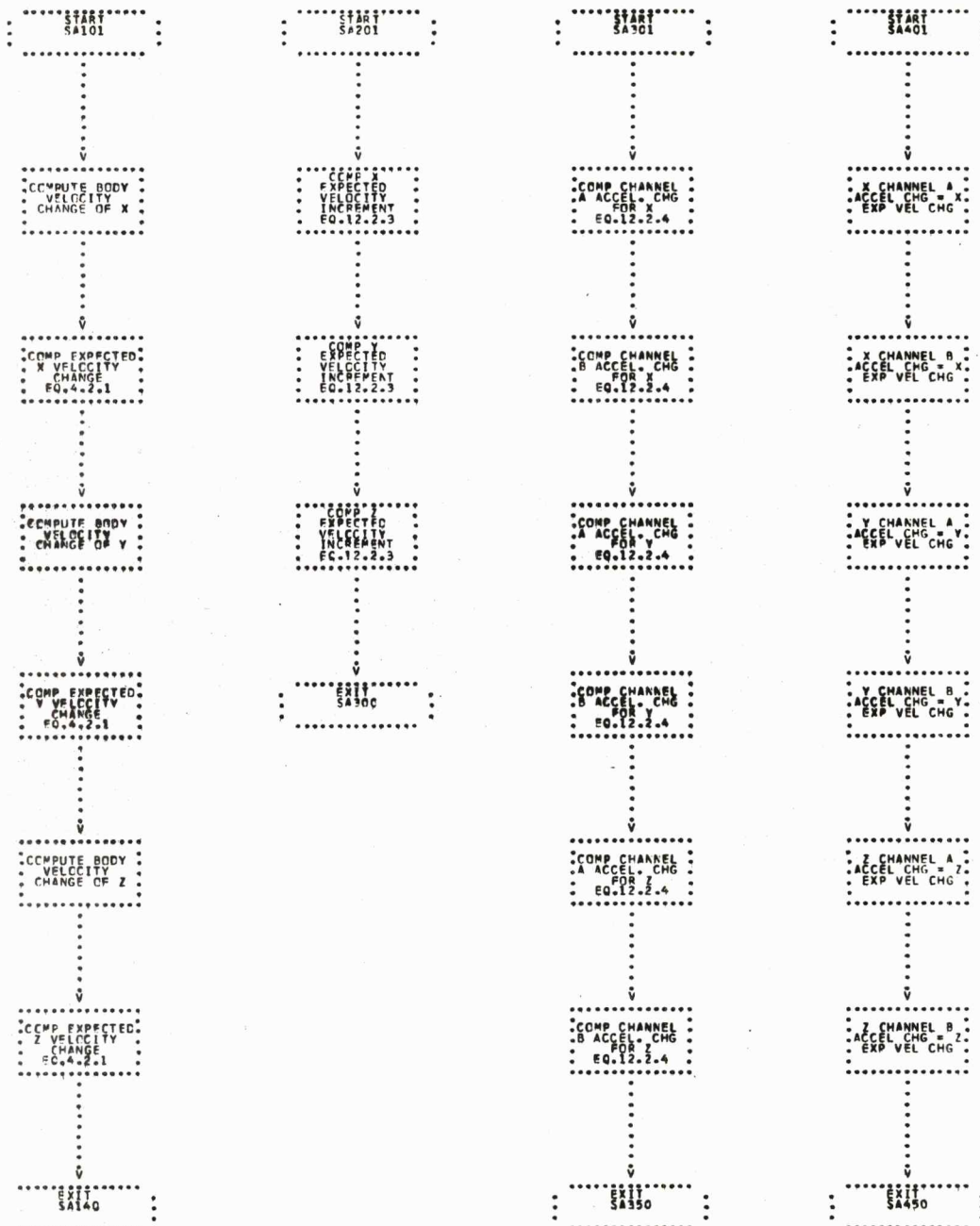
This module is executed only in the Flight Simulation Mode of program operation.

SIMULATED ACCELEROMETERS (SA)

1

M.SA00





2.56 SINE-COSINE UTILITY (SC)

Function

The function of the Sine-Cosine module is to compute the sine and cosine of an input angle by means of an algorithm which utilizes a polynomial approximation to the sine function.

Entries

U.SC00 Entered by any module.

External

None.

Input Data

See Notes.

Output Data

See Notes.

Status Words

None.

Telemetry

None.

Initial Conditions

None.

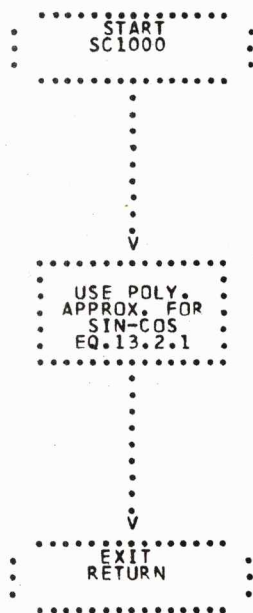
Notes

Upon entry, the input angle must be in the accumulator in pirads scaled B0.

Upon exit, the sine of the angle will be in accumulator and the cosine of the angle will be in the PQ scaled B0.

SINE-COSINE UTILITY (SC)





2.57 SCALED DIVIDE UTILITY (SD)

Function

The function of the Scaled Divide Utility module is to compute as accurately as possible a scaled quotient of two specified numbers which are not identically scaled.

Entries

U.SD00 Entered by any module.

External

None.

Input Data

D.VUTB+1 Scaling of the numerator (scaled B25)

D.VUTB+2 Scaling of the denominator (scaled B25)

D.VUTB+3 The desired scaling of the quotient (scaled B25)

See notes.

Output Data

See Notes.

Status Words

None.

Telemetry

None

Initial Conditions

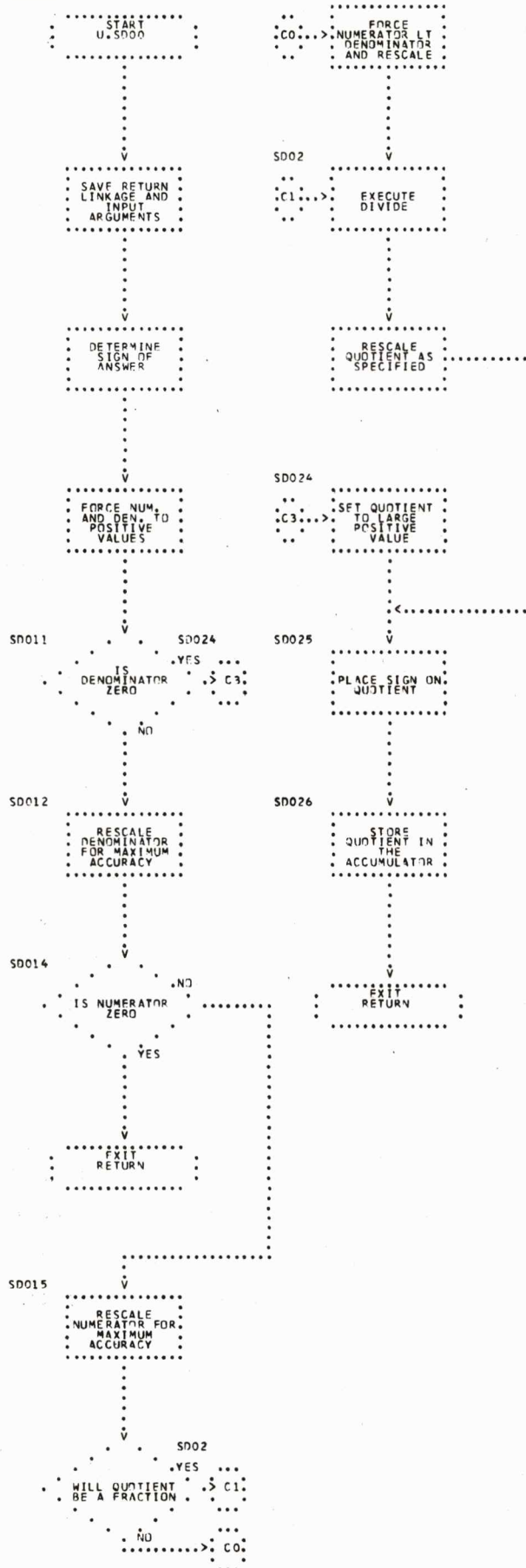
None.

Notes

Upon entry, the numerator must be in the accumulator, and the denominator must be in the PQ register.

The quotient scaled as specified will be in the accumulator on exit.

U.SD00



2.58 STEERING MISALIGNMENT CORRECTION (SM)

Function

The function of the Steering Misalignment Correction module is to calculate the steering misalignment correction terms in both the pitch and yaw planes.

Entries

M.SM00 Entered by the IGM module.

External

U.FD00 Fixed Point Divide Utility
U.SC00 Sine-Cosine Utility
U.TR00 Real Time Telemetry Delay Utility

Input Data

D.VCZA	Average of present and past Minor Loop yaw command chi at the time of major computer cycle accelerometer read	*
D.VCYA	Average of present and past Minor Loop pitch command chi at the time of major computer cycle accelerometer read	*
D.VDT	Elapsed time between current and previous accelerometer reads in seconds	*
D.VMOF	Unsmoothed value of M/F	
D.VYDD	Y component of platform measured acceleration	*
D.VZDD	Z component of platform measured acceleration	*
D.VXDD	X component of platform measured acceleration	*
D.PSMC	Steering misalignment past computation cycle inhibit flag	*
D.GSMC	Steering misalignment correction variable HOP constant	

Output Data

D.VSCZ	Yaw steering misalignment correction	*
D.VSCY	Pitch steering misalignment correction	*

Status Words

D.VMC5 Mode Code 25

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.VSCZ	2560	Yaw steering misalignment correction
D.VSCY	2564	Pitch steering misalignment correction

Initial Conditions

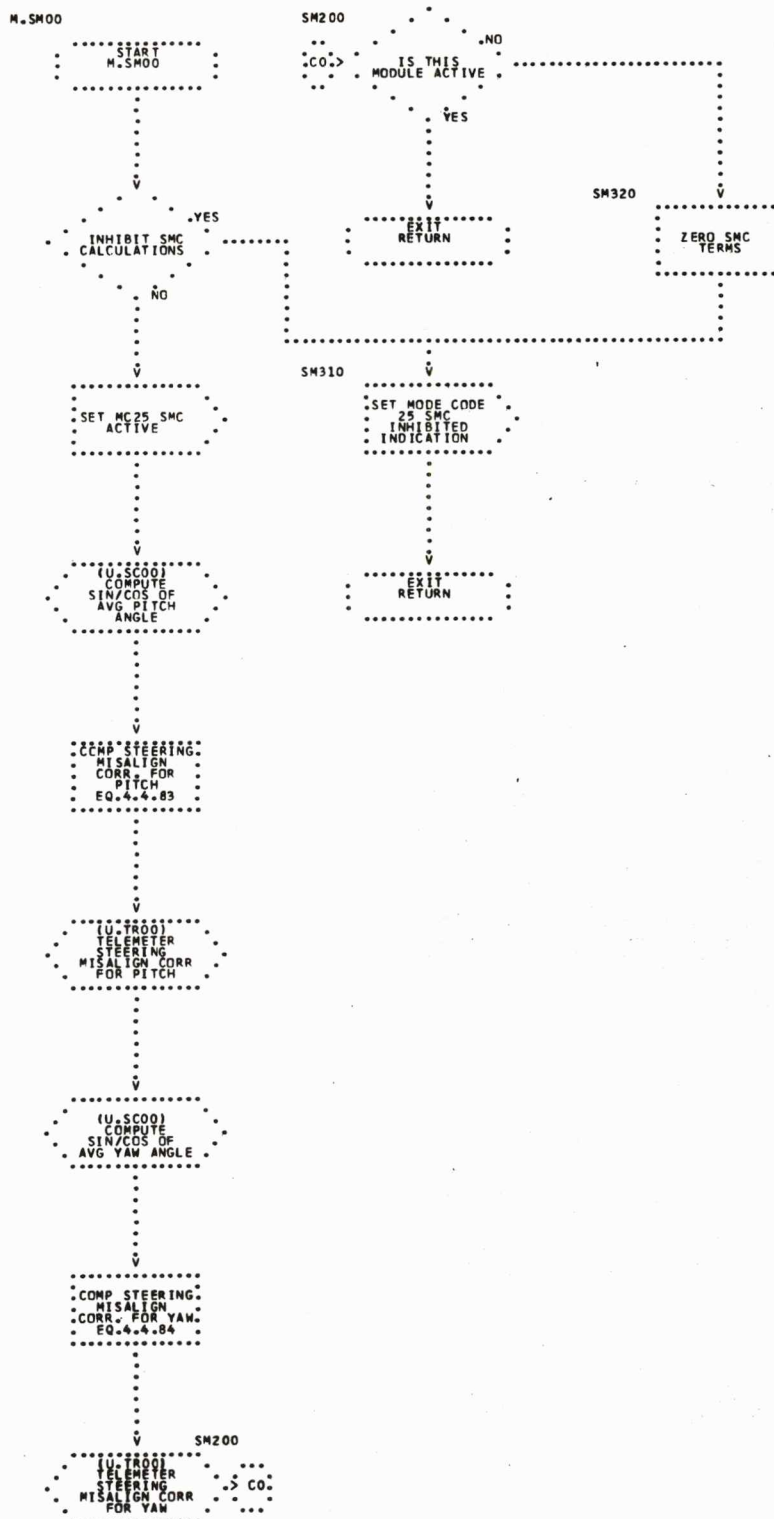
D.PSMC, D.VSCZ, and D.VSCY are initially equal to zero. D.VSCG must be set to proper value at start of mission.

Notes

This module is called as a subroutine of the IGM module. The Events Processor module activates this module by setting the HOP constant D.GSMC to go to M.SM00. When IGM bypasses this module, D.GSMC is set to go to M.IG40.

STEERING MISALIGNMENT CORRECTION (SM)

1



2.59 SQUARE ROOT UTILITY (SR)

Function

The Square Root Utility computes the square root of an input argument.

Entries

U.SR00 Entered from any module.

External

None.

Input Data

See Notes.

Output Data

See Notes.

Status Words

None.

Telemetry

None.

Initial Conditions

None.

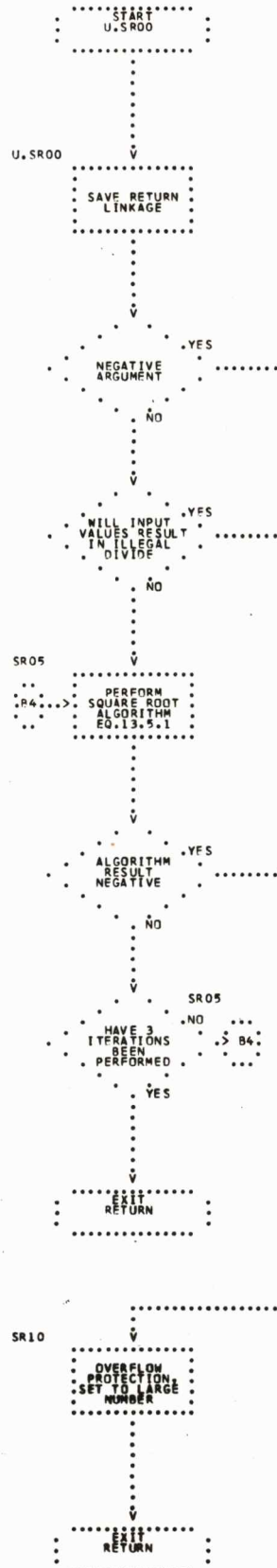
Notes

Upon entry the input argument must be in the accumulator and the first guess of the square root in the utility temporary location, D.VUTB.

The octal value of the first guess must not be less than one-half of the octal value of the input argument; otherwise, an illegal divide will occur which will cause the square root output to be set to the largest possible value. *

Upon exit, the square root of the input argument is in the accumulator. *

SQUARE ROOT UTILITY (SR)



2.60 SWITCH SELECTOR PROCESSOR (SS)

Function

The function of this module is to issue switch selector commands to specified control distributors as a function of time.

Entries

- M.SS00 Entry to the switch selector check from any module requesting an alternate or special switch selector sequence
- M.SS05 Entry from Timer 1 to switch selector check routine
- M.SS10 Entry from the Time Base Set module for proper initialization at each time base change
- M.SS20 Entry to issue forced reset
- M.SS30 Entry from Timer 1 to test for a switch selector hung stage
- M.SS40 Entry from Timer 1 to issue a switch selector stage and address
- M.SS50 Entry from Timer 1 to verify a switch selector address
- M.SS55 Entry to check for time to issue the read command after having issued the stage and complemented address
- M.SS60 Entry from Timer 1 to issue a switch selector read
- M.SS70 Entry from Timer 1 to reset a switch selector
- M.SS80 Entry from Timer 1 to issue a switch selector stage and complement address

External

- M.PC80 Compressed Data Store Utility *
- E.GP08 Entry to the Timer 1 Halt and Reschedule Executive Routine
- U.TRO0 Real Time Telemetry Delay Utility
- E.IHBT Inhibit interrupts
- E.ENBL Enable interrupts
- E.QT Read Command (M.SS55)
- E.QT Read Reset (M.SS70)

E.QT	Complement Stage and Address (M.SS80)
E.QT	Read Command (M.SS60)
E.QT	Verify Address (M.SS50)
E.QT	Switch Selector Check (M.SS05)
E.QT	Hung Stage Test (M.SS30)
E.QT	Issuance of Switch Selector Stage and Address (M.SS40)

Input

D.FLT	Flight Mode indicator	*
D.VTRR	Elapsed total time in current time base	*
D.VRTC	Real time clock reading at last time update	*
D.VTGB	Accumulated ground bias time	*
D.FTUP	Time update waiting flag	*
D.VDCT	Time from GRR (in seconds positioned for data compression)	
V.GBIA	Time base bias to be implemented	*
D.FPHS	Phase indicator flag:	*
	(-) = Boost	*
	(+) = Orbit (any bits except sign bit)	*

Output

D.VTGB	Accumulated ground bias time	*
D.VTRB	Elapsed time in current time base including ground bias time updates	
F.SSAC	Switch Selector active flag	
V.GBIA	Time base bias to be implemented	
D.FWV	Flag which indicates state of water valve (open or close)	

Status Words

D.VASW	Switch Selector Request Status
D.VICR	Internal Control Register
D.FILE	Flight Program Status

D.VMC4 Mode Code 24

D.VMC5 Mode Code 25

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
V.SSTL	0075	Switch Selector Function Indication at Time the Read Command is issued
V.SSFB	0500	Switch Selector Feedback in error

Initial Conditions

The residual data location reserved for use in conjunction with the Switch Selector EXM instruction must be initialized for flight at mission start. All switch selector time biases will be initialized at mission start as well as all table pointers.

Notes

- (1) This module will guarantee all switch selector DOM telemetry.
- (2) All switch selector functions and their associated time of issuance are contained in a set of Switch Selector tables referenced by this module for processing.

Switch Selector Nominal and Alternate Event Tables

- A. Nominal Event Tables
 - 1. The events per mission ICD are to be entered in a nominal event table.
 - 2. The switch selector program is to have the capability to issue events other than the nominal table.
- B. Alternate Event Tables
 - 1. The events are to be entered with delta times referenced to the time the alternate event table is initiated.
 - 2. All alternate event tables are to be terminated when a new time base is initiated.
- C. Event Table Classification (See Table 1)
 - 1. Class 0
 - a. The nominal event table is considered as class 0.
 - b. All alternate event tables may be started when the Switch Selector module is operating from the class 0 table.
 - 2. Class 1
 - a. Class 1 tables will be started immediately upon request.
 - b. If a switch selector is active, a special entry is made to the forced reset operation and the alternate sequence is started.
 - c. The interrupted sequence(s) will not be reentered, and upon completion, the Class 1 alternate sequence will assume a class 0 priority level.
 - d. The nominal S-IVB cutoff switch selector has the highest priority of all switch selector commands.
 - 3. Class 2
 - a. A class 2 sequence interrupts the current switch selector sequence and must be issued in its entirety.
 - b. If a switch selector is active, a special entry is made to the forced reset operation and the alternate sequence is started.
 - c. Upon completion of the class 2 event table control is returned to the table that was in effect when the class 2 table was initiated.

- d. All events that are late because the class 2 sequence was initiated will be issued at the maximum switch selector rate until normal timing can be resumed.
- 3. Class 3
 - a. Class 3 events are to be issued at maximum switch selector rate.
 - b. Class 3 tables may be started when the switch selector program is operating from a class 0, or class 4 table.
 - c. Upon completion of the class 3 event table, control is returned to the table that was in effect when the class 3 table was initiated.
 - d. All events that are late because the class 3 sequence was initiated will be issued at the maximum switch selector rate.
- 4. Class 4
 - a. Class 4 tables are to be concurrent event tables with no restriction on number of events.
 - b. Class 4 tables may be started when the switch selector program is operating from any class table.
 - c. The events of a class 4 table are to be merged with the events of the table that the switch selector is operating from.
 - d. The concurrent table (Class 4) has the lowest priority when the times of the read operation are equivalent.

Switch Selector Table Classes

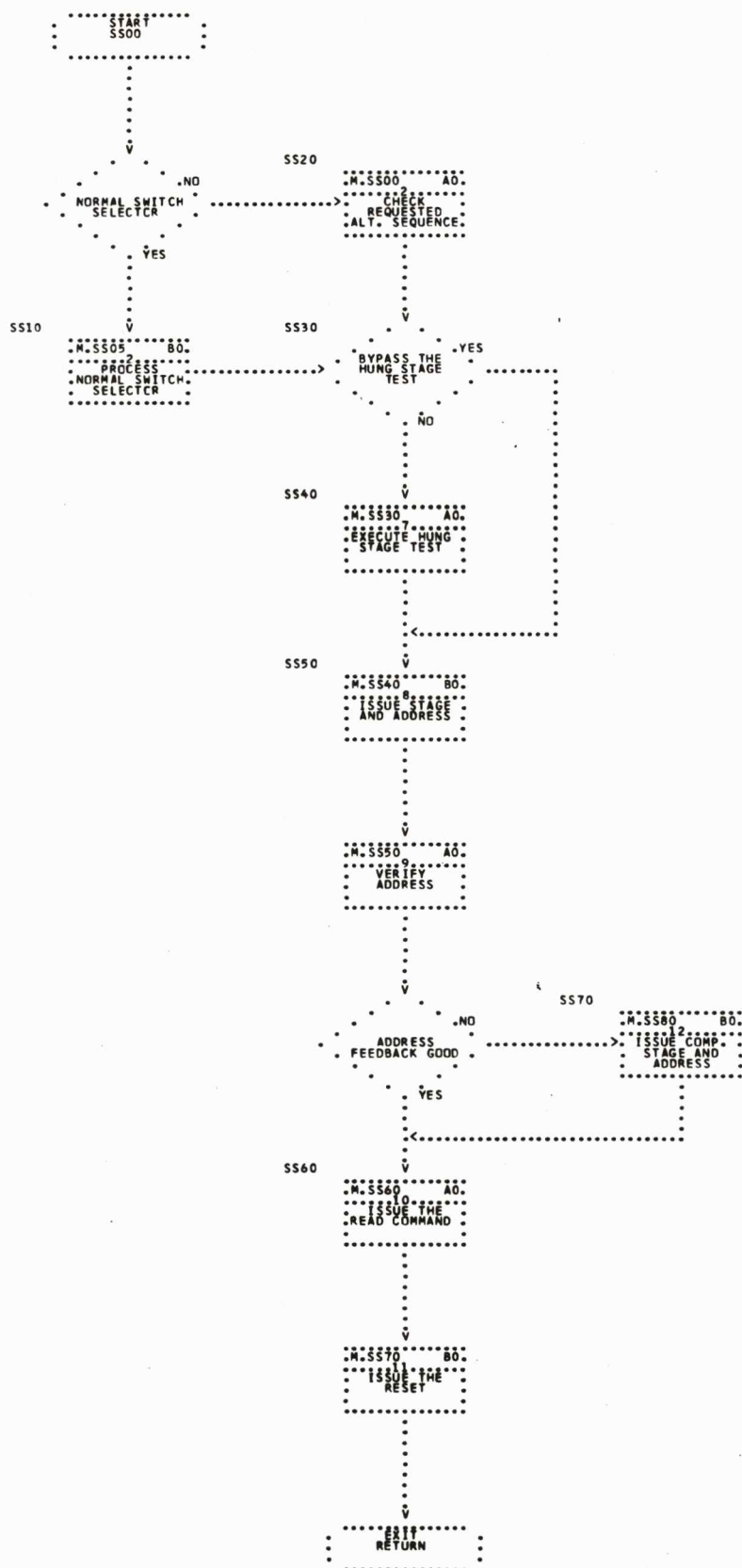
Class 0 Nominal event table

Class 1 Alternate event table of highest priority

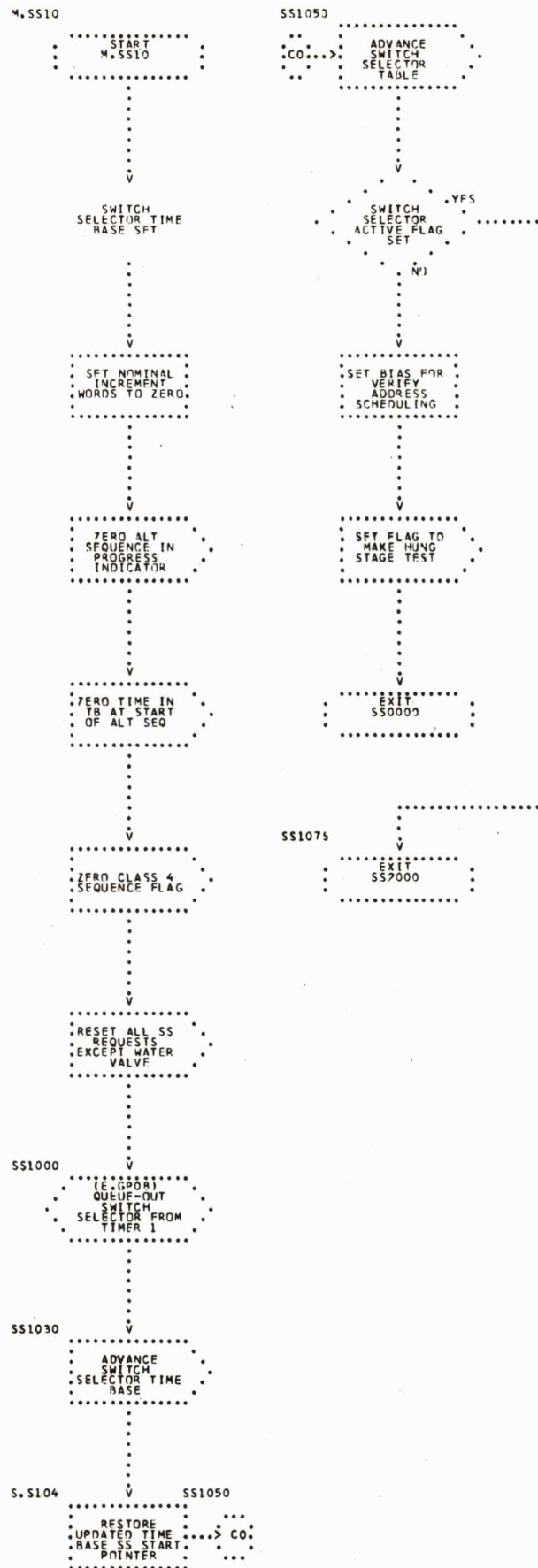
Class 2 Alternate event table of high priority

Class 3 Alternate event table of one or two events

Class 4 Alternate event table of a finite number of events
(concurrent table)







SWITCH SELECTOR PROCESSOR (SS)

M. SS20

START
M. SS20

V

SWITCH
SELECTOR
FORCED RESET

V

SS2000

ISSUE FORCED
RESET

V

SET FLAG TO
BYPASS HUNG
STAGE TEST

V

SS2UPD A0
IB
SWITCH
SELECTOR TIME
UPDATE

V

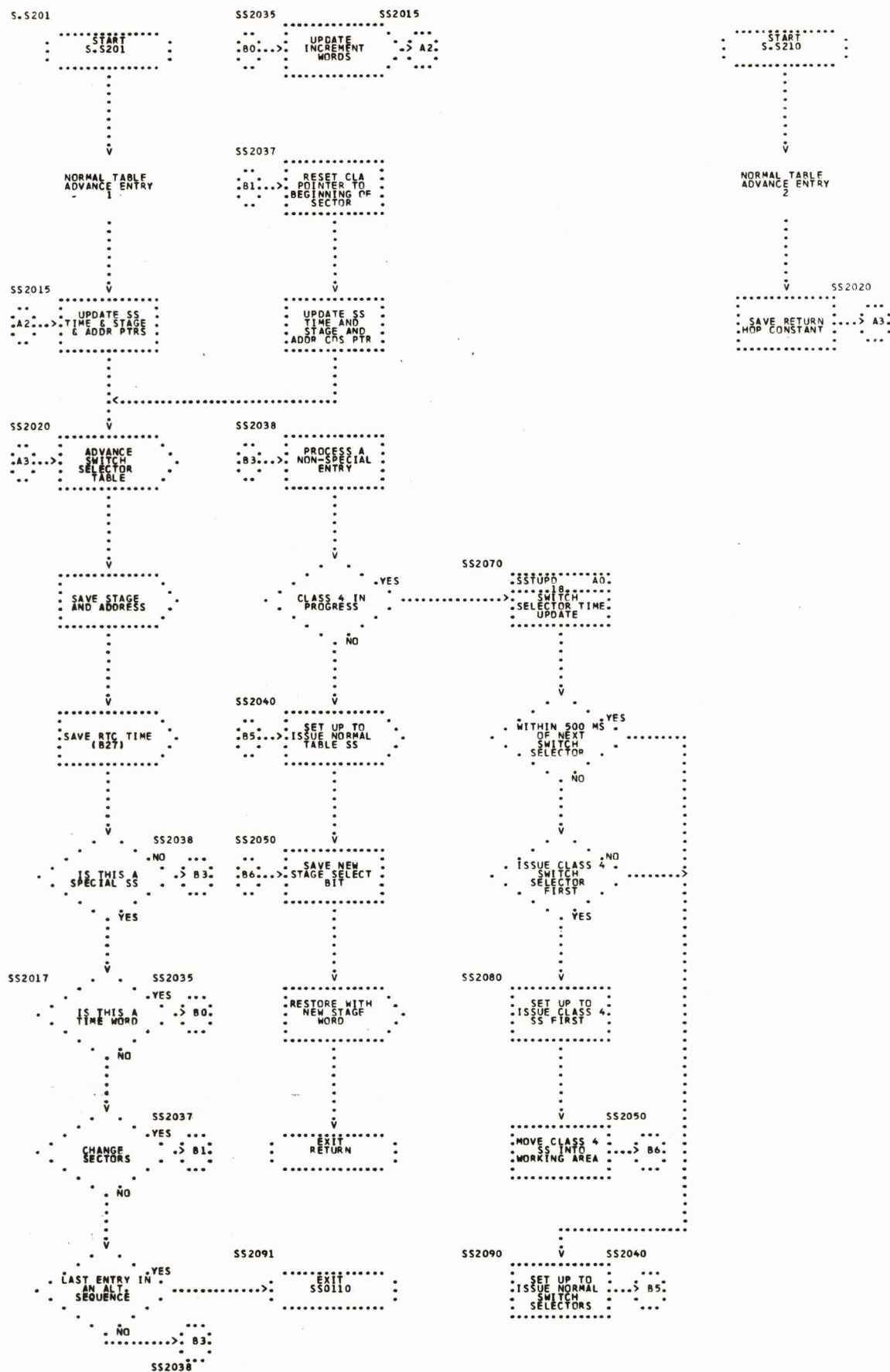
(F. DT)
SCHEDULE
SWITCH SELECT
CHECK
(M. SS05)

V

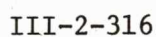
SET BIAS FOR
VERIFY
ADDRESS
SCHEDULING

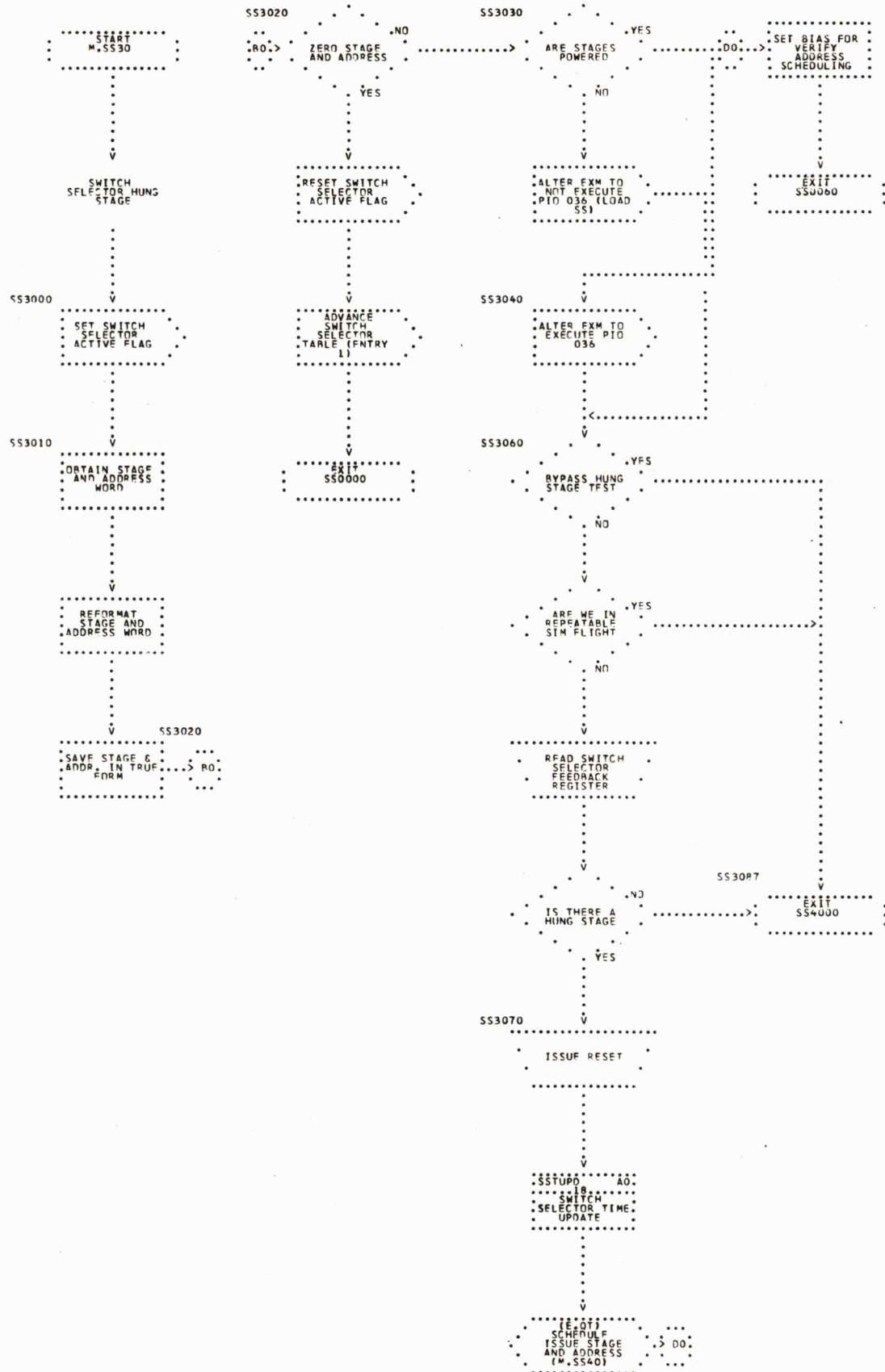
V

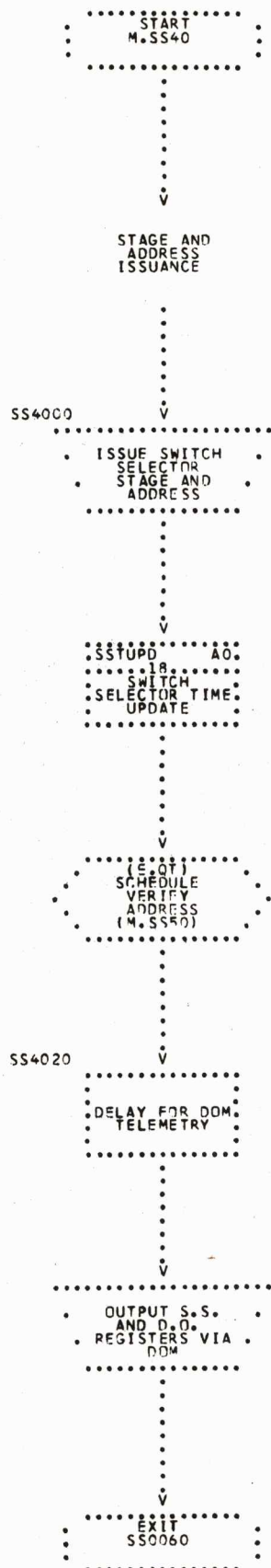
EXIT
SS0060

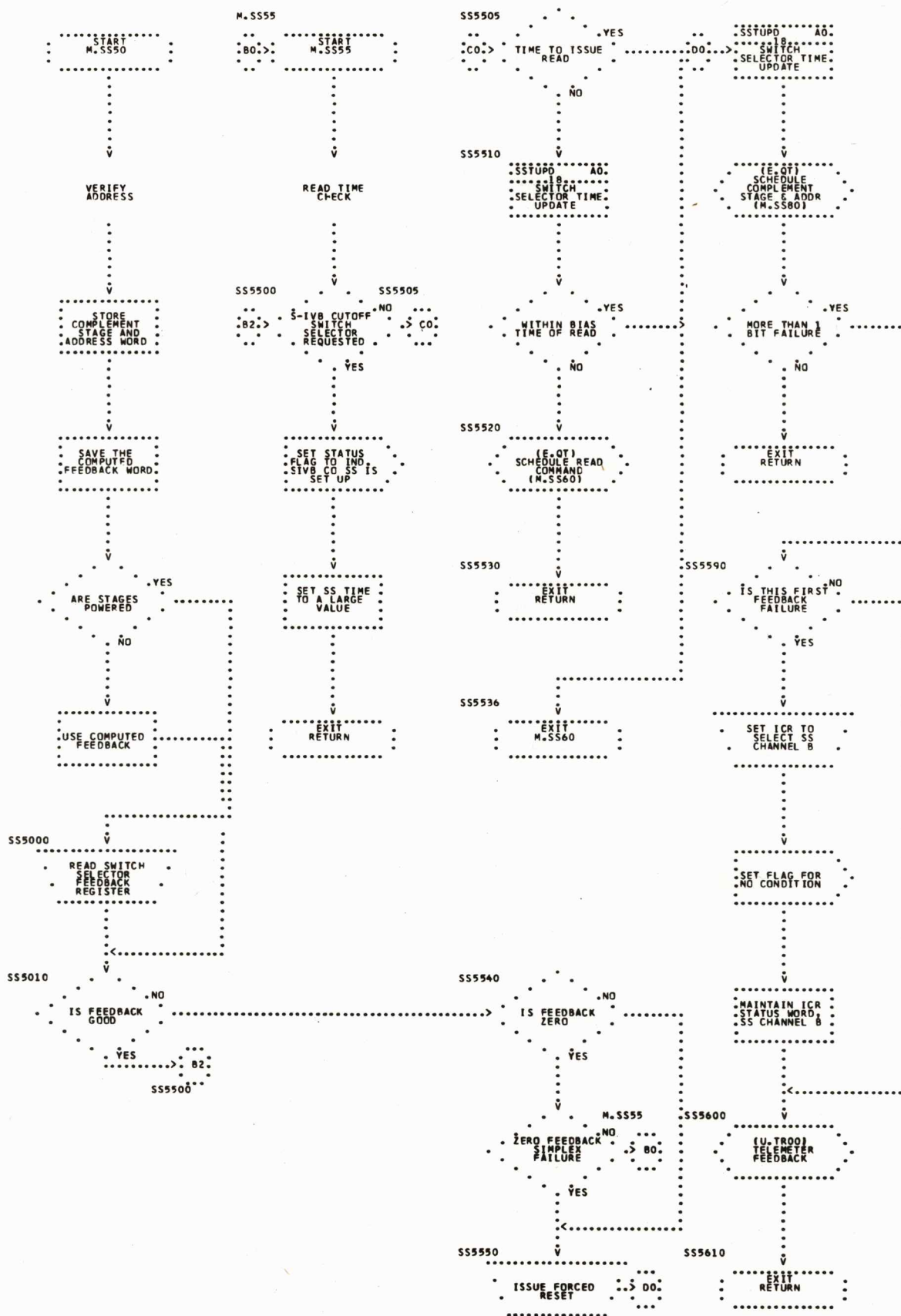


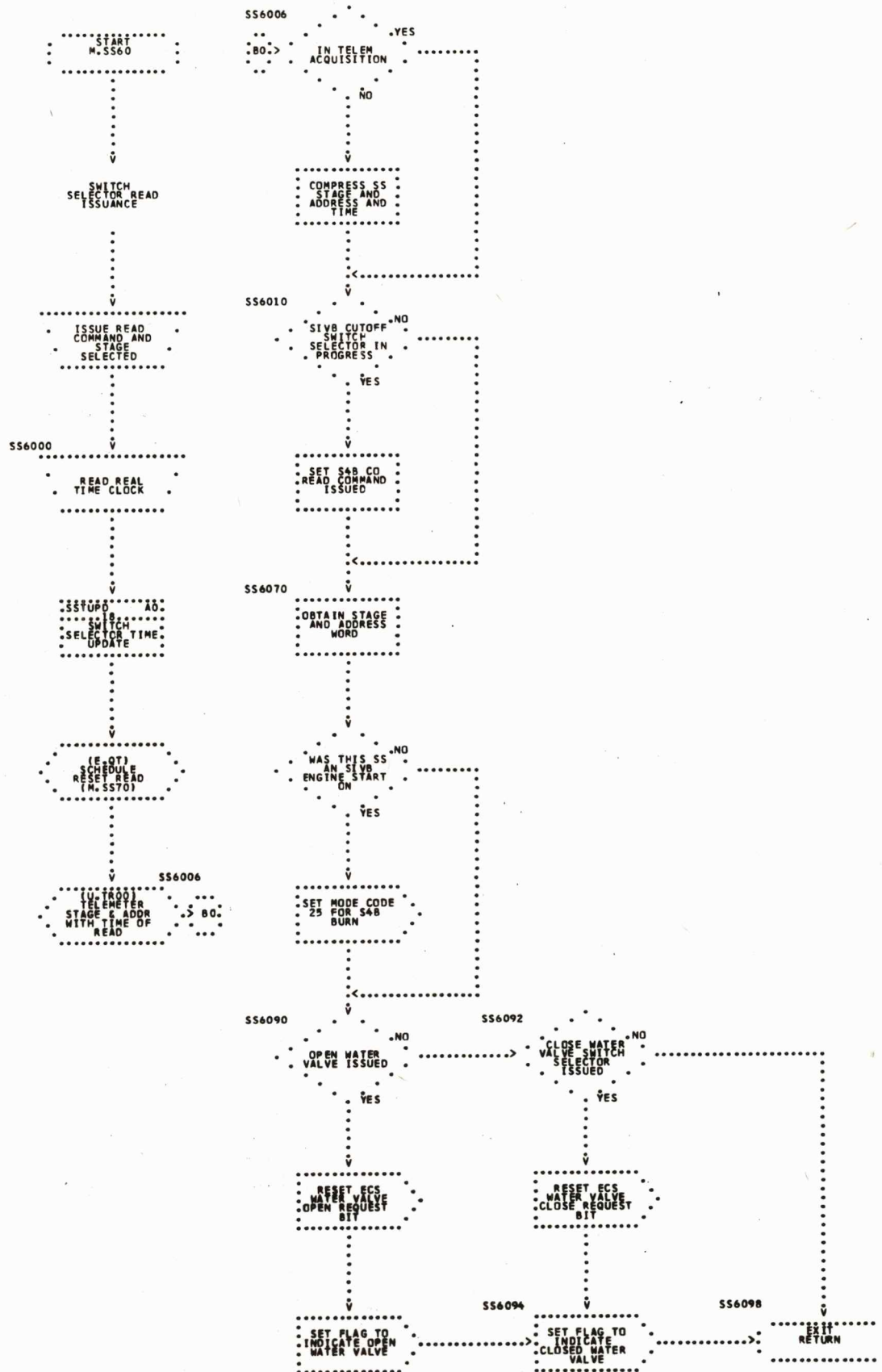
6

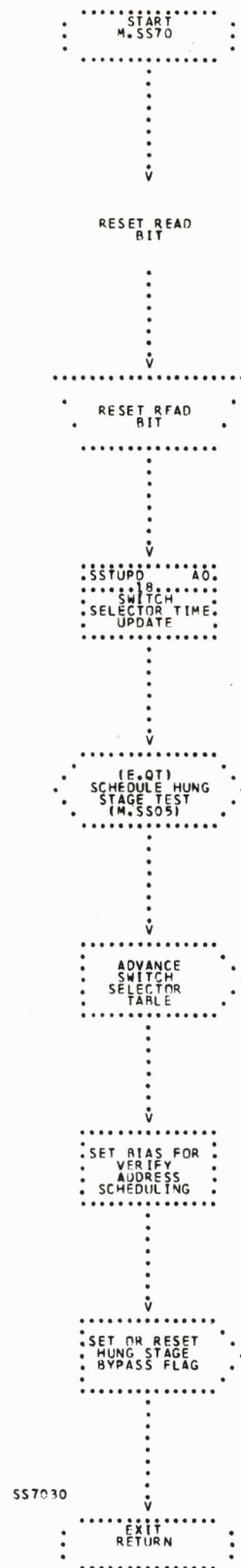


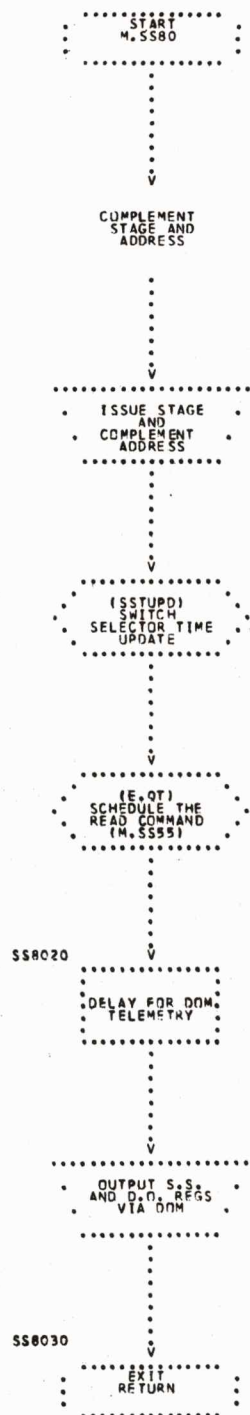


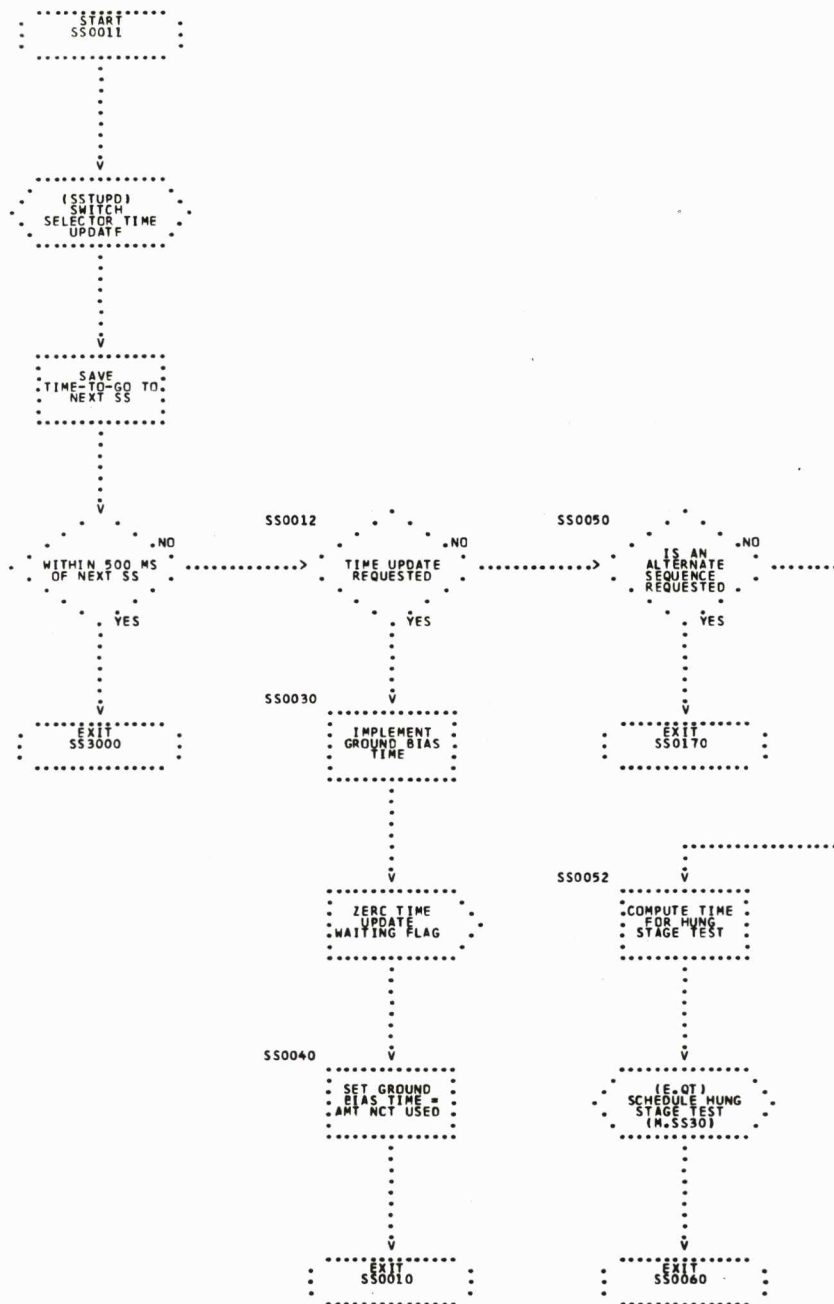


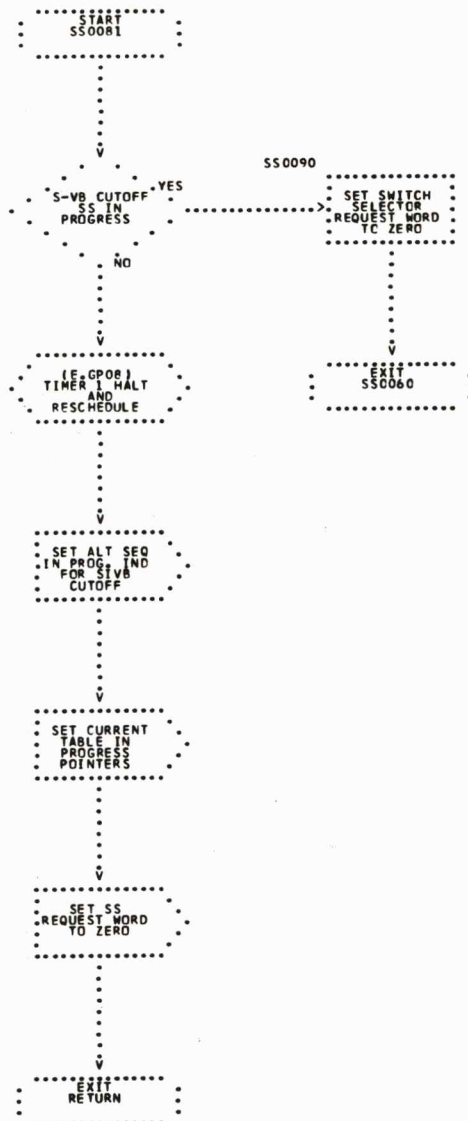




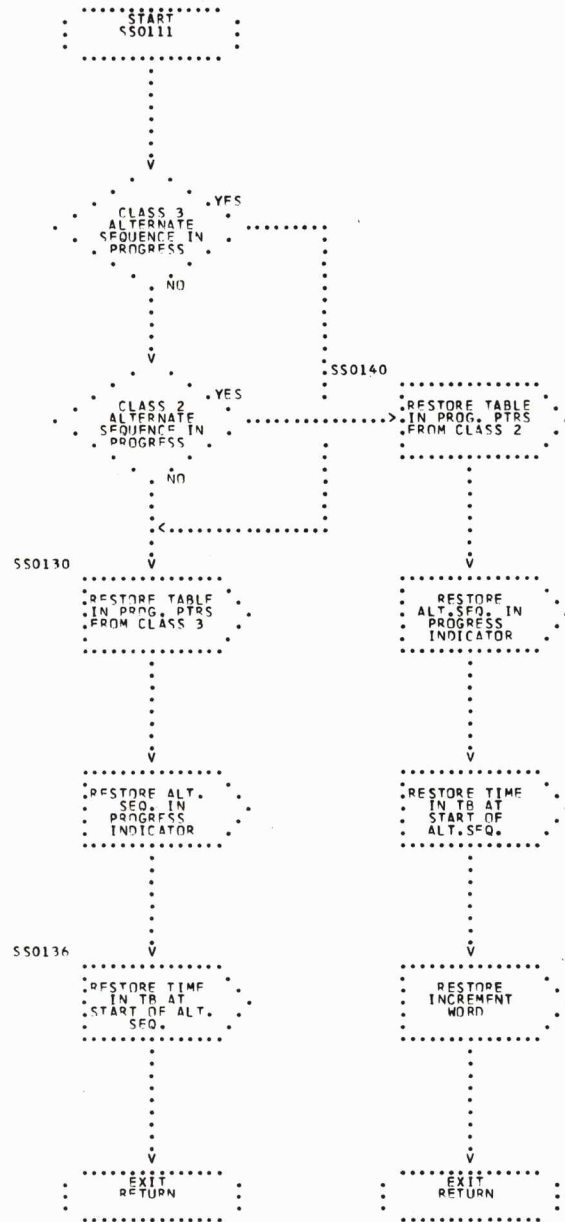


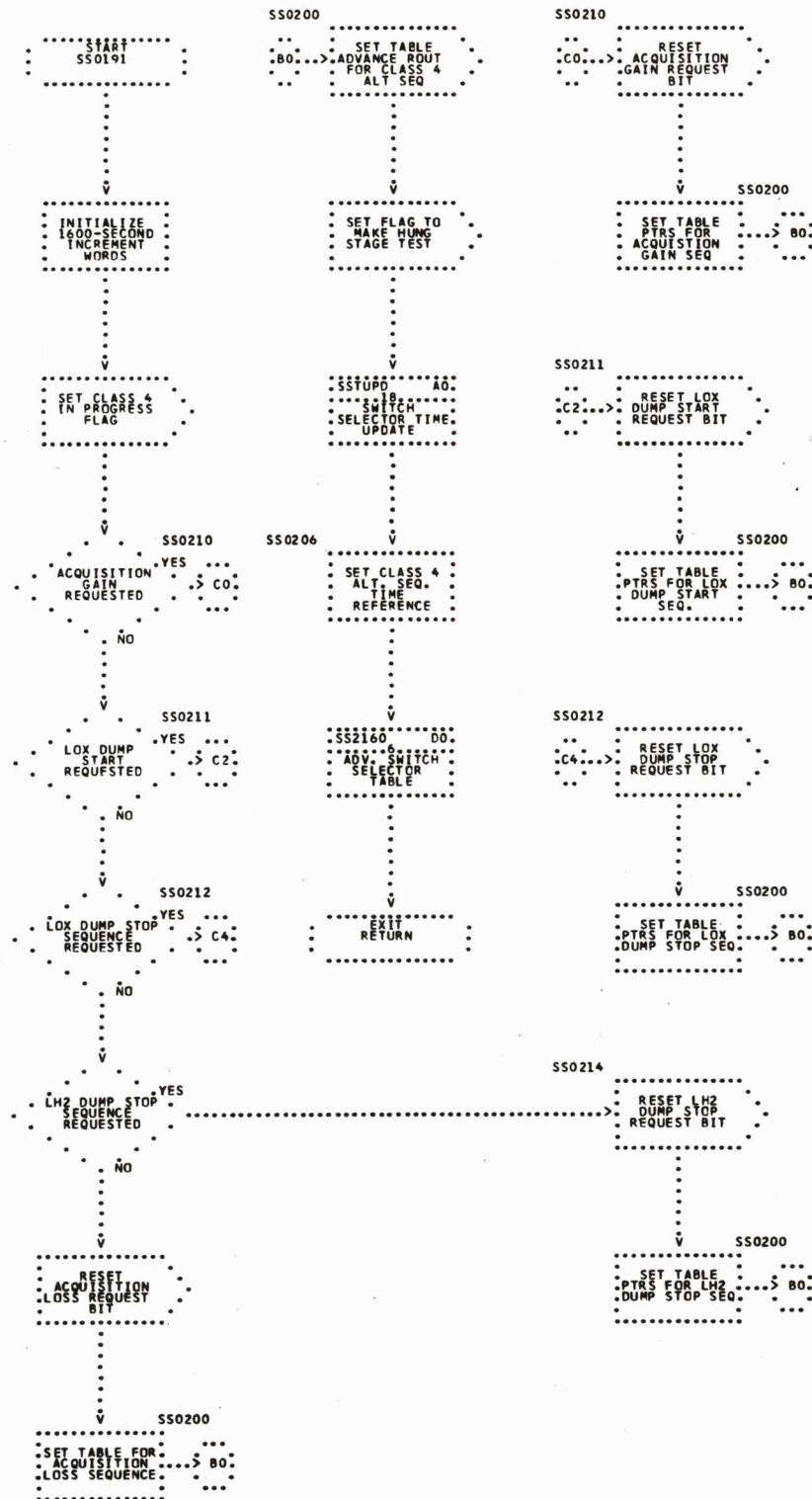


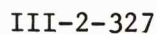




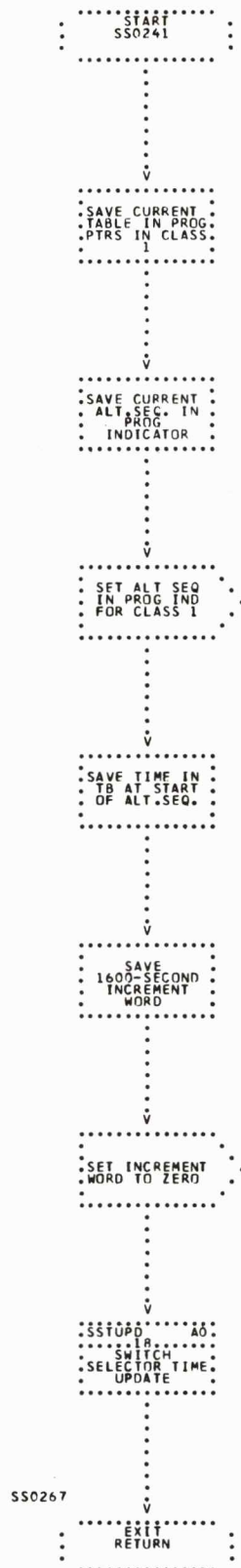
SWITCH SELECTOR PROCESSOR (SS)











2.61 M/F SMOOTHING (ST)

Function

The function of the M/F Smoothing module is to compute and filter the reciprocal value of measured acceleration.

Entries

M.ST00 Entered from the F/M Calculation module,

External

U.FD00 Fixed Point Divide Utility

U.TR00 Real Time Telemetry Delay Utility

Input Data

D.VFOM Total vehicle acceleration determined from accelerometer readings. *

Output Data

D.VMOF Unsmoothed value of M/F *

D.VMFS Filtered value of current M/F

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>	
D.VMOF	2150	Unsmoothed value of M/F	*
D.VMFS	2154	Filtered value of current M/F	*

Initial Conditions

The current and three previous M/F values (D.VMOF, D.VMF1, ..., D.VMF3), the current and three previous M/F filtered values (D.VMFS, D.VMS1, ..., D.VMS3), and the eight filter coefficients must be initialized at start of mission to appropriate constants for the S-IVB stage.

Notes

This module is called as a subroutine of the F/M Calculations module. To activate this module, the HOP constant D.GMFS is set to go to M.ST00. To bypass this module, D.GMFS is set to go to D.V300.

The output of the M/F filter, D.VMFS, will be limited to a maximum value of $.25 \text{ sec}^2/\text{m}$. There will also be a rate of change limit per computation cycle of P.MFSR.

This module keeps a three computation cycle history of previous M/F values and filtered M/F values. The following program variables are used to keep these histories:

D.VMF1	}	History of previous M/F values
D.VMF2		
D.VMF3		
D.VMS1	}	History of previous filtered M/F values
D.VMS2		
D.VMS3		

M/F SMOOTHING (ST)

1

M. ST00

START
M. ST00

ST10

UPDATE M/F
HISTORIES FOR
USE IN M/F
FILTER

(U. F000)
COMPUTE RECIP
OF CURRENT
VEHICLE ACCEL

(U. T000)
TELEMETER
RECIPROCAL OF
F/F

ST20

COMPUTE
SMOOTHED M/F
EQ. 4.2.11

ST40

UPDATE (M/F)'S
HIST FOR USE
IN M/F FILTER

LIMIT M/F
FILTER
OUTPUT RATE
& MAGNITUDE

(U. T000)
TELEMETER
SMOOTHED
VALUE OF M/F

EXIT
RETURN

2.62 TIME BASE SET (TB)

Function

This module will perform the following functions upon entry:

1. Access the Control Program Time Update routine to update the elapsed time in mission and current phase and reset elapsed time in current time base to zero.
2. Set start time of current time base equal to updated elapsed time in mission.
3. Access the table advance routines provided by the Switch Selector Processing, Ladder Ramp Processor, and Events Processor modules to position their respective table pointers for the new time bases.

Entries

M.TB00 Entered once by each time base module defined for the mission to initiate the proper time base.

External

M.EP05 Events Processor
M.LR20 Ladder Ramp Processor
M.SS10 Switch Selector Processing
U.TR00 Real Time Telemetry Delay Utility

E.GP09 Control Program Time Update

Input Data

D.VTMR Time in mission at start of reference (RTC scaled B27)
D.VERT Time error associated with last time update

Output Data

D.VTI	Time from GRR that the current time base was set	*
D.FTBC	Time base change flag	
D.VTB	Time elapsed in current time base	
D.VTBB	Time base biased elapsed time	*
D.VTGB	Accumulated ground bias time	
D.VTBI	Indication of Time Base Initiation	

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>	
D.VTBI	0455	Indication of Time Base Initiation	*

Initial Conditions

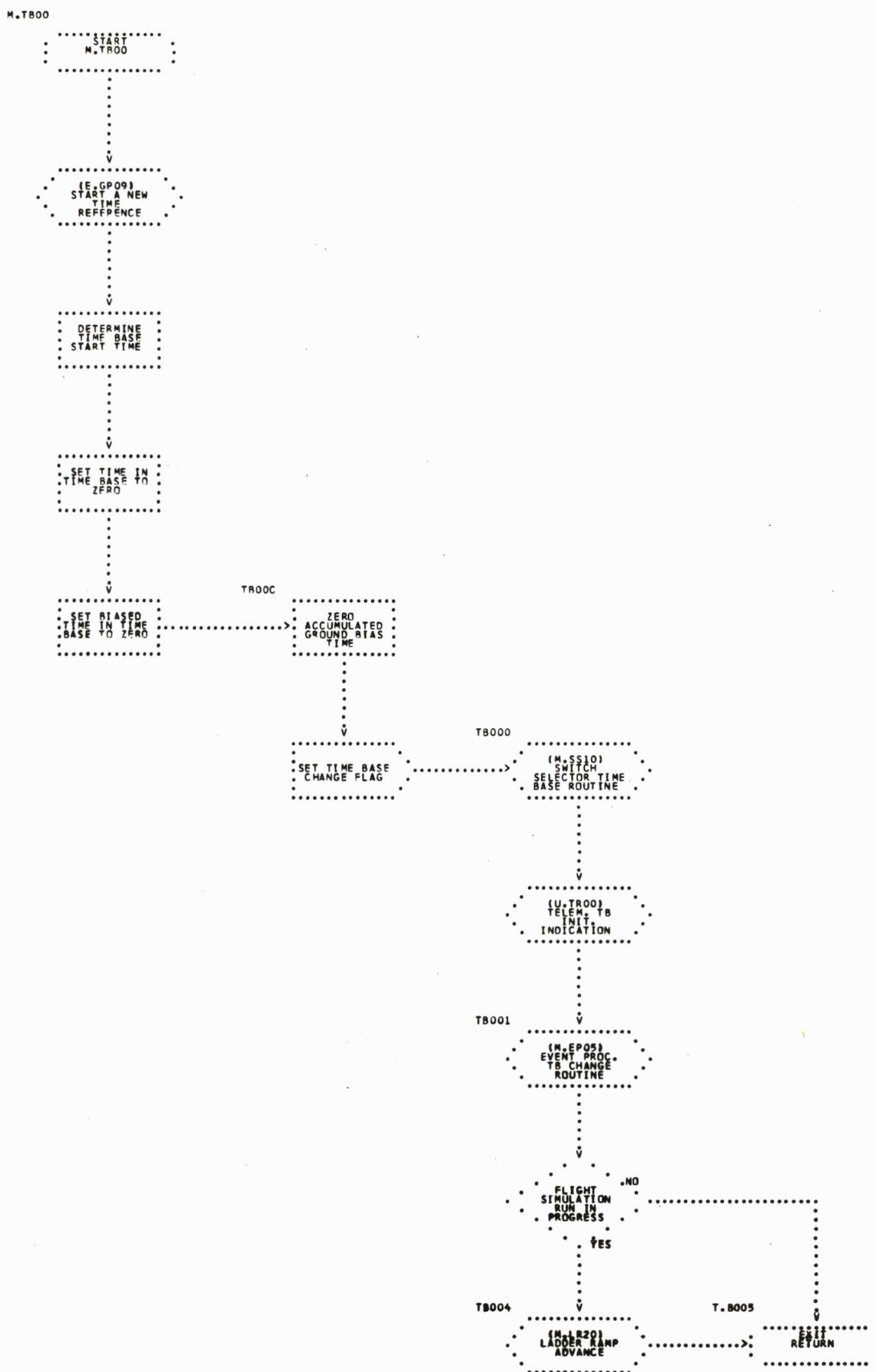
The variable HOP (G.LRTB) must be initialized to bypass or access M.LR20, the ladder ramp time base entry during flight or flight simulation, respectively.

Notes

Each entry may be called once and only once by each mission defined time base module to initiate the respective time base.

Interrupt protection must be in effect upon entry to this module. It is the responsibility of the calling module to provide the required interrupt protection during the execution of the Time Base Set module.

M.T800



2.63 TIME BASE 1 (TB1)

Function

This module initiates Time Base 1 when accessed by the Liftoff Search module or when entered on a Timer 2 interrupt scheduled by Events Processor in the repeatable Flight Simulation Mode.

Entries

M.TB15 Entered by the Liftoff Search module when the liftoff discrete is detected, or from Events Processor in repeatable Flight Simulation Mode.

External

M.TB00 Time Base Set

U.D010 Discrete Output Register Set/Reset Utility

E.ENBL Enable interrupts

Input Data

D.GTBM Minor Loop entry HPC

D.VTI Time from GRR that the current time base was set

Output Data

D.TBID Time base indicator

D.KT1 Time Base 1 start time from GRR

D.GML Minor Loop entry HOP constant used by the Timer 1 Scheduler

D.GTBD Discrete Processor time base HOP constant

*

Status Words

D.VMC5 Mode Code 25

Telemetry

None.

Initial Conditions

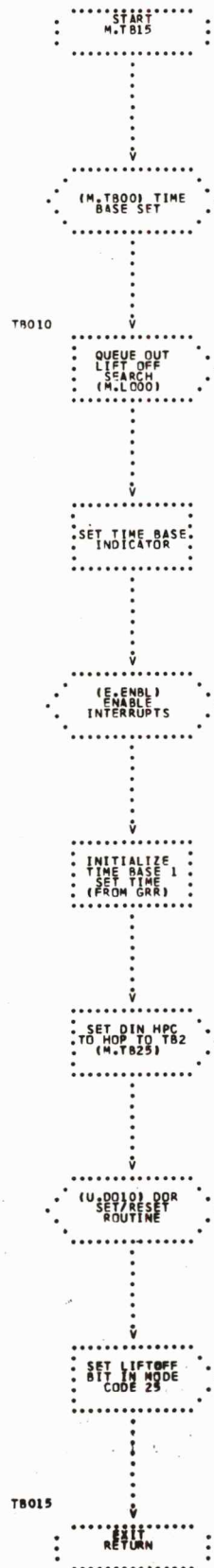
None.

Notes

The Time Base 1 module alters the Timer 1 HOP constant value, D.GML, to M.ML00 to deactivate the Liftoff Search module and to activate the Minor Loop module when Time Base 1 is initiated.

TIME BASE 1 (TB1)

M.TB15



Function

The Time Base 2 module will initiate Time Base 2 on the occurrence of either the Low Level Sensors Dry "A" (INT2) or the Low Level Sensors Dry "B" (INT6) in Flight Mode, or when entered on a Timer 2 interrupt scheduled by Events Processor in the Repeatable Flight Simulation Mode.

Entries

- M.TB20 Entered by the Interrupt Processor when the Low Level Sensors Dry "A" Interrupt is detected.
- M.TB25 Entered from Timer 2 when scheduled by the Events Processor module in Repeatable Sim Flight.
- M.TB27 Entered by the Interrupt Processor when the Low Level Sensors Dry "B" Interrupt is detected.

External

- M.TB00 Time Base Set
- U.TR00 Real Time Telemetry Delay Utility
- E.ENBL Enable interrupts
- E.IHBT Inhibit interrupts
- E.MASK Inhibit INT2 and INT6 processing

Input Data

- D.VTEX RTC reading at the time the external interrupt was recognized
- D.VTI Time from GRR that the current time base was set *
- D.VZDS Z component of velocity vector, in S-system *

Output Data

- D.KT2 Time Base 2 start time from GRR *

D.VTRS RTC reading to be used by the Control Program Time Update routine
 at a new time base start

D.TBID Time Base indicator

D.GTBD Discrete Processor time base HOP constant

D.VTBI Indication of Time Base Initiation

Status Words

D.VMC5 Mode Code 25

D.VMC7 Mode Code 27

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.TBBU	0465	Time Base 2 Repeatable telemetry
D.VINT	0441	Interrupt 2 telemetry
D.VINT	0441	Interrupt 6 telemetry

Initial Conditions

None.

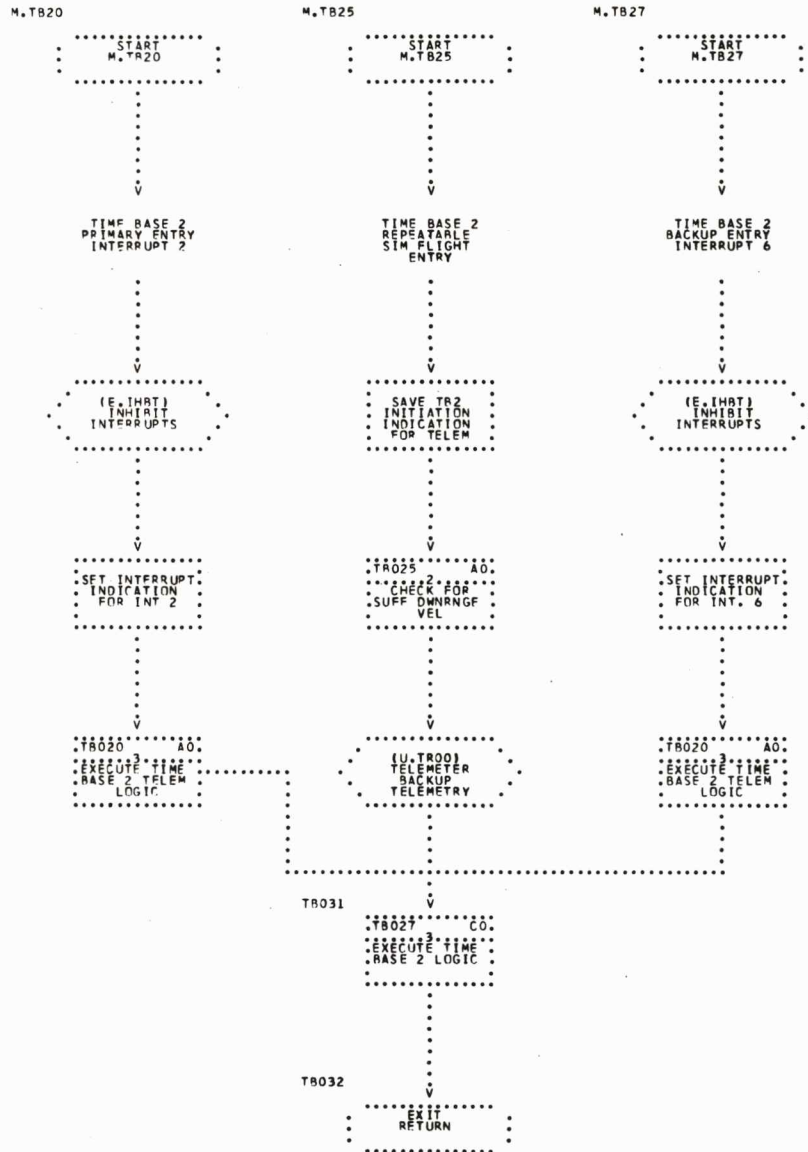
Notes

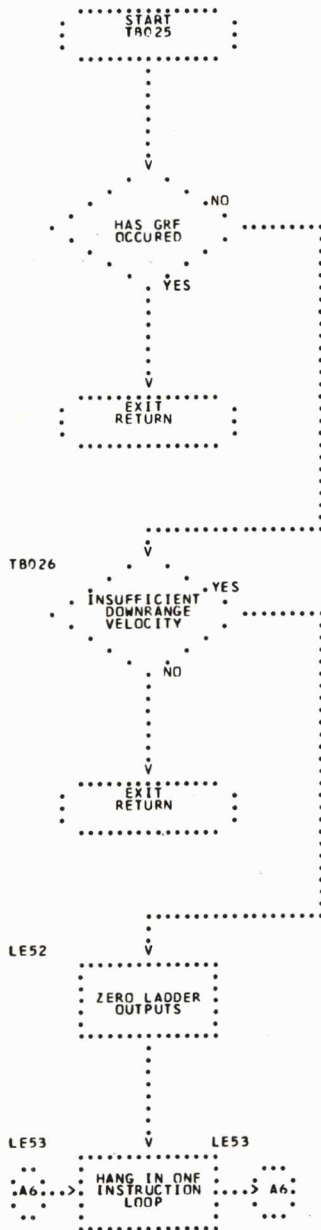
The downrange velocity reading test module provides a safeguard against starting Time Base 2 on the pad should Time Base 1 be started without liftoff. Furthermore, if Time Base 2 is not established, no subsequent time bases can be started. If guidance reference failure has occurred, this test will be bypassed and Time Base 2 started upon recognition of either INT2 or INT6.

*

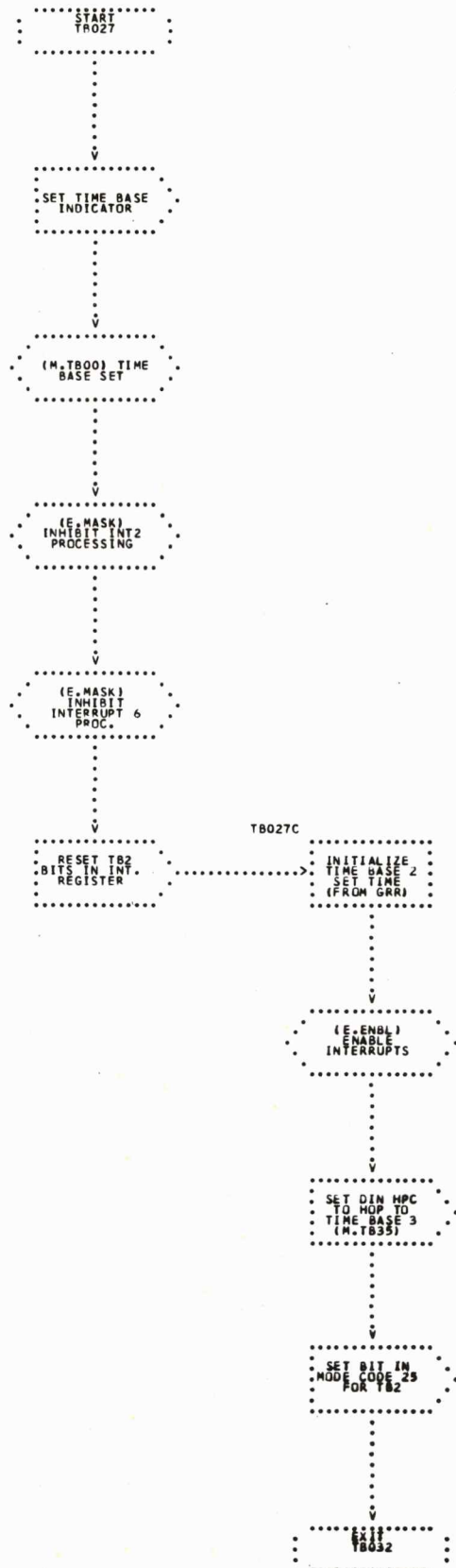
Once entered, this module will inhibit the Low Level Sense Interrupt for the remainder of the mission.

TIME BASE 2 (TB2)





TIME BASE 2 (TB2)



(THIS PAGE HAS BEEN LEFT BLANK INTENTIONALLY)

2.65 TIME BASE 3 (TB3)

*

Function

This module will initiate Time Base 3 when either the S-IB Outboard Engines Cutoff "A" (INT 5) or S-IB Outboard Engines Cutoff "B" (DI23) occurs, or if a preset time in Time Base 2 elapses. In the Repeatable Flight Simulation * Mode, it will initiate Time Base 3 on the occurrence of a scheduled Timer 2 interrupt.

Entries

M.TB30 Entered from the Interrupt Processor on occurrence of the S-IB Outboard Engine Out interrupt (INT5).
M.TB35 Entered from Discrete Processor or a Timer 2 in flight mode, or a Timer 2 in Flight Simulation Mode.

External

M.TB00 Time Base Set
E.MASK Inhibit interrupt 5 processing
E.IHBT Inhibit interrupts
E.ENBL Enable interrupts
U.TR00 Real Time Telemetry Delay Utility

*

Input Data

D.VTEX RTC reading at the time the external interrupt was recognized
D.VTI Time from GRR that the current time base was set
F.T3BU Time Backup Entry Flag

*

Output Data

D.KT3 Time Base 3 start time from GRR
D.VTRS RTC reading to be used by the Control Program Time Update routine at a new time base start

*

D.TBID	Time base indicator	
D.GTBD	Discrete Processor time base HOP constant	
D.VTBI	Indication of Time Base Initiation	
D.VDPT	Mask word which specifies which time base discretes are to be processed	
D.VDPM	Mask word which specifies which DI's are to be processed when they change from OFF to ON	* *

Status Words

D.VMC5 Mode Code 25

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.VINT	0441	Time Base 3 interrupt telemetry
D.TBBU	0465	Time Base 3 discrete backup telemetry
V.T3TW	0475	Time Base 3 time backup telemetry

Initial Conditions

None.

Notes

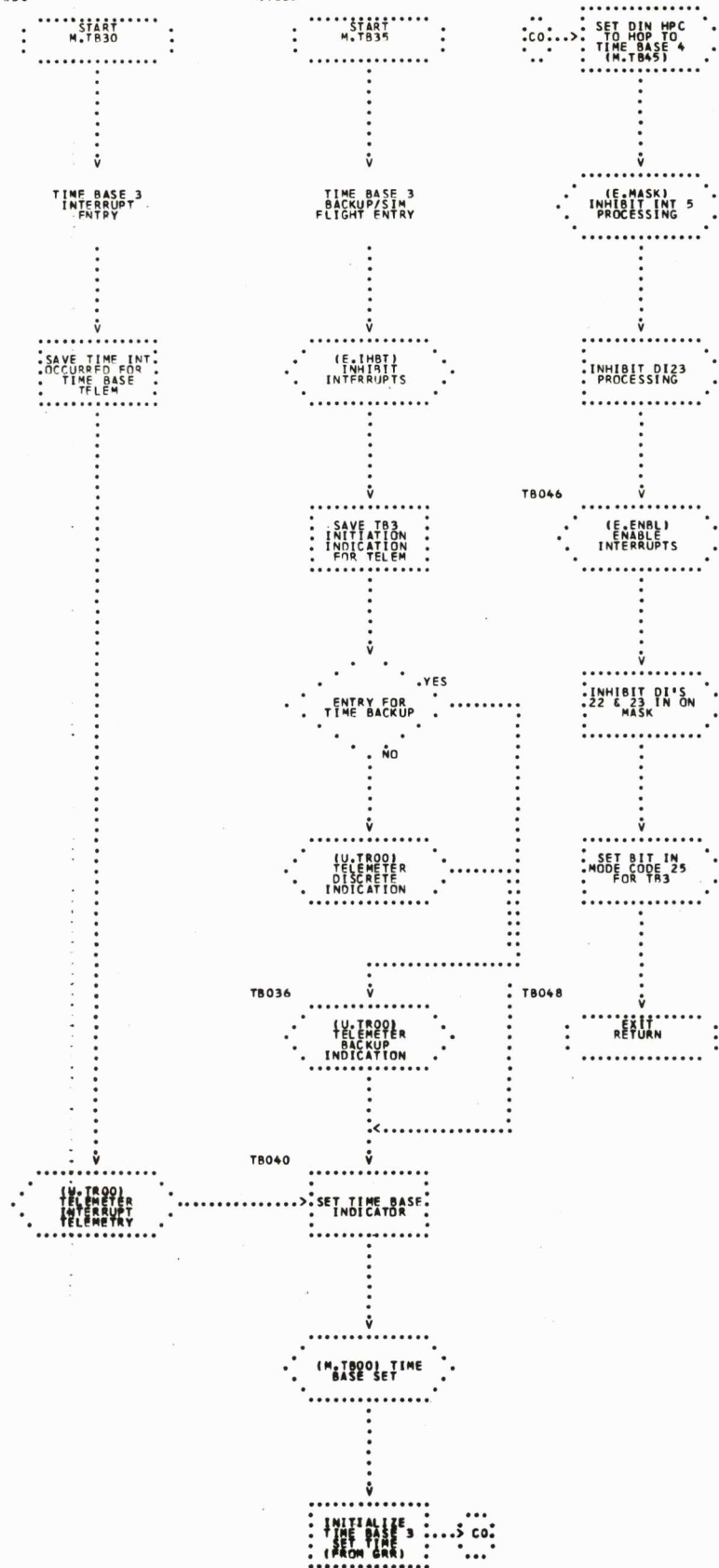
This module cannot be executed until the S-IB Propellant Depletion Interrupt (INT5) inhibit is removed. This inhibit will only be removed when the S-IB Outboard Engine Cutoff Enable switch selector function has been issued.

Once entered, this module will inhibit the S-IB Propellant Depletion interrupt for the remainder of the mission.

TIME BASF 3 (TB3)

M.TB30

M.TB35



2.66 TIME BASE 4 (TB4)

Function

This module will check S-IVB cutoff conditions necessary to initiate Time Base 4. When sufficient conditions have occurred, Time Base 4 will be initiated.

Entries

M.TB40 Entered from the Interrupt Processor on occurrence of the S-IVB Engine Out interrupt (INT4).

M.TB45 Entered by the Discrete Processor when the S-IVB Engine Out discrete (DI5) is detected. Entered in the repeatable Flight Simulation Mode for Timer 2 when scheduled by Events Processor module.

M.TB47 Entered from the Non-Interrupt Sequencer.

External

M.TB00 Time Base Set

U.TR00 Real Time Telemetry Delay Utility

E.ENBL Enable interrupts

E.IHBT Inhibit interrupts

E.Q Time Base 4 (M.TB47)

E.MASK Inhibit interrupt 4 processing

E.Q S-IVB Cutoff Prediction (M.HS00)

Input Data

D.VTEX RTC reading at the time the external interrupt was recognized.

D.VTI Time from GRR that the current time base was set *

D.VVSQ Sum of the squares of X, Y, and Z accelerometer changes

D.VDPT Mask word which specifies which time base discretes are to be processed

Output Data

D.VDPT Mask word which specifies which time base discretes are to be processed

D.VTRS RTC reading to be used by the Control Program Update routine at a new time base start

D.TBID Time base indicator

D.KT4 Time Base 4 start time from GRR *

D.GMLC Cutoff logic HOP constant (M.C000)

D.VTBI Indication of Time Base Initiation

Status Words

D.VMC5 Mode Code 25

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.VINT	0441	Interrupt 4 telemetry
D.VDIT	0465	DI 5 telemetry

Initial Conditions

None.

Notes

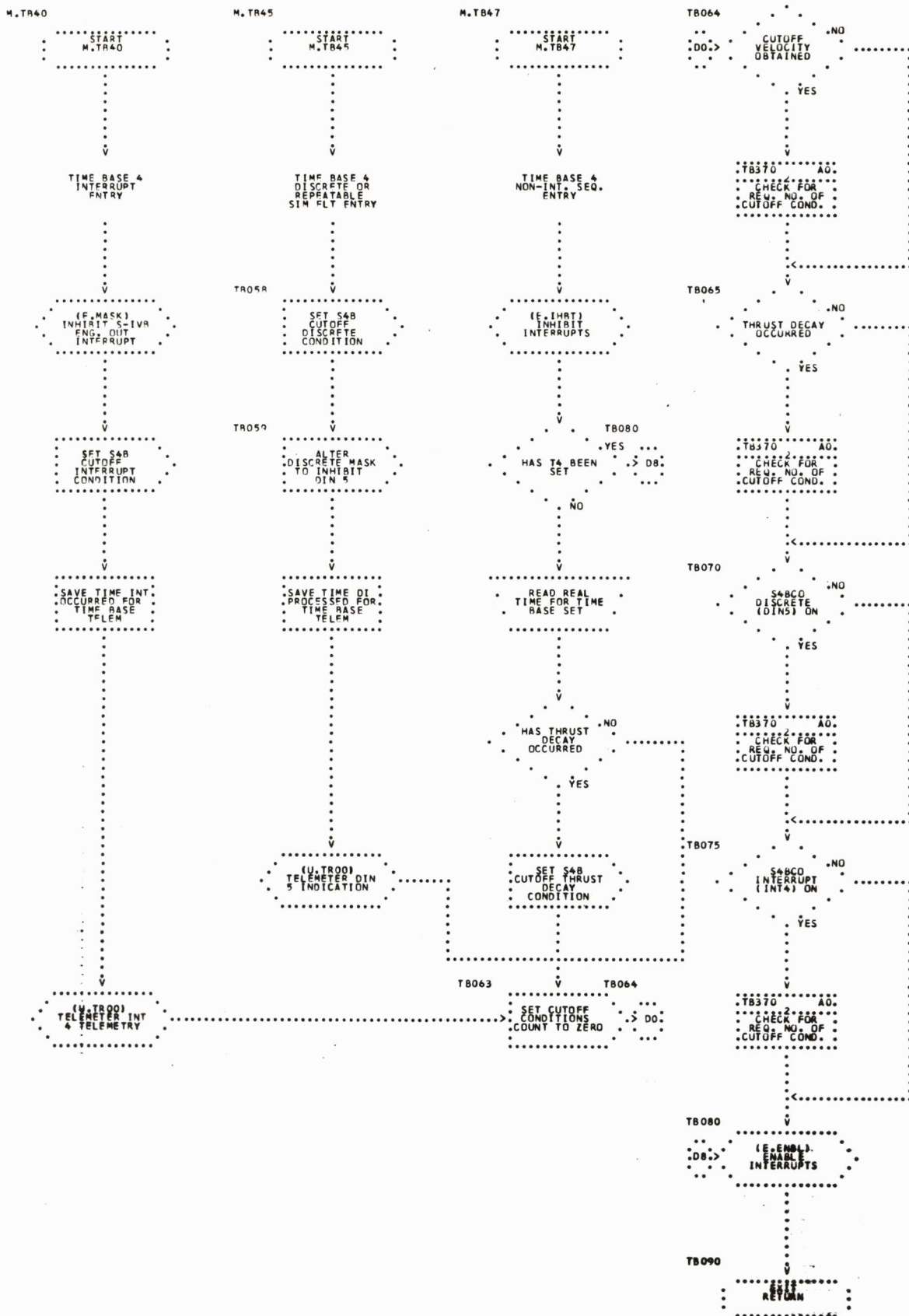
All entries to this module cannot be executed until the S-IVB Engine Out interrupt (INT4) and S-IVB Engine Out "A" discrete (DI5) inhibits are

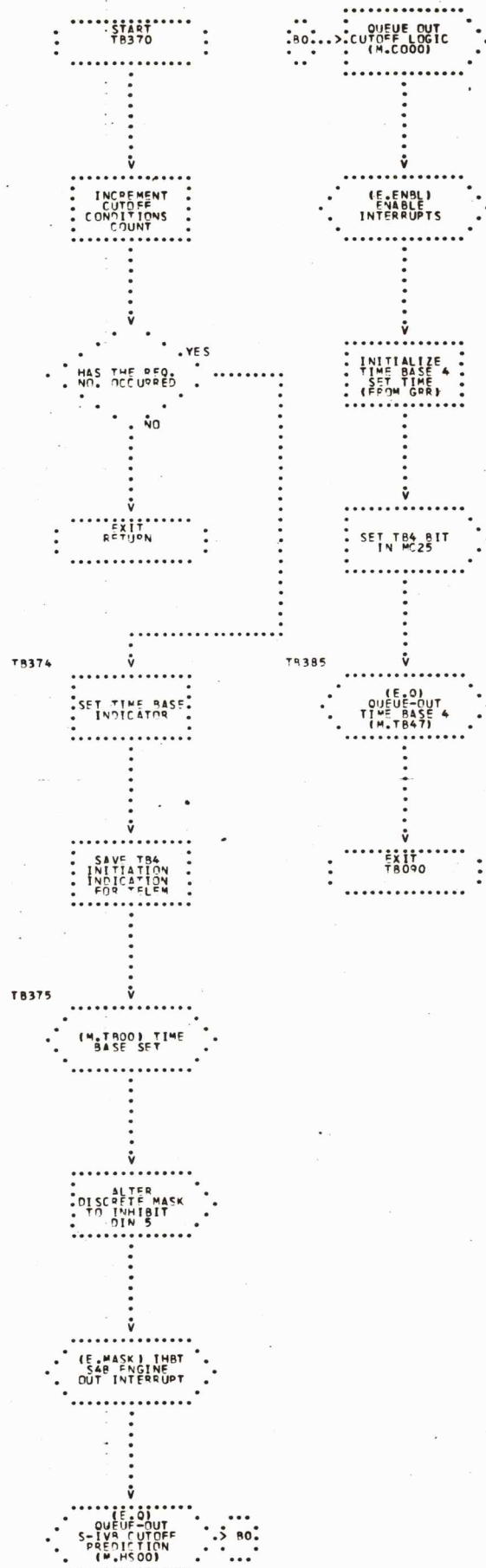
removed and the Non-Interrupt Sequencer entry (M.TB47) is queued-in. This will be accomplished at a mission defined time in Time Base 3.

Once this module initiates the appropriate time base, it will inhibit the S-IVB Engine Out interrupt and discrete and queue-out the backup entry M.TB47.

Interrupt protection must be in effect upon entry to this module at M.TB45.

TIME BASE 4 (TB-4)





2.66.1 TIME BASE 5 (TB5)

Function

This module will start Time Base 5 when the time in Time Base 4 exceeds the time, T_{DSS} , included in the S-IVB/IU De-Orbit DCS command. This module has no backup for the DCS command, and, in the absence of an acceptable command, will not be executed.

Entries

M.TB50 Entered from Timer 2 when scheduled by the DCS S-IVB/IU De-Orbit Command.

External

M.TB00 Time Base Set
U.SSRQ Request LOX Dump Start
U.SSRQ Request Safing Sequence
E.IHBT Inhibit Interrupts
E.ENBL Enable Interrupts
E.Q Time Base 5 (M.TB50)
E.QT Alternate Sequence Requester (M.AS00)
E.Q Alternate Sequence Requester (M.AS00)
E.Q Nominal Maneuver Processor (M.MP50)
E.Q DCS Commanded Maneuver Processor (M.MP55)

Input Data

D.VTI	Time from GRR that the current time base was set	*
D.VTLD	Duration time of LOX dump included in De-Orbit command	
D.VTHD	Duration time of LH2 dump included in De-Orbit command	
F.MANT	Maneuver type flag of current maneuver in progress	
V.RAM	Inertial Roll angle in pirads	*

V.SPM	Inertial Pitch angle in pirads or sine of pitch delta from local horizontal	*
		*
V.SYW	Inertial Yaw angle in pirads or sine of yaw delta from local horizontal	*
		*
V.CPM	Cosine of pitch delta from local horizontal	*
V.CYW	Cosine of yaw delta from local horizontal	*

Output Data

D.VTRS	RTC reading to be used by the Control Program Update routine at a new time base start	
D.VTBI	Indication of Time Base initiation	
D.TBID	Time Base indicator	
D.KT5	Time Base 5 start time from GRR	*
G.OV00	Orbit Velocity HOP constant (M.OV00)	*
G.GL00	Acquisition Gain/Loss HOP constant (M.GL00)	
D.ELVR	Delta velocity check allow flag	
G.AC00	APS attitude control test HOP constant (M.AC00)	
D.VXDM	X component of measured velocity	*
D.VYDM	Y component of measured velocity	*
D.VZDM	Z component of measured velocity	*
D.YREF	Inertial Y-axis reference angle for commanded maneuver	*
D.ZREF	Inertial Z-axis reference angle for commanded maneuver	*
D.XREF	Inertial X-axis reference angle for commanded maneuver	*
F.MTYP	Commanded maneuver type flag	*
F.TSOM	Time to implement commanded maneuver	*
V.MV6	Maneuver type flag for Time Base 5 maneuver	
V.MV6+1	Roll maneuver parameter for Time Base 5 maneuver	
V.MV6+2	Pitch maneuver parameter for Time Base 5 maneuver	
V.MV6+3	Yaw maneuver parameter for Time Base 5 maneuver	
V.MV6+4	Pitch maneuver parameter for Time Base 5 maneuver	
V.MV6+5	Yaw maneuver parameter for Time Base 5 maneuver	
G.DOGM	Dump dependent event table time	
D.VACX	X accelerometer reading	
D.VACY	Y accelerometer reading	

D.VACZ Z accelerometer reading
D.VDPM Mask word which indicates which DIN's are to be processed
 when they change from OFF to ON
V.DOFF Mask word which indicates which DIN's are to be processed
 when they change from ON to OFF

Status Words

D.VMC7 Mode Code 27
D.VLNK DCS Status Word

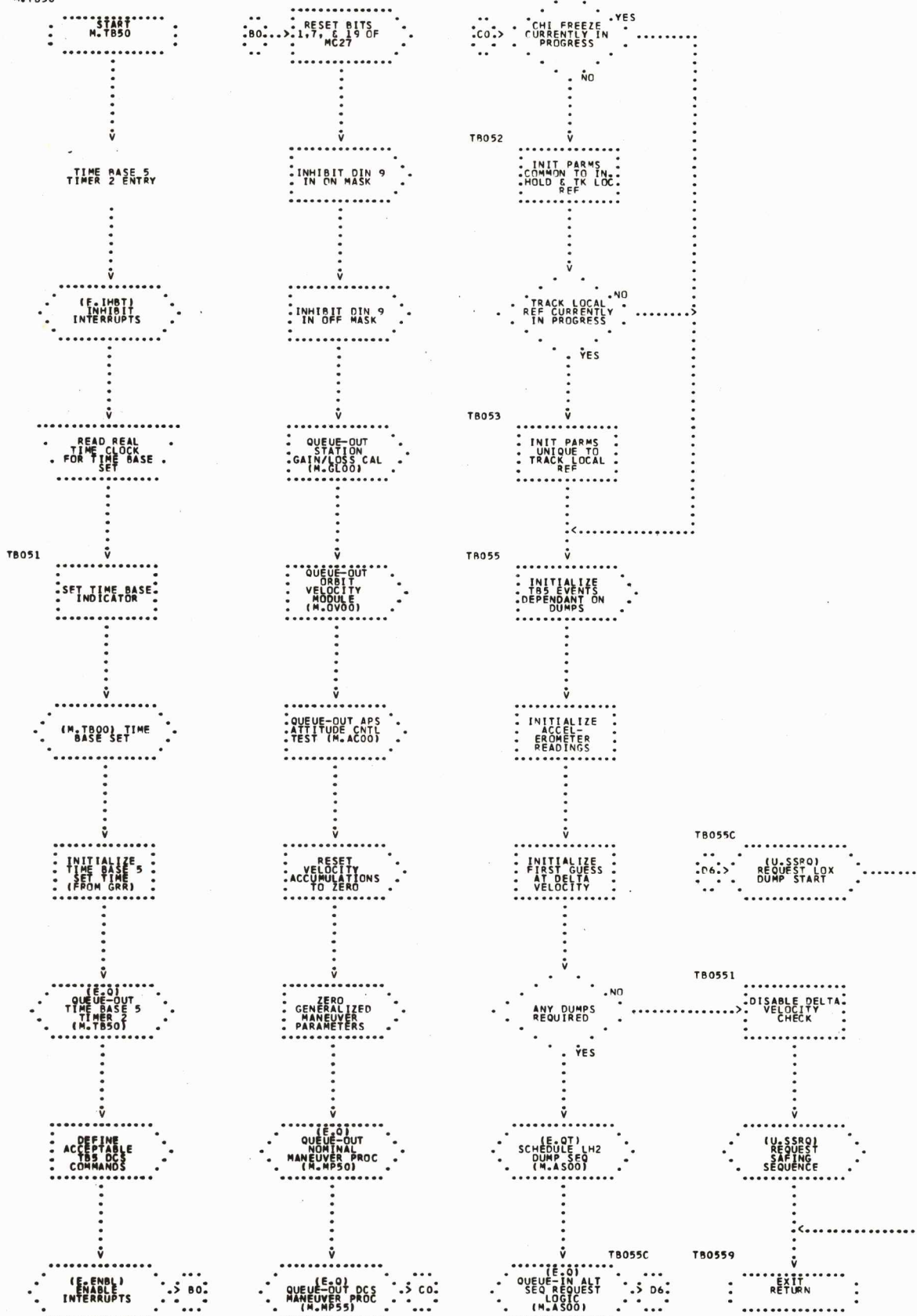
Telemetry

None.

Notes

This module will request the LOX dump start alternate sequence if the transmitted parameter T_{LDD} is non zero. If T_{LDD} is zero, this module will request the safing sequence and no other Time Base 5 alternate sequences will be requested.

M.TB50



2.67 REAL TIME TELEMETRY DELAY UTILITY (TR)

Function

The function of the Real Time Telemetry Delay Utility module is to guarantee that telemetry items are not output faster than the maximum rate allowed. When the time for telemetry occurs, the module sets the LVDA Mode Register to the required bit configuration and returns to the calling program module which immediately issues the telemetry.

Entries

U.TR00 Entered by any module.

U.TR10 Dummy entry to bypass minor loop error telemetry
if guidance failure occurs.

*

*

External

None.

Input Data

None.

Output Data

D.VTD Time of last telemetry output

Status Words

None.

Telemetry

None.

Initial Conditions

Time of last telemetry output (D.VTD) must be initialized to zero at start of mission.

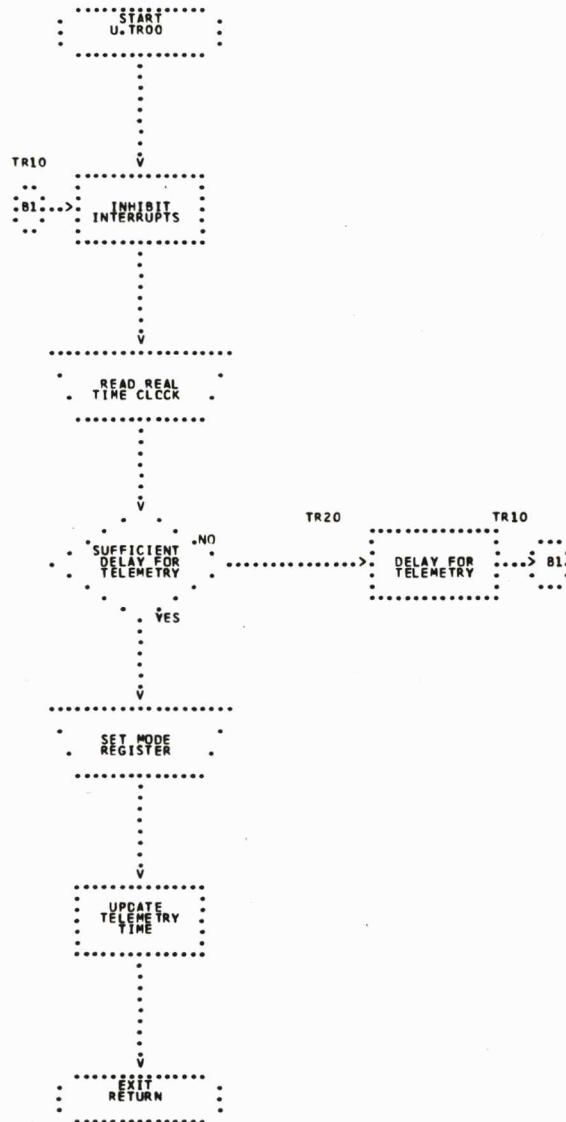
Notes

This module should be accessed only through the telemetry macro, TELM, to assure the entry and return requirements of the module are met.

If the TELM macro is not used, the data sector containing the required mode register bit configuration for the item to be telemetered must be in effect on entry to this module. Upon return the next two instructions executed in the calling program module must be CLA and PIO to complete the telemetry operation while interrupt protection is still in effect.

All interrupts are inhibited (MPY instruction) during the test for telemetry output time. When time for telemetry occurs, interrupt protection is extended (DIV instruction) over the total telemetry output operation. This includes the first two instructions executed in the calling module upon return. Interrupts are periodically allowed during the delay loop.

REAL-TIME TELEMETRY DELAY UTILITY (TR)



2.68 SIMULTANEOUS MEMORY ERROR RECOVERY (TS)

Function

This module will attempt recovery from a Simultaneous Memory Error (TLC) interrupt and provide pertinent data for analysis.

Entries

M.TS00 Entered from the Interrupt Processor on occurrence of the TLC interrupt (INT9).

External

M.PA00 Phase Activator
M.PC80 Data Compression Store Utility
M.SS20 Switch Selector Processing
E.GP07 Timer 2 Halt and Reschedule
E.GP08 Timer 1 Halt and Reschedule

Input Data

D.VTEX	Time of external interrupt	
G.L2R1	Interrupt Processor return HOP constant	
D.FPHC	Phase change in progress indicator	
F.SSAC	Switch Selector active flag	
D.VDCT	Time from GRR (in seconds positioned for data compression)	*
D.GTB4	HOP constant to Data Compression Table 4	
V.BTC	Computation cycle count	
D.FPHS	Phase indicator flag	*
G.NSCT	Non-Interrupt sequencer control table pointer	
D.VTMM	Elapsed time in mission from GRR to last time update	*
D.VMLT	Execution time of next Minor Loop	*

G.L2R1	Interrupt Processor return HOP constant
D.VTD	Time of last telemetry output
V.BTC	Computation cycle count
D.VMLT	Execution time of next Minor Loop

D.VEMR	Error Monitor Word
D.VMC5	Mode Code 25
D.VICR	Internal Control Register
D.VDOR	Discrete Output Register
D.FIL1	Interrupt Level 1 in progress flag

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>
D.VTEX	0441	TLC code and RTC
G.L2R1	0461	TLC HOP save

None.

This module resets the Interrupt Processor's HOP constant to return to the beginning of the Non-Interrupt Sequencer table, terminates any switch selector in progress, and halts and reschedules Timer 1 and Timer 2. It returns to the Interrupt Processor unless the TLC occurred during a phase change in which case it exists to the Phase Activator to restart phase initialization.



2.69 TIME TILT GUIDANCE (TT)

Function

The function of the Time Tilt Guidance Module is to generate steering commands (guidance chi's) as a function of time from liftoff through the S-IB burn.

Entries

M.TT00 Entered from the Non-Interrupt Sequencer.

External

M.CC10 Chi Computations

U.FD00 Fixed Point Divide Utility

*

E.Q Chi Computations (M.CC10)

E.Q Time Tilt Guidance (M.TT00)

Input Data

D.VTAS Total time elapsed in mission

*

D.VEOT Time of S-IB engine out

*

D.KT1 Time Base 1 start time from GRR

*

D.VXS X component of radius vector, in S-system

*

D.GTRY Distance from geocentric radius to top of gantry

Output Data

D.VGCY Guidance Chi Y (Desired pitch gimbal angle)

*

D.VGCZ Guidance Chi Z (Desired yaw gimbal angle)

*

Status Words

D.VMC5 Mode Code 25

Telemetry

None.

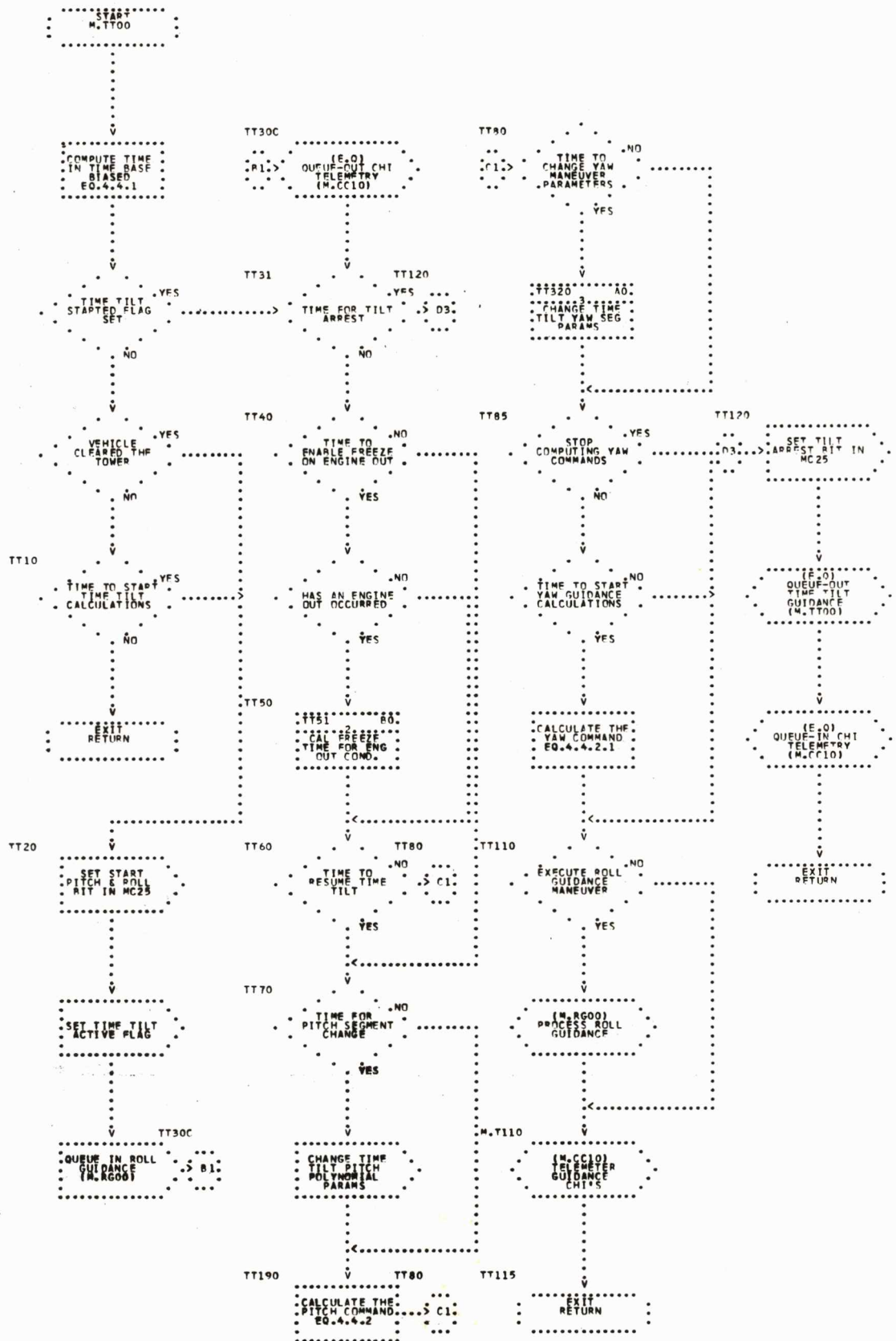
Initial Conditions

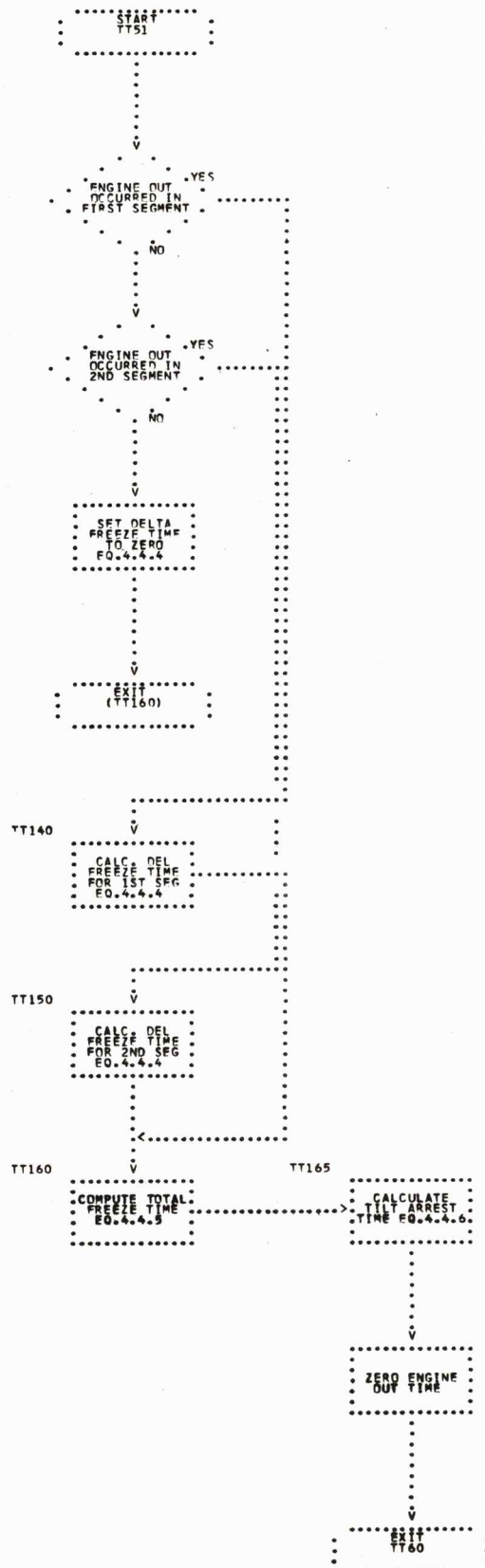
D.VTAR, V.TTST, and V.TTF must be initialized to pre-stored values at mission start.

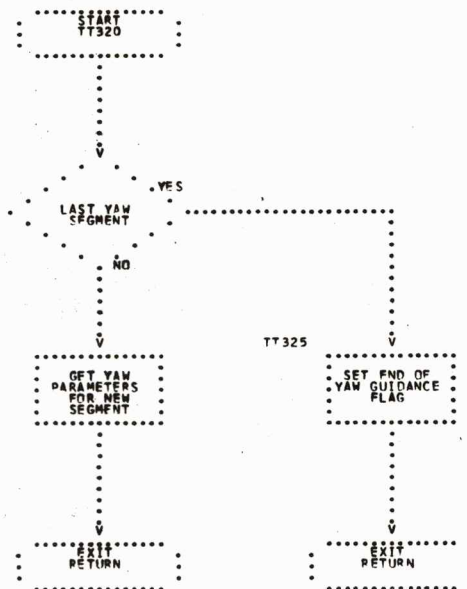
D.VEOT and F.TTA must be initialized to zero at mission start.

Notes

On indication from the Discrete Processor module that engine out has occurred, the Time Tilt Guidance module will freeze guidance chi's for a specified time, and then continue normal time tilt guidance. Variable scaling is incorporated through the use of pseudo-variables. Initialization of the various pitch time tilt segments and yaw time tilt segments for the Time Tilt Guidance module will be executed as a function of time via a time test in the Time Tilt Guidance module.







(THIS PAGE HAS BEEN LEFT BLANK INTENTIONALLY)

2.70 PHASE II TIME UPDATE (TU)

Function

This module will access the Control Program Time Update routine and compute the navigation entry time. It will telemeter all time quantities and form the data compression time word.

Entries

M.TU00 Entered at least once per second from the Phase II Control module.

External

E.GP20 Control Program Time Update

U.TROO Real Time Telemetry Delay Utility

E.ENBL Enable interrupts

E.IHBT Inhibit interrupts

Input Data

D.VTMM	Elapsed time in mission from GRR to last time update	*
D.VTGB	Accumulated ground bias time	
D.VRTC	Real time clock reading at last time update	*
D.VERT	Time error associated with last time update	
D.VTI	Time from GRR that the current time base was set	

Output Data

D.VACT	Real Time Clock reading associated with D.VTAS or D.VNTM	*
D.VTB	Time elapsed in current time base	*
D.VTBB	Time base biased elapsed time	
D.VNTM	Time in mission from GRR of navigation extrapolation	*
D.VDCT	Time from GRR (in seconds positioned for data compression)	*

Status Words

None.

Telemetry

<u>Item</u>	<u>PIO Tag</u>	<u>Description</u>	
D.VTI	0561	Current time base start time	
D.VTB	0031	Time base elapsed time	
D.VTBB	0030	Time base biased time	
D.VNTM	0100	Time in mission from GRR of navigation extrapolation	*
D.VACT	0174	Real Time Clock reading associated with D.VTAS	

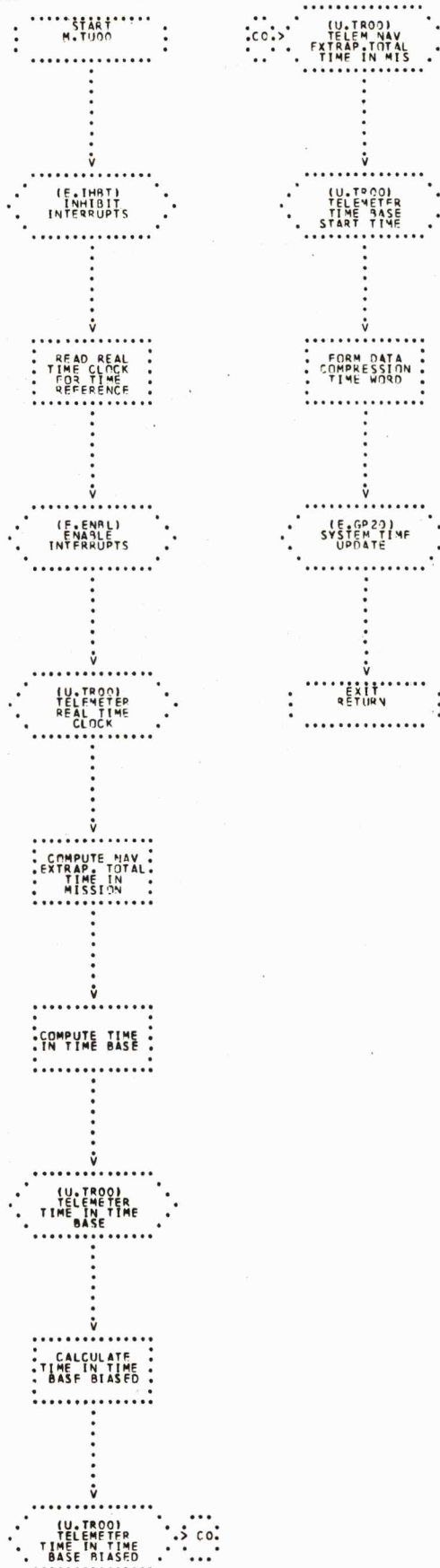
Initial Conditions

None.

Notes

The data compression time is set so that the least significant bit is worth 1/2 second and there are ten leading zeros.

M.TU00



2.71 PHASE I TIME UPDATE (UM)

Function

This module will update system time and access the Minor Loop Error Monitor module.

Entries

M.UM00 Entered from Timer 2 once every 0.8 seconds during Phase 1.

External

M.EC00 Minor Loop Error Monitor

E.GP20 Control Program Time Update

E.QT Phase I Time Update (M.UM00)

Input Data

D.VTMM Elapsed time in mission from GRR to last time update

*

Output Data

None.

Status Words

None.

Telemetry

None.

Initial Conditions

None.

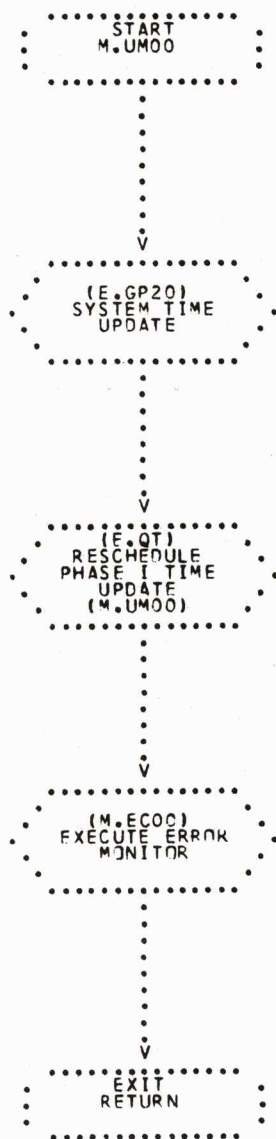
Notes

None.

PHASE I TIME UPDATE (UM)

1

M.UM00



2.72 VECTOR CROSS PRODUCT UTILITY (VC)

Function

The function of the Vector Cross Product Utility module is to compute the cross product of two specified vectors.

Entries

U.VC00 Entered by any module.

External

None.

Input Data

See Notes.

Output Data

D.VUTB+1	First component of cross product
D.VUTB+2	Second component of cross product
D.VUTB+3	Third component of cross product

Status Words

None.

Telemetry

None.

Initial Conditions

None.

Notes

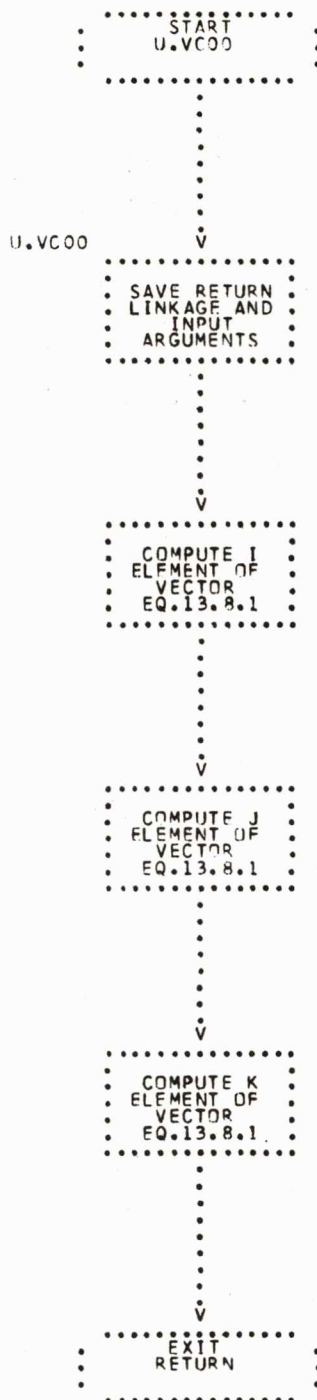
Upon entry the accumulator must contain a computer word of the form CLA A; MPY B where A and B are initial memory locations of the A and B vectors respectively.

Upon exit:

$$D.VUTB+1 = A(2) * B(3) - A(3) * B(2)$$

$$D.VUTB+2 = A(3) * B(1) - A(1) * B(3)$$

$$D.VUTB+3 = A(1) * B(2) - A(2) * B(1)$$



2.73 VECTOR DOT PRODUCT UTILITY (VD)

Function

The function of the Vector Dot Utility is to compute the dot product of two specified vectors.

Entries

U.VD00 Entered by any module.

U.VD01 Entered by the Matrix Multiply Utility module.

External

None.

Input Data

See Notes.

Output Data

See Notes.

Status Words

None.

Telemetry

None.

Initial Conditions

None.

Notes

Upon entry a computer word is to be in the accumulator with the following format:

(1) Syllable 1 = CLA P

where P is location of first of the three VEC locations for one vector

(2) Syllable 0 = MPY Q

where Q is location of first of the three VEC locations for the second vector

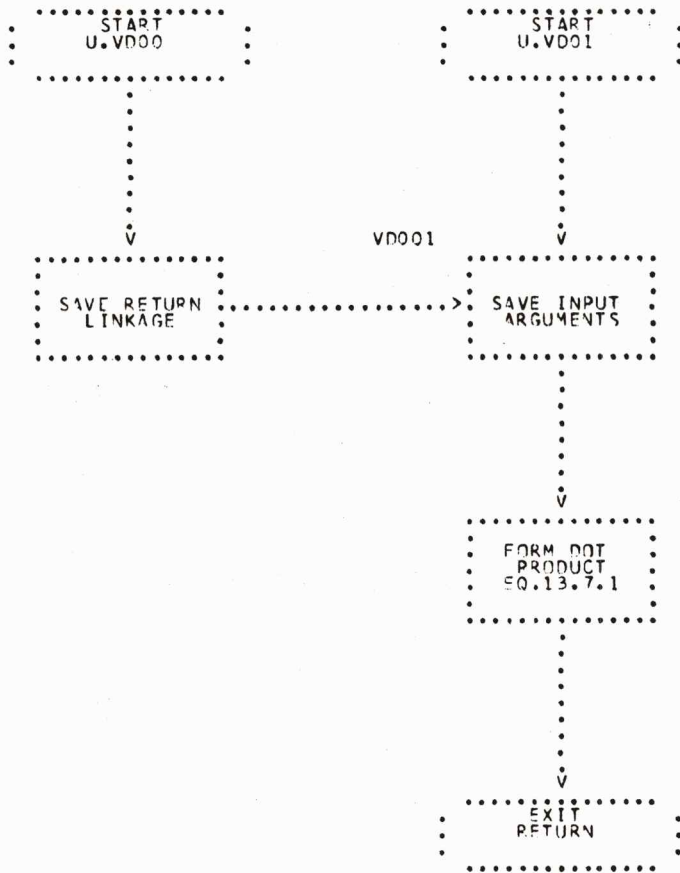
Upon exit the dot product $P(1) * Q(1) + P(2) * Q(2) + P(3) * Q(3)$ will be in the accumulator.

VECTOR DOT PRODUCT UTILITY (VD)

1

U.VD00

U.VD01



2.74 WATER VALVE PROCESSOR (WM)

Function

This module will request the issuance of the ECS Water Valve switch selectors to open or to close the water valve.

Entries

M.WM00 Entered initially from Timer 2 at GRR + mission dependent time and subsequently entered at fixed time intervals.

External

E.QT Water Valve Processor (M.WM00) *

U.SSRQ Switch Selector to close ECS Water Valve (M.SS00)

U.SSRQ Switch Selector to open ECS Water Valve (M.SS00)

Input Data

D.FWV Flag which indicates state of water valve (open or close)

D.FLT Flight Mode indicator *

Output Data

None.

Status Words

D.VASW Switch Selector Request Status

Telemetry

None.

Initial Conditions

D.FWV must be initially set to indicate a closed water valve.

Notes

This module is inhibited until GRR + mission dependent time at which time the discretes are read and the next entry into the module is scheduled to occur 300 seconds from now.

If the water valve is already in the desired state, no switch selector is requested.

If a DCS Command has inhibited issuance of the ECS Water Valve switch selectors, no switch selector is requested.

```

graph TD
    WM000[START  
M.WM00] --> WM010{E.O.T  
SCHEDULE NFXT  
WATER  
METHOD  
MODULE ENTRY}
    WM010 --> WM010_1{ARE WE IN  
REPEATABLE  
FLIGHT SIM}
    WM010_1 -- YES --> WM010_2[EXIT  
RETURN]
    WM010_1 -- NO --> WM210{U.SSRQ  
REQUEST SS  
CLOSE WATER  
VALVE}
    WM210 --> WM010_1
    WM010_2 --> WM010_3{DISCRETE 10  
OR 11 ON}
    WM010_3 -- YES --> WM220{ECS WATER  
VALVE OPEN}
    WM010_3 -- NO --> WM210
    WM220 -- YES --> WM220_1{U.SSRQ  
REQUEST SS  
OPEN WATER  
VALVE}
    WM220_1 --> WM220
    WM220 -- NO --> WM210
    WM210_1 --> WM225[EXIT  
RETURN]
    WM225 --> WM210_1
    WM210_2 --> WM200[EXIT  
RETURN]
    WM200 --> WM210_2

```

(THIS PAGE HAS BEEN LEFT BLANK INTENTIONALLY)

SECTION 3

VARIABLE CROSS REFERENCE

3.1 INTRODUCTION

Table 3-1 gives the correspondence between the symbols defined in Section 15 of Part I and the variables used in the flight program. This cross reference provides the means for relating the program listing to the equations and other requirements in Parts I and II.

Table 3-2 is a cross reference for relating the program variables and the EDD symbols in reverse of Table 3-1.

Table 3-1 Variable Cross Reference

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
\bar{A}	D.VFOM	6	m/sec ²	
A'	V.GA1	7	m/sec ²	
A''	V.GA2	6	m/sec ³	*
A _{c0}	V.ACZR	18	m/sec	*
\bar{A}_D	D.VDMX D.VDMY D.VDMZ in S-system	4	m/sec ²	
a _e	P.AE	23	meters	
A _z	D.FRAZ	0	pirads	
A _{zL}	P.F1AZ	0	pirads	*
A ₀₀	P.AP00	3	pirads	
A ₀₁	P.AP01	5		
A ₁₀	P.AP10	3		
A ₁₁	P.AP11	5	pirads ⁻¹	
A ₁	V.A1		ladder units	
A ₂	V.A2		ladder units	
A ₃	V.A3		ladder units	
A ₄	V.A4		ladder units	
A ₅	V.A5		ladder units	
A ₆	P.CPL		ladder units	*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
b	P.B	15	meters	
B ₁₁	P.B11	1		
B ₁₂	P.B12	8	seconds	
B ₂₁	P.B21	1		
B ₂₂	P.B22	8	seconds	
BIAS	V.BIAS	8	m/sec	
BIAS _X	D.VDX	18	m/sec	*
BIAS _Y	D.VDY	18	m/sec	*
BIAS _Z	D.VDZ	18	m/sec	*
BIAS1	P.B1A1	8	m/sec	
BIAS2	P.B1A2	8	m/sec	
BIAS3	P.B1A3	8	m/sec	
BIAS4	P.B1A4	8	m/sec	
BIAS5	P.B1A5	8	m/sec	
BIAS6	P.B1A6	8	m/sec	
BN ₁	P.BN1	10	seconds	
BN ₅	Events Table	27	seconds	*
BU3	P.T1T	27	seconds	
BU4	Events Table	27	seconds	*
B _{X0}	V.BX0	0	m/sec ²	
B _{X01}	V.BX01	0	m/sec ²	*
B _{Y0}	V.BY0	0	m/sec ²	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
B _{Z0}	V.BZ0	0	m/sec ²	
B _{Z01}	V.BZ01	0	m/sec ²	*
\bar{C}	N/A			
C _{ar1}	P.CAR1	8	seconds	
C _D	N/A			
C _{D1}	P.CD1	4		
C _{D2}	P.CD2	4		
C _{D3}	P.CD3	4		
C _{D4}	P.CD4	4		
C _{D5}	P.CD5	4		
CKSMIB	V.CS16			
C ₂	N/A	1		
C ₄	N/A	-2		
CP _{M1}	N/A	0		*
CP _{M2}	V.MV2+4	0		*
CP _{M3}	V.MV3+4	0		*
CP _{M4}	V.MV4+4	0		*
CP _{M5}	V.MV5+4	0		*
CP _{M6}	V.MVE6+4	0		*
CP _{M7}	V.MVE7+4	0		*
CP _{M8}	V.MVE8+4	0		*
CPITCH	V.CPM	0		*
CPM45	N/A			*
CPM67	N/A			*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
CY _{M1}	N/A	0		*
CY _{M2}	V.MV2+5	0		*
CY _{M3}	V.MV3+5	0		*
CY _{M4}	V.MV4+5	0		*
CY _{M5}	V.MV5+5	0		*
CY _{M6}	V.MVE6+5	0		*
CY _{M7}	V.MVE7+5	0		*
CY _{M8}	V.MVE8+5	0		*
CYAW	V.CYW	0		
CYM45	N/A			*
CYM67	N/A			*
Cα1	V.CA1	0		
Cα2	V.CA2	0		
Cα3	V.CA3	0		
Cα4	V.CA4	0		
D	P.D	-19		
d _A	N/A		meters	*
D _P	N/A	24	meters	*
D _y	N/A	23	meters	*
DCRTC	N/A		RTC	*
DELVR	D.DLVR }	14	meters/sec	*
	D.ELVR }	6	meters/sec	*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol1</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
DPIT1	V.DPIT	0	pirads	*
DPIT2	V.EPIT	0	pirads	*
DROLL1	V.DROL	0	pirads	*
DROLL2	V.EROL	0	pirads	*
DVSQR	V.DLVP	6	m/sec	*
DYAW1	V.DYAW	0	pirads	*
DYAW2	V.EYAW	0	pirads	*
\overline{F}	D.VFOR	19	newtons	
$\overline{F'}$	V.XSP } V.YSP } V.ZSP } in S-system			
F_b	D.VFOR	19	newtons	
F'_b	D.VFRP	19	newtons	
F_{EO}	D.VEOF	0		
F'_{EO}	D.VEFP	0		
F_0	P.F0	19	newtons	
F_1	P.F1	19	newtons	
F_2	P.F2	19	newtons	
F_{2I}	P.F2I	19	newtons	
F_3	P.F3	19	newtons	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
F ₁₀	P.PA0	3	pirads	*
F ₁₁	P.PA1	- 1	pirads/sec ²	*
F ₁₂	P.PA2	- 7	pirads/sec ³	*
F ₁₃	P.PA3	-14	pirads/sec	*
F ₂₀	P.PB0	3	pirads	*
F ₂₁	P.PB1	- 1	pirads/sec ²	*
F ₂₂	P.PB2	- 7	pirads/sec ³	*
F ₂₃	P.PB3	-14	pirads/sec	*
F ₃₀	P.PC0	3	pirads	*
F ₃₁	P.PC1	- 1	pirads/sec ²	*
F ₃₂	P.PC2	- 7	pirads/sec ³	*
F ₃₃	P.PC3	-14	pirads/sec	c
(F/M)	D.VFOM	6	m/sec ²	
(F/M) _c	D.VFMC	6	m/sec ²	
(F/M) _{cx}	D.VFMX	6	m/sec ²	
(F/M) _{cy}	D.VFMY	6	m/sec ²	
(F/M) _{cz}	D.VFMZ	6	m/sec ²	
FM0	P.FOV0	6	m/sec ²	
FM2	P.FOV1	6	m/sec ²	
FM4	P.FOV2	6	m/sec ²	
$\overline{G}(\overline{G}_P)$	<div> D.VGX } D.VGY } D.VGZ } </div>	4	m/sec ²	
\overline{G}^*	N/A		m/sec ²	*
G _H	P.HGRA	6	m/sec ²	
\overline{G}_L	V.GRV	6	m/sec ²	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
\overline{G}_T	N/A	5	m/sec ²	
GANTRY	V.GTRY	23	meters	
GOMTYP	F.MTYP			
h	V.ALT	19	meters	
H	P.H	-17		
h_1	P.ALT1	19	meters	
h_2	P.ALT2	19	meters	
i	V.I	0	pirads	
i_s	N/A	0		
J	P.J	- 9		
J_p	N/A	24		
j_s	N/A			
J_y	N/A	23		
J_1	N/A	21	meters	*
J_3	N/A	20	meters	*
J'_3	N/A	23	meters	*
K_D	P.KD	-9	m ² /Kg	
K_p	N/A	-2		

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>
k_S	N/A		
K_y	N/A	-2	
K_1	N/A	1	radians
K_2	N/A	-1	rad/sec
K_3	V.K3	1	radians
K_4	N/A	-2	rad/sec
L_p	N/A	13	m/sec
L_y	N/A	13	m/sec
L'_y	N/A	13	m/sec
L_1	N/A	13	m/sec
L_3	N/A	13	m/sec
L'_3	N/A	13	m/sec
LML	P.LML	0	ladder units *
M	D.VMAS	15	Kg
M'	D.VMAS	15	Kg
M'_c	N/A	15	Kg
M'_{c1}	N/A	15	Kg

*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
\dot{M}	D.VMFR	13	Kg/sec	
\dot{M}'	D.VFMP	13	Kg/sec	
M_{cc}	D.VMAS	15	Kg	
M_p	D.VMAS	15	Kg	
M_{GR}	P.MGR	13	Kg	
M_0	P.M0	15	Kg	
\dot{M}_0	P.MD0	13	Kg/sec	*
M_1	P.M1	15	Kg	
\dot{M}_1	P.MD1	13	Kg/sec	
M_2	P.M2	15	Kg	
\dot{M}_2	P.MD2	13	Kg/sec	
\dot{M}_{2G}	P.MD2G	2	Kg/sec	*
\dot{M}_3	P.MD3	13	Kg/sec	
\dot{M}_{3G}	P.MD3G	2	Kg/sec	*
(M/F)	D.VMOF	-2	sec^2/m	
(M/F) S	D.VMFS	-2	sec^2/m	
(M/F) S1	D.VMFS	-2	sec^2/m	
(M/F) S2	D.VMS1	-2	sec^2/m	
(M/F) S3	D.VMS2	-2	sec^2/m	
(M/F) S4	D.VMS3	-2	sec^2/m	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variables</u>	<u>Scaling</u>	<u>Units</u>
(M/F) ₁	D.VMOF	-2	sec ² /m
(M/F) ₂	D.VMF1	-2	sec ² /m
(M/F) ₃	D.VMS2	-2	sec ² /m
(M/F) ₄	D.VMF3	-2	sec ² /m
(M/F) ₀	P.MVFO	-2	sec ² /m
[MBS]	D.VM(j,k) j=1,2,3 k=1,2,3	0	
[MBS _a]	V.MBA1 ⁻ V.MBA9	0	
[MEG]	N/A		
[MES]	N/A		
MFK1	P.S4K1	0	
MFK2	P.S4K2	0	
MFK3	P.S4K3	0	
MFK4	P.S4K4	0	
MFK5	P.S4K5	0	
MFK6	P.S4K6	0	
MFK7	P.S4K7	0	
MFK8	P.S4K8	0	
[MGA]	N/A		
[MG4]	N/A		
MLKRC1	V.CG10 V.CG11	16	gimbal units *
MLKRC3	V.CG20 V.CG21	16	gimbal units *
MLKRC5	V.CG30 V.CG31	16	gimbal units *

Table 3-1. Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variables</u>	<u>Scaling</u>	<u>Units</u>
MLK2	V.ML20 } V.ML21 } V.ML30 } V.ML31 }	11	gimbal units *
MLK12	V.ML10 } V.ML11 }	11	gimbal units *
MLR	D.VMLD	25	ms
ML6HUN	V.OCK	0	pirads *
[MSG]	V.MA1 - } V.MA9 }	0	
MSK5	D.VM05	25	ladder units
MSK6	D.VM06	25	ladder units
MSK16	D.VM16	25	ladder units
MSLIM1	D.VLM1	0	pirads
MSLIM2	D.VLM2	0	pirads
[MSV]	N/A		
MS04DT	D.VIMR	0	m/comp cycle
MS15DT	N/A		
MS25DT	D.VMLR	0	1/comp cycle
[MS4]	D.VAP1 - } D.VAP9 }	0	*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
[M4V]	N/A			
NTSTB2	Events Table	27	clock pulses	
NTSTB3	Events Table	27	clock pulses	
NUPTIM	D.VNUT	15	seconds	
P	V.ONP	-8		*
P _{bx}	V.PBX	1		
P _{by}	V.PBY	1		
P _{bz}	V.PBZ	1		
P _C	V.IGPC	10	seconds	
P _{CMR}	P.CO	10	seconds	*
P ₁	N/A	30		
P ₃	N/A	31		
P ₃₄	V.ONP	-3		*
PF	D.VPF	1		
PF'	D.VPFP	1		
PF ₀	D.VPFO	1		
PF ₁	D.VPFI	1		
PITCH	V.PIT	0	ladder units	*
Q _P	N/A	29		

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
Q_y	N/A	30		
Q_1	N/A	30		
Q_3	N/A	30		
$\overline{R(R_p)}$	$\left. \begin{array}{l} D.VXS \\ D.VYS \\ D.VZS \end{array} \right\}$	23	meters	
$\overline{R_L}$	P.RB	23	meters	
$\overline{R_{OG}}$	D.VOGR	23	meters	
R_{STA}	P.R	27	meters	
$\overline{R_T}$	P.COR	23	meters	*
$\overline{R_1}$	N/A			
$\overline{R_4}$	N/A	23	meters	*
RA_{M1}	N/A			*
RA_{M2}	V.MV2+1	0	pirads	*
RA_{M3}	V.MV3+1	0	pirads	*
RA_{M4}	V.MV4+1	0	pirads	*
RA_{M5}	V.MV5+1	0	pirads	*
RA_{M6}	V.MVE6+1	0	pirads	*
RA_{M7}	V.MVE7+1	0	pirads	*
RA_{M8}	V.MVE8+1	0	pirads	*
RAM45	N/A			*
RAM67	N/A			*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
ROLL	V.ROL	0	ladder units	*
ROLLA	V.RAM	0	pirads	*
ROV	P.ROV			
RTCX	D.VRCX	18	.05 m/sec ²	*
RTCY	D.VRCY	18	.05 m/sec ²	*
RTCZ	D.VRCZ	18	.05 m/sec ²	*
S	V.ONS	-19		*
S _p	N/A	24		
S _y	N/A	23		
S ₁	N/A	21		
S ₃	N/A	20		
S ₃₄	V.ONS	-6		*
SF	V.XSF V.YSF V.ZSF }	-10	pirads/gimbal units	
SIBEOB	N/A			
SMCG	P.SMG1	0	rad/sec	*
SMCY	D.VSCY	0	pirads	
SMCZ	D.VSCZ	0	pirads	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD</u> <u>Symbol</u>	<u>Program</u> <u>Variable</u>	<u>Scaling</u>	<u>Units</u>	
SP _{M1}	N/A			*
SP _{M2}	V.MV2+2	0		*
SP _{M3}	V.MV3+2	0		*
SP _{M4}	V.MV4+2	0		*
SP _{M5}	V.MV5+2	0		*
SP _{M6}	V.MVE6+2	0		*
SP _{M7}	V.MVE7+2	0		*
SP _{M8}	V.MVE8+2	0		*
SPITCH	V.SPM	0		*
SPM45	N/A			*
SPM67	N/A			*
SQ1	P.SQ1	14	m/sec	
SQ2	P.SQ2	1		
SY _{M1}	N/A			*
SY _{M2}	V.MV2+3	0		*
SY _{M3}	V.MV3+3	0		*
SY _{M4}	V.MV4+3	0		*
SY _{M5}	V.MV5+3	0		*
SY _{M6}	V.MVE6+3	0		*
SY _{M7}	V.MVE7+3	0		*
SY _{M8}	V.MVE8+3	0		*
SYAW	V.SYW	0		*
SYM45	N/A			*
SYM67	N/A			*
S4IGTM	Events Table	27	sec	*
SaCβ1	V.SC1	0		*
SaCβ2	V.SC2	0		*
SaCβ3	V.SC3	0		*
SaCβ4	V.SC4	0		*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
SaSB1	V.SS1	0		*
SaSB2	V.SS2	0		*
SaSB3	V.SS3	0		*
SaSB4	V.SS4	0		*
T*	N/A	10	seconds	
T _{ar}	D.VTAR	8	seconds	
T _{as}	D.VTAS	15	seconds	
T _{asc}	D.VTAS	15	seconds	
T _{asp}	D.VTAS	15	seconds	
T _c	N/A	1	seconds	*
T _{CO}	N/A		seconds	*
T _d	P.TDTN	15	seconds	
T _D	V.TD	15	seconds	*
T _{DAY}	V.TDAY	10	days	
T _{DSS}	D.VTDS	9	minutes	*
T _{EO}	D.VEOT	8	seconds	
T _{EVNT}	T.TEVT	15	seconds	
T _{far}	P.TFAR	8	seconds	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
T _{f1}	P.FAL1	8	seconds	
T _{f2}	P.FAL2	8	seconds	
T _{GMT}	D.PGMT	17	seconds	*
T _{GRR} (T _{GRR0})	D.GRRT	17	seconds	
T _{HDD}	D.VTHD	12	seconds	*
T _{HSL}	P.HSL1	10	seconds	
T _L	V.PLOT	17	seconds	*
T _{LDD}	D.VTLD	12	seconds	*
T _{LS}	N/A		seconds	*
T _{LWC}	V.TLWC	17	seconds	
T _{RNTL}	F.TSOM	15	seconds	
T _{RTC}	Events Table	27	seconds	*
T _{SMC}	Events Table	27	seconds	*
T _{SO}	P.TT1	15	seconds	*
T _{SOM}	F.TSOM	15	seconds	
T _{SON}	N/A		seconds	*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
T _{SS}	N/A		seconds	*
T _{S1}	P.SEG2	15	seconds	
T _{SIE0}	P.TSE0	15	seconds	*
T _{S2}	P.SEG3	15	seconds	
T _{TMA1}	V.TTM1	15	seconds	
T _{TMA2}	V.TTM2	15	seconds	
T _{TMA3}	V.TTM3	15	seconds	
T _{TMA4}	V.TTM4	15	seconds	
T _{T2}	P.TT2	15	seconds	
T _{1i}	D.VT1I	10	seconds	
T ₁₀	P.T10	10	seconds	
T _{3i}	D.VT3I	10	seconds	
T _{4M2}	V.T4M2	15	seconds	*
T _{4M3}	V.T4M3	15	seconds	*
T _{4M4}	V.T4M4	15	seconds	*
T _{4M5}	V.T4M5	15	seconds	*
T _{4M6}	V.T4M6	15	seconds	*
T _{4M7}	V.T4M7	15	seconds	*
T _{4M8}	V.T4M8	15	seconds	*
TB	D.VTB	15	seconds	
TBA	V.TBA	15	seconds	
TBB	D.VTBB	15	seconds	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
TBL	V.TBL	15	seconds	
TI	D.VTI	15	seconds	
TLCK	D.SMCK			
TLPAST	D.VACT	25	RTC	
TLTSEC	D.VNTM	15	seconds	
TM	P.TWM	27	seconds	
TMEFRZ	V.DFTR	25	RTC	
TYAW01 - TYAW25 }	D.TYAW - D.TYAW+24 }	8	seconds	*
TYP _{M1}	V.MV1			*
TYP _{M2}	V.MV2			*
TYP _{M3}	V.MV3			*
TYP _{M4}	V.MV4			*
TYP _{M5}	V.MV5			*
TYP _{M6}	V.MVE6			*
TYP _{M7}	V.MVE7			*
TYP _{M8}	V.MVE8			*
T0	N/A		seconds	*
T1	D.KT1	15	seconds	
T2	D.KT2	15	seconds	
T2 _{IECO}	Events Table	27	seconds	*
T3	D.KT3	15	seconds	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
$T3_{FM}$	Events Table	27	seconds	*
$T3_{FMC}$	Events Table	27	seconds	*
$T3_{IGM}$	Events Table	27	seconds	*
$T3_{Pu}$	Events Table	27	seconds	*
$T4$	D.KT4	15	seconds	
$T5$	D.KT5	15	seconds	*
$T5_{GRD}$	P.T5GD	27	seconds	*
U_1	N/A	37		
U_3	N/A	38		
V	D.VVS	14	m/sec	
$\bar{V}(\bar{V}_p)$	<div style="display: inline-block; vertical-align: middle;"> D.VXDS D.VYDS D.VZDS </div>	14	m/sec	
V'	D.VV1	14	m/sec	
V''	D.VV2	14	m/sec	
V_{ex1}	P.VEX1	13	m/sec	
V_{ex3}	P.VEX3	13	m/sec	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
V_{GRD}	P.VGD1	14	m/sec	
\overline{V}_r	V.VR	14	m/sec	
\overline{V}_T	D.VGVT	14	m/sec	
X_A	N/A		meters	*
X_B	N/A		meters	*
\ddot{X}_{DS}	N/A		m/sec ²	*
X_G	N/A		meters	*
\ddot{X}_{Gg}	N/A		m/sec ²	*
X_{LS}	N/A	23	meters	
\dot{X}_m	D.VXDM	14	m/sec	
\ddot{X}_m	D.VXDD	6	m/sec ²	
\dot{X}_0	P.XD0	14	m/sec	
X_P	N/A			*
X_{ref}	D.XREF	0	pirads	
X_S	D.VXS	23	meters	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
\dot{X}_S	D.VXDS	14	m/sec	
\ddot{X}_S	D.VXDD	6	m/sec ²	
\dot{X}_{Sg}	V.IGX	14	m/sec	
\ddot{X}_{Sg}	D.VGX	4	m/sec ²	
X_V	N/A	23	meters	
\dot{X}_V	N/A	14	m/sec	*
\ddot{X}_V^*	N/A	4	m/sec ²	*
\ddot{X}_{VgT}	N/A	5	m/sec ²	
X_{VT}	V.GXVT	23	meters	
\dot{X}_{VT}	N/A	14	m/sec	
X_4	V.TLX4	23	meters	
\dot{X}_4	N/A	14	m/sec	
Y_A	N/A		meters	*
Y_B	N/A		meters	*
\ddot{Y}_{DS}	N/A		m/sec ²	*
Y_G	N/A		meters	*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
\ddot{Y}_{Gg}	N/A		m/sec ²	*
Y_{LS}	N/A		meters	*
\dot{Y}_m	D.VYDM	14	m/sec	
\ddot{Y}_m	D.VYDD	6	m/sec ²	
\dot{Y}_0	N/A		m/sec ²	*
Y_p	N/A			*
Y_{ref}	D.YREF	0	pirads	
Y_S	D.VYS	23	meters	
\dot{Y}_S	D.VYDS	14	m/sec	
\ddot{Y}_S	D.VYDD	6	m/sec ²	
\ddot{Y}_{Sg}	V.IGY	14	m/sec	
\ddot{Y}_{Sg}	N/A		m/sec ²	*
Y_V	N/A	23	meters	
\dot{Y}_V	N/A	14	m/sec	
\ddot{Y}_{Vg}^*	N/A	4	m/sec ²	
\ddot{Y}_{VgT}	N/A	5	m/sec ²	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
Y_{VT}	N/A		meters	*
\dot{Y}_{VT}	N/A	14	m/sec	
Y_4	V.TLY4	23	meters	
\dot{Y}_4	N/A	14	m/sec	
YAW	V.YAW	0	ladder units	*
YAWC01- YAWC25 }	D.YAWC- D.YAWC+24 }	0	pirads	* *
YAWLIM	N/A		deg	
Z_A	N/A		meters	*
Z_B	N/A		meters	*
\ddot{Z}_{DS}	N/A		m/sec ²	*
Z_G	N/A		meters	*
\ddot{Z}_{Gg}	N/A		m/sec ²	*
Z_{LS}	N/A		meters	*
\dot{Z}_m	D.VZDM	14	m/sec	
\ddot{Z}_m	D.VZDD	6	m/sec ²	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
z_0	N/A		m/sec	*
z_p	N/A			*
z_{ref}	D.ZREF	0	pirads	
z_s	D.VZS	23	meters	
\dot{z}_s	D.VZDS	14	m/sec	
\ddot{z}_s	D.VZDD	6	m/sec ²	
\dot{z}_{sg}	V.IGZ	14	m/sec	
\ddot{z}_{sg}	N/A	4	m/sec ²	
z_v	N/A	23	meters	
\dot{z}_v	N/A	14	m/sec	
\ddot{z}_{vg}^*	N/A	4	m/sec ²	
\ddot{z}_{vgT}	N/A	5	m/sec ²	
z_{vT}	N/A		meters	*
\dot{z}_{vT}	N/A	14	m/sec	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
Z_4	V.TLZ4	23	meters	
\dot{Z}_4	N/A	14	m/sec	
α	V.CSAT	0	pirads	*
α_{ACTX}	V.ACTX	0	pirads	*
α_{ACTZ}	V.ACTZ	0	pirads	*
α_f	N/A			
α_{LH}	N/A		pirads	*
δ_2	N/A			
ΔA	D.VDAX D.VDAY D.VDAZ }	18	.05 m/sec	
ΔB	D.VDBX D.VDBY D.VDBZ }	18	.05 m/sec	
ΔL_3	N/A	13	m/sec	
$\overline{\Delta R}$	N/A		meters	*
$\overline{\Delta R}_{SAV}$	V.XDPR V.YDPR V.ZDPR }	17	meters	
ΔT	D.VDT	2	seconds	
$\Delta T'$	D.VDTP	12	pulses	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
ΔT_b	N/A		seconds	*
$\Delta T'_E$	D.VDTE	15	seconds	
ΔT_{EVNT}	N/A	15	seconds	
ΔT_f	V.DFT	8	seconds	
ΔT_N	N/A		seconds	*
ΔT_{N1}	P.CCT0	2	seconds	
ΔT_{N2}	P.CCT1	2	seconds	
ΔT_{N3}	P.CCT3	2	seconds	
ΔT_{N4}	P.CCT4	2	seconds	
ΔT_{N5}	P.CCT5	2	seconds	
ΔT_{N6}	N/A	2	seconds	
$\Delta T'_3$	N/A	10	seconds	
ΔV	N/A		pulses	*
$\overline{\Delta V}$	N/A		m/sec	*
ΔV_b	V.GDVB	14	m/sec	
ΔV_{cx}	D.VDAX } D.VDBX }	18	m/sec	*
ΔV_{cy}	D.VDAY } D.VDBY }	18	m/sec	*
ΔV_{cz}	D.VDAZ } D.VDBZ }	18	m/sec	*
ΔV_f	N/A			

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
ΔV_{fx}	D.VFX	18	m/sec	
ΔV_{fy}	D.VFY	18	m/sec	
ΔV_{fz}	D.VFZ	18	m/sec	
ΔV_{gx}	N/A		m/sec	*
ΔV_{gy}	N/A		m/sec	*
ΔV_{gz}	N/A		m/sec	*
$\overline{\Delta V}_{SAV}$	V.XDSP V.YDSP V.ZDSP	7	m/sec	
$\overline{\Delta V}_V$	N/A	14	m/sec	*
$\overline{\Delta V}'_V$	N/A	14	m/sec	*
ΔV_x	D.VXDM	14	m/sec	
ΔV_y	D.VYDM	14	m/sec	
ΔV_z	D.VZDM	14	m/sec	
ΔX	N/A		pirads	*
ΔX_{mf}	N/A		meters	*
ΔX_{Sg}	N/A		meters	*
ΔX_V	N/A	24	meters	
$\dot{\Delta X}_V$	N/A	14	m/sec	
$\dot{\Delta X}'_V$	N/A	14	m/sec	
ΔY	N/A		pirads	*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
ΔY_{mf}	N/A		meters	*
ΔY_{Sg}	N/A		meters	*
ΔY_v	N/A	23	meters	
$\dot{\Delta Y}_v$	N/A	14	m/sec	
$\dot{\Delta Y}'_v$	N/A	14	m/sec	
ΔZ	N/A		pirads	*
ΔZ_{mf}	N/A		meters	*
ΔZ_{Sg}	N/A		meters	*
ΔZ_v	N/A		meters	*
$\dot{\Delta Z}_v$	N/A	14	m/sec	
$\dot{\Delta Z}'_v$	N/A	14	m/sec	
$\Delta \theta_{xe}$	N/A		pirads	*
$\Delta \theta_{ye}$	N/A		pirads	*
$\Delta \theta_{ze}$	N/A		pirads	*
$\Delta \chi'_x$	D.VDCX	0	pirads	
$\Delta \chi'_y$	D.VDCY	0	pirads	
$\Delta \chi'_z$	D.VDCZ	0	pirads	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
ϵ_2	P.EP2	10	seconds	
θ_R	N/A		pirads	*
θ_T	V.THET	0	pirads	
θ_x	D.VTHX	0	pirads	
θ'_x	N/A		pirads	*
θ_{xa}	D.VCXA	0	pirads	
θ_{xac}	D.VTXA	0	pirads	
θ_{xe}	D.VTHX	0	pirads	
$\dot{\theta}_{xe}$	V.DELX	0	pirads/sec	*
θ_{xp}	D.VTX0	0	pirads	
θ_y	D.VTHY	0	pirads	
θ'_y	N/A		pirads	*
θ_{ya}	D.VCYA	0	pirads	
θ_{yac}	D.VTYA	0	pirads	*
θ_{ye}	D.VTHY	0	pirads	
$\dot{\theta}_{ye}$	V.DELY	0	pirads/sec	*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
θ_{yp}	D.VTY0	0	pirads	
θ_z	D.VTHZ	0	pirads	
θ'_z	N/A		pirads	*
θ_{za}	D.VCZA	0	pirads	
θ_{zac}	D.VTZA	0	pirads	
θ_{ze}	D.VTHZ	0	pirads	
$\dot{\theta}_{ze}$	V.DELZ	0	pirads/sec	*
θ_{zp}	D.VTZ0	0	pirads	
λ	V.LAMB	0	pirads	
λ_0	N/A		pirads	
$\dot{\lambda}$	V.LAMD	-15	pirads/sec	
μ	P.MU	49	m^3/sec^2	
π	P.PI	2		*
ρ	V.RHO	-22	Kg/m^3	*
ρ_c	P.RHO	-22	Kg/m^3	*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
ρ_0	P.A0	-22	Kg/m ³	
ρ_1	P.A1	-37	Kg/m ⁴	
ρ_2	P.A2	-54	Kg/m ⁵	
ρ_3	P.A3	-72	Kg/m ⁶	
ρ_4	P.A4	-91	Kg/m ⁷	
ρ_5	P.A5	-110	Kg/m ⁸	
τ	N/A		seconds	*
τ_1	V.TU1	10	seconds	
τ_{10}	P.TU10	10	seconds	
τ_3	V.TU3	10	seconds	
τ_{3K}	N/A		seconds	*
τ_{3M}	N/A		seconds	*
τ_{3N}	P.TAU3	10	seconds	
ϕ_H	P.HLAT	0	pirads	*
ϕ_i	N/A		pirads	*
$\dot{\phi}_i$	N/A		pirads/sec	*
ϕ_{iT}	N/A		pirads	*
ϕ_L	N/A		pirads	*
ϕ'_L	N/A		pirads	*
ϕ_P	N/A		pirads	*

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
ϕ_{S0}	N/A		pirads	*
ϕ_T	N/A		pirads	*
$\dot{\phi}_T$	N/A		pirads/sec	*
χ	N/A		pirads	*
$\tilde{\chi}$	N/A		pirads	*
χ_x	D.VGCX	0	pirads	
χ'_x	D.VCCX	0	pirads	
χ''_x	D.VCCX	0	pirads	*
χ'_{xp}	N/A		pirads	*
χ_y	D.VGCY	0	pirads	
$\tilde{\chi}_y$	N/A		pirads	*
χ'_y	D.VCCY	0	pirads	*
χ''_y	D.VCCY	0	pirads	*
χ'_{yp}	N/A		pirads	*
χ_{y4}	N/A		pirads	*
χ_z	D.VGCZ	0	pirads	

Table 3-1 Variable Cross Reference (Continued)

<u>EDD Symbol</u>	<u>Program Variable</u>	<u>Scaling</u>	<u>Units</u>	
\sim χ_z	N/A		pirads	*
χ'_z	D.VCCZ	0	pirads	
χ''_z	D.VCCZ	0	pirads	*
χ'_{zp}	N/A		pirads	*
χ_{z4}	N/A		pirads	*
ω_E	P.OMEG	-14	pirads/sec	

Note 1: The definition provided by the dictionary consists only of the primary event which occurs at the corresponding time. Refer to the Event Sequence Timeline Table in Part II.

Table 3-2 Program Variable Cross Reference

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
D.DLVR	DELVR	14	m/sec	*
D.ELVR	DELVR	6	m/sec	*
D.FRAZ	A_z	0	pirads	
D.GRRT	$T_{GRR}(T_{GRR0})$	17	seconds	*
D.KT1	T1	15	seconds	
D.KT2	T2	15	seconds	
D.KT3	T3	15	seconds	
D.KT4	T4	15	seconds	
D.KT5	T5	15	seconds	*
D.PGMT	T_{GMT}	17	seconds	*
D.SMCK	TLCK			*
D.TYAW- D.TYAW+24 }	T_{YAW01-} T_{YAW25} }	8	seconds	*
D.VACT	TLPAST	25	RTC	*
D.VAP1- D.VAP9 }	[MS4]	0		
D.VCCX	χ'_x	0	pirads	
D.VCCY	χ'_y	0	pirads	
D.VCCZ	χ'_z	0	pirads	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
D.VCXA	\ominus_{xa}	0	pirads	
D.VCYA	\ominus_{ya}	0	pirads	
D.VCZA	\ominus_{za}	0	pirads	
D.VDAX } D.VDAY } D.VDAZ }	ΔA	18	.05 m/sec	
D.VDBX } D.VDBY } D.VDBZ }	ΔB	18	.05 m/sec	
D.VDCX	$\Delta x'_x$	0	pirads	*
D.VDCY	$\Delta x'_y$	0	pirads	*
D.VDCZ	$\Delta x'_z$	0	pirads	*
D.VDMX } D.VDMY } D.VDMZ } in S-system	\bar{A} D	4	m/sec ²	
D.VDT	ΔT	2	seconds	
D.VDTE	$\Delta T'_E$	15	seconds	
D.VDTP	$\Delta T'$	12	pulses	
D.VDX	BIASA _X	18	m/sec	
D.VDY	BIASA _Y	18	m/sec	
D.VDZ	BIASA _Z	18	m/sec	
D.VEFP	F'_{EO}	0		

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
D.VEOF	F_{EO}	0		
D.VEOT	T_{EO}	8	seconds	
D.VFMC	$(F/M)_c$	6	m/sec^2	*
D.VFMP	\dot{M}'	13	Kg/sec	
D.VFMX	$(F/M)_{cx}$	6	m/sec^2	
D.VFMY	$(F/M)_{cy}$	6	m/sec^2	
D.VFMZ	$(F/M)_{cz}$	6	m/sec^2	
D.VFOM	$\bar{A}, (F/M)$	6	m/sec^2	*
D.VFOR	\bar{F}, F_b	19	newtons	*
D.VFRP	F'_b	19	newtons	
D.VFX	ΔV_{fx}	18	m/sec	
D.VFY	ΔV_{fy}	18	m/sec	
D.VFZ	ΔV_{fz}	18	m/sec	
D.VGCX	χ_x	0	pirads	
D.VGCY	χ_y	0	pirads	
D.VGCZ	χ_z	0	pirads	
D.VGVT	\bar{V}_T	14	m/sec	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
D.VGX	\ddot{X}_{Sg}	4	m/sec ²	
D.VGX } D.VGY } D.VGZ }	$\bar{G}, (\bar{G}_P)$	4	m/sec ²	*
D.VLM1	MSLIM1	0	pirads	*
D.VLM2	MSLIM2	0	pirads	
D.VM (j, k) } j = 1, 2, 3 } k = 1, 2, 3 }	[MBS]	0		
D.VMAS	M, M', M _{cc} , M _p	15	Kg	*
D.VMFR	\dot{M}	13	Kg/sec	
D.VMFS	(M/F) _S	-2	sec ² /m	
D.VMOF	(M/F) ₁	-2	sec ² /m	*
D.VMF1	(M/F) ₂	-2	sec ² /m	
D.VMF2	(M/F) ₃	-2	sec ² /m	
D.VMF3	(M/F) ₄	-2	sec ² /m	*
D.VMFS	(M/F) _{S1}	-2	sec ² /m	
D.VMS1	(M/F) _{S2}	-2	sec ² /m	
D.VMS2	(M/F) _{S3}	-2	sec ² /m	
D.VMS3	(M/F) _{S4}	-2	sec ² /m	
D.VMLD	MLR	25	ms	
D.VMLR	MS25DT	0	1/comp cycle	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>
D.VMOF	(M/F)	-2	sec ² /m
D.VM05	MSK5	25	ladder units
D.VM06	MSK6	25	ladder units
D.VM16	MSK16	25	ladder units
D.VNTM	TLTSEC	15	sec
D.VNUT	NUPTIM	15	sec
D.VOGR	$\overline{R_{OG}}$	23	meters
D.VPF	PF	1	
D.VPFP	PF'	1	
D.VPFO	PF ₀	1	
D.VPF1	PF ₁	1	
D.VRCX	RTCX	18	.05 m/sec ² *
D.VRCY	RTCY	18	.05 m/sec ² *
D.VRCZ	RTCZ	18	.05 m/sec ² *
D.VSCG	SMCG/ π	-2	pirads/sec *
D.VSCY	SMCY	0	pirads
D.VSCZ	SMCZ	0	pirads
D.VTAR	T _{ar}	8	sec
D.VTAS	T _{as} , T _{asc} , T _{asp}	15	sec

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
D.VTB	TB	15	sec	
D.VTDS	T _{DSS}	9	minutes	*
D.VTHD	T _{HDD}	12	seconds	*
D.VTHX	θ_x, θ_{xe}	0	pirads	*
D.VTHY	θ_y, θ_{ye}	0	pirads	*
D.VTHZ	θ_z, θ_{ze}	0	pirads	*
D.VTI	TI	15	seconds	
D.VTLD	T _{LDD}	12	seconds	*
D.VTXA	θ_{xac}	0	pirads	
D.VTX0	θ_{xp}	0	pirads	
D.VTYA	θ_{yac}	0	pirads	
D.VTY0	θ_{yp}	0	pirads	
D.VTZA	θ_{zac}	0	pirads	
D.VTZ0	θ_{zp}	0	pirads	
D.VT1I	T _{1i}	10	seconds	
D.VT3I	T _{3i}	10	seconds	
D.VV1	V'	14	m/sec	
D.VV2	V''	14	m/sec	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
D.VVS	V	14	m/sec	
<div> D.VXDS D.VYDS D.VZDS </div>	$\bar{V} (\bar{V}_p)$	14	m/sec	*
<div> D.VXS D.VYS D.VZS </div>	$\bar{R} (\bar{R}_p)$	23	meters	
D.VXDD	\ddot{X}_m, \ddot{X}_s	6	m/sec ²	*
D.VXDM	$\dot{X}_m, \Delta V_x$	14	m/sec	*
D.VXDS	\dot{X}_s	14	m/sec	
D.VXS	X_s	23	meters	
D.VYDD	\ddot{Y}_m, \ddot{Y}_s	6	m/sec ²	*
D.VYDM	$\dot{Y}_m, \Delta V_y$	14	m/sec	*
D.VYDS	\dot{Y}_s	14	m/sec	
D.VYS	Y_s	23	meters	
D.VZDD	\ddot{Z}_m, \ddot{Z}_s	6	m/sec ²	*
D.VZDM	$\dot{Z}_m, \Delta V_z$	14	m/sec	*
D.VZDS	\dot{Z}_s	14	m/sec	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
D.VZS	Z_S	23	meters	
D.V1MR	MS04DT	0	minor loop/ comp cycle	
D.XREF	X_{ref}	0	pirads	
D.YAWC- D.YAWC+24 }	YAWC01- YAWC25 }	0	pirads	*
D.YREF	Y_{ref}	0	pirads	
D.ZREF	Z_{ref}	0	pirads	
F.MTYP	GOMTYP			
F.TSOM	T_{RNTL}, T_{SOM}	15	seconds	
P.AE	a_e	23	meters	
P.ALT1	h_1	19	meters	
P.ALT2	h_2	19	meters	
P.AP00	A_{00}	3	pirads	
P.AP01	A_{01}	5		
P.AP10	A_{10}	3		
P.AP11	A_{11}	5	pirads ⁻¹	
P.A0	ρ_0	-22	Kg/m ³	
P.A1	ρ_1	-37	Kg/m ⁴	*
P.A2	ρ_2	-54	Kg/m ⁵	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>
P.A3	ρ_3	-72	Kg/m ⁶
P.A4	ρ_4	-91	Kg/m ⁷
P.A5	ρ_5	-110	Kg/m ⁸
P.B	b	15	meters
P.BN1	BN ₁	10	seconds
P.BIA1	BIAS1	8	m/sec
P.BIA2	BIAS2	8	m/sec
P.BIA3	BIAS3	8	m/sec
P.BIA4	BIAS4	8	m/sec
P.BIA5	BIAS5	8	m/sec
P.BIA6	BIAS6	8	m/sec
P.B11	B ₁₁	1	seconds
P.B12	B ₁₂	8	
P.B21	B ₂₁	1	seconds
P.B22	B ₂₂	8	
P.CAR1	C _{ar1}	8	seconds
P.CCT0	ΔT_{N1}	2	seconds
P.CCT1	ΔT_{N2}	2	seconds
P.CCT3	ΔT_{N3}	2	seconds
P.CCT4	ΔT_{N4}	2	seconds
P.CCT5	ΔT_{N5}	2	seconds
P.CD1	C _{D1}	4	
P.CD2	C _{D2}	4	
P.CD3	C _{D3}	4	
P.CD4	C _{D4}	4	
P.CD5	C _{D5}	4	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
P.CLAT	ϕ_L	0	pirads	*
P.CO	P_{CMR}	10	seconds	
P.COR	\overline{R}_T	23	meters	*
P.D	D	-19		
P.CPL	V.A6		ladder units	*
P.EP2	ϵ_2	10	seconds	
P.FAL1	T_{f1}	8	seconds	
P.FAL2	T_{f2}	8	seconds	
P.F0	F_0	19	newtons	
P.FOV0	FM0	6	m/sec ²	
P.FOV1	FM2	6	m/sec ²	
P.FOV2	FM4	6	m/sec ²	
P.F1	F1	19	newtons	
P.F1AZ	A_{ZL}	0	pirads	*
P.F2I	F_{2I}	19	newtons	
P.F2	F_2	19	newtons	
P.F3	F_3	19	newtons	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
P.H	H	-17		
P.HGRA	G_H	6	m/sec^2	
P.HLAT	ϕ_H	0	pirads	*
P.HSL1	T_{HSL}	10	seconds	*
P.J	J	-9		
P.KD	K_D	-9	m^2/Kg	
P.LML	LML	0	ladder units	*
P.MD0	M_0	13	Kg/sec	
P.MD1	\dot{M}_1	13	Kg/sec	
P.MD2	\dot{M}_2	13	Kg/sec	
P.MD2G	\dot{M}_{2G}	2	Kg/sec	*
P.MD3	\dot{M}_3	13	Kg/sec	
P.MD3G	\dot{M}_{3G}	2	Kg/sec	*
P.MGR	M_{GR}	13	Kg	
P.MU	μ	49	m^3/sec^2	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
P.MVFO	$(M/F)_0$	-2	sec^2/m	
P.M0	M_0	15	Kg	
P.M1	M_1	15	Kg	
P.M2	M_2	15	Kg	
P.OMEG	ω_E	-14	pirads/sec	
P.PA0	F_{10}	3	pirads	*
P.PA1	F_{11}	-1	pirads/sec	*
P.PA2	F_{12}	-7	pirads/sec^2	*
P.PA3	F_{13}	-14	pirads/sec^3	*
P.PB0	F_{20}	3	pirads	*
P.PB1	F_{21}	-1	pirads/sec	*
P.PB2	F_{22}	-7	pirads/sec^2	*
P.PB3	F_{23}	-14	pirads/sec^3	*
P.PC0	F_{30}	3	pirads	*
P.PC1	F_{31}	-1	pirads/sec	*
P.PC2	F_{32}	-7	pirads/sec^2	*
P.PC3	F_{33}	-14	pirads/sec^3	*
P.PI	π	2		*
P.R	R_{STA}	27	meters	
P.RB	$\overline{R_L}$	23	meters	
P.RHO	ρ_c	-22	Kg/m^3	*

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
P.ROV	ROV			
P.SEG2	T_{S1}	15	sec	
P.SEG3	T_{S2}	15	sec	
P.SMG1	SMCG	0	rad/sec	*
P.SQ1	SQ1	14	m/sec	
P.SQ2	SQ2	1		
P.S4K1	MFK1	0		
P.S4K2	MFK2	0		
P.S4K3	MFK3	0		
P.S4K4	MFK4	0		
P.S4K5	MFK5	0		
P.S4K6	MFK6	0		
P.S4K7	MFK7	0		
P.S4K8	MFK8	0		
P.TAU3	τ_{3N}	10	seconds	
P.TDTN	T_d	15	seconds	
P.TFAR	T_{far}	8	seconds	
P.TSE0	T_{SIE0}	15	seconds	*
P.TT1	T_{S0}	15	seconds	*
P.TT2	T_{T2}	15	seconds	
P.TU10	τ_{10}	10	seconds	
P.TWM	TM	27	seconds	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
P.T1T	BU3	27	seconds	
P.T10	T ₁₀	10	seconds	
P.T5GD	T5 _{GRD}	27	seconds	*
P.VEX1	V _{ex1}	13	m/sec	
P.VEX3	V _{ex3}	13	m/sec	
P.VGD1	V _{GRD}	14	m/sec	
P.XD0	X ₀	14	m/sec	
T.TEVT	T _{EVNT}	15	seconds	
V.ACZR	A _{c0}	18	m/sec	*
V.ACTX	α ACTX	0	pirads	*
V.ACTZ	α ACTZ	0	pirads	*
V.ALT	h	19	meters	
V.A1	A ₁		ladder units	*
V.A2	A ₂		ladder units	*
V.A3	A ₃		ladder units	*
V.A4	A ₄		ladder units	*
V.A5	A ₅		ladder units	*
V.BIAS	BIAS	8	m/sec	
V.BX0	B _{X0}	0	m/sec ²	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
V.BX01	B_{X01}	0	m/sec^2	*
V.BY0	B_{Y0}	0	m/sec^2	
V.BZ0	B_{Z0}	0	m/sec^2	
V.BZ01	B_{Z01}	0	m/sec^2	*
V.CA1	$C_{\alpha 1}$	0		
V.CA2	$C_{\alpha 2}$	0		
V.CA3	$C_{\alpha 3}$	0		
V.CA4	$C_{\alpha 4}$	0		
V.CG10 } V.CG11 }	MLKRC1	16	gimbal units	*
V.CG20 } V.CG21 }	MLKRC3	16	gimbal units	*
V.CG30 } V.CG31 }	MLKRC5	16	gimbal units	*
V.CPM	CPITCH	0		*
V.CSAT	α	0	pirads	*
V.CS16	CKSMIB			
V.CYW	CYAW	0		
V.DCX	$\Delta\chi_x$	0	pirads	
V.DCY	$\Delta\chi_y$	0	pirads	
V.DCZ	$\Delta\chi_z$	0	pirads	
V.DELX	$\dot{\theta}_{xe}$	0	pirads/sec	*
V.DELY	$\dot{\theta}_{ye}$	0	pirads/sec	*

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>Edd Symbol</u>	<u>Scaling</u>	<u>Units</u>	
V.DELZ	$\dot{\theta}_{ze}$	0	pirads/sec	*
V.DFT	ΔT_f	8	seconds	
V.DFTR	TMEFRZ	25	RTC	
V.DLVP	DVSQR	6	m/sec	*
V.DPIT	DPIT1	0	pirads	*
V.DROL	DROLL1	0	pirads	*
V.DYAW	DYAW1	0	pirads	*
V.EPIT	DPIT2	0	pirads	*
V.EROL	DROLL2	0	pirads	*
V.EYAW	DYAW2	0	pirads	*
V.GA1	A'	7	m/sec ²	*
V.GA2	A''	6	m/sec ³	*
V.GDVB	ΔV_b	14	m/sec	
V.GRV	$\overline{G_L}$	6	m/sec ²	
V.GTRY	GANTRY	23	meters	
V.GXVT	X_{VT}	23	meters	
V.I	i	0	pirads	
V.IGPC	P_C	10	seconds	

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>
V.IGX	\dot{X}_{Sg}	14	m/sec
V.IGY	\dot{Y}_{Sg}	14	m/sec
V.IGZ	\dot{Z}_{Sg}	14	m/sec
V.K3	K_3	1	radians
V.LAMB	λ	0	pirads
V.LAMD	$\dot{\lambda}$	-15	pirads/sec
V.MA1- V.MA9 }	[MSG]	0	
V.MBA1- V.MBA9 }	[MBS _a]	0	
V.ML10 V.ML11 }	MLK12	11	gimbal units *
V.ML20 V.ML21 V.ML30 V.ML31 }	MLK2	11	gimbal units *
V.MV1	TYP _{M1}		*
V.MV2	TYP _{M2}		*
V.MV2+1	RA _{M2}	0	pirads *
V.MV2+2	SP _{M2}	0	*
V.MV2+3	SY _{M2}	0	*
V.MV2+4	CP _{M2}	0	*
V.MV2+5	CY _{M2}	0	*
V.MV3	TYP _{M3}		*

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program</u> <u>Variable</u>	<u>EDD</u> <u>Symbol</u>	<u>Scaling</u>	<u>Units</u>	
M.MV3+1	RA _{M3}	0	pirads	*
V.MV3+2	SP _{M3}	0		*
V.MV3+3	SY _{M3}	0		*
M.MV3+4	CP _{M3}	0		*
V.MV3+5	CY _{M3}	0		*
V.MV4	TYP _{M4}			*
V.MV4+1	RA _{M4}	0	pirads	*
V.MV4+2	SP _{M4}	0		*
V.MV4+3	SY _{M4}	0		*
V.MV4+4	CP _{M4}	0		*
V.MV4+5	CY _{M4}	0		*
V.MV5	TYP _{M5}			*
V.MV5+1	RA _{M5}	0	pirads	*
V.MV5+2	SP _{M5}	0		*
V.MV5+3	SY _{M5}	0		*
V.MV5+4	CP _{M5}	0		*
V.MV5+5	CY _{M5}	0		*
V.MVE6	TYP _{M6}			*
V.MVE6+1	RA _{M6}	0	pirads	*
V.MVE6+2	SP _{M6}	0		*
V.MVE6+3	SY _{M6}	0		*
V.MVE6+4	CP _{M6}	0		*
V.MVE6+5	CY _{M6}	0		*
V.MVE7	TYP _{M7}	00	pirads	*
V.MVE7+1	RA _{M7}	0		*
V.MVE7+2	SP _{M7}	0		*
V.MVE7+3	SY _{M7}	0		*
V.MVE7+4	CP _{M7}	0		*
V.MVE7+5	CY _{M7}	0		*
V.MVE8	TYP _{M8}			*
V.MVE8+1	RA _{M8}	0	pirads	*

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
V.MVE8+2	SP _{M8}	0		*
V.MVE8+3	SY _{M8}	0		*
V.MVE8+4	CP _{M8}	0		*
V.MVE8+5	CY _{M8}	0		*
V.ONP	P	-8		*
V.ONP	P ₃₄	-3		*
V.ONS	S	-19		*
V.ONS	S ₃₄	-6		*
V.PBX	P _{bx}	1		
V.PBY	P _{by}	1		
V.PBZ	P _{bz}	1		
V.PIT	PITCH	0		
V.PLOT	T' _L	17	seconds	
V.RAM	ROLLA	0	pirads	*
V.RHO	ρ	-22	Kg/m ³	*
V.ROL	ROLL	0	ladder units	*
V.SC1	S α C β 1	0		
V.SC2	S α C β 2	0		
V.SC3	S α C β 3	0		
V.SC4	S α C β 4	0		

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
V.SPM	SPITCH	0		*
V.SS1	S α S β 1	0		
V.SS2	S α S β 2	0		
V.SS3	S α S β 3	0		
V.SS4	S α S β 4	0		
V.SYW	SYAW	0		*
V.TBA	TBA	15	seconds	
V.TBL	TBL	15	seconds	
V.TD	T _D	15	seconds	
V.TDAY	T _{DAY}	10	days	*
V.THET	θ_T	0	pirads	
V.TLWC	T _{LWC}	17	seconds	
V.TLX4	X ₄	23	meters	
V.TLY4	Y ₄	23	meters	
V.TLZ4	Z ₄	23	meters	
V.TTM1	T _{TMA1}	15	seconds	*
V.TTM2	T _{TMA2}	15	seconds	*
V.TTM3	T _{TMA3}	15	seconds	*
V.TTM4	T _{TMA4}	15	seconds	*

Table 3-2 Program Variable Cross Reference (Continued)

<u>Program Variable</u>	<u>EDD Symbol</u>	<u>Scaling</u>	<u>Units</u>	
V.TU1	τ_1	10	seconds	
V.TU3	τ_3	10	seconds	
V.T4M2	T_{4M2}	15	seconds	*
V.T4M3	T_{4M3}	15	seconds	*
V.T4M4	T_{4M4}	15	seconds	*
V.T4M5	T_{4M5}	15	seconds	*
V.T4M6	T_{4M6}	15	seconds	*
V.T4M7	T_{4M7}	15	seconds	*
V.T4M8	T_{4M8}	15	seconds	*
V.VR	\bar{V}_r	14	m/sec	
V.XDPR } V.YDPR } V.ZDPR }	$\overline{\Delta R}_{SAV}$	17	meters	
V.XDSP } V.YDSP } V.ZDSP }	$\overline{\Delta V}_{SAV}$	7	m/sec	
V.XSF } V.YSF } V.ZSF }	SF	-10	pirads/ gimbal units	*
V.XSP V.YSP V.ZSP	$\overline{F'}$			* * *
V.YAW	YAW	0	ladder units	*
V.OCK	ML6HUN	0	pirads	*

SECTION 4

REQUIREMENTS/MODULE CROSS REFERENCE

4.1 INTRODUCTION

Table 4-1 shows the correspondence between the program requirements in Part I and the flight program module in which the requirement is implemented. The requirements are identified by the Part I paragraph number in which that requirement is established. Some modules (e.g., Phase Initialization) satisfy the requirements listed in several sections of Part I and thus appear more than once in the table.

The information in Table 4-1 provides a means for locating the flight program segment where a particular function is implemented.

Table 4-1 Requirement/Module Cross Reference Table

<u>REQUIREMENTS</u>	<u>MODULE</u>
<u>Section 2</u>	
2.3 Transformation Matrices	Matrix Multiply Utility (MM) Matrix Transpose Utility (MT)
<u>Section 3</u>	
3.3.2.2 Repositioning	Digital Command System (DS)
3.4 Flight Program Initialization	Phase Initialization (IN)
<u>Section 4</u>	
4.2.1 Accelerometer Data Format	Accelerometer Read (AR)
4.2.2 Accelerometer Data Processing	Accelerometer Processing (AP)
4.2.3 Velocity Accumulation	Accelerometer Processing (AP)
4.2.4 F/M Calculations	F/M Calculations (DV)
4.2.5 M/F Smoothing Calculations	M/F Smoothing (ST)
4.2.6 (F/M) _c Acceleration	Accelerometer Read (AR)
4.3 Boost Navigation	Boost Navigation (NE)
4.3.2 Gravitation Acceleration	Gravitation Acceleration (GR)

*

Table 4-1 Requirement/Module Cross Reference Table (Continued)

<u>REQUIREMENTS</u>	<u>MODULE</u>
4.4.1 First Stage Guidance	Time-Tilt Guidance (TT)
4.4.1.1 Roll Maneuver	Roll Guidance (RG)
4.4.2 Iterative Guidance Mode	Iterative Guidance Mode (IG)
4.4.2.4 Terminal Steering and Cutoff	S-IVB Cutoff Prediction (HS) Cutoff Logic (CO)
4.4.4 Steering Misalignment Correction	Steering Misalignment Correction (SM)
4.4.5 CHI Computations	CHI Computations (CC)
<u>Section 5</u>	
5.2 Orbital Processing Rates	Phase II Control (CM) Phase II Time Update (TU)
5.3 Orbital Initialization	Phase Initialization (IN)
5.4 Orbital Navigation	Orbit Navigation (NZ) Navigation Extrapolation (NI)
5.4.2 Acceleration Models	Orbit Accelerations (OA) Gravitation Acceleration (GR) *
5.4.3 Navigation Update	Digital Command System (DS) Navigation Update Implementation (NU)

Table 4-1 Requirement/Module Cross Reference Table (Continued)

<u>REQUIREMENTS</u>	<u>MODULE</u>	
5.5 Orbital Guidance	Orbit Guidance (OG)	
	Maneuver Processor (MP)	*
5.5.5 Nominal Control Switchover Capability	Discrete Input 9 (DP 3)	
5.5.7 DCS Commanded Functions	Digital Command System (DS)	
5.6 Telemetry Acquisition and Loss	Acquisition Gain/Loss (GL)	
	Compressed Data Telemetry (CT)	
	Switch Selector Processing (SS)	*
<u>Section 7</u>		
7.2.2 Time Base 1	Liftoff Search (LO)	
	Time Base 1 (TB1)	
	Time Base Set (TB)	
7.2.3 Time Base 2	Time Base 2 (TB2)	
	Time Base Set (TB)	
7.2.4 Time Base 3	Time Base 3 (TB3)	
	Time Base Set (TB)	
	Discrete Processor (DP)	
7.2.5 Time Base 4	Time Base 4 (TB4)	
	Time Base Set (TB)	
	Discrete Processor (DP)	
7.2.6 Time Base 5	Time Base 5 (TB5)	*
	Time Base Set (TB)	*
	Digital Command System (DS)	*

Table 4-1 Requirement/Module Cross Reference Table (Continued)

<u>REQUIREMENTS</u>	<u>MODULE</u>
7.3 Discrete Outputs	Discrete Output Register Set/Reset Utility (DO)
7.4 Discrete Inputs	Discrete Processor (DP) Discrete Inputs 14 and 15 (DP1) Discrete Input 9 (DP 3) Water Valve Processor (WM)
<u>Section 8</u>	
8.2 Minor Loop	Minor Loop (ML)
8.2.2.1 Disagreement Bit Processing	Disagreement Bit Processing (DG)
8.3 Minor Loop Support	Minor Loop Support (MS) Minor Loop Error Monitor (EC) APS Attitude Control Test (AC) *
<u>Section 9</u>	
9.2 Command Execution Operations	Switch Selector Processor (SS)
<u>Section 10</u>	
10.3 DCS Command Verification	Digital Command System (DS)
10.4 DCS Commands	Digital Command System (DS)
10.4.2 Navigation Update	Navigation Update Implementation *

Table 4-1 Requirement/Module Cross Reference Table (Continued)

<u>REQUIREMENTS</u>	<u>MODULE</u>	
10.4.3 Generalized Switch Selector	Switch Selector Processing (SS)	*
10.4.4 Memory Dump	Memory Dump Telemetry (DT)	
10.4.6 Execute Generalized Maneuver	Maneuver Processor (MP)	*
10.4.7 Return to Nominal Timeline	Maneuver Processor (MP)	*
10.4.12 Ladder Magnitude Limit	APS Attitude Control Test (AC)	*
10.4.13 S-IVB/IU De-orbit	Time Base 5 (TB5)	*
	Alternate Sequence Requester (AS)	*
	Switch Selector Processing (SS)	*
10.4.14 Compressed Data Dump	Compressed Data Telemetry (CT)	*
10.4.15 Remove Inhibit on the Extraction Maneuver	Maneuver Processor (MP)	*
		*
<u>Section 11</u>		
11.2 Telemetry System Interface	Real Time Telemetry Delay Utility (TR)	
11.4.2 LVDA Telemetry	Digital Output Multiplexer	*
	Telemetry (OD)	*
11.5 Data Compression	One Second Data Compression (PC01) Sixty Second Data Compression (PC60)	

Table 4-1 Requirement/Module Cross Reference Table (Continued)

<u>REQUIREMENTS</u>	<u>MODULE</u>
	Fifty Second Data Compression (PC50)
	One Hundred Second Data Compression (PC99)
	Data Compression Position (PC70)
	Compressed Data Telemetry (CT)
	Data Compression Store Utility (PC80)
 <u>Section 12</u>	
12.2 Flight Simulation	Simulated Accelerometers (SA)
	Simulated Platform Gimbal Angles (PG)
	Ladder Ramp Processor (LR)
 <u>Section 13</u>	
13.2 Sine-Cosine Algorithm	Sine Cosine Utility (SC)
13.3 Arctangent Algorithm	Arctangent Utility (AT)
13.4 Natural Logarithm Algorithm	Natural Logarithm Utility (LN)
13.5 Square Root Algorithm	Square Root Utility (SR)
13.7 Vector Dot Product	Vector Dot Product Utility (VD)
13.8 Vector Cross Product	Vector Cross Product Utility (VC)

(THIS PAGE HAS BEEN LEFT BLANK INTENTIONALLY)

PART IV

TABLE OF CONTENTS

Paragraph	Title	Page
N/A	EDD CONFIGURATION CONTROL PAGE LISTING	ii
SECTION 1:	ASTP MISSION DEPENDENT EVENTS	-- *
1.1	INTRODUCTION	IV-1-1
1.2	EVENTS PROCESSOR TIMED EVENTS TABLE	1
1.3	EVENTS PROCESSOR SUBMODULES	8

PART IV

EDD CONFIGURATION CONTROL PAGE LISTING, REVISION K

NOTE: The following Flight Program Change Requests (FPCR's) have been authorized and are incorporated in Revision K:

FPCR's 210-1, -5, -8, and -11.

(Revision K is a complete reprinting of Parts III and IV of the EDD:
Part IV is a completely new section containing only Section I.)

SECTION 1

ASTP MISSION DEPENDENT EVENTS

*

1.1 INTRODUCTION

This section contains a detailed description of the ASTP mission dependent events required for this mission. *

1.2 EVENTS PROCESSOR TIMED EVENTS TABLE

The Events Processor Timed Events Table contains each event which is active in the ASTP mission. The table has the following categories for each time base in the mission: *

1. Time in Time Base the event is to occur
2. Events Processor Submodule to be executed
3. Flight mode the event is executed in
4. Description of the event

EVENT PROCESSOR TIMED EVENTS TABLE (Continued)

TIME BASE	TIME IN TIME BASE	EVENT PROCESSOR SUBMODULE (L.EXXX)	FLT (F), NON-REP (N), OR REP (R)	EVENT DESCRIPTION
T0	0.0	L.E285	F,N,R	Schedule Water Valve Processing
T0	0.0	L.E30	F,N,R	Schedule first Thrust Misalignment Vector Implementation
T0	0.0	L.E40	F,N,R	1. Set Backup Accelerometer Profile Parameters 2. Set Engine Out Factor and Performance Factor History to 1.0 3. Disable Performance Factor calculation 4. Set constant acceleration biases for S-IB stage
T0	17.0	L.E290	R	Start Time Base 1
T0	17.4	L.E10	F,N	Queue in Liftoff Search (M.L000)
T0	150.0	L.E17	F,N	Stop looking for Time Base 1
T1	0.0	L.E25	F,N,R	Set Nominal Number of Minor Loops per Minor Loop Support
T1	0.0	L. E20	F,N,R	Queue in Time Tilt (M.TT00)

EVENT PROCESSOR TIMED EVENTS TABLE (Continued)

TIME BASE	TIME IN TIME BASE	EVENT PROCESSOR SUBMODULE (L.EXXX)	FLT (F), NON-REP (N), OR REP (R)	EVENT DESCRIPTION	
T1	0.0	L.EVNT	F,N,R	Enable Thrust Event Calculations	
T1	3.0	L.E21	F,N,R	Queue in F/M Calculations (M.DV00)	
T1	3.0	L.E22	F	Enable DI 14 and 15 in the ON Mask	
T1	10.0	L.EPFC	F,N,R	Enable Performance Factor Calculation	
T1	10.0	L.E66	F,N,R	Set Accelerometer RTC's	*
T1	127.9	L.E62	F,N	Enable INT 2 and INT 6	
T1	132.9	L.E290	R	Start Time Base 2	*
T2	2.9	L.E73	F,N,R	Set Accelerometer RTC's	*
T2	3.0	L.EVNT	F,N,R	Enable Thrust Event Calculations	
T2	3.0	L.E370	F,N,R	Set F/M to a Constant	
T2	4.5	L.E65	F,N	Enable DI 23 and INT 5	
T2	6.0	L.E290	R	Start Time Base 3	
T2	13.0	L.E360	F,N	Start Time Base 3 on Time Backup	
T3	0.0	L.E72	F,N,R	1. Set Roll Ladder Magnitude Limit 2. Queue in APS Attitude Control Test	
T3	0.0	L.E74	F,N,R	Disable Accelerometer Zero Test	

EVENT PROCESSOR TIMED EVENTS TABLE (Continued)

TIME BASE	TIME IN TIME BASE	EVENT PROCESSOR SUBMODULE (L.EXXX)	FLT (F), NON-REP (N), OR REP (R)	EVENT DESCRIPTION
T3	0.0	L.E375	F,N,R	Set F/M to a Constant
T3	0.0	L.E365	F,N,R	Set Accelerometer Zero Test Term
T3	0.0	L.E355	F,N,R	1. Queue out Time Tilt (M.TT00) 2. Queue in Chi Telemetry (M.CC10)
T3	0.0	L.EVNT	F,N,R	Enable Thrust Event Calculations
T3	6.08	L.EVNT	F,N,R	Enable Thrust Event Calculations
T3	6.08	L.E96	F,N,R	1. Queue in F/M Calculations (M.DV00) 2. Queue in M/F Smoothing (M.ST00)
T3	6.08	L.E95	F,N,R	1. Set Accelerometer RTC's 2. Enable Accelerometer Zero Test
T3	7.1	L.EPFC	F,N,R	Enable Performance Factor Calculations
T3	10.0	L.E135	F,N,R	Queue in TB4 N.S Entry
T3	10.0	L.E142	F,N	Enable DI 5 and INT 4
T3	35.0	L.E105	F,N,R	1. Queue in IGM (M.IG00) 2. Queue out Chi Telemetry (M.CC10)
T3	35.0	L.E115	F,N,R	Set Nominal Number of Minor Loops per Minor Loop Support
T3	60.53	L.E110	F,N,R	Queue in Steering Misalignment
T3	328.1	L.EVNT	F,N,R	Enable Thrust Event Calculations

*

*

IV-I-4

Rev. K

EVENT PROCESSOR TIMED EVENTS TABLE (Continued)

TIME BASE	TIME IN TIME BASE	EVENT PROCESSOR SUBMODULE (L.EXXX)	FLT (F), NON-REP (N), OR REP (R)	EVENT DESCRIPTION
T3	500.0	L.E295	R	Start Time Base 4
T4	0.0	L.E70	F,N,R	1. Set Accelerometer RTC's 2. Disable Accelerometer Zero Test
T4	0.0	L.E130	F,N,R	1. Queue out M/F Smoothing (M.ST00) 2. Queue in DOM Telemetry (M.OD00) 3. Queue in Minor Loop Support (M.MS00) in Flight Mode Only
T4	0.0	L.E145	F,N,R	1. Set F/M Constant 2. Enable following DCS Commands: Generalized Manuever, Time Base Update, Generalized Switch Selector, Terminate, Inhibit Water Control Valve Logic, Return to Timed Maneuver Sequence, S-IVB/IU De-orbit, Ladder Magnitude Limit, and Extraction Maneuver Enable 3. Change Roll Magnitude Limit 4. Change Pitch, Yaw, and Roll Magnitude Rate Limits

*

EVENT PROCESSOR TIMED EVENTS TABLE (Continued)

TIME BASE	TIME IN TIME BASE	EVENT PROCESSOR SUBMODULE (L.XXX)	FLT (F) NON-REP (N), OR REP (R)	EVENT DESCRIPTION
				5. Set and Reset CRP 6. Enable DCS Interrupt (INT8) in Flight and Non-Repeatable Modes Only 7. Queue Out IGM (M.IG00) 8. Set Nominal Number of Minor Loops per Minor Loop Support 9. Set DCS Flag to Not in Progress 10. Set DCS Inhibit Removed Bit in MC27 11. Change APS Attitude Control Test Limits
T4	0.0	L.EVNT	F,N,R	Enable Thrust Event Calculations
T4	0.0	L.E160	F,N,R	1. Set HOP Constant to go to Maneuver Processor 2. Set Initial Guess at R 3. Set S/C not in Control 4. Queue in Orbit Guidance (M.OG00) 5. Queue out Chi Telemetry (M.CC10) 6. Zero Steering Misalignment Parameters 7. Reset SMC Active Bit in MC25

*

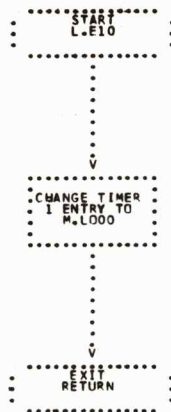
EVENT PROCESSOR TIMED EVENTS TABLE (Continued)

TIME BASE	TIME IN TIME BASE	EVENT PROCESSOR SUBMODULE (L.EXXX)	FLT (F), NON-REP (N), OR REP (R)	EVENT DESCRIPTION	
T4	5.0	L.E172	F,N,R	Enable DI 9	
T4	15.0	L.E280	F,N,R	Queue in Phase Activator (M.PA00) to Terminate Phase 1 and Start Phase 2	
T4	55.0	L.E300	R	Queue Out Acquisition Gain/Loss (M.GL00)	
T4	60.0	L.E305	R	Repeatable Terminate	*
T5	0.0	L.E50	F,N	Queue in Maneuver Processor for Time Base 5 Orbit Guidance	
T5	0.0	L.E52	F,N	Set Pitch and Yaw Ladder Magnitude Limits to Maximum	* *
T5	31.0	L.E53	F,N	Queue in Orbital Velocity Calculations	
T5	T _{SS} *+2905.0	L.E55	F,N	Enable Generalized Maneuver DCS Commands after Propellant Dump Completion	* *
*T _{SS} as used above is defined to be:					
$T5 + T_{LDD} + T_{HDD} + 63.5$					
where T _{LDD} and T _{HDD} are included in the S-IVB/IU De-Orbit DCS Command.					

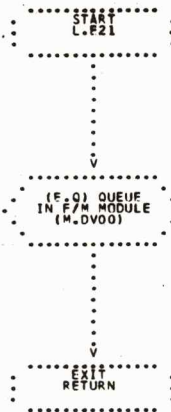
1.3 EVENTS PROCESSOR SUBMODULES

This section contains the flowcharts of each Events Processor Submodule active in the ASTP mission.

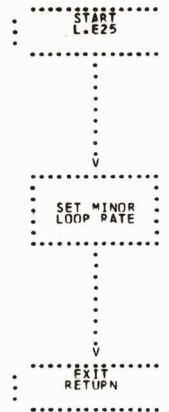
L.E10



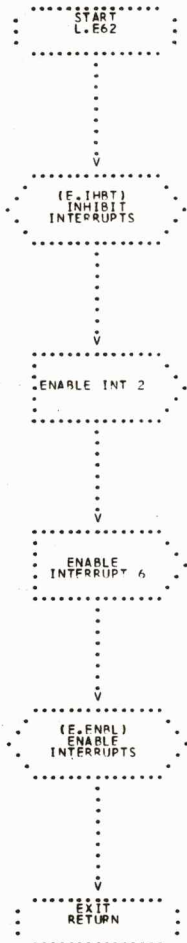
L.E21



L.E25



L.E62



L.E17



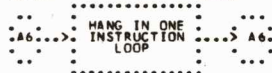
L.E22



L.E30



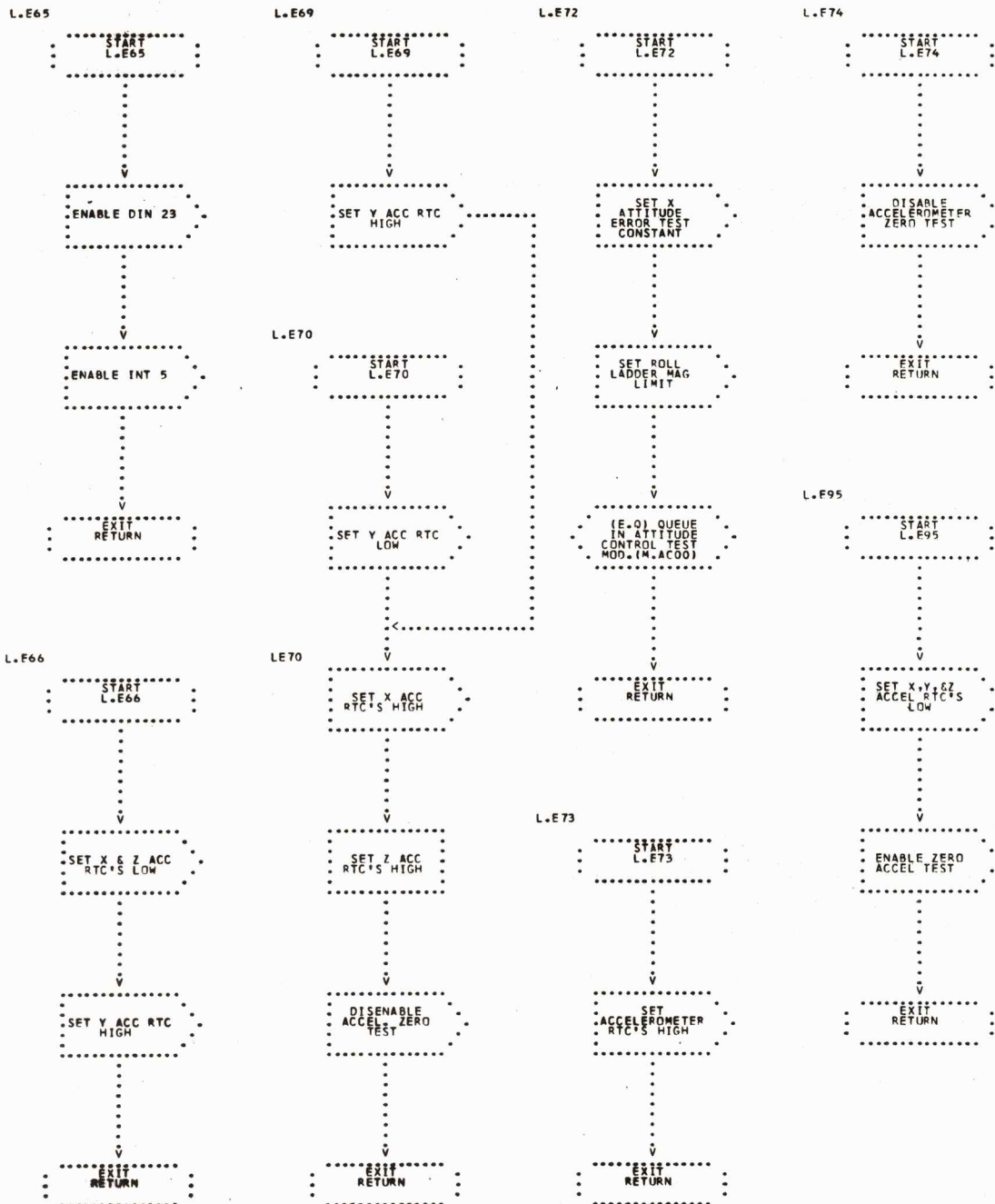
TB025

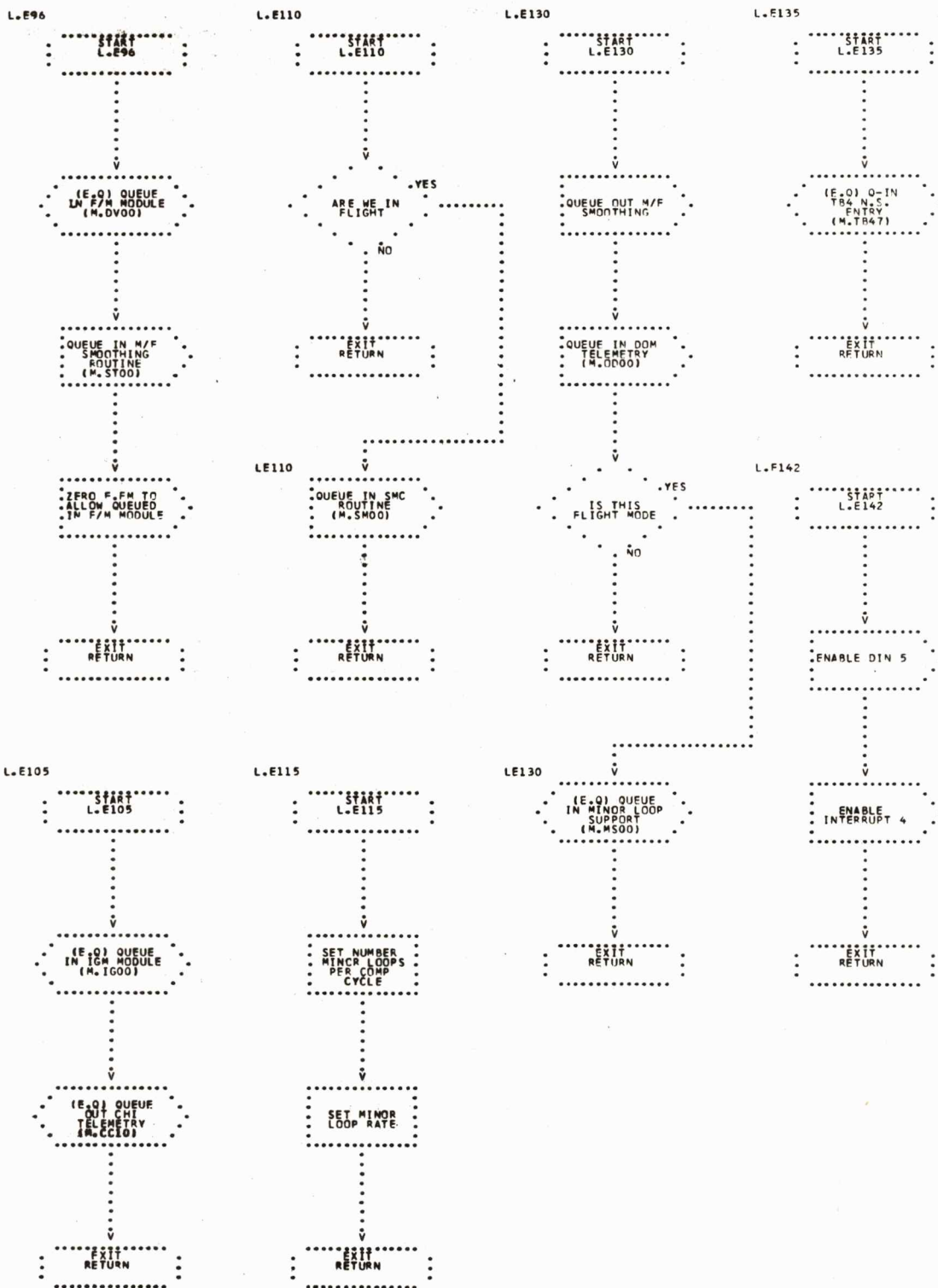


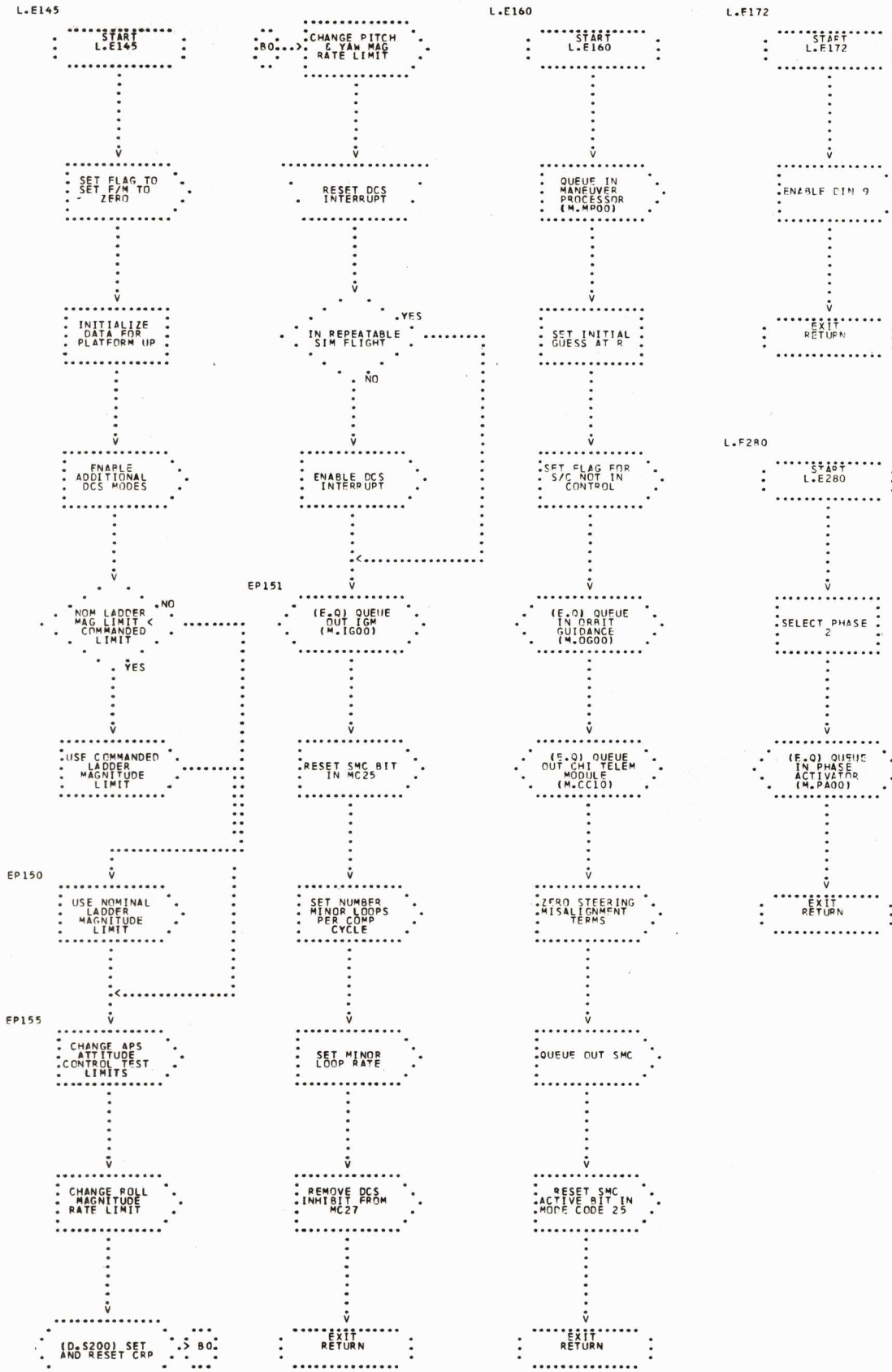
TB025

L.E20





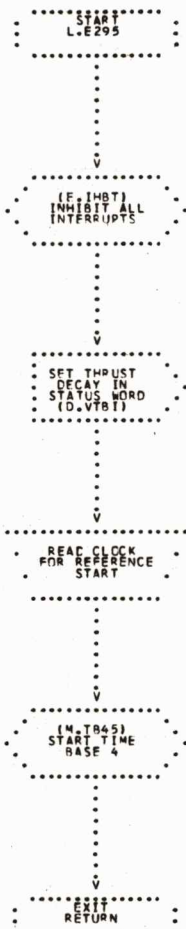




L.E285



L.E295



L.E305



L.E355



L.E290



L.E300



LE306



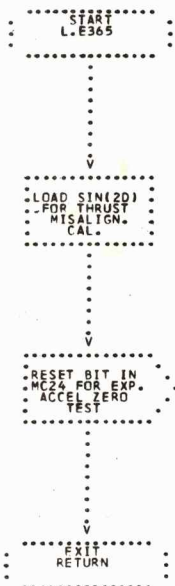
LE306

L.E360

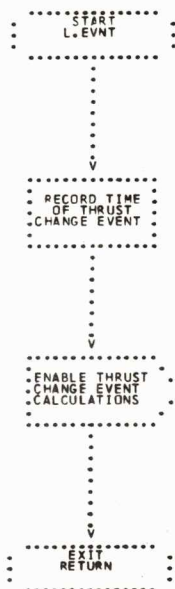


LE360

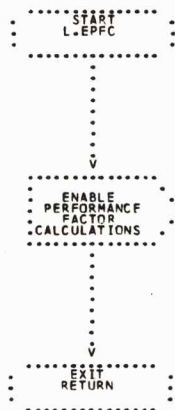
L.E365



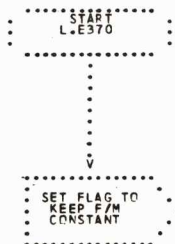
L.FVNT



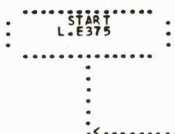
L.EPFC



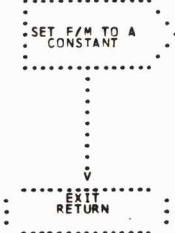
L.E370



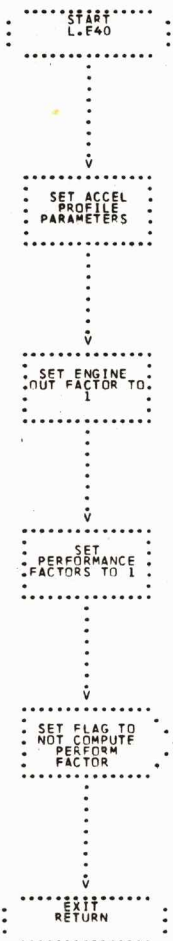
L.F375



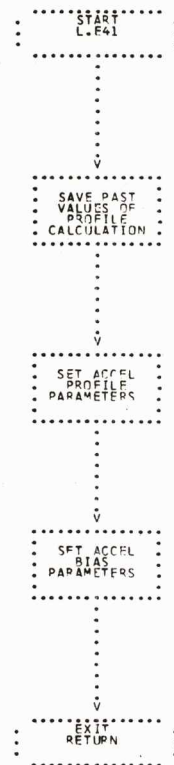
L.E375



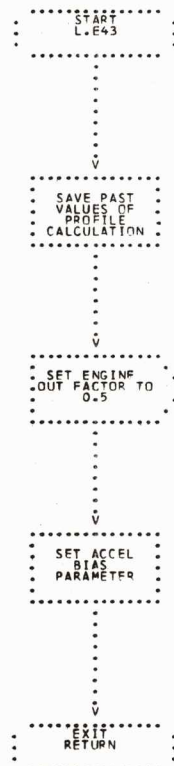
L.E40



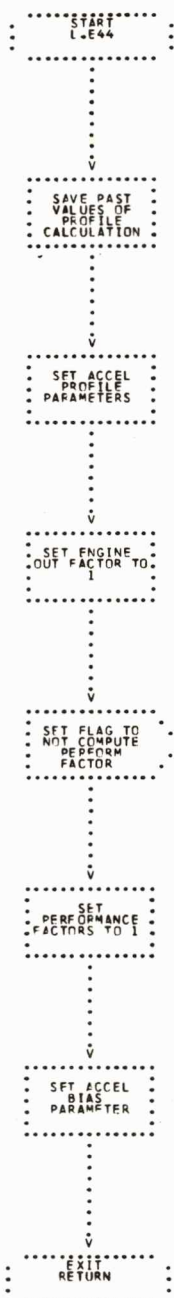
L.E41



L.E43



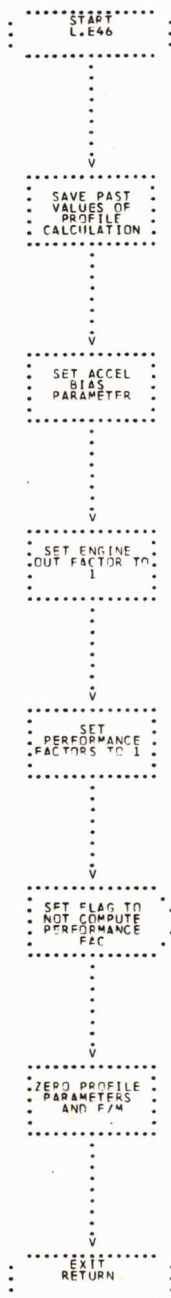
L.E44



L.E45



L.E46



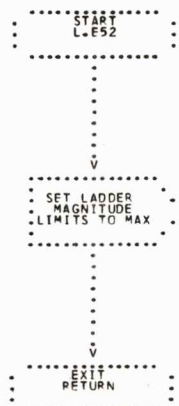
L.E455



L.E50



L.E52



L.E55



L.E53

