5520.5-205

PROGRAMMED GUIDANCE EQUATIONS FOR COLOSSUS 3 COMMAND MODULE EARTH ORBITAL AND LUNAR PROGRAM

NAS 9-8166

20 APRIL 1971

BASED ON ARTEMIS 72 PROGRAM

Prepared for
MISSION PLANNING AND ANALYSIS DIVISION
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
MANNED SPACECRAFT CENTER
HOUSTON, TEXAS

WORKING PAPER

Change Page Record

The original issue of 5520.5-205 represents a complete re-issue of 5520.5-177, updated to the contents of the Colossus 3 (Artemis 72) program. Those pages which were changed in content from the Colossus 2E description (5520.5-177) are identified by "Rev. O" on the same line as the page number.

ABSTRACT

The information presented in this document on the Colossus 3 (manned Apollo Command Module earth orbital and lunar) guidance program was produced with the intention that it be used together with a symbolic tabulation of the program. The information is divided into a series of separate segments called "writeups", each of which describes a basic area of guidance computations within the program. In order to assist the user in finding the computations for a particular quantity, summaries of each of the writeups, plus a brief discussion of the program computation flow, have been included. At the end of the document is an alphabetical list of the routines in the writeups, together with their location by page number in the individual writeups.

The program listing from which the "programmed equations" information was generated is identified as "Artemis 72", which is being wired into computer memory ropes for the "J" missions (starting with Apollo 15). Identification of areas of change since the previous flight program was considerably aided by the results of a symbolic comparison program developed by Flight Software Branch.

Because of the purposes for which the information in this document was prepared, and the methods used in its production, this material under no circumstances should be considered as definitive information on the Colossus program. Contrary to an analogous document produced by MSC for the LM program, this document has not been approved by NASA management. Since this document has not been approved by NASA, therefore, it should never be used or cited as a reference for the equations in the Colossus computer program.

CONTENTS

Page

Introduction 1

Summary of Individual Writeups 3

Program Computation Flow 9

Notation and Terminology 31

Major Variables 41

Channel Assignments 67

Meanings of MODREG Values 73

Special Program Routines 75

Checklist and Option Codes 77

Error Codes 79

Job Priorities 85

Erasable Memory Prelaunch Load 89

Writeups

- Attitude Maneuvers ATTM-1
- Boost Computations BOOS-1
- Burn Control BURN-1
- Conic Routines CONC-1
- Coordinate Transformations COOR-1
- Data Input/Output DATA-1
- Display Interface Routines DINT-1
- Display Computations DISP-1
- Digital Autopilot Entry Routines DPEN-1
- Digital Autopilot Interface Routines DPIR-1

CONTENTS (Continued)

Page

Digital Autopilot RCS Routines DPRC-1

Digital Autopilot TVC Routines DPTV-1

Entry Computations ENRY-1

Entry Preparation ENTP-1

General Program Control GENP-1

IMU Computations IMUC-1

Inflight Alignment INFA-1

Mathematical Functions MATH-1

Measurement Incorporation MEAS-1

Minimum Key Rendezvous MINK-1

Noun Definitions NNDF-1

Optics Computations OPTC-1

Orbital Integration ORBI-1

Orbital and Rendezvous Navigation ORVN-1

Prelaunch Alignment PREL-1

Rendezvous Computations REND-1

Return to Earth Computations RTER-1

Steering Computations STER-1

Telemetry TELE-1

Testing Routines TEST-1

Uplink Processing UPLK-1

Verb Definitions VBDF-1

Index of Routines X-1

Introduction

Under the auspices of TRW Systems MTCP task A-201 ("Support of Apollo Guidance Document Review," J. R. Garman, task monitor), a review of certain portions of the Apollo Guidance Computer program identified as "Colossus 3" has been conducted. This program is a considerably modified version of the "Colossus 2E" program used on the "H3" (Apollo 14) mission, and is intended for use with the Command Module Computer on the "J" missions. The major purpose of this review has been to provide more effective support to the Flight Software Branch of MSC in the identification and analysis of various program performance features, and to permit more effective and rapid review of published computer program documentation.

During similar reviews of the AS-202, AS-204, AS-501, LM-1, Sundisk, Sundance, and previous programs of the Colossus (lunar CSM) series, it was found desirable to assemble a set of working-paper information on the equations which seem to have been programmed for these flights. This material has proven to be useful to a few groups associated with these flights, in that it can be used to help bridge the gap between the extreme detail of the program listing and what appears in the approved computer program documentation. Consequently, the material on the following pages has been assembled in a fashion similar to that done for previous flights, and follows the same general format.

Certain aspects of the program are quite complex, and this programmed guidance equation material should not be considered as a substitute for actual study of the program symbolic listing itself. To assist in such a study, the binary scale factors of equation quantities have been included, as well as the program notation employed for constants (the notation for variables follows that, in general, used in the program), and except in a few instances divisions have been made in accordance with the symbolic tags assigned to program steps in the listing. It should be realized that no complete set of approved computer program documentation was available against which the programmed equations could be validated, and in the interests of timely publication the review of the assembled document against the program assembly has not been as detailed as would be desired.

The program assembly listing which was used to prepare the programmed equations information bears the heading print:

REVISION 072 OF AGC PROGRAM ARTEMIS BY NASA 2021114-011,
19:41 FEB. 26,1971

The function of virtually all the program steps of interest to the flight is described on the following pages or, for general computer system control (executive and waitlist system, interpretive language, and similar items) in 3420.5-27 (see Notation and Terminology). Only isolated omissions of unreferenced constants and non-executed program steps have been made.

It is emphasized that the material on the following pages is intended to be an aid to the review of the symbolic listing, not a substitute for it. Definitive guidance equation information can and should be provided only in approved computer program documentation, available through the appropriate MSC channels.

Summary of Individual Writeups

The information on the programmed equations has been divided into a number of separate entities called "writeups". Each writeup is intended to be complete within itself except for definition of certain frequently used quantities (described in "Major Variables") and perhaps for reference to routines in other writeups (whose location may be found through use of the "Index of Routines"). Familiarity with the information in "Notation and Terminology" is necessary in order to understand the material presented in the individual writeups, since a somewhat specialized type of notation is necessary in order to describe adequately the computations within the guidance program. Individual writeups are assigned four-letter codes, with page numbers restarted at the beginning of each writeup, to facilitate updating. The writeups are arranged in the alphabetical order of these four-letter codes, and are also arranged in that sequence below.

Attitude Maneuvers (ATTM)

This writeup contains the computations performed to determine the axis about which spacecraft rotation should take place and the magnitude of the rotation required to go from the present vehicle attitude to a final specified attitude. It also shows the attitude computations which are performed to provide a general-purpose vehicle attitude pointing capability.

Boost Computations (BOOS)

This writeup contains the computations performed when the liftoff signal is received, including the control for computing display parameters during the boost phase of flight, the calculations to drive the error needle displays, and the computations associated with the Saturn mode of operation of the Digital Autopilot (DAP). Also included are the equations used for backup Saturn steering after liftoff, as well as those for backup S4B cutoff for TLI.

Burn Control (BURN)

This writeup contains the control computations associated with most of the pre-thrust and thrusting programs, except for P37 (in Return to Earth Computations). This includes the logic associated with the generation of the various displays and some of the computations associated with the programs themselves (other computations are in the Rendezvous Computations and Steering Computations writeups). The control for sequencing through the automatic rendezvous sequence is in the Minimum Key Rendezvous writeup.

Conic Routines (CONC)

This writeup contains the various conic subroutines that are grouped together in the program and in official equation information, including the Lambert and Kepler routines.

Coordinate Transformations (COOR)

This writeup contains the various coordinate transformation subroutines in the program, entered by the using routines as necessary. Also included are the routines for determining lunar and solar position, for the transforming between latitude/longitude/altitude and a position vector, and for converting between planetary and inertial coordinates.

Data Input/Output (DATA)

This writeup contains the computations performed in response to a keyboard (DSKY) or uplink input, as well as those routines that permit internal use of the display system. The computations associated with most of the nouns and special-purpose verbs, as well as the display interface routines, are subjects of separate writeups.

Display Interface Routines (DINT)

This writeup contains the various display interface routines used by internal callers of the display system in order to achieve proper sequencing of the displays according to their priorities, and to achieve various special-purpose displays with a minimum of complication to the "mission program" making use of the display system.

Display Computations (DISP)

This writeup contains the computation of those variables whose major (or sole) purpose is display for crew review or action (time of flight, special rendezvous displays, entry preparation displays, antenna pointing display, etc.).

Digital Autopilot Entry Routines (DPEN)

This writeup contains the computations associated with the performance of the digital autopilot (DAP) during the entry phase of the flight, including the selection of appropriate RCS jets for the CM and the timing of their firing duration.

Digital Autopilot Interface Routines (DPIR)

This writeup includes those computations performed in support of several portions of the DAP (such as derivation of mass properties information), as well as those influencing both the DAP and the non-DAP portions of the program (such as engine gimbal drive test). Also included are the computations associated with the crew use of DAP parameters (verb 46 and verb 48).

Digital Autopilot RCS Routines (DPRC)

This writeup contains those computations performed for the service module RCS (Reaction Control System) DAP. The jet selection logic (with the effect of jet failure information) and the processing of manual controller information is also shown.

Digital Autopilot TVC Routines (DPTV)

This writeup contains those computations associated with the TVC (Thrust Vector Control) DAP, used with the SPS engine. This includes the computation of the trim estimation and the TVC DAP roll-control computations (which, of course, use the RCS jets).

Entry Computations (ENRY)

This writeup contains the computations performed for entry, once the maneuver to the entry attitude has been completed, as well as the entry display computations. The entry-DAP support computations performed at a two-second rate are also shown.

Entry Preparation (ENTP)

This writeup contains the computations performed when preparation for entry is started, including generation of displays for entry monitoring purposes and request for separation.

General Program Control (GENP)

This writeup contains those computations which have an influence upon the performance of the complete program. Included are the equations for starting accelerometer sampling, powered flight navigation ("Average-G"), the TIME4 interrupt control logic, fresh start and restart computations, processing of a verb 37 (program change) input to start the new program, pre-standby and post-standby computations, and the alarm and software restart routines.

IMU Computations (IMUC)

This writeup contains the computations directly associated with the IMU (Inertial Measurement Unit), including accelerometer sampling, powered flight and free-flight accelerometer and gyro compensation, the IMU monitoring functions performed by the TIME4 interrupt logic (checks for status, alarms, and gimbal lock), and the IMU mode change and gyro torquing routines.

Inflight Alignment (INFA)

This writeup contains the computations required to perform the various IMU alignment functions during flight, as well as the automatic star selection routine. Alignment prior to launch is described in the Prelaunch Alignment writeup.

Mathematical Functions (MATH)

This writeup contains the various trigonometric, logarithm, and square root functions that are included in the program.

Measurement Incorporation (MEAS)

This writeup contains the computations which are performed to update the state vector based on optics or VHF range measurements, the cislunar midcourse navigation (P23) computations, the rendezvous tracking data processing routine (R22), and the routine to reduce the W matrix to 6x6. Other navigation logic is in Orbital and Rendezvous Navigation.

Minimum Key Rendezvous (MINK)

This writeup contains those routines associated exclusively with the minimum key rendezvous (automatic rendezvous) logic, including the program sequencing logic, the logic for W matrix automatic reinitialization, and the special alignment program computations. P79 logic is also shown.

Noun Definitions (NNDF)

This writeup contains the computations and constants directly associated with the decoding of the various nouns used with the display system. Also included in this writeup is the information reflected in the noun tables incorporated in the program, and a list of the program internal references to individual nouns.

Optics Computations (OPTC)

This writeup contains the computations associated with the optics system, including those for monitoring optics status, changing the position of the optics, performing optics marks and processing them, and computing optics pointing information.

Orbital Integration (ORBI)

This writeup contains the orbital integration and related control (interface and driver) functions that are included in the program for precision (Encke) integration, and the interface and driver logic for conic integration (the Kepler routine is in Conic Routines). Also included are the routines to transfer between powered flight and free flight (and between free flight and powered flight, i.e. Average-G) navigation.

Orbital and Rendezvous Navigation (ORVN)

This writeup contains the computations associated with orbital navigation and rendezvous navigation (except R22 which is in Measurement Incorporation, as are some W matrix routines). Included is the control logic associated with the generation of displays and initialization of the W matrix (except for the Minimum Key Rendezvous material on automatic W-matrix initialization).

Prelaunch Alignment (PREL)

This writeup contains the computations performed prior to launch to erect the platform and, by gyrocompassing, align it to the proper orientation for launch. Also included is the logic for processing a change to the required launch azimuth (initiated by a verb 78). The computations shown are provided independently of those for IMU calibration (in Testing Routines), even though some duplication of the resulting information is necessary, for convenience in description.

Rendezvous Computations (REND)

This writeup contains the calculations associated with the performance of the pre-thrust and thrusting guidance of the rendezvous programs (the Lambert routine itself, however, is in Conic Routines). The program control for these pre-thrust and thrusting programs is in the Burn Control writeup.

Return to Earth Computations (RTER)

This writeup contains the computations for the Return to Earth program (P37).

Steering Computations (STER)

This writeup contains the guidance computations associated with the SPS burn program (P40), some of which are also used with the RCS burn program (P41). This includes the cross-product steering routine, computation of preferred IMU alignment and burn attitude, the thrust monitoring computations (logic based on monitor outputs is in Burn Control), etc. The velocity-required logic for Lambert guidance is in Rendezvous Computations.

Telemetry (TELE)

This writeup contains the computations performed when a telemetry interrupt is received, and the logic associated with the generation of the "erasable memory dump" downlist (verb 74). Also included is a brief summary of the information which is telemetered.

Testing Routines (TEST)

This writeup contains the computations performed for checking the computer erasable and fixed memory (including the memory sum check): these are the only computer self-check computations in the program. Also included are the computations associated with the prelaunch calibration of the IMU that are provided in fixed memory (limited mainly to drift testing). These latter computations are shown independently of those for prelaunch alignment, even though some duplication of the resulting information is necessary, for convenience in description. In addition, information from an earlier program (Sundisk) was used to supply an indication of expected values of erasable memory constants and program steps to be used with the drift testing.

Uplink Processing (UPLK)

This writeup contains the computations performed in response to receipt of the special-purpose uplink/DSKY verbs (70-73), and the logic associated with the incorporation of a state vector update.

Verb Definitions (VBDF)

This writeup contains the computations associated with the decoding of most of the special-purpose verbs (except those in DPIR and UPLK). Only the initial computations performed after the verb is decoded are shown in this writeup (such as the setting of a control cell and the establishing of the job), with subsequent calculations given in another writeup (such as Display Computations).

Program Computation Flow

Because of the nature of the Apollo Guidance Computer hardware (and software) design, there is no unique "flow" of computations that takes place. Instead, various calculations can proceed virtually independently of one another at different cycle rates. For some purposes, however, that do not require detailed simulation or understanding of the computer calculations, the simplification of a straight-line flow can be made. The following paragraphs outline briefly the sequence of computations which takes place with such a simplification, and are intended to orient the user to some of the contents of this document. A review of the segment on "Summary of Individual Writeups" will also prove useful.

Prelaunch

When power is first applied to the computer, one of the early functions which is performed is the setting of the computer's initial conditions in erasable memory. This is accomplished by means of an input of V36E, which causes the computations starting at "SLAP1", page GENP-8, to be entered. Following this, the various erasable memory presets required for the flight (such as constants stored in erasable memory for flexibility) can be performed either by DSKY or Uplink means. An uplink interrupt causes "UPRUPT", page DATA-1, to be entered; a main panel DSKY key depression causes "KEYRUPT1", on the same page, to be entered; and a navigation panel DSKY key depression causes "MARKRUPT", page OPTC-9, to be entered (where checks are first made for an optics mark or mark reject). All three data inputs eventually cause "CHARIN", page DATA-2, to be established to process the input information.

Following this, the IMU can be made ready for flight, making use of computations starting at "IMUMON", page IMUC-3. Next, an input of V37 E 01E will start program 01 (P01), on page PREL-1 ("GTSCPSS"). This causes the initial conditions for the prelaunch erection and gyro compassing routines to be set. In addition, the IMU CDU's are zeroed, coarse aligned to the desired IMU angles for the flight, and then switched to fine align. At the end of this sequence, some more initial conditions are set and the program number on the DSKY changed to 02, whereupon "SLEEPIE" (page PREL-3) is entered.

"SLEEPIE" is the standard end point for the prelaunch gyro compassing routine (vertical erection is performed for 640 seconds at the start of P02, and for 320 seconds after an azimuth change input). It controls a half-second waitlist call of "ALLOOP" to read and reset accelerometers and establish "ALFLT" (page PREL-4). Every five seconds, "EARTHR*" is entered to correct for earth rotation effects and initiate pulse torquing of the gyros (page PREL-6).

While in P02, changes to the azimuth towards which the stable member is being aligned (i.e. the "launch azimuth") may be achieved by V 78 E, which causes "AZMTHCG1", page PREL-6, to be established. A new launch azimuth may then be loaded. After completion of the routine (either by a PRO or V34E), a flag is set that is checked every 5 seconds in "ALFLT". If any change in azimuth is detected, the Z-axis gyro is pulse torqued as specified by the angle change, and then the gyro compassing program is essentially reinitialized (by transfer to "ESTIMS", page PREL-2), although the vertical erection interval will be only 320 seconds instead of the 640 seconds at the start of P02.

Also while P02 is running, P03 can be initiated (the gyrocompassing computations continue) in order to perform an optical verification of azimuth. P03 can be entered by V 65 E, which causes "GCOMPVER", page PREL-7, to be established. After completion of the optics marks and displays, P02 again becomes displayed (by entrance to "GCOMP5", page PREL-9).

Boost

At the beginning (in "ALFLT") and end (in "SLEEPIE") of each half-second gyro-compassing cycle in P02 or P03, a check is made for liftoff by performing "CHKCOMED", page PREL-3. If the liftoff input channel bit indicates liftoff, or if a V 75 E (backup liftoff, performing "LFTFLGON", page VBDF-12) has been received, then the prelaunch cycling is halted and "P11", page BOOS-1, is entered.

"P11" updates ephemeris time to have an origin of liftoff time (and zeros the computer clock), changes the program number display to 11, gets rid of the backup liftoff verb (or other DSKY input) if it is on the DSKY, initializes position and velocity based on pad location data, and computes an initial value for [REFSMMAT] , the reference-to-stable-member matrix. It also performs some initialization functions for the computation of position and velocity every two seconds, which is a process starting at "READACCS", page GENP-2. The value of liftoff time may be "updated" by using verb 70 (in P27, as described on page UPLK-10) if desired, after P11 has been terminated and P00 entered.

Every two seconds during the boost phase, and whenever else Average-G computations are running, "READACCS" is entered. This routine causes accelerometer data to be sampled (and accelerometers reset) via "PIPASR" (page IMUC-1). This task also causes "SERVICER" to be established (page GENP-4), which checks for excessive (saturated) accelerometer output, compensates accelerometer data for scale factor and bias errors via "1/PIPA" (page IMUC-1: gyro compensation during powered flight is also handled there), updates position and velocity using "CALCRVG" (page GENP-4), and then transfers to "VHHDOT", page BOOS-2. "VHHDOT" computes quantities for display purposes, and causes them to be displayed by noun 62 (see page NNDF-26) unless some other display has been specified by the crew. This two-second cycle continues until manual specification of another program is provided (by V37 means) by the crew.

Shortly after "P11" was first entered, the present CDU angles are sampled and used as the "desired" CDU angles for driving a display (in roll, pitch, yaw body coordinates) on the FDAI attitude error needles. Starting a desired length of time after liftoff deduced (the time interval is specified by an erasable memory constant), "ATERJOB", page BOOS-3, begins using nominal roll rate information and a stored pitch polynomial to compute the desired CDU angles. These computations are halted a desired length of time (again specified by an erasable memory constant) after they were started. They are performed with a low "priority", meaning that they will be done only if no other computation (such as Average-G) must be done. In addition, a delay of $\frac{1}{4}$ second is imposed between the end of one pass through these equations and the start of the next pass. Backup automatic steering of the Saturn using the output of the polynomial (with errors considered zero at initiation) can be achieved by setting appropriate spacecraft switches.

Manual inputs for Saturn control can be accomplished by first keying in V 48 E and loading DAPDATR1 with 3xxxx (page DPIR-2) to specify the Saturn DAP configuration. By next keying in V 46 E, "STABLISH", page DPIR-1, is entered, which in turn causes "SATSTKON", page BOOS-5, to be entered. This in turn causes "REDOSAT" to be entered, which finally causes "SATSTICK" (page BOOS-6) to be entered every 0.1 second to generate outputs based on RHC (rotational hand controller) settings. When "SATSTKON" is entered, flagword bits are set which cause "ATERJOB" to halt.

Coast

When it is decided that Average-G computations should be terminated at the end of Saturn thrusting, this can be accomplished by V37 E 00E, which will cause "V37", page GENP-15, to be entered. Since Average-G will be running, "ISITP00", page GENP-18, will set a flagword bit (bit 1 of FLAGWRD1) to 0, causing "READACCS" to set the exit from the Average-G equations to transfer to "AVGEND" (page GENP-3). There, the state vector from the last Average-G cycle is loaded into the "permanent" state vector for free flight of the CSM (in "AVETOMID", page ORBI-4), and then, after some other settings, "CANV37", page GENP-19, is entered, in order to continue processing of the program whose number was specified by V37 input.

The "CANV37" computations cause a master initialization routine, "INITSUB" (page GENP-17), to be performed to set several flagword bits and do other functions. Then, since program 00 was specified ("P00"), a restart group is set so that "STATINT1" will be established. Restart groups are a software design feature included mainly for the purpose of providing a capability to resume computations in the event of a transient difficulty such as a power glitch, but they also can be used to achieve proper program performance control under normal conditions. Computations similar to those performed when the computer was initialized at turn-on are then performed (via entrance to "GOPROG2", page GENP-12), and these include establishing "STATINT1".

"STATINT1", page ORBI-1, is the job intended to keep the CSM state vector from becoming too "old", while minimizing the buildup of errors due to the computation. It is entered every 10 minutes (more precisely, it is entered 10 minutes after its previous pass has been completed). This looping can be halted (or looping for an excessive integration time within the orbital integration package) and P00 selected by an input of V96E (which sets a flagword bit checked at the start of "STATINT1" and also in "TESTLOOP", page ORBI-12). If "STATINT1" detects that the bit is set, it is reset and no additional ten-minute waitlist call is made (hence integration remains suspended until do e.g. V37 E 00E again).

If the flagword bit (QUITFLAG, bit 5 of FLAGWRD9) indicating a V96E is not set, then further checks are made before deciding to perform an integration. These are done by setting some flagword bits in "STATINT1" that are analyzed in "TESTLOOP": these preclude starting an integration of the CSM state vector unless it is at least four "time steps" (a variable time depending on distance of vehicle from primary body) delayed from the present time. If integration is started, it will continue until the state vector is within one time step of the "present" time (measured at the start of "STATINT1"), but no integration for a fraction of a time step is done. After the CSM state vector has been integrated (if necessary), then the LM state vector is integrated to the same time as the CSM state vector, so as to keep the two state vectors "together". Integration of the LM state vector, however, is bypassed if a V44E input has set SURFFLAG, indicating that the LM is on the lunar surface (bit reset by V45E). While this integration is going on (both CSM and LM), bit 1 of FLAGWRD2 is set to inhibit processing of a V37 input that would interrupt the integration after the CSM state vector has been integrated, thus leaving the two state vectors out of synchronism with one another. Unless a V96E is provided, the ten-minute checks continue indefinitely, until some other program is specified.

Uplink

Four verbs (70 - 73) are provided for loading the computer erasable memory with parameters from the ground via the uplink (the same verbs could be used for input via the DSKY, of course). These verbs, which are accepted only if the present program is P00, P02, or P20 in a non-rendezvous (i.e. tracking) mode, cause P27 to be displayed via computations starting on page UPLK-1. The function of each of the verbs is described starting on page UPLK-10.

Processing of a LM or CSM state vector update is handled by verb 71 (see page UPLK-11), with the first word in the sequence (after those associated with control of P27 performance) serving to identify not only which state vector is involved, but also whether it should be considered earth-centered or moon-centered. The computations performed to incorporate the state vector are in "INTWAKEU", page UPLK-6. Since the erasable memory general purpose update verbs (71 and 72) include specification of absolute erasable memory cells, considerable flexibility in the uplink information is possible. Aside from some constraints on erasable memory bank changes (verb 71 only), no software lockouts are included, so that internal computer program control cells can be modified (such as flagwords). To change particular flagword bits (for addresses above 30_8), however, it would be more satisfactory to make use of noun 07 (see page NNDF-21). Channels (of address 30_8 and below) can also use the noun, or else noun 10.

External Delta V Burn (SPS)

Preparations for an External Delta V burn (for which a velocity increment and ignition time are specified by inputs to the computer) can begin by the performance of an uplink load of the target parameters (and, if desired, an update of the state vector). Performance of the External Delta V pre-thrust program (P30) is then started by input of V37 E 30 E, causing "P30", page BURN-1, to be entered. There, after setting two program control flagword bits (associated with P20), a display of ignition time is produced. This display is generated in the program by loading the accumulator with the proper bit pattern for verb 06 and noun 33 (bits 14-8 contain the verb and bits 7-1 the noun), and then transferring to display interface routine "VNFLASH". This routine (page DINT-6) sets a flagbit indicating that it was entered (for use in processing the response), retains information on the verb and noun desired, and eventually progresses to "MAKEPLAY" (page DINT-7), where the actual display generation functions are initiated. Return is not made to the computations on BURN-1 until a suitable response to the display is provided.

The "MAKEPLAY" computations cause internal communication cells to be set for generation of the information required by the verb-noun pattern: verb 06 means display in decimal and noun 33 (page NNDF-22) references T_{ig}, the ignition time. After the display is produced ("HMSOUT", page NNDF-3, is entered due to noun table information, causing "DSPDECWD", page DATA-25, to load DSPTAB cells via "DSPIN", page DATA-27, for subsequent loading, once every 40 ms, in "DSPOUTSB", page GENP-7, into hardware output channel 10), the job is suspended ("put to sleep") in "ENDIDLE", page DINT-16, awaiting a crew response.

In general, responses to flashing displays are implemented in the program as a return to one of three addresses depending on the nature of the response. These are referred to as "terminate" (a crew input of V34E), "proceed" (a crew input of V33E or depression of the PRO DSKY button), or "otherwise". The "otherwise" return can be caused by a variety of DSKY inputs, although in general only one of them would be appropriate to the display presented: these include completion of a data entry, input of a recycle verb (V32E), or inputs of verbs 50, 51, 53, 59, 97, or 99. It should be recognized that the "otherwise" term is a document convention (in this document), and does not signify, for example, some sort of continuous program looping until a proceed/terminate response is given.

The various DSKY responses all exit to "RECALTST" (page DATA-33), where the job previously put to sleep is awakened. It starts executing at "IDLERET1", page DINT-16, where a check is made that if a data load verb was received, then the accompanying noun was what was requested originally: if not, the original display is regenerated. Following this, logic in "ENDRET", for the "VNFLASH" entrance (which set the VNFLAG bit), determines on DINT-18 where the return should go. A terminate transfers directly to "GOTOPOOH" (without returning to the caller), a proceed transfers to calling address +1, and an "otherwise" to calling address -1 (to pick up verb/noun pattern again). If the VNFLAG is not set, as true for e.g. "GOFLASH" (from e.g. "P76ER77", page BURN-33), then returns for terminate, proceed, and otherwise go to calling address +1, +2, and +3 respectively. For the "VNFLASH", since the terminate response is handled on DINT-18 rather than by the calling routine, the action for terminate is indicated in parentheses. The "GOTOPOOH" routine, page GENP-15, causes a request for a new program to be generated ("V 37 E" is effectively produced by the software).

If the response was a "proceed", then the program goes on to the next display (noun 81), as indicated on page BURN-1 by "proceed". The noun 81 display process closely parallels that for noun 33: noun 81 (page NNDF-28) causes a display of the velocity increment in local vertical coordinates. It should be noted that, by convention, this information is in the DELVLV$\underline{C}$ erasable memory cells: no coordinate transformations are done by the DSKY output routines. The scale of the display is XXXX.X fps, as implemented by constant $K_{sfot_{10}}$, page NNDF-17, with a suitable scaling shift. The terminate and "otherwise" responses cause a similar effect as for the first display, while "proceed" again causes computations to be started up on page BURN-1.

Following a proceed response to the second display, the XDELVFLG bit is set 1, indicating that an External Delta V burn has been targeted. After this setting has been accomplished, a P20 control bit (used to control updates) is reset and then the CSM state vector integrated to the input (noun 33) ignition time via integration driver routine "CSMPREC", page ORBI-2, entered with desired time in T_{decl} and exiting with position vector in $\underline{R}_{att}$ and velocity vector in $\underline{V}_{att}$. This information is used to retain the state vector at ignition for use by "S40.1" (page STER-1), and also to compute the apocenter and pericenter altitudes expected after the burn for display by noun 42 (page NNDF-23). Following the response to this display, a display of N45 (number of marks, time until ignition, and middle gimbal angle) is generated: the middle gimbal angle is set to -0.02°(by convention) if the orientation of the IMU is not known, and is computed in "GET+MGA" (page DISP-1) of the IMU orientation is known.

The N45 display uses a verb 16 instead of a verb 06: this means a monitoring function, giving an update (with new data if available) due to entrance to "MONDO" (page DATA-23) once a second from logic in "MONREQ" (same page). The middle register of N45 (R2) has data that is updated once a second also, namely T_{togo}, computed as $(T_{now} - T_{ig})$ by "CLOKTASK" (page BURN-28). After a PRO response to the N45 display, the "GOTOPOOH" routine, page GENP-15, is entered to request a new program as described previously for V34E responses to the display. Although there is no lockout requiring P30 to be performed prior to the burn program, subsequent computations make use of quantities computed in P30 (before the N42 display), so P30 should not be bypassed.

Assuming the IMU is on and aligned satisfactorily, after P30 the DAP parameters can be loaded via R03 (Routine 03), which is initiated by V48E (entering "DAPDISP", page DPIR-2). After loading the components of a particular noun, the calculations to incorporate this data in the parameters actually employed in the DAP are made. It should be noted that changes in the input masses of CSM (or LM, if LM is indicated as being attached) will not have an effect on the vehicle mass properties (moments of inertia and total vehicle weight used in estimating acceleration in the P40/P41 programs) unless a specification of the RCS DAP (LM-off or LM-on) is made in R03. R03 should be completed before P40 is selected, since some of the parameters loaded (trim estimates as well as vehicle weight) in R03 are used in the P40 computations.

Assuming the SPS burn program (P40) is desired, input of V37 E 40 E will cause the computations at "P40CSM", page BURN-24, to be entered. There, after doing some initialization and checking that the IMU is aligned (via "R02BOTH", page IMUC-1), routine "S40.1", page STER-1, is entered to rotate the velocity change input data (computed by P30) by half the estimated central angle of travel during the maneuver, and to compute for optional display the velocity change information in local vertical coordinates (via "GET.LVC", page DISP-1).

The "S40.1" routine exits to "S40.2,3" (page STER-2), which loads cells for the initial attitude required for the burn, as well as computes and loads a "preferred IMU orientation" (which makes gimbal angles read zero at ignition, and hence includes the effect of the pitch and yaw trims entered in Routine 03). This routine exits to "P40SXTY", page BURN-24, where a bit is set indicating that a preferred IMU orientation is available (for use in P52/P54 to generate the initial option code display). Then, after specifying that the maneuver should be of a "VECPOINT" (as opposed to 3-axis) type, "R60CSM" (page ATTM-2) is entered to perform the maneuver to the ignition attitude. P52 or P54 could be selected at this point to realign the IMU if desired. If one of these programs is entered (by V37 means), of course, it would be necessary to reselect P40 again.

Completion of the attitude maneuver routine causes a checklist code 0204_8 display to be produced, requesting a decision as to whether an SPS engine gimbal drive test should be done (PRO) or bypassed (ENTR). Response to this flash causes "S40.6" (page DPIR-4) to be entered to disable optics driving (since same computer counter cells used to control position of SPS engine bell) and to set the SPS engine to its trim value in pitch and yaw (as was specified by R03) after doing, if specified, the gimbal trim test. After completion of this routine, an interface cell is set causing "CLOKTASK" (which was started at the start of P40 for optional display) to establish "CLOCKJOB" which in turn displays non-flashing V06N40 that is updated once a second (by the "CLOCKJOB" computations) with time from ignition. Then the "MIDTOAV1" computations (page ORBI-5) are entered to integrate the state vector to T_{ig} - 29.96 seconds, the time when Average-G cycling is to be started. If the integration is not completed at least 12.5 seconds before the Average-G cycling is to be started, then an alarm is generated (pattern 1703_8) and ignition time is automatically slipped to 29.96 seconds from the time to which the state vector has been integrated (no crew response is required for this to occur).

Five seconds before Average-G is to be started, i.e. at T-34.96 seconds, the DSKY is blanked (except for the program number) due to "TIGBLNK" (page BURN-26) establishing "P40BLNKR" (page BURN-26). Five to six seconds later, depending on the phasing of "CLOKTASK" with respect to the "countdown" to ignition, the V06N40 display is restored again via "TIGAVEG" (page BURN-26), which sets NVWORD1 to the desired verb-noun pattern (causing "CLOKTASK" to establish "CLOCKJOB").

Five seconds before nominal ignition, NVWORD1 is set to - 400_8 in "TIG-5". The next time "CLOCKJOB" is entered (i.e. within a second), this negative non-zero setting will cause "CLOCPLAY" to be entered. This routine sets several display interface routine control bits, including the "BURNREQ" bit. When "NV50DSP" (page DINT-13) detects these bits, it causes a verb 99 to be placed on the DSKY to write over the original verb 06 used to generate a decimal display. This verb 99 permits an ENTR response to go to the "otherwise" return (via "GOLOADLV", page VBDF-8, per page VBDF-22): this response means that no SPS burn is desired, but post-burn display information should be supplied. A PRO response, on the other hand, means that engine ignition is authorized: this causes "V99P" (page BURN-30) to be entered in order to ignite the engine either promptly (if nominal ignition time has already arrived) or, by setting a flagword bit, at nominal ignition time if it has not yet arrived.

Five seconds after "TIG-5" was entered, namely at the original ignition time (or the time updated as a result of slipping ignition to complete the integration), "TIG-0" is entered (page BURN-27). This routine either enters "IGNITION"(same page) at once, if crew authorization has been received, or merely sets a bit that causes "V99P" to enter "IGNITION" if authorization received after the nominal ignition time. The "IGNITION" routine samples the outer gimbal angle for roll hold in the roll TVC DAP, turns on the SPS engine, if a "short" burn was deduced (from a job established by "TIG-5") sets a waitlist task to turn off the engine, disables the RCS DAP, and then delays 0.4 seconds. At the end of this interval, "TVCDAPON" is caused to be entered in 0.01 seconds to get the TVC DAP running (as discussed below); 1.6 seconds later, namely at 2 seconds from ignition, the ullage jets are turned off (they must be initiated manually, and remain at whatever setting they were left when the RCS DAP was disabled, since the "disabling" does not affect the RCS DAP TIME6 logic (see below)) and, if the burn is not "short" (i.e. predicted to be more than 6 seconds), bit 11(STEERSW) of FLAGWRD2 is set 1 to enable steering and computation of time-to go. This concludes the engine ignition sequence.

After P40 returned from the R60 attitude maneuver, the exit from the Average-G computations (AVEGEXIT) was set to "S40.8" (page STER-5). When "TIGAVEG" is entered 29.96 seconds before nominal ignition time, transfer is made to "PREREAD" (page GENP-1): this causes free-flight gyro bias compensation to be halted, clears the accelerometers, loads the Average-G state vector cells with information computed previously ("MIDTOAV1" results), and calls "READACCS" in 2 seconds (i.e. at nominal ignition - 27.96 seconds). "READACCS" performs in the same manner as described previously for the Boost phase, but instead of having "SERVICER" transfer to "VHHDOT", it transfers to "S40.8" (because of the AVEGEXIT setting).

"S40.8" (page STER-5) comprises the "guidance equations" for the flight. The first function that is done is the performance of "UPDATEVG" (page STER-4): for an External Delta V burn, that routine merely updates $\underline{V}_g$ based on the accelerometer output (DELVREF). "S40.8" next checks for the magnitude of the accelerometer output being below a pad-load limit (as it would be expected to be until the engine is turned on). If it is, and if the STEERSW bit is zero, the computations cease (STEERSW is reset in e.g. "INITSUB") until the next two-second cycle. The program remains in this mode of updating $\underline{V}_g$ after Average-G until the pad-load accelerometer output limit is exceeded. When this happens, the computer's internal estimate of the weight of the CSM vehicle (CSMMASS) is updated by two seconds' worth of nominal SPS mass flow rate, and, if STEERSW is still zero, the computations again cease. If a short burn is done (predicted duration in "S40.13", page STER-7, below 6 seconds), then STEERSW will remain zero; otherwise, it is set 1 in "IGNITION" 2 seconds after ignition.

After the STEERSW bit gets set, and assuming the accelerometer output is sufficient, CSMMASS gets updated, a TVC DAP quantity used for trim estimation is reset, and, after a check for thrust in the proper direction, a value of T_{go} is computed. If this value (which includes an erasable memory constant for thrust decay effects) is at least 4 seconds, then a new guidance steering command, OMEGAC, is generated for use by the TVC DAP. When it becomes below 4 seconds, "S40.81", page BURN-31, is entered. There, the guidance steering command is set 0 and the engine cutoff computations are called at the proper time.

If "S40.8" finds a single point below the accelerometer output threshold, with the STEERSW bit set, then thrust fail procedures are indicated (since the failure could be in the inertial sensor rather than the engine, however, no engine cutoff command is generated at this point). After a check for a bit set only for response to a fail condition, the guidance steering command is set 0, a DAP cell used for trim updates is zeroed, NVWORD1 is set -0, and STEERSW is set 0. The STEERSW setting reduces "S40.8" to the $\underline{V}_g$ (and possibly CSMMASS) update mode, while the NVWORD1 = -0 setting causes "CLOCKJOB" (page BURN-29) to generate a similar display to that for engine ignition authorization, except that the verb, instead of being 99, is now 97 (and, of course, different program branches are taken). An ENTR to the V97N40 display causes the T_{togo} display to be set to +59 59 (for reasonable times from launch), the engine to be shut off, the TVC DAP to be shut down, the RCS DAP to be started, a new "short" burn estimate via "S40.13" to be made, and finally, about 2½ seconds after the ENTR, a V99N40 display to appear (all this is done in "V97E", page BURN-29). The effect of the ENTR, therefore, could be considered analogous to returning to the T_{ig}-5 point, although of course the RCS DAP attitude hold reference probably would be different from that left by R60. If a PRO response is provided (indicating that the engine is on), then "V97P" is entered: this resets STEERSW back to 1 and sets bit 6 of FLAGWRD1 to 1 for 2 seconds to inhibit the "S40.8" failure action. A V34E response to the V97 causes engine shutdown, followed by startup of the RCS DAP and entrance to "GOTOPOOH". It should be noted that a crew response to the V97 flash is necessary before any action is taken by the software to shut down the engine or disable the TVC DAP (the software does, however, zero guidance steering commands and disable trim updates by the DAP).

The normal engine cutoff routine is "ENGINOFF" (page BURN-31), which shuts off the engine, waits $2\frac{1}{2}$ seconds, starts up the RCS DAP (after a time delay to avoid interference with TVC DAP shutdown), and finally establishes "POSTBURN" (page BURN-32) to provide a V16N40 display. After a PRO response to this display, AVEGEXIT is loaded with "CALCN85" (page DISP-2), which likewise enters "UPDATEVG" but which then transforms the velocity-to-be-gained information into RCS control coordinates (rotated $7\frac{1}{4}$ degrees from body axes) via "S41.1" (page STER-8). This information is used to drive a V16N85 display initiated by "TIGNOW" (page BURN-32). After a response to this display, "GOTOPOOH" is entered (which, via "INITSUBA", restores the RCS DAP deadband to the value last specified by the crew), thus ending the External Delta V sequence.

External Delta V Burn (RCS)

The initial preparations for an External Delta V burn using the RCS instead of SPS propulsion system can follow the same procedures as described for the SPS burn, since selection of a propulsion system is not made until the burn program is chosen. Following P30, input of V37 E 41 E will cause the computations at "P41CSM", page BURN-32, to be entered. There, after doing some initialization, "P40S/F", page BURN-24, is entered in order to take advantage of the similarity between the P40 and P41 burn initialization computations. Following return from the maneuver routine, the P41 sequence starts a display of the velocity-to-be-gained in RCS control coordinates, using V06N85. This display is initialized in "P40SXTY", and is updated once a second (by "CLOKTASK" establishing "DYNDISP", page BURN-28). Following this, the "MIDTOAV1" routine is entered to request the state vector at 29.96 seconds before ignition, with return options as for the SPS case, and the DSKY is blanked by "TIGBLNK" 5 seconds before Average-G turn-on, again as for the SPS case.

When "TIGAVEG" (page BURN-26) is entered to start Average-G at 29.96 seconds before the nominal ignition time, a check is made to conclude that P41 (as opposed to P15 or P40) is running. If it is, the DSKY display is made V16N85, and "TTG/0" is called at the nominal ignition time. The updating of the N85 information by "DYNDISP" is no longer done, since the updates are now based on the exit of Average-G to "CALCN85" (set in "P40SXTY" after return from R60) which, as discussed for this computation in the final P40 display, does its own coordinate transformation (as well as entering "UPDATEVG" to update velocity-to-be-gained based on accelerometer outputs). The only functions of "TTG/0" (page BURN-33) are to establish "TIGNOW" (page BURN-32), causing the V16N85 display to flash, and to disable the cycling of "CLOKTASK" by setting a flagword bit to 0. Termination of the display (by a terminate or PRO response) causes the "GOTOPOOH" routine to be entered (as in the P40 case).

Monitoring of thrust applied, with Average-G running, can also be done in P47, which starts at "P47CSM", page BURN-33. There, after checking for IMU alignment ("R02BOTH"), routine "MIDTOAV2", page ORBI-6, is entered. This routine performs computations similar to "MIDTOAV1" (described for the SPS case), except that it uses the "ignition slipped" logic, requesting for each orbital integration time step an integration to 12.5 seconds after the present time, and halting (in general) when the given time step successfully completes the integration. This effectively permits Average-G to be turned on "as soon as practical" after entrance to P47. "TIGON", page BURN-33, is called to start Average-G, to set the Average-G exit to "CALCN83", and to generate in "P47BODY" (page BURN-33) a V16N83 display of velocity change. "CALCN83", page DISP-1, updates the velocity change (converting it into RCS control coordinates for the N83 display), and also computes the quantities displayed by N62 (for optional crew observation of them by a manual DSKY input): these quantities are the same as those supplied by the P11 Boost program, except the altitude base can be the landing site (for moon-centered computations).

Rendezvous Targeting

Instead of having a fixed externally provided velocity change to be achieved by a burn, as is the case for P30, a capability is also provided to have the burn program parameters determined internally in order to achieve rendezvous. A capability also exists to perform targeting for the "other vehicle" (i.e. LM) by input of program P7x (x = 2-5) rather than P3x. The various rendezvous programs can each be sequenced manually, or an automatic sequence, reducing the number of crew keystrokes required, can be performed.

If a manual input of V37 E 3x E is made (x = 1-6), or if P79 is keyed in, from a "base configuration" of no rendezvous programs running (e.g. from P00), and if the REFSMFLG is set (IMU orientation known), then "V37" (page GENP-15) exits to "REND3OS", page MINK-1. There, an initiation of the P20 program is performed by setting the AUTOSEQ bit = 1, entering "AUTOSET" (page MINK-2), transferring to "AUTO37" to complete the processing normally done for a V37 (page GENP-16), and finally starting at "PROG20", page ORVN-1. There, since the AUTOSEQ bit is 1, displays are suppressed and "NDUTINPT", page ORVN-3, transfers to "AUTOCHK", page MINK-1. There, since AUTOSEQ bit is 1, transfer is made to the address in AUTPOINT, which "AUTOSET" had retained as the return address to "REND3OS". Hence "AUTO37" is again entered, this time with the desired program equal to TEMPMM, i.e. in range of 81-86 (due to the start of "REND3OS"). If P79, that number used.

"AUTO37" again performs the standard V37 processing, this time with P79 or the program number of 81-86. The table on page GENP-39 shows that in this case transfer is made to these program starting points ("Pxy"), all of which are in the Minimum Key Rendezvous writeup. If P31 originally keyed in, for example, then "P81", page MINK-3, is entered. There, the first thing that is done is performance of "MINKDISP", page MINK-1, which first writes over the "81" of the program number with "31". Then a checklist code display is generated (checklist 00017_8), requesting a decision as to whether the "automatic" or "manual" sequence is desired. An ENTR means the manual sequence, causing AUTOSEQ bit to be set 0 and return to "P81". There, via "AUTOSET", the P31 computations (page BURN-3) are entered. The final display response there (exit from "VN1645", page BURN-18), transfers to the second line of "GOTOPOOH", page GENP-15. There, a performance of "AUTOCHK", MINK-1, will reveal that the AUTOSEQ bit is zero and hence the V37 display is flashed in the same fashion as described previously for e.g. P30 termination.

Except for P79, the other rendezvous targeting programs mentioned can use a similar "manual" mode of operation. P31, P32, P33, and P36 use the External Delta V interface with the burn programs, while the P34 and P35 targeting programs use another interface.

This other interface is employed when it is desired to arrive at a certain point at a certain time. One program making use of this "Lambert" scheme is P37, to which the complete Return to Earth Computations (starting on page RTER-1) writeup is devoted. These Lambert targeting programs reset a flagword bit (XDELVFLG) to 0 to indicate to P40/P41 that a Lambert burn, as contrasted with External Delta V, has been targeted. As previously mentioned, the rendezvous programs in the BURN writeup making use of the Lambert capability are P34 and P35.

The major Lambert computation package is entered via "INITVEL", page REND-8, which includes various scaling and other initialization functions before entering "LAMBERT" (page CONC-5). If a Lambert burn is specified, "UPDATEVG" (page STER-4) causes an accumulation of velocity increment information to be maintained until the Lambert computation is complete, and then updates the velocity-to-be-gained vector: as for the External Delta V case, "S40.8" continues to be performed for steering and T_{go}. If "UPDATEVG" discovers that a Lambert solution has been generated (checking a cell that is also set negative in "P40SXTY" for P40/P41), then "S40.9" (page STER-6) is established. This job enters "INITVEL" to do the Lambert calculations, and then, after correcting for earth oblateness gravity effects on the target (if necessary), sets the checking cell NBRCYCLS negative to indicate to "UPDATEVG" on its next pass that a new solution is available. In addition to this (comparatively slight) change in the burn calculations, "S40.1" must use a different procedure to calculate the required thrust direction for the burn: this is done in "S40.1B", page STER-1.

The P79 computations are not actually a "targeting" set for a burn, but instead are employed for final phase monitoring. As previously mentioned, a manual input of P79 causes "P79", page MINK-3, to be entered. There, "STARTAUT", MINK-2, is entered to set some initial conditions before returning (due to AUTPOINT loading in "P79") to "P791". At this point, the AUTPOINT cell is set to "P79A". As a result of this, and the setting of PCMANFLG to 1 at the start of "P79", the P20 computations initiated at "PIKUP20" (page ORVN-4) will transfer via "AUTOCHK" to "P79A". There, the DSKY display of R31 is initiated, and then return made to "P20TRACK", page ORVN-5.

If, instead of an ENTR to the checklist 00017_8 display, a PRO response had been given (again considering the P31 example), then "MINKDISP" would have entered "STARTAUT", where after some initializations a return (via AUTPOINT) to "P81" would have been done. There, via "AUTOSET", the usual V37 program-change logic would, as in the ENTR case, have led to "P31" with eventual exit to the second line of "GOTOPOOH". This time, however, the bit AUTOSEQ would still be 1, so that "AUTOCHK", page MINK-1, instead of returning to "GOTOPOOH" for the V37 display would have transferred to the address specified by AUTPOINT (i.e. the address loaded by "AUTOSET", the 4th line of "P81" on page MINK-3). Here, transfer to "BURNHOW", MINK-2, is done to determine the burn program to be selected. If the required velocity change is below 7.0 fps, then P41 is selected; otherwise, P40 is selected (in either case, AUTPOINT now contains the return address from "BURNHOW").

Regardless of which burn program is chosen, "TIGNOW", page BURN-32, exits (for the nominal sequence) to the second line of "GOTOPOOH", where again entrance to "AUTOCHK", since AUTOSEQ bit = 1, causes transfer to the AUTPOINT information, which by now has the address of the 5th line of "P81". This causes "AFTERBRN", MINK-6, to be entered, which saves the proper return address and establishes "HARTBURN" (same page). Here, computations are made of the proper estimate of the burn that was done: if it is deduced that the CSM did the burn (SPS burn or a reasonable deflection of the translation hand controller during Average-G, both of which set BURNFLAG), then the velocity increment is set 0; otherwise, it is derived depending on the program anticipated to have been used by the other vehicle. In any case, P76 is then entered ("P76ER77", page BURN-33) to update the other vehicle's state vector and then, again via exit to 2nd line of "GOTOPOOH", returning to the AUTPOINT address (now the 6th line of "P81"). This causes transfer to the second line of "P82" (page MINK-3), where an analogous sequence is started, this time using P32. The subsequent sequencing of the programs, ending with P79 (at the end of "P85"), can be seen from the other material in the MINK writeup. The P8x programs are not intended for direct DSKY entrance, but instead represent a convenient way to make use of the V37 processing logic: they can be considered as the "driver" for the various pre-thrust, thrust, and post-thrust (P76) programs, with the pointer as to where the present sequence has progressed being contained in the AUTPOINT cell.

IMU Alignment

IMU orientation determination is accomplished (for use of normal optics) by P51, which starts at "P51", page INFA-1. After a check that the IMU is on (if it is not, alarm 0210_8 is generated and "GOTOPOOH" is entered), an option is available to coarse align to 0° for all 3 IMU gimbal angles; otherwise, routine "R53" (page OPTC-20) is entered twice to obtain sighting information to a pair of manually selected bodies (one of 37 stars stored in fixed memory, the sun, the earth, or the moon (positions likewise computed by CMC)): if a code of 00 for the "celestial body" is entered, then a unit vector is requested by N88 input (see "PLANET", page INFA-9). Following sighting of the two bodies and specification of their identification, routine "R54" (page INFA-13) is entered to display the difference between the measured angle between them and the "theoretical" angle. After a response to this display(indicating acceptance), the reference-to-stable-member matrix ([REFSMMAT]) deduced from the sightings is computed and stored, a bit set (REFSMFLG) indicating that a known IMU alignment (valid [REFSMMAT]) is considered to exist, and "GOTOPOOH" is entered. If backup optics is used, V37 E 53 E is keyed in, which causes "P51" to be entered too, but which causes "P51B" (page INFA-2) to perform "R56" (page OPTC-22) to obtain the sighting data instead of "R53". Aside from this difference, the computations associated with P51 and P53 are the same.

Once the IMU has been aligned (the REFSMFLG set), then P52 (starting at "PROG52", page INFA-3), for use of normal optics, can be used to perform realignment. Four options are available for this program: the "preferred orientation" (computed by P40 or P41, or uplinked: the PFRATFLG bit is used only to generate the initial option code display); the "nominal orientation" (local vertical at a specified time); the "[REFSMMAT] orientation"(to get rid of the effects of uncompensated IMU drift since the previous alignment);

or the "landing site orientation" (tied to RLS site stored in computer, at a specified time). All except the third can do coarse alignment to the new angles (if any angle change exceeds 1^{o}); all then perform an automatic star selection ("P52C", page INFA-6) if desired and then end by entering "R51", page INFA-11, to complete the processing of observation data. Instead of doing an IMU coarse alignment, it is also possible to change the IMU orientation solely by gyro pulse torquing, in "GYCRS", page INFA-14.

"R51" begins by displaying the star code selected for the first star: this may be changed, or some other body loaded, if desired. "PLANET" is then entered to compute the unit vector to this body, and then "R52" (OPTC-14) is entered. This routine performs automatic optics positioning (if optics in proper mode), and requests marks via "R53" when optics first switched out of computer control. Shaft stop avoidance logic is included in "OPTTEST", page OPTC-1. After two bodies have been sighted and the "R54" display approved, gyro torquing angles are displayed in "R55" (page INFA-14). If these are approved, the IMU gyros are pulse torqued so as to make the information now in [REFSMMAT] (which was updated as necessary when the coarse align check of "CAL53A", page INFA-11, was done) indeed reflect the orientation of the IMU in inertial space. If the backup optics is used, V37 E 54 E is keyed in, which causes "PROG52" to be entered too, but which causes "R56" (as in P53) rather than "R52" to be selected. Aside from this difference (no automatic optics pointing is provided since R52 is not entered), the computations associated with P52 and P54 are the same.

Tracking and Rendezvous Navigation

Considerable elaborateness is included in the program in order to allow tracking and, if desired, rendezvous navigation computations to be carried out while other computations are running. These computations are conducted under the auspices of program P20, which starts at "PROG20", page ORVN-1. The automatic entrance to this program for Rendezvous Targeting purposes (AUTOSEQ bit = 1) has already been discussed. A manual selection (from e.g. P00) of V37 E 20 E, of course, also causes "PROG20" to be entered. In this circumstance, the first display provided requests a choice to be made of the desired tracking option: #0 is LM with "VECPOINT" (axis) pointing; #1 is celestial body with "VECPOINT"; #2 is rotation about a body vector; #4 is LM with three-axis control; and #5 is celestial body with three-axis control. Following this, an N78 display (in "DOV6N78", page ORVN-1) for specification of the body axis of interest is provided, followed by an N79 display of the desired tracking deadband and, for #2, the maneuver rotation rate. This is the final display in the initialization sequence for options #0 or #4 (the "rendezvous tracking" options).

Options #1, #2, and #5 are considered to be "universal tracking", as opposed to "rendezvous", options, and result in the setting of the UTFLAG (bit 9 of FLAGWRD8). Operation in this mode of the program involves only special spacecraft attitude control logic, and allows some special program performance features (such as entrance to P27) that are excluded from the rendezvous mode. Hence it is important to be explicit in identifying which mode of operation of P20 is taking place if the performance of the CMC software in some circumstance is of interest.

If option #2 was specified, the next display is of V06N34 (in "P200PT"), to indicate the time when the maneuver is to be started. This is the final display in this option before entrance to "NDUTINPT", page ORVN-3. If option #1 or #5, however, was specified then "V1N70DSP", ORVN-3, is entered so that the desired body can be specified via the V01N70 display (and, for the "planet" option of 00, the V06N88 display). Exit is then made to "NDUTINPT", where the UTFLAG is found 1 from "P200PT". As a result, "STATINT1" is established and then exit to "PIKUP20", page ORVN-4, is made. The "STATINT1" routine, for periodic state vector updating, is the same routine that is established by the V37 logic if P00 is selected, and has been previously described. The "PIKUP20" routine, also entered due to restart group settings e.g. after Average-G termination, after checking flags exits (since UTFLAG = 1) to "CALLR6X".

"CALLR6X", page ATTM-21, determines whether the option selected was 1/5 or 2. If #1/5, exit is to "R66CSM", page ATTM-22, which enters "R61CSM", page ATTM-14, every half second. There, the required attitude to point the specified vehicle axis at the specified point is determined ("UTAREA1", page ATTM-19, plus, for 3-axis maneuver, "UTOPT45" on ATTM-20). This loop is continued until interrupted (e.g. by resetting the TRACKFLG bit). If, on the other hand, option #2 was chosen, then "CALLR6X" delays until the N34-specified time is reached, and then enters "R67", page ATTM-22. There, the desired rate is specified and then the terminal part of the standard R60 automatic maneuver routine ("NEWANGL", page ATTM-11) is entered. The program stays in the one-second maneuvering loop until some other action occurs (the "predicted time of maneuver completion" is set to a large number), or an update of some matrix elements is necessary (about once every 34 minutes). No jet "forced firing" occurs at startup.

For the rendezvous options, after the response to the V06N79 display exit is to "NDUTINPT", with, however, the UTFLAG = 0. As a result, both vehicle state vectors are integrated to the present time, the "R22" measurement data incorporation routine is established, and "PIKUP20" is entered. Here, "R61CSM" (ATTM-14) is entered, followed by entrance to the "R52" optics pointing routine which causes "R61CSM" to be initiated periodically to maintain the proper vehicle attitude.

The "R61CSM" routine, page ATTM-14, enters "CRS61.1", page ATTM-15, to do the necessary computations. If it is concluded that a maneuver is necessary (gimbal angle change exceeds 10^o or, for "VECPOINT", pointing angle error with respect to DAP deadband center, <u>not</u> present attitude, exceeds 10^o), then exit to "STKTEST", page ATTM-17, is made. There, logic is performed to cause an estimate of the maneuver time to be made by the R60 computations, and then the relative positions of the CSM and LM are recomputed for about 20 seconds after the end of this time interval (via "RCYCLR61", page ATTM-18), so as to minimize the effects of relative geometry changes during the maneuver. If a maneuver is still indicated, and is appropriate (determined by other bit checks), then "R61CSM" will cause "R60CSM" to be entered to carry out the maneuver. At the end of the maneuver ("ENDMANUV"), if both AUTOSEQ and TRACKFLG bits = 1, an exit from the R60 logic is made. Special restart provisions are made on ATTM-3 ("CHKLINUS"/"RELINUS") to handle the case of e.g. target program selection after P20 initiated (although the automatic rendezvous logic should make this an unlikely procedure).

The optics pointing computations, "R52", start on OPTC-14. For the LM tracking case, "R52H" periodically enters "R61CSM" and then "R52D" to update the optics pointing angles. Due to the setting of the R21MARK bit in "PIKUP20", optics marks can be taken at any time, causing entrance to "MARKRUPT", page OPTC-9, where, from "MARKDIF", optics marks are stored in an interface buffer, "PUTMARK", for use by R22.

The measurement incorporation portion of P20 is "R22", which is established by "NDUTINPT" and which starts on page MEAS-8. Once entered, this job runs with priority 26_8, and therefore takes precedence over the R52 loop that runs with priority 14_8 (due to "PIKUP20"). It should also be noted that the restart group settings (e.g. 2.7 and 1.11) for both R22 and the "PIKUP20" area are only priority 10_8, whereas V37 processing establishes jobs with priority 13_8 ("V37XEQ", page GENP-22). As a result, even though for most program changes the TRACKFLG is reset in "R00", if tracking is still appropriate for the new program (such as P30 or P21), the new program will be initiated, will set the TRACKFLG = 1, and will pause at its first display before the effect of the P20 restart groups is implemented. Hence when "PIKUP20", for example, checks TRACKFLG it would be 1 if the P20 computations are allowed to run in the background of the newly selected program. After some initial conditions, R22 enters "WAITONE", page MEAS-17, causing a 4 second delay in R22 performance. Then, if R61 is not doing a maneuver and updates are allowed, return is to "REND1", page MEAS-9, where the interface optics buffer "PUTMARK" loads is checked. If it contains mark data, it is sampled for use and the R22CAFLG bit is set. This bit is employed in "MKREJECT", OPTC-11, to determine if optics mark processing is being carried out: if set, then the REJCTFLG bit, checked at the end of "REND7", is set. Hence if it is desired to reject an optics mark, it is necessary to do so before the R22 computations have processed it (assuming the mark was not so erroneous that the N49 excessive update display is triggered). VHF ranging marks are also obtained periodically in the R22 loop by "REND3", page MEAS-9, if authorized (V87E or "STARTAUT", page MINK-2) and if the estimated separation is below the 327.67 nmi interface saturation level (i.e. if RANGFLAG, controlled at the start of "CRS61.1", is set).

If computations at the end of "REND7" (page MEAS-13) conclude that either the position or the velocity state vector update is "excessive" (bigger than tolerances stored in erasable memory), then "RENDISP", page MEAS-16, is entered. This routine establishes "RENDISP2" with priority 27_8 (same page) to generate a priority display using V06N49 and awaits crew response. Thanks to the "WAITONE" logic (for optics and for radar except when waiting for radar sample to accumulate), it is known that the R60 priority display is not on the DSKY, and to avoid having it be initiated "RENDISP" performs a short loop checking a communication cell with "RENDISP2" to determine when a reply is received (and of what type). R22 and R52 cycling can be halted by V56E (causing "TRACKTRM", page VBDF-9, to be entered) or by selection of P00. Cycling is suspended (but can be started up again) if the TRACKFLG = 0 (e.g. entrance to a burn program). Backup marks can be initiated by keying V54E, causing "R23CSM", page OPTC-13, to be established. If the AUTOSEQ bit is set, "REND4" (MEAS-11) exits to "AUTOW" (MINK-7) to check for automatic W matrix initialization.

Other Navigation Programs

Another navigation program is P21 (which starts on page ORVN-5). This program merely gives the latitude, longitude, and altitude at a specified time (incremented automatically by 10 minutes if desired). Other quantities are also computed for optional display. A similar program, P29 (ORVN-20), may be used to derive the time (after an input time) when the spacecraft will arrive at an input value of longitude.

A "cislunar midcourse navigation" program, P23, is provided which uses sextant trunnion angle as the sole measurement input (to measure the angle between a star and a known "landmark" or the horizon). This program starts on page MEAS-18, and requires a trunnion angle calibration via "R57" (page OPTC-23). If it is desired to incorporate a series of marks in this program with the option of being able to restore the unmodified state vector, this can be accomplished by performing V66E before the start of the marks (to put the CSM state vector into the "LM" cells), and then, after the marks, performing V47E (which moves the unupdated "LM" cells back into the CSM "permanent" state vector).

A fourth navigation program is P22, which can be used to perform state vector modification based on the tracking of known or unknown landmarks (on earth or moon): the "landing site" vector can be selected, or a landmark loaded manually (only the manual option is allowed for earth). P22 starts on page ORVN-7 ("PROG22"). After some initial presets and selection of sighting options, "R52" is entered to perform optics pointing to the desired landmark (after R52 starts, N43 could be used to change the coordinates towards which the optics is pointing). After completion of the mark sequence, "DOV5N71" is entered (page ORVN-8) to display the landmark selection for crew approval, and then "S22.1", page ORVN-11, is entered to do the actual processing of the observation data. The data is placed on the downlink (in "MARKDIF", page OPTC-10) as it is received.

"S22.1" performs some initialization settings, including different W matrix initialization for known and unknown landmarks, and then processes the mark data. A flagword bit is set for the processing of the first non-offset mark (22DSPFLG, bit 13 of FLAGWRD2) that causes "S22BOX12" (page ORVN-16) to display via V06N49 the magnitude of the first update for crew approval: if it is approved, the remainder of the updates (including the second for that mark) are incorporated with no check on their validity (contrary, for example, to R22). Following the incorporation of the last mark, an option is provided (in "S22I=N", page ORVN-17) to update the landing site $RL\underline{S}$ based on the derived position of the landmark being tracked (which could have been specified as the serial number of 01, meaning $RL\underline{S}$). A manual input to the N89 display will not be effective in changing what will be loaded into $RL\underline{S}$, since although it changes the N89 cells, it does not change the vector information used to derive the $RL\underline{S}$ setting. Finally, the W matrix is reduced to 6x6 (landmark correlation information is not carried over from one landmark to the next, of course) in "9DWTO6DW" (page MEAS-5), and then a new set of marks can be obtained. If desired, the W matrix values can be observed, and initialized to different values, by input of V67E, causing "V67CALL", page MEAS-4, to be entered.

Another program that is provided in connection with navigation is P24 (starting at "PROG24", page ORVN-19). This program provides a rate-aided optics drive and downlinking of the mark information as it is taken. A capability is also provided to update the estimate of landmark location used for the optics commands based on optics mark information, although this capability can be disabled if desired by suitable erasable memory initializations.

Entry

Input of V37 E 61 E will permit "P61" (page ENTP-1) to be entered. Average-G is turned on via "S61.1" (page ENTP-4) which causes "MIDTOAV2" to be entered (the same routine as used by P47) in order to get the Average-G computations started (generally in 12.5 seconds from the beginning of the last time step). After this is done, the IMU alignment is checked in "S61.1A" (page ENTP-5), and then a N61 display of target parameters is supplied. Response to this display causes "NEWRNVN", page ENTP-1, to be entered to sample the state vector from the same two-second Average-G cycle (if the sampling is unsuccessful, the process is repeated). Routine "S61.2", page DISP-15, is entered to compute a number of quantities for display purposes, including Entry Monitor System (EMS) information. The N63 display (which includes time from EMS interface altitude) is updated every 2 seconds in "SERVICER" (page GENP-4), and is also corrected for the computation delay since sampling of state vectors. In addition, a V32E response to the display will cause the display parameters to be recomputed, and the N60 display to reappear. A PRO response causes "P62" to be entered.

"P62", page ENTP-2, can either be entered from "NEWRNVN" or can be selected directly by V37 means (see page GENP-39). It performs "S61.1" again (if from "NEWRNVN", Average-G is already running), starts the entry DAP going ("READGYMB", page DPEN-1), and then starts a flash of checklist code 00041_8 to request separation. A PRO response to this display causes "CM/DAPON" to be entered to activate entry DAP computations (page DPEN-1), and then "P62.1", page ENTP-3, to be entered. Here, after another N61 display of target information, a display of entry angles (via N22) as computed by "P62.3", page ENTP-4, is provided (unless the trim angle already is less than 45^o, in which case P63 is entered immediately). Logic in "EXDAP", page DPEN-11, causes "WAKEP62" (page ENTP-4) otherwise to establish "P63".

"P63", page ENTP-4, changes the program number display to P63, sets the verb/noun display to V06N64 (for use in "ENDEXIT", page ENRY-16), and causes "STARTENT", page ENRY-2, to be entered. This routine sets some initial conditions for the entry program and then causes the computations to enter "SCALEPOP" (page ENRY-3) and then "INITROLL" (page ENRY-5). The "INITROLL" computations monitor the status of the bit indicating 0.05 g (the bit is set/reset in "TARGETNG", entered from "SCALEPOP"), and when it first becomes one changes the program number display to 64 (and the verb-noun to 0674). If the velocity at this point is less than 27,000 fps (K_{vfnl}), the exit address from "SCALEPOP" is set to "KEP2" (page ENRY-12): this causes exit to P67 (via "PREFINAL", page ENRY-12) when the drag exceeds 6.5 fps^2. If velocity is above 27,000 fps, then a more complete set of entry computations can be done (the setting of INRLSW keeps the 27,000 fps check from being performed more than once).

"PREDICT3", page ENRY-13, is the "final phase" of the entry computations, and is entered until the velocity is below 1000 fps, when "P67.1" (page ENRY-16) is entered instead to display V16N67 (computed by "P67.2", same page) until a response is received, when the "GOTOPOOH" routine is entered to stop the computations.

Digital Autopilots

Four distinct digital autopilots are included in the computer program: the Saturn DAP (mentioned in the Boost phase discussion, and used only to generate outputs based on hand-controller deflections); the RCS DAP (used for free-flight CSM control); the TVC DAP (used for SPS thrusting, and including a roll control section using RCS jets); and the Entry DAP. A capability also exists to "turn off" all DAP's; by performing routine R03 (V48E, causing "DAPDISP", page DPIR-2, to be entered) followed by V46E (causing entrance to "STABLISH", page DPIR-1), manual initiation of the Saturn, RCS, or no DAP can be accomplished.

RCS DAP operation is started by verb 46, as mentioned, which causes "STABLISH" to perform "RCSDAPON" (page DPRC-1). This routine sets T5LOC (the address to which transfer is made when computer clock TIME5 overflows) to "RCSATT", and causes TIME5 to overflow in 0.6 seconds. When the clock overflows, "RCSATT", page DPRC-1, is entered. There, provided that G&N control of the spacecraft is set, the DAP is initialized (via "REDAP", page DPRC-4). This includes initialization of the Kalman filter used for determination of the spacecraft attitude rate: while this initialization is taking place (about 1.1 seconds to obtain the necessary ten first-differences of CDU angles), no jets are fired, not even those which might be commanded by manual inputs (such as the translational controller). The complete DAP "computation cycle" is 0.1 seconds, and involves the sequence shown on page DPRC-31. Every other pass through the DAP (or every 200 ms), the attitude error needles on the FDAI are updated (using computations performed in "KMATRIX", page DPRC-3) based either on internal DAP-generated error signals or on the difference between the "desired" attitude and the present CDU angles. Selection of the type of display, and its optional initialization to present attitude, is made by verbs 60-63. Automatic operation for the performance of attitude maneuvers, also used for performance of rotational commands (in "T5PHASE2", page DPRC-6, logic), is controlled by the quantity HOLDFLAG, described on page DPIR-14. The selection of the jets to be used, and the checks for translation inputs, is made starting at "JETSLECT", page DPRC-12, and includes compensation for jet failures that have been reported via R03 (see pages DPIR-28 and DPIR-29). Actual control of the jet on-time is accomplished by "T6START", page DPRC-22.

Transition from the RCS DAP to the TVC DAP is handled by the program, and cannot be accomplished manually. As mentioned in the discussion of the SPS burn program, the generation of translation-control inputs to the computer for ullage generation must be done by the crew. When "IGNITION" (page BURN-27) is entered to turn on the SPS engine, it halts the performance of the RCS DAP (by setting T5LOC to an idling location). 0.4 seconds later, T5LOC is set to "TVCDAPON", with an associated delay of 0.01 seconds. Consequently, 0.41 seconds after ignition "TVCDAPON", page DPTV-1, will be entered. This routine sets a number of DAP cells zero, and causes "TVCINIT1", page DPTV-1, to be entered 0.01 seconds later. There, other non-zero initial conditions are set, and "MASSPROP", page DPIR-9, is entered to compute mass properties of vehicle (in "FIXCW", same page) after initializing cells for "FIXCW" that remain undisturbed during the TVC DAP operation (hence, to save execution time, "FIXCW" is entered during burn to update vehicle mass properties). "TVCINIT1" exits to "TVCINIT4" (page DPTV-3), which calls "TVCEXEC"in 0.51 seconds, or about 0.93 seconds after ignition, and also causes "DAPINIT", page DPTV-3, to be entered a DAP half-cycle later (for LM-off, 20 ms; for LM-on, 40ms).

"DAPINIT" samples CDU_y and CDU_z for use in forming the past values of these quantities (to obtain in turn rate information), and sets T5LOC to "PITCHDAP" with a corresponding delay of a DAP cycle (for LM-off, 40 ms, bringing the total time since ignition to about 0.48 seconds). "PITCHDAP", page DPTV-6, calls (via T5LOC setting) "YAWDAP" (page DPTV-8) in a DAP half-cycle; "YAWDAP", in turn, causes "PITCHDAP" to be entered in a DAP half-cycle: for LM-off, this causes SPS actuator output information to be updated every 40 ms. Guidance commands start the first entrance to "S40.8" after STEERSW = 1 (setting to 1 done 2 seconds after ignition, if a "long" burn).

"TVCEXEC", page DPTV-4, is entered (due to "TVCINIT4" waitlist setting) for the first time at about 0.93 seconds after ignition. Here, a roll attitude error is computed and the FDAI error display is updated. Every 10 seconds, "FIXCW" is entered to update the vehicle mass-properties related information (see CSMMASS on page DPIR-11 for the updating history of the cell used as the independent variable for these computations). Provision is made for updating the pitch and yaw trim estimation after a suitable time interval has elapsed (for LM-on, no time delay is imposed before the first correction). In addition, every 0.5 second ("TVCEXEC" keeps calling itself at that rate until the TVC DAP is halted), 0.03 seconds after "TVCEXEC", the TVC roll DAP, starting at "ROLLDAP" (page DPTV-11) is entered to compute a jet on-time for roll attitude hold about the outer gimbal angle value sampled in "IGNITION". Contrary to the RCS DAP, no jet failure information is used; instead, quads are alternated separately for both positive and negative torques.

When the engine cutoff command is generated, "SPSOFF", page DPIR-4, is entered to turn off the engine. In addition, new trim values, if appropriate, are also loaded. For the normal cutoff via "ENGINOFF" (page BURN-31), "RCSDAPON" is entered 2.5 seconds later, causing "RCSATT" to be entered 0.6 seconds later to start the RCS DAP as described previously. If "S40.8" detects a thrust failure, trim updates are halted (and mass updates there while the indicated failure persists), but it is not until a response to the V97 display in "CLOCKJOB" is received that there is a possibility of having the software cut off the engine and TVC DAP (barring some phenomena as a fresh start, of course).

As mentioned with the Entry phase, entrance to P62 starts the performance of the gimbal angle sampling for the entry DAP; after crew response to the request for separation confirmation, "CM/DAPON", page DPEN-1, is entered to disable the RCS DAP (by setting T5LOC to "T5IDLOC", page DPIR-1) and to set some initial conditions and flagword bits to permit "READGYMB", which is performed as a waitlist task every 0.1 second, to continue on to "BODYRATE", page DPEN-3. There, T5LOC is set to "ATTRATES" (page DPEN-4) with a corresponding delay of 0.01 second. In "ATTRATES", a check is made of the entry guidance bit indicating 0.05g's. If drag is below this level, "EXDAP", page DPEN-11, is entered; if above it, then pitch and yaw rate damping is performed and channel 5 loaded appropriately.

If "READACCS" (page GENP-2) senses that the entry DAP is active, it saves some angles computed by "READGYMB" and "ATTRATES", and sets a waitlist call for "SETJTAG" (page DPIR-1) to set a quantity positive non-zero. This setting, which occurs once every 2 seconds, causes "EXDAPIN" (page DPEN-5) to continue with computations leading to a strategy for firing roll jets. The appropriate pattern is loaded into channel 6 in "JETCALL2" (page DPEN-10), and waitlist calls are set for replacement of the pattern (if necessary). Checks are made in "GETON2", page DPEN-7, that the sum of the waitlist-call times will not exceed 2 seconds. Following this, "CM/FDAIR" (page DPEN-10) is entered to display errors on the FDAI or to save quantities for downlink transmission (if the "SETJTAG" task is not done, then "CM/FDAIR" is entered directly from "EXDAPIN").

Telemetry

Every 20 ms (nominally, for the high bit rate), a signal is received from the telemetry system which causes "DODOWNTM", page TELE-1, to be entered. Five telemetry lists, plus a "dump" of all cells in erasable memory, can be sent, as described on pages TELE-9 through TELE-36. Most changes to the selection of the downlist sent are made as part of the V37 sequence (programs with their downlists are given on page GENP-39), although changes also take place elsewhere (such as at the end of "NEWRNVN" for the transition from P61 to P62, at the start of "P11", and in "V73UPDAT", page UPLK-1, for uplink verbs causing P27).

Notation and Terminology

Because of some of the special design features of the Apollo Guidance Computer, a set of special notation and terminology has been found useful in describing the equations programmed for this computer. In most cases, this notation and terminology follows that which seems to be employed by the G&N contractor, and also follows that which was used in documents previously prepared on Block 1 and Block 2 programs.

The following document may be found useful for supplemental information on the symbolic listing, and for a more detailed discussion of the computer hardware and general computer system control:

> **3420.5-27, Revision 2, "Apollo Guidance Program Symbolic Listing Information for Block 2," dated 20 November 1969. Referenced here as merely "3420.5-27".**

Copies of that document, together with revision information as it is published, may be obtained from the Flight Software Branch of MSC.

Numbers

A General

The guidance computer is designed with a memory word length of 15 bits (plus a sixteenth bit, not sensed by the program, used to achieve "odd parity", i.e. an odd number of binary ones in the total 16-bit word). It is also designed as a fractional machine, so that all numbers in the computer are less than one: "true values" greater than 1.0 are accommodated by suitable scaling, as described below. There is no hardware floating point capability in the computer, although a facsimile to floating point computations is sometimes used (and identified as "quasi-floating point") in the program, particularly when the quantities involved can have a large dynamic range.

The 15-bit word is divided into a sign bit and 14 magnitude bits, with the bits numbered from the sign (#15) to the least significant magnitude bit (#1), as shown:

Value:	Sign	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	2^{-9}	2^{-10}	2^{-11}	2^{-12}	2^{-13}	2^{-14}
Number:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

Once the value of a number in the computer register (evaluated using the bit weights shown) has been determined, its "true value" may be determined by multiplying this value by some power of two which is called the "scale factor":

> The scale factor of a quantity is the power of two by which the number in the computer (considered as a fraction in the range between -1 and +1) must be multiplied to obtain its true value. For convenience, the scale factor is shown as "Bxx", where "xx" is some positive or negative integer.

For example, if a word has a scale factor B14, this means that the individual bits have a weight 2^{14} (or 16384) times the values shown in the above table. A counter in the program with this scale factor, therefore, would have its least significant magnitude bit (or least increment) equal to 1 ($2^{-14} \times 2^{14} = 1$). See Appendix A of 3420.5-27 for more details.

In many instances, the word length of 15 bits, permitting a number to be expressed to 1: ± 16384, is insufficient to give the required precision in the computations. In these instances, a double precision number (stored in two consecutive memory cells) is used. Scale factor information for double precision numbers has the same interpretation as for single precision numbers, and the least significant half of the word has weights that are 2^{-14} times the weights shown above. A quantity that is double precision with a scale factor of B28, for example, would have a least significant bit of 1 and a maximum value of $(2^{28} - 1)$, or 268 435 455. In a few cases (usually involving time information), triple precision quantities are required, which follow similar rules.

In general, unless otherwise specified words can be considered to be double precision. The following subscripts are used in the programmed equations to indicate the type of number involved, in cases where this information is needed:

dp: Double precision.

sp: Single precision.

tp: Triple precision.

vn: Verb-noun pattern (for DSKY display): first two digits are the verb (bits 14-8 of the word) and last two digits are the noun (bits 7-1 of the word).

x, y, z: Vector components.

0, 1, 2 (on vector variables): Vector components, different significance than x, y, z. For example, x, y, z might refer to IMU coordinates and 0, 1, 2 to body (roll, pitch, yaw) coordinates. Not used with TS.

2 (on numbers): Binary value.

8 (on numbers): Octal value.

Numbers with no subscript are quoted in decimal.

In addition to the scale factor information, it is necessary to know the units in which quantities are expressed. Times are usually in units of "centi-seconds" (0.01 seconds). Powered flight navigation equations give position in units of meters and velocity in units of meters/centi-second. Angles are usually expressed in units of revolutions (one revolution is 360°). Where not reasonably obvious, units for the quantities are given as part of their definition.

B Constants

Fixed scalar constants are denoted by K_j, where j is a subscript of arbitrary type (lower case letters and/or numbers) selected for its mnemonic usefulness (generally similar to the program notation itself). An additional subscript in capital letters may also appear: this quantity is a variable, and is used for indexing purposes.

Constants stored in erasable memory (so they can be changed without hardware implications) are denoted by C_j, where j is again a subscript of arbitrary type. If an initial condition is necessary for some cell, but it is subsequently updated by the program, the quantity is given the designation of a variable.

For the comparatively few constants which are vectors (or which can be treated as vectors, such as the IMU calibration constants), the notation employed is an underline under the K or C (i.e. $\underline{K}_j$ or $\underline{C}_j$) except in a few cases where mnemonic usefulness dictated some other presentation (such as unit$\underline{X}$ for a unit vector in the X direction). Matrices are designated by their program notation, enclosed in square brackets (no special notation is used to distinguish between matrices which are constants and those which are stored in erasable memory).

C Variables

Variables are generally designated by several upper case capital letters, sometimes with subscripts (the single letters K and C, however, are reserved for constants). Lower case letters sometimes appear with the symbol for a variable, and have the following special meanings:

d: A division indication appears as part of the symbol in the program (LdD appears in the listing as L/D).

i: A suitable capital letter is substituted as defined in the related equation information for "i"("j" also used).

m: A minus sign appears as part of the symbol in the program (mTPER appears in the listing as -TPER).

p: A plus sign appears as part of the symbol in the program (pMGA appears in the listing as +MGA). **If two "p" letters are in the symbol, these signify parentheses in the symbol in the listing.**

An underline of the final character of a symbol means a quantity with several components, frequently a vector but sometimes a quantity having vector-type properties, such as gyro compensation about different axes. Individual components of a vector are identified by a final subscript quantity (an additional subscript level or separated by a comma from the other subscript information) using the same symbol as for the vector, but without the underline: TS_x, for example, would be the X component of the vector $T\underline{S}$.

A variable sometimes is used as a subscript itself, in which case it performs an indexing or address specification function. The capital letter E with such a subscript means the contents of the (generally erasable memory) cell whose address is given by the contents of the cell with the indicated symbol: E_{TS}, for example, is identical to VAR if TS contains the address of VAR. This notation, of course, signifies use of indirect addressing.

Matrices are designated by enclosing their identifications in square brackets. The matrix may have its own special symbol (such as [REFSMMAT]), or it may be designated by the symbol of its first vector component (the first row of the matrix), but without the underline: $[X_{sm}]$ is the matrix composed of $\underline{X}_{sm}$, $\underline{Y}_{sm}$, and $\underline{Z}_{sm}$. Some special notation for the W matrix information, to permit more convenient comparisons with official guidance equation information, is described on pages MEAS-37 and MEAS-38.

Interpretive language operations in the guidance computer (see 3420.5-27) permit convenient manipulations to be performed on 3 x 3 matrices, the elements of which are each stored double precision in sequential erasable memory cells (all elements of first row, then second row, and then third row). Individual elements of a matrix are designated by the matrix symbol followed by a subscript between 0 and 8: $REFSMMAT_3$, for example, is the first element in the second row of [REFSMMAT]; $REFSMMA\underline{T}_3$ would be the vector formed of ($REFSMMAT_3$, $REFSMMAT_4$, and $REFSMMAT_5$), i.e. the second row of [REFSMMAT] (which gives the stable member Y axis in reference coordinates). The program notation for $REFSMMAT_3$ would be "REFSMMAT +6", since elements are stored double precision with the most significant half of $REFSMMAT_0$ at address "REFSMMAT".

D Addresses

The value of an address is designated by the symbol used in the program, enclosed in quotation marks: without the quotation marks, the contents of the cell with that symbol would be indicated. In order to improve the clarity of the presentation, many of the program-step symbols are omitted from the equations, and others may not precisely correspond to the program step actually identified by the symbol in question. Computer addresses themselves must occasionally be mentioned: these are in CADR or 2CADR formats (single precision and double precision respectively, as described in 3420.5-27). Erasable and fixed memory CADR formats differ.

A number of subroutines are used within the program, each of which require information on the "main" program to which program control must be transferred at the end of the subroutine. For clarity in showing the computation flow, the retention of the necessary return address information is shown explicitly in a few places. The "return address" is the address to which control is to be transferred after completion of the subroutine; the "calling address" is the address from which transfer to the subroutine was made. In many cases, the return address can be considered to be one greater than the calling address (ignoring special provisions for program-step changes between fixed memory banks), but in some instances such as generation of program alarms, the cell following the calling address contains information pertinent to the subroutine (such as the alarm pattern), and therefore the return address may be several address locations after the calling address. To save program steps, in some cases the complement of the address may actually be used by the program for storage purposes, but this fact, since it has no effect on the computation flow, is not indicated in the programmed equations.

Program Control

There are three types of program entities that may be considered to take place within the program. These may be classified as:

a) A Subroutine, which performs a certain function and then returns control to the program entity which called it (Subroutines, of course, may in turn call other Subroutines).

b) A Task, which is a short sequence of computations performed based either upon a time criterion or upon some external signal.

c) A Job, which is a program entity (such as a routine to compute a large attitude maneuver, the processing of a keyboard or uplink input character, or the performance of a Lambert routine during powered flight guidance) of long duration (compared to a Task) which must be done in a definite sequence. For example, during powered flight the accelerometer data must be corrected for biases before the state vector updating computations are performed, and the state vector updated before steering commands are generated for the DAP (digital autopilot), and therefore these functions all form part of the same Job.

Time-dependent Tasks are implemented by a "waitlist" system (see section VIIA of 3420.5-27), for which the programmer merely specifies the time delay (from "now") when he wants a computation done, and the starting address of that computation. This time delay has a least increment of one centi-second (0.01 second). Unless interrupts are inhibited, a program interrupt (which can be caused by a signal from the telemetry system, the uplink, either of the two

DSKY's, completion of a radar input, two DAP waitlist-type clocks, the "T4RUPT" clock, or the waitlist clock for the CM) causes termination of the computations for the Job and performance of the Task. Tasks, however, are not subject to interruption by other Tasks, but instead continue to completion. There is also a hardware monitoring function which can cause a "hardware restart" that could, if necessary, interrupt a task: in general, however, such restarts should not be encountered.

Jobs are sequenced with the aid of a priority system (see section VIIB of 3420.5-27), and are performed only if no Tasks must be performed (provided that interrupts are not inhibited). If no "productive" computations are required, then a routine called the "dummy job" is performed. This routine checks periodically for the availability of a Job to be performed, and of course is subject to interruption for a Task. A Job can be "established" (put into a list to be selected when its priority is sufficiently high) by another Job or by a Task. A Job can be "put to sleep" (by temporarily setting its priority negative) until some event takes place: the occurrence of the event can "awaken" the job by setting its priority positive again. Jobs can optionally be assigned a set of working storage cells called a "VAC area" (see section VIIB of 3420.5-27). Addresses within this VAC area are referred to as "relative addresses" (relative to the start of the area assigned to the particular job), or as "push-down list addresses". Address 12D, for example, would be the 12th address from the start of the job's VAC area (the "D" emphasizes that the number is quoted in decimal, since octal is also used in the program).

The following program control terms are employed:

Awaken a Job: Cause a Job (if any) with the indicated starting address to be restored to its original priority after a period of being "asleep", during which the performance of the Job was suspended by making its priority negative.

Call "XXXX" in yy seconds: Cause a Task with starting address "XXXX" to be entered into the waitlist system to be performed in yy seconds from the present time (yy has a least increment of one centi-second).

Delay yy seconds: Cause the present string of computations to be suspended for yy seconds, and then restart at next line. Delays for a waitlist Task are implemented by a waitlist call to the following step, and then an End of task; delays for a Job are achieved by using the "DELAYJOB" routine.

End of job: Terminate the performance of a Job, and transfer control to the "executive" system to initiate performance of the Job which has the highest priority of those remaining. If no Jobs are to be performed, the "dummy job" routine is entered.

End of task: Terminate performance of the waitlist-initiated Task, and transfer control to a routine which checks for other waitlist Tasks which may be due, causing resumption of previous computations (cf. Resume) if there are no such Tasks.

Establish "XXXX": Enter a Job with starting address "XXXX" in the executive system priority list to be performed when appropriate (the priority of the Job is also indicated at the point where it is established). A Job can optionally be established with or without a working storage (VAC) area, but this option selection generally is omitted in the programmed equations information.

If: Carry out the indicated manipulations provided that the indicated conditions are satisfied. Should the conditions not be satisfied, or after performing the indicated manipulations, continue on in sequence (unless, of course, a manipulation involved transfer of program control). To avoid a proliferation of program tags, the manipulations are indicated by indentations, with if necessary additional indentations for additional conditions. In a few cases, it was necessary to continue the indentation function to the following page, and in such cases (such as the top of page ATTM-16) the most recent "If" controlling the performance of the routine is repeated in parentheses in order to ensure that proper interpretation of the information is made.

Perform "XXXX": Cause a Subroutine with starting address "XXXX" to be entered. The specific memory cell to which the Subroutine returns control depends in some cases upon the purpose of the routine. May be a "Proceed to" in program listing.

Proceed to "XXXX": Cause the program step with address "XXXX" to be the next one executed, and continue the performance of the program from that point. If no address is specified (as is true for some Subroutines which have multiple return addresses), "proceed" merely means continue with the next equation in sequence. May be a "Perform" in program listing.

Proceed to address specified by XXXX: Cause the program step whose address is stored <u>in</u> cell XXXX to be the next one executed, and continue the performance of the program from that point.

Put to sleep: Cause the present Job's computations to be suspended (by making its priority negative) until some event takes place causing the Job to be "awakened". Differs from End of job and Establishing the Job again in that the VAC area (and Job Register Set) are retained. When a Job is put to sleep, a starting address identification is provided for use in awakening the Job (and also to specify the starting point for the Job when it is performed).

Resume: Start performing again the computations (in a Job) that were interrupted to perform the Task. Used only for Tasks not initiated by waitlist means, cf. End of task.

Return: Proceed to the Subroutine's calling program, from which transfer to the Subroutine was made via a Perform.

In addition to the above terms, frequent use is made of special program control functions to accommodate crew use of the display system. In general, an action requested by the computer program can have three crew responses: a terminate verb, a proceed command (verb 33 or the PRO button), or a recycle verb (verb 32)/data enter. Uses of the display system are buffered by the routines described in the Display Interface Routines writeup, such as "GOFLASH" (entered e.g. from bottom of ATTM-1). These routines cause the display to be activated and then put the Job to sleep awaiting a response. The three responses are indicated by the terms "terminate", "proceed", and "otherwise", and are shown as the result after crew response. In addition, an immediate return (such as "GOFLASHR" use on ATTM-1) may be available, in which case the "Perform", rather than "Proceed to", program control term is used. This immediate return may be used for restart protection and/or to set bits for blanking one of the display registers (or more). Finally, there are the special "VNFLASH"/"VNFLASHR" interface routines, discussed earlier (page 13), for which the terminate response is a direct exit, rather than via the caller: to indicate this (on e.g. BURN-1), the terminate action, for format consistency, is still shown, but is enclosed in parentheses.

Operations

Several mathematical manipulations are available to the programmer and are indicated either by special characters or by lower case symbols. The trigonometric, logarithm, and square root functions are described in the Mathematical Functions writeup.

1) cos X: Cosine of X.

2) $\cos_{sp}$ X: Single precision cosine of X.

3) $\cos^{-1}$ X: Arc cosine of X.

4) cycle: Shift in a cyclic fashion, with bits "spilling out" of one end of the register appearing at the other end. For example, bit 14 cycled left 2 places, since the word length of the computer is 15 bits for data, becomes bit 1, as does the same bit cycled right 13 places (or, in this case, shifted right 13 places).

5) limit: Cause the maximum value of a quantity to be less than the value indicated (unless otherwise specified, the magnitude is limited, with sign information preserved). Frequently done by means of a check for overflow.

6) modulo: Form a quantity, for X modulo Y, equivalent to the value of Y times the remainder from (X/Y). For example, 380° modulo one revolution is 20°.

7) overflow: Exceed the capacity of the computer register (i.e. the maximum value of the number allowed by its scaling). The interpretive language (section VI of 3420.5-27) has a special cell which is set if such an overflow is encountered, and which may be sensed to cause program branching. The divide instruction in the interpretive language, if the numerator exceeds the denominator (using the numbers in the computer register), sets the answer to the maximum capacity of the computer register with the proper sign.

8) quasi-floating point: Carry out a computation (usually involving a division) by, in general, normalizing both numerator and denominator before performing the division (with suitable provisions to avoid division overflow), and then shifting the result the appropriate number of places. Normalization involves shifting a number so that there are no leading magnitude zeros, and counting the number of shifts required.

9) set X = Y and Y = X: Exchange the contents of "X" and "Y".

10) sgn (X): Complement the result if X is negative, and otherwise leave the result alone. Unless otherwise specified, if X = 0 the result is also left alone (i.e. 0 is a "positive" number).

11) shift: Shift in a non-cyclic fashion, with exclusion of the sign bit: bits "spilling out" of the least significant end of the register (for shifting right) are lost. Cf. "cycle". Vacated bits are set to sign bit.

12) sign agreement: Force the signs of the various parts of a multiple-precision word to be the same.

13) sin X: Sine of X.

14) $\sin_{sp}$ X: Single precision sine of X.

15) $\sin^{-1}$ X: Arc sine of X.

16) unit$\underline{X}$, or unit($\underline{A}$ + $\underline{B}$): Form or use a unit vector from the vector information specified. The scale factor of the unit vector when formed is B1. The magnitude of the vector is left in push-down list address 36D and its square in 34D. Operation "overflows" if all vector components, before forming the unit, are less than 2^{-21} in magnitude as stored in the computer register: if a cross product of 2 unit vectors is involved, this is a "true value" of 2^{-19} (about 0.002 mr).

17) $\log_e$ X: Natural logarithm of X.

18) Vector manipulations:

   * ($\underline{A}$ * $\underline{B}$): Vector cross product.

   · ($\underline{A}$ · $\underline{B}$): Vector dot product.

18) Vector manipulations (cont):

$(\underline{A}\ \ \underline{B})$: Form individual components of result from the individual components of A and B(e.g. Y component of result is $A_y\ B_y$)

$|\underline{A}|$: Length of $\underline{A}$.

$|\underline{A}|^2$: Square of length of $\underline{A}$.

$\underline{A} = (a_1\ , a_2, a_3)$: Form a vector $\underline{A}$ from the given components.

$[A]\ \underline{B}$: Premultiply the vector $\underline{B}$ by the matrix $[A]$ (interpreter operation MXV). The X component of the answer is equal to the dot product of the first row of $[A]$ and $\underline{B}$, etc.

$\underline{B}\ \ [A]$: Postmultiply the vector $\underline{B}$ by the matrix $[A]$(interpreter operation VXM). The X component of the answer is equal to the dot product of the first column of $[A]$ and $\underline{B}$, etc. Same as premultiplication by the transpose of $[A]$.

19) Scalar manipulations:

+ : If space before sign, addition operation. If no space before sign, increment to address (single precision unless otherwise specified). MPAC+0 is the most significant part of $MPAC_{dp}$, while MPAC +1 would be the addition of 1 to the contents of MPAC. In a subscript, the second meaning applies.

$|X|$: Absolute value of X.

$B = (b_1, b_2)$:Form a double precision number B, with most significant half b_1 and least significant half b_2.

$\sqrt{X}$: Square root of X.

$\cap$ $(X \cap Y)$: Form the logical product one bit at a time (a "masking" order). If a given bit in both X and Y is a binary one, then the result is a binary one; otherwise, the result is a binary 0.

20) A bit is "set to 1" or "set to 0" by forcing its value to be 1 or 0 respectively.

21) The difference between two angles (expressed in twos complement form, see 3420.5-27) can be performed in such a way as to give a ones complement result, noted where significant as "ones complement difference formed".

22) Loading of registers of less than full-word capacity (such as EBANK which has only 3 bits) is generally indicated for clarity as if the loading were done by a masking operation.

Major Variables

The following quantities are used at several points in the program, and therefore have not been included in the list of "Quantities in Computations" at the end of each writeup.

CD$\underline{U}$: Single precision twos complement value of the computer hardware CDU (Coupling Data Unit) registers used with the IMU (Inertial Measurement Unit). Scale factor B-1, units revolutions. Other CDU's are used with the optics, but unless so specified reference to "CDU" should be interpreted as the IMU CDU's.

DEL$\underline{V}$: Value of sampled accelerometer output, scale factor B14, units of accelerometer counts (5.85 cm/sec per count). When loaded during flight, only the most significant half of each component is set to the accelerometer output value, with the least significant half set zero. After compensation (in "1/PIPA"), the least significant half can become non-zero.

DSPTAB: Set of 12 single precision erasable memory cells containing "display table" information. DSPTAB+0 through DSPTAB+10 are used to control the display registers and other numerical displays on the DSKY (Display and Keyboard assembly), with the format given on page DATA-40. DSPTAB+11 is used to drive individual indicators with bit assignments given in the table below: required changes for the values of these indicators are flagged for output by setting bit 15 of DSPTAB+11 to 1.

DSPTEM1, DSPTEM2: Cells used for communication purposes with the display routines (several nouns are associated with the cells to permit displays of appropriate parameters). DSPTEM1 also "NORMTEM1".

E: Notation employed for a quantity selected by the address information given by the subscript on E. $E_{TS_{dp}}$ would mean a double precision word whose address is given by the contents of cell TS.

FLAGWRDi (i = 0 - 9): Set of ten words whose individual bits are used for program control purposes. FLAGWRD4 is used only with the Display Interface Routines logic, and hence is defined in that writeup, while the others are defined in the table below. As specified by Mr. J. R. Garman of MSC, where applicable the flagbit mnemonics are the same as those used by W. C. Koelsch in his Sundance programmed guidance equations document. In other cases, special mnemonics have been created for the individual bits, with the objective of having the mnemonic suggest the significance of the bit when the bit is 1. More detailed discussions of individual bits can be found in suitable downlink functional description documents.

FLGWRD10, FLGWRD11: Words whose individual bits are used for program control purposes. See table below.

I, i, j: Index parameters (not necessarily explicitly contained in the program) used as a convenience in describing the performance features of the equations. Their use and significance are defined (unless obvious) where they are used.

MAX: The maximum capacity of a computer cell (octal 37777_8 for a single precision +MAX and octal 40000_8 for -MAX). Decimal equivalent is dependent on scale factor of quantity involved (if scaled B0, value for single precision is $1 - 2^{-14}$ and for double precision $1 - 2^{-28}$).

MODREG: Single precision cell containing the value of the current computer program number (or "major mode"), generally displayed also on the DSKY (the quantity is a decimal, not octal, number). A setting to -0 is made if a fresh start is performed, thus inhibiting display of the quantity.

MPAC: Multiple-precision (or multiple-purpose) accumulator, a set of seven cells assigned to each job. Frequently used for temporary storage purposes. MPAC+2 is a communication cell with internal DSKY programs (contains the machine address of the quantity to be manipulated).

$\underline{R}$: Position state vector, scale factor B29, units meters. Program notation is "RN".

$\underline{R}_{att}$, $\underline{R}_{att1}$: Communication cells with the orbital integration package containing the value of the position vector after completion of the integration (i.e. "at time" specified). $\underline{R}_{att}$ always has scale factor B29, units meters (in push-down list location 0D); $\underline{R}_{att1}$ has scale factor B29 meters for earth-centered computations and B27 meters for moon-centered computations (stored in push-down list location 14D). The use of the push-down list permits several jobs to employ the orbital integration package (with "INTSTALL" logic performing the necessary sequencing).

[REFSMMAT]: Reference to stable member matrix, elements with scale factor B1. The reference coordinate system "is defined by the line of intersection of the mean earth equatorial plane and the mean orbit of the earth (the ecliptic) at the nearest beginning of the Besselian year in which the mission takes place. The X-axis is along the ascending node of the ecliptic on the equator (the equinox), the Z-axis is along the mean earth north pole, and the Y-axis completes the right handed triad." Loaded for certain options of the inflight alignment routines, and initially in "P11". The first row gives the X stable-member axis in reference coordinates, so that pre-multiplication of a vector in reference coordinates by $\lceil$REFSMMAT$\rceil$ gives the same vector in IMU coordinates.

S1, S2: Interpretive language registers (push-down list locations 40D and 41D) used as communication cells for single-precision address information.

T_{att}: Communication cell with orbital integration package (cf. $\underline{R}_{att}$) containing the value of time tag associated with the state vector output, in push-down list location 12D with scale factor B28, units centi-seconds. Program notation is "TAT".

T_{decl}: Communication cell with orbital integration package (cf. $\underline{R}_{att}$) containing the value of time to which integration is to be performed. Stored in push-down list location 32D, scale factor B28, units centi-seconds.

T_{evt}: Time of major event (e.g. liftoff, engine on, or engine off, whichever took place most recently), scale factor B28, units centi-seconds.

T_{ig}: Value of "ignition time" (displayed by N33), scale factor B28, units centi-seconds. Also used by "CLOKTASK" to compute a "time-to-go" as $T_{now} - T_{ig}$ (T_{togo}, displayed by several nouns). Hence after ignition is replaced by predicted cutoff time; for an ENTR response to the V97N40 thrust-fail display, the most significant half is set to a negative quantity, causing the "time-to-go" display to be +59 59 (for T_{now} above about 9 minutes).

T_{now}: Present value of the computer "clock" (registers TIME2, TIME1) hardware erasable memory cells, scale factor B28, units centi-seconds. Reset 0 in "P11", but can be updated by verb 55, verb 70, or verb 73.

T_{pptm}: Time tag associated with $\underline{R}$ and $\underline{V}$, scale factor B28, units centi-seconds. Program notation is "PIPTIME" (since it often is the value of the clock when accelerometers were sampled).

THETA$\underline{D}$: Single precision twos complement value of the "desired" IMU CDU values, scale factor B-1, units revolutions. Can be loaded or displayed by noun 22, and is used as one of the display options for attitude error needles with RCS DAP (the cells themselves are <u>not</u> used internally by the DAP's: THETAD$\underline{X}$ is used by the RCS DAP, which is a different parameter than THETA$\underline{D}$). Cells also have notation (CPHI, CTHETA, CPSI).

TIME1, TIME2: See T_{now}.

TS (and variants, such as T$\underline{S}$, TS_1, etc.): Real or dummy temporary storage cells, used for convenience in describing the performance of the equations. When used as a communication cell between routines (such as specification of verb-noun pattern to a display routine) generally would be a computer hardware register, such as the accumulator, in the actual program.

UNIT$\underline{R}$: Value of unit$\underline{R}$, scale factor B1.

$\underline{V}$: Velocity state vector, scale factor B7, units meters/centi-second. Program notation is "VN".

$\underline{V}_{att}$, $\underline{V}_{att1}$: Communication cells with the orbital integration package containing the value of the velocity vector after completion of the integration (i.e. "at time" specified). $\underline{V}_{att}$ always has scale factor B7, units meters/centi-second (in push-down list location 6D); $\underline{V}_{att1}$ has scale factor B7 meters/centi-second for earth-centered computations and B5 meters/centi-second for moon-centered computations (stored in push-down list location 20D). See $\underline{R}_{att}$.

unit$\underline{X}$, unit$\underline{Y}$, unit$\underline{Z}$: Unit vectors in X, Y, and Z directions respectively, scale factor B1.

X1, X2: Interpretive language index registers (push-down list locations 38D and 39D) sometimes used as address communication cells or as counters (scale factor B14). Can be used to select cells or shift amounts in interpretive language (X1 frequently is -2 for earth, -10 for moon; X2 frequently is 0 for earth and +2 for moon). If used as address communication cells, complementing in loading and subsequent use sometimes omitted for clarity.

$\underline{X}_{dc}$, $\underline{Y}_{dc}$, $\underline{Z}_{dc}$ (or $\left[X_{dc}\right]$): Unit vectors, scale factor B1, used to define desired IMU orientation for communication with routines in Coordinate Transformations, or to define navigation base orientation. Notation also $\underline{X}_{nb}$ etc. in program.

$\underline{X}_{sm}$, $\underline{Y}_{sm}$, $\underline{Z}_{sm}$ (or $\left[X_{sm}\right]$): Unit vectors, scale factor B1, used to define the orientation of the "stable member" (i.e. IMU).

Meaning of bits in DSPTAB+11

Bit	Meaning
9	Program alarm.
8	Tracker alarm (optics). Also used for VHF data-good information (set e.g. by "LIGHTON").
6	Gimbal lock (controlled by "GLOCKMON").
4	No attitude.

Meaning of bits in FLAGWRDO

FLAGWRDO also has notation "STATE". Initialized to 00000_8 by a fresh start; bit 1 is set 0, with other bits left alone, in "INITSUB". Address is 0074_8.

Bit	Symbol	Meaning
15		Not assigned.
14	JSWITCH	Bit set to 1 within orbital integration package to indicate that extrapolation of W matrix (rather than state vector) being performed.
13	MIDFLAG	Bit set to 1 within orbital integration package to indicate that integration with solar and secondary-body perturbations is required: set if altitude above primary body exceeds 800 km.
12	MOONFLAG	Bit set to 1 within orbital integration package to indicate that state vector is moon-centered (if 0, is earth-centered).
11	FARHOR	Bit set to 1 in P23 if sighting on "far" horizon (as contrasted with "near" horizon) is done. Program notation "NORFHOR".
10	ZMEASURE	Bit set to 1 in P23 if the measurement and state-vector bodies are different (earth is body and moon-centered state vector, for example), and set 0 if they are the same.
9	NEEDLFLG	Bit set 1 (by V62E or V63E) or 0 (by V61E) to indicate that FDAI attitude error display should be of total attitude error (if 1) or DAP "following error" if 0. See bit 6 of FLAGWRD9.
8	IMUSE	Bit set 1 to indicate IMU "in use" (an alarm is generated if IMU is turned off with bit 1).
7	RNDVZFLG	Bit set 1 to indicate that P20 in rendezvous mode (options #0 or #4) has been enabled, to control proper selection of restart information when Average-G is stopped, and to serve other functions. P20 may be dormant even if bit is 1.
6	SGTMK	Bit set 1 to indicate that sighting marks have been initiated. Program notation "R53FLAG".
5	F2RTE	Bit set 1 in P37 to indicate than an iteration to provide a result for time-critical mode (in "GAMDV25") is desired.

Bit	Symbol	Meaning
4	CYC61FLG	Bit set 1 in "STKTEST" (part of R61) to cause R60 computations to exit to "RCYCLR61" after determining required maneuver time for use in computing a new CSM/LM relative position.
3	FREEFLAG	Bit used for temporary program control purposes, such as a communication bit with "R54" that is left at 0 if a V32E response is received, and otherwise is left at 1.
2	AMOONFLG	Bit used to indicate if Average-G state vector is moon-centered (value is 1) or earth-centered (value is 0). Set and reset in "MIDTOAV2".
1	P29FLAG	Bit set 1 in "P29", used to permit some computations common to P21 to be performed. Used in "HOPALONG" to determine if in first iteration (where bit reset).

Meaning of bits in FLAGWRD1

Bit 12 of FLAGWRD1 is left alone, and other bits set 0, by a fresh start; bit 6 is set 0, with other bits left alone, in "INITSUB".

Bit	Symbol	Meaning
15	2JETSFLG	Bit set 1 in R03 to indicate that two-jet X translation is desired (0 means 4 jets), ignoring failure information. Program notation "NJETSFLG".
14	STIKFLAG	Bit set 1 in RCS DAP if RHC commands provided or if $\|MGA\| > 75^\circ$. Used to bypass automatic maneuvers. Bit reset 0 by V58E and in "R00".
13	ERADCOMP	Bit set 1 to specify that coordinate computation should use a "model" radius for earth and moon: for earth, it is computed based on Fischer model, and for moon a mean lunar radius (1738.09 km) is employed. If bit is 0, the earth Saturn launch site radius (6,373,338 meters) or moon LM "launch site" (i.e. RLS landing site) radius is used. Program notation is "ERADFLAG".
12	NODOP01	Bit set 1 at the start of "P11", to inhibit selection of P01 subsequently.
11	RCSBURN	Bit set 1 if an RCS burn (P41) is specified, and set 0 if SPS burn (P40) is specified. Program notation "ENG2FLAG".
10	LMTRG	Bit set 1 to indicate that the target being sighted is the LM. Program notation "TARG1FLG".
9	LMKTRG	Bit set 1 to indicate (with bit 10 = 0) that the target being sighted (R52) is a landmark. If bits 10 and 9 are both 0, the target is a star. Program notation "TARG2FLG".
8	CSMUPDAT	Bit set 1 to specify that CSM state vector is to be updated by navigation measurements (a value of 0 means LM state vector). Can be set to 1 by V81E and to 0 by V80E. Program notation "VEHUPFLG".
7	UPDATFLG	Bit set 1 to indicate that updating of state vector by (optics or VHF) marks is allowed, and set to 0 to indicate that such updating is not allowed.

Bit	Symbol	Meaning
6	IDLEFAIL	Bit set to 1 for 2 seconds in "V97P", to inhibit the engine fail detection logic of "S40.8" from being triggered after a PRO response to the V97N40 display (for the two-second interval).
5	TRACKFLG	Bit set to 1 to indicate that tracking via P20 is allowed (in either the rendezvous or universal tracking modes, see bit 7 of FLAGWRD0 and bit 9 of FLAGWRD8). A setting to 0 means that the P20 tracking should not be running.
4	MARKFLG	Bit set 1 for an optics mark (except in R21 and R57), used in the mark reject logic (except R21) to determine whether to accept the mark reject input or generate a program alarm.
3	ITER1SW	Bit set 1 to indicate that first Lambert iteration is being performed, and then reset 0. Program notation "SLOPESW".
2	GUESSSW	Bit set 1 to indicate to Lambert iteration that no starting value is available; otherwise, it is set 0. Program notation "GUESSW".
1	AVEGFLAG	Bit set 1 when "READACCS" (accelerometer sampling and Average-G computations) is initiated, and reset (in "ISITP00") to halt computations.

Meaning of bits in FLAGWRD2

FLAGWRD2 is initialized to 00000_8 by a fresh start; bits 12, 11, and 9 are set 0, with other bits left alone, in "INITSUB".

Bit	Symbol	Meaning
15	DRIFTFLG	Bit set 1 to enable free-flight gyro drift compensation.
14	R21MARK	Bit set 1 to indicate that reception of optics r/v primary marks is enabled, used to control the location in which mark information is stored by "MARKDIF".
13	22DSPFLG	Bit set 1 in P22 to cause a V06N49 display of the state vector update from the first mark of a set, and then reset 0.
12	P21FLAG	Bit set 1 in P21 after a "base vector" has been obtained: subsequent evaluations make use of the base vector instead of the present memory permanent state vector.
11	STEERSW	Bit set 1 to permit P40 cross-product steering and P15/P40 time-to-go computations to be performed.
10	SKIPVHF	Bit set 1 in "STARTSB2" as a flag for "VHFREAD" that a restart has taken place. The resulting radar interrupt is ignored.
9	IMPULSW	Bit set 1 in "S40.13" if a burn duration of less than 6 seconds is determined. Used in "IGNITION" to set waitlist call for cutoff and bypass the setting of the STEERSW bit.
8	XDELVFLG	Bit set 1 to specify that External Delta V guidance, as contrasted with "Lambert Aimpoint", has been targeted for P40/P41.
7	FIRSTFLG	Bit set 1 to indicate that a pass through "S40.9" has not yet been accomplished (set 1 at the start of "S40.1", and then reset 0 at the end of "S40.9").
	HAVEELEV	Bit set 1 in P34/P74 to indicate that elevation angle is available, so T_{tpi} should be computed and displayed. A zero value means the opposite. Bit set 0, if 1, on final pass in minimum key rendezvous mode. Program notation is "ETPIFLAG".

Bit	Symbol	Meaning
6	FINALFLG	Bit set 1 to indicate that final pass through rendezvous targeting computations is to be performed (set in P37 to cause proper exit from the V16N45 display).
5	LMACTFLG	Bit set 1 to indicate that LM is "active" vehicle (i.e. the one to do the maneuver) in P72-P75. Also set in "P83" for computation of LM P76 information. Program notation "AVFLAG".
4	PFRATFLG	Bit set 1 to indicate to P52/P54 that a "preferred attitude" has been computed and is available. It is set in P40/P41 for burn attitude, and in the P36 minimum key rendezvous sequence after the plane-change information is available. The bit is used to select the initial V04N06 option display. A "preferred attitude" can be uplinked and used without setting the bit (cells are also used for P27 uplink buffer).
3	P24MKFLG	Bit set 1 in P24 when mark taken that can be used to update landmark coordinates. Reset for mark reject or after being used for the updating.
2	CALCMAN2	Bit set 1 to specify that maneuver starting procedure should be done in R60 attitude maneuver package, and then reset.
1	NODOV37	Bit set 1 to cause V37 inputs of other than 00, unless in TVC mode, to be ignored. Set e.g. while periodic (P00 or P20 tracking) integration is done, while the P76/P77 integration is done, when P06 is done, or when entry DAP is started. Program notation is "NODOFLAG".

Meaning of bits in FLAGWRD3

Bit 13 of FLAGWRD3 is left alone, and other bits set 0, by a fresh start; bits 14 and 9 are set 0, with other bits left alone, in "INITSUB".

Bit	Symbol	Meaning
15	V50N18FL	Bit set 1 by V37 selection of a new program (except P00), and by V58E. It is set 0 upon completion of "R61CSM". If bit is 1 and attitude error sufficient, then R60 is entered from R61.
14	GLOKFAIL	Bit set 1 if "CALCGA" computes a middle gimbal angle in excess of 60° (reset in "INITSUB" as well as in prelaunch programs).
13	REFSMFLG	Bit set 1 if a meaningful [REFSMMAT] is considered to be available.
12	LUNLATLO	Bit set 1 if lunar coordinate transformations are to be used to obtain or process longitude and latitude; a value of 0 means earth-centered coordinate transformations. Program notation is "LUNAFLAG".
11	P22MKFLG	Bit set 0 at start of P22 and P24, and set 1 after N71 response (P22) or after first mark (P24). Can be used to indicate that words 18-35a of the program 22 downlist are at least partially updated (raw time, optics angles, and IMU CDU angles), although P22 and P24 both load downlist cells as marks are obtained.
10	VFLAG	Bit left at 1 by star selection routine if a suitable pair of stars has not been found.
9	POOFLAG	Bit set 1 to flag to orbital integration logic that doing periodic (P00 or P20 tracking) integration, in order to enable the logic checks for this type of integration. Bit reset in "INITSUB".
8	PRECIFLG	Bit set 1 for various orbital integration entrances except from "STATINT1" (periodic P00 or P20 tracking) for CSM. It is used in "TESTLOOP" to identify the special "STATINT1" integration, and bypass any integrations of less than a time step. Hence if bit is 1, integration is done up to the specified time (which may require a fraction of a time step).
7	CULTFLAG	Bit set 1 in star selection routine if star is occulted (by earth, sun, or moon), or in "SXTANG1" for trunnion "true" angle outside 0-90° range.
6	ORBWFLAG	Bit set 1 if W matrix considered satisfactory for orbital navigation (used for P22 and P23, as contrasted with P20, purposes).

Bit	Symbol	Meaning
5	STATEFLG	Bit set 1 if the "permanent state vector" of the CSM or LM is to be updated by the orbital integration package (used and reset in "A-PCHK"). Permanent state vector also updated if bit 1 (WMATINT) of FLAGWRD3 = 1.
4	CONICINT	Bit set 1 if conic integration is to be performed by orbital integration package (if 0, Encke integration is done). Program notation "INTYPFLG".
3	CSMINTSW	Bit set 1 if CSM state vector is to be integrated by orbital integration package (if 0, LM state vector is integrated). Can enter at "INTEGRVS" with vector supplied in separate communication cells. Program notation "VINTFLAG".
2	9DIMWMAT	Bit set 1 if W matrix to be considered 9x9 for integration purposes (bit 9 of FLAGWRD5 gives dimension for measurement incorporation); if 0, matrix considered 6x6. Program notation "D6OR9FLG".
1	WMATINT	Bit set 1 if W matrix is to be integrated by orbital integration package (done for each time step after orbital integration of state vector). If 0, no W matrix integration is to be done. Program notation "DIMOFLAG".

For bits in FLAGWRD4, see Display Interface Routines.

Meaning of bits in FLAGWRD5

FLAGWRD5 is initialized to 00200_8 (bit 8 = 1) by a fresh start; bits 12, 10, and 7 are set 0, with other bits left alone, in "INITSUB".

Bit	Symbol	Meaning
15	DSKYFLAG	Bit set 1 in "KEYCOM" if a DSKY (not uplink) input is received. If the bit is 0, as it is after a fresh start, then the drive of DSKY outputs in "T4RUPT" is usually bypassed (except for the remainder of the 0.12-second interval after DSPTAB+11 is transmitted).
14	RETROFLG	Bit set 1 in P37 if present vehicle orbit is retrograde (to cause proper polarity of N81 R1 display).
13	SLOWFLG	Bit set 1 if a "slow down" maneuver desired in P37 (indicated by input of a negative velocity change).
12	P23CALIB	Bit set 1 to indicate that a calibration mark (i.e. R57) is being obtained for P23, and reset in "ENDR57". Program notation is "V59FLAG".
11	FSTINCRP	Bit set 1 if the first incorporation of measurement information (in P22 or R22) from optics data is being made, and set 0 for second. In R22 for VHF, only one incorporation is made. Program notation is "INCORFLG".
10	NEWTFLAG	Bit set 1 in "HOPALONG" (part of P29) if another pass through a portion of the P21 computations is required, used to cause exit to proper point in the iteration loop.
9	DMENFLG	Bit set 1 if W matrix 9x9 for incorporation purposes (0 for 6x6, including initial cycle for processing of P22 data from a given set of marks).
8	CMCCOMP	Bit set 1 (as part of fresh start), but not subsequently referenced. Program notation "COMPUTER" (formerly used to distinguish between CMC and LGC computations).
7	ENGONFLG	Bit set 1 when the SPS engine-on command is generated, and set 0 when it is removed. Note that it does not indicate that thrust has been detected. Bit used e.g. to restore the engine-on command after a hardware restart.

Bit	Symbol	Meaning
6	3AXISFLG	Bit set 1 if attitude maneuver package ("KALCMANU") provided with three angles to which maneuver is to be done (e.g. options #4 or #5 of P20, or "R62DISP"). If bit reset, maneuver calculated using "VECPOINT".
5	BKUPLO	Bit set 1 when V75E (backup liftoff) is received. Program notation is "GRRBKFLG".
4		Not assigned.
3	NOSOLNSW	Bit set 1 if a conic routine (e.g. Lambert or time-radius) unable to provide a solution, and set 0 if a solution generated. Program notation is "SOLNSW".
2		Not used (assigned notation "MGLVFLAG").
1	RENDWFLG	Bit set 1 if W matrix considered satisfactory for rendezvous navigation (R22 of P20).

Meaning of bits in FLAGWRD6

FLAGWRD6 is initialized to 00004_8 by a fresh start; bit 13 is set 0, with other bits left alone, in "INITSUB". Also is "CM/FLAGS".

Bit	Symbol	Meaning
15	DAPBIT1	DAP status bit (see bit 14).
14	DAPBIT2	DAP status bit, used with bit 15 to indicate the DAP which is activated, and to select the appropriate starting point if a hardware restart:

Bits 15-14	DAP Activated
00_2	None (or entry)
01_2	RCS DAP
10_2	TVC DAP
11_2	Saturn DAP

Bit	Symbol	Meaning
13	ENTRYDSP	Bit set 1 to cause display generation in entry program based on verb-noun in ENTRYVN (in "ENDEXIT"). Reset 0 to inhibit this display (e.g. while "P65.1" display is provided).
	STRULLSW	Bit set 1 in "IGNITION" if steering is to be enabled 2 seconds after SPS ignition, and set 0 if it is not to be (i.e. short burn, for which ullage termination is the only function at that time).
12	CMDAPARM	Bit set 1 to "arm" the entry DAP (to allow "entry firings and calculations").
11	GAMDIFSW	Bit set 1 if GAMDOT (see Entry Computations) is to be calculated (as the difference of present and previous angle information).
10	GONEPAST	Bit set 1 to indicate that lateral control calculations are to be omitted during entry. It is initialized to 1 in "STARTENT" and reset 0 in "INITROLL" after 0.05g sensed; in "PREDICT3" (P67) it is a "latched" version of bit 8 of FLAGWRD7, indicating target overshoot.
9	RELVELSW	Bit set 1 in entry equations to cause earth-relative velocity to be used.
8	EGSW	Bit set 1 in entry equations to indicate P67.

Bit	Symbol	Meaning
8(cont)	LNDKNOWN	Bit set 1 if a "known" landmark is being observed in P22, and 0 if "unknown". Program notation "KNOWNFLG".
7	NOSWITCH	Bit set 1 in "UPCONTRL" (entry) if drag exceeds 140 fps^2, in order to inhibit a lateral switch in "LIMITL/D".
6	HIND	Bit set 1 in entry equations to specify that "HUNTEST" iteration is to be performed.
5	INRLSW	Bit set 1 in entry equations after 0.05g sensed, to indicate termination of the period when initial roll attitude is to be held.
4	LATSW	Bit set 0 in entry equations to force the entry DAP to "roll over the top" (rather than shortest route).
3	05GSW	Bit set 1 when drag is over 0.05g, and reset 0 when less than that amount. Program notation ".05GSW". Bit set to an initial condition of 1 in fresh start to bypass TTE computation in "SERVICER" (reset e.g. in "NEWRNVN" for P61, and also in "P62" before entry DAP started).
2	CMDSTBY	Bit set 1 when entry DAP computations are not to be inhibited (i.e. when entry DAP "activated"). Program notation "CM/DSTBY".
1	GYMDIF	Bit set 1 in entry DAP if CDU differences and body rates are to be computed (as difference of present and previous angles). Program notation "GYMDIFSW".

Meaning of bits in FLAGWRD7

FLAGWRD7 is initialized to 00000_8 by a fresh start; bits 13, 12, and 11 are set 0, with other bits left alone, in "INITSUB".

Bit	Symbol	Meaning
15	TERMIFLG	Bit set 1 to terminate performance of "R52": set at end of R53 and (unnecessarily) R56.
14	ITSWITCH	Bit set 1 in P34/P74 to force another iteration in the computation of TPI time. Program notation is "ITSWICH". Bit also set in P33/P73.
13	IGNFLAG	Bit set 1 when nominal ignition time reached.
12	ASTNFLAG	Bit set 1 when crew authorization for ignition has been received.
11	TIMRFLAG	Bit set 1 to permit "CLOKTASK" to continue (loops at a one-second rate to update time-to-go, T_{togo}).
10	NORMSW	Bit set 1 to specify that $\underline{\mathrm{UN}}$ is provided as an input from "INITVEL" package to Lambert; if 0, Lambert computes its own value (in "GEOM").
9	RVSW	Bit set 1 to inhibit calculation of final state vector by "COMMNOUT" (for e.g. time-theta or time-radius routines).
8	GONEBYTG	Bit set 1 in entry computations if an overshoot of the target is determined to exist. It is updated each computing interval to provide sign information for entry display of target angle. Program notation is "GONEBY".
7		Not assigned.
6	V37FLAG	Bit set 1 if the Average-G package ("READACCS" etc.) is running. If bit is 1, then V37 processing halts to wait for Average-G termination via "AVGEND".
5		Not assigned.

Bit	Symbol	Meaning
4	UPLOCKFL	Bit set 1 to lock out further uplink inputs (except error reset) after a complement (k $\bar{k}$ k) failure detected in uplink information.
3	VERIFLAG	Bit whose value is complemented when confirmation of P27 update (a V33E proceed response to F V21 NO2) is received.
2	LMATTCH	Bit set 1 in RO3 if DSKY input indicates CSM and LM are attached (not otherwise used).
1	TFFSW	Bit set 1 if calculation of perigee time is to be made, and set 0 if time of free fall is to be obtained.

Meaning of bits in FLAGWRD8

Bits 12, 11, and 8 of FLAGWRD8 are left alone, and other bits set 0, by a fresh start. FLAGWRD8 is left alone in "INITSUB".

Bit	Symbol	Meaning
15	RPQFLAG	Bit set 1 in "INTEGRV" and "INTEGRVS" (at start of a use of the integration package) to indicate that position vector to secondary body not calculated; set 0 in "ACCOMP" when computed.
14	NEWLMFLG	Bit set 1 in R52 when done in P24 after a new landmark location derived from measurement data, and used to bypass rate computation subsequently in order to avoid improper result.
13	NEWIFLG	Bit set 1 by various entrances to integration package to indicate the initial pass through the integration, and set 0 in "TESTLOOP" after the logic concludes that performing periodic integration of the CSM state vector (in P00 or P20 tracking): effect is to inhibit start of integration of this type unless time tag at least 4 time steps behind present time.
12	CMOONFLG	Bit set 1 if CSM state vector in lunar sphere (i.e. moon-centered coordinates), and set 0 if earth-centered. Can be uplinked in either system (see UPSVFLAG in Uplink Processing), and changed in "ORIGCHNG" (bit 12 of FLAGWRD0, which is eventually copied into bit 12 of FLAGWRD8 for a permanent state vector update).
11	LMOONFLG	Bit set 1 if LM state vector in lunar sphere (i.e. moon-centered coordinates), and set 0 if earth-centered. See bit 12 of FLAGWRD8.
10	ADVTRK	Bit set 1 in "S22N7071" (part of P22) to indicate advanced ground track mode of R52 is required (bit reset to 0 for normal R52 entrances).
9	UTFLAG	Bit set 1 to indicate that P20 in tracking mode (options #1, #2, or #5) has been enabled, to control proper selection of restart information when Average-G is stopped, and to serve other functions. P20 may be dormant even if bit is 1.
8	SURFFLAG	Bit set 1 by V44E and 0 by V45E, used to indicate (if 1) that LM on lunar surface. If 1, conic or precision "integration" attempts will branch in "INTEGRV" to use the planetary inertial orientation subroutine, based on $\underline{RLS}$, to determine LM state vector.

Bit	Symbol	Meaning
7	INFINFLG	Bit set 1 to indicate that no conic solution can be obtained (set several places in "GETX" Universal Variable Subroutine).
6	ORDERSW	Control bit used in "ITERATOR", expected to remain 0.
5	APSESW	Bit set 1 in time-radius routine if desired radius outside of pericenter-apocenter range, and otherwise set 0.
4	COGAFLAG	Bit set 1 if calling routine finds that COGA computation overflows (in Conic Routines).
3	V96ONFLG	Bit set 1 when a V96E (inhibit P00 integration) input received, and reset 0 when P00 integration is done. See bit 5 of FLAGWRD9. Could also be used to inhibit P20 tracking periodic integration, if P20 selected after the V96E.
2	R67FLAG	Bit set 1 in "R67" when the specified (via N34) time for start of rotation in option #2 of P20 has arrived, used to control logic in automatic maneuver routine.
1	360SW	Bit set 1 in "GETX" package if transfer angle near 360°.

Meaning of bits in FLAGWRD9

FLAGWRD9 is initialized to 00000_8 by a fresh start; bits 15 and 11 are set 0, with other bits left alone, in "INITSUB".

Bit	Symbol	Meaning
15	SWTOVER	Bit set 1 in "SWICHOVR" (entered for V46E with LM-on TVC DAP) when the manually initiated change to the "low bandwidth" LM-on TVC DAP is made. Bit also initialized to 0 in "TVCINIT1".
14	P24FLAG	Bit set 1 when P24 is started, and reset 0 in "AUTO37". Used to control performance of R52 and logic of mark/mark reject processing.
13	V82EMFLG	Bit set 1 if state vector used in R30 is moon-centered and 0 if earth-centered.
12	MAXDBFLG	Bit set 1 in R03 if maximum deadband selected, and set 0 if minimum deadband selected. Used in "FIXDB" and "INITSUBA" to restore the RCS DAP deadband to this selected value.
11	V94FLAG	Bit set 1 in P23 while a V94E input is allowed (provide a "recycle" to cause R60 to be done).
10	SAVECFLG	Bit set 1 in P23 before start of N71 display (after the mark): it allows use of a portion of the same coding ("saves cells").
9	VHFRFLAG	Bit set 0 by V88E (or "V37") and to 1 by V87E and "STARTAUT". If both this bit and bit 9 of FLGWRD10 are one, R22 attempts to process VHF range data periodically.
8	VHFSOURC	Bit set 1 by R22 if the data source is VHF ranging, and set 0 if it is optics. Program notation is "SOURCFLG".
7	R22CAFLG	Bit set 1 while R22 is processing an optics mark, used if a mark reject is received to determine appropriate action.
6	N22ERNDS	Bit set 1 by V62E and 0 by V63E, causing data selected for total attitude FDAI error display in RCS DAP (see bit 9 of FLAGWRD0) to be THETAD (N22) if bit is 1 and CPHIX (N17, loaded with CDU by V60E) if the bit is 0. Program notation "N22ORN17".

Bit	Symbol	Meaning
5	QUITFLAG	Bit set 1 by V96E, to cause exit from the integration loop in "TESTLOOP". If the bit is 1, the periodic (P00 and P20 tracking)integration is terminated in "STATINT1" (where bit is reset without another waitlist call for "STATEINT").
4	R31FLAG	Bit set 1 if R31 (V83E or P79) computations are selected, and set 0 if R34 (V85E) is selected.
3	MID1FLAG	Bit set 1 in "MIDTOAV1" to indicate that integration to the specified input time is desired, and reset if entrance to "MIDTOAV2" is made, or if the specified time is less than 12.5 seconds in the future after any time step (as deduced in "MIDTOAV1" or "CKMID2"). A reset value means to integrate to a certain time in the future (generally 12.5 seconds).
2	MIDAVFLG	Bit set 1 in "MIDTOAV2" (at a point after "MIDTOAV1" enters it) to flag the fact that the coasting flight to powered flight logic is desired. Causes "DIFEQ+2" to transfer to "CKMID2" at the completion of a time step. Bit reset in "INTEXIT" at completion of integration.
1	AVEMIDSW	Bit set 1 in "AVETOMID" if "permanent" CSM state vector must be integrated for W-matrix purposes, and reset in "INTEXIT". It is used to inhibit loading of CSM state vector cells at the end of the integration, since these cells (which otherwise would be used for downlink) contain the final results from the Average-G computations.

Meaning of bits in FLGWRD10

Bit 11 of FLGWRD10 is left alone, and other bits set 0, by a fresh start. FLGWRD10 is left alone in "INITSUB".

Bit	Symbol	Meaning
15	PCMANFLG	Bit set 1 for minimum key rendezvous operation of P36 and P79, causing "PIKUP20" to exit after doing "R61CSM".
14	INTINUSE	Bit set 1 to indicate that the "INTSTALL" orbital integration routine has accepted a request for a use of the integration package. Program notation is "INTFLAG".
13	INTGRAB	Bit set 1 to indicate that a restart group phase change has taken place while the routine using the integration package (which previously set bit 14) is performing the computations. Program notation is "REINTFLG".
12	REJCTFLG	Bit set 1 if a mark reject is received when optics in primary P20 mode and bit 7 of FLAGWRD9 = 1. Bit checked just before incorporation of optics mark information into the state vector. Can also be set (due to similar logic) by V86E.
11	HDSUPFLG	Bit used in "STARTAUT" to determine the appropriate initial condition for AZIMANGL (if 1, an angle of 0^o is used).
10	BURNFLAG	Bit used in "HARTBURN" to determine if the CSM did thrusting. Set 0 at start of P40/P41, and set 1 if SPS engine ignition done or if translation hand controller found deflected when "SERVXT1" entered.
9	RANGFLAG	Bit set in "CRS61.1" based on the conic estimate of CSM/LM separation, used if 1 to allow a VHF sample to be obtained (intended to lock out attempts for ranges above the saturated data link of 327.67 nmi).
8	P35FLAG	Bit set 1 in "P35/P75B" during the final computation cycle of P35 and P75; it is used in "AUTOW" to control W matrix initialization, and is reset in "REND5C".
7	AUTOSEQ	Bit set 1 in "REND30S", and left at that value for a PRO to the 00017_8 checklist display in "MINKDISP". If 1, indicates the automatic rendezvous mode, also known as minimum key rendezvous, is in effect.

Bit	Symbol	Meaning
6		Not assigned.
5	MANEUFLG	Bit set 1 in "VN1645" after final PRO response (except in P36), and reset in "REND12" after mark has been incorporated. Used in "AUTOW" logic to indicate that the first mark after a burn opportunity is being processed. Can also be set in "STARTAUT".
4	PTV93FLG	Bit set 1 in "AUTOW" if W matrix initialization after the next burn is desired. Can also be set in "STARTAUT".
3	TPIMNFLG	Bit set 1 in "S34/35.5" on final pass (of P34/P74 and P35/P75). Used in "AUTOW" W matrix initialization logic. Set 0 at start of P34/P74.
2	FULTKFLG	Bit whose value can be changed by use of the display initiated by V57, and whose value is used to control "AUTOW" logic if TPIMNFLG and MANEUFLG bits both 1. A value of 1 for the bit means that do <u>not</u> have both optics and VHF sensors available.
1	PCFLAG	Bit set 1 at the start of P36, for use in controlling program logic, including minimum key rendezvous P52 sequencing.

Meaning of bits in FLGWRD11

FLGWRD11 is initialized to 00000_8 by a fresh start, and is left alone in "INITSUB".

Bit	Symbol	Meaning
15	S32.1F1	Bit set 1 in "CSI/B1" if CSI velocity change is excessive (P31/P32/P72), and used to halt the iteration if another excessive magnitude encountered. It is the equation document flag "S1".
14	S32.1F2	Bit set 1 to indicate the first cycle through the P31/P32/P72 target iteration loop, and then reset 0. It is equation document flag "S2".
13	S32.1F3A	Computation status flag for P31/P32/P72 (see bit 12).
12	S32.1F3B	Computation status flag for P31/P32/P72, used with bit 13 to control computation sequencing. The pair corresponds to equation document flag "S3". The initial condition is 01_2; a tentative alarm generation sets to 00_2; a 50 fps velocity change sets to 11_2; and an angle error sign change sets to 10_2 (and sets velocity change to 10 fps).
11-9		Not assigned.
8	AZIMFLAG	Bit set 1 for options #4 and #5 of P20, used in "R63COM1" to control nature of computation (and in other places as necessary to distinguish between 3-axis and "VECPOINT" solutions).
7	HAFLAG	Bit set to 1 in "P31RECYC" and reset in "P32/P72C" after successful solution reached (used to control branching back to P31 computations).
6	CSISFLAG	Bit included in logic assignments to allow multiple CSI targeting in minimum key rendezvous mode. Due to coding error in "P82", however, P32 computations would always be expected to see the bit as 0.
5-1		Not assigned.

Channel Assignments

Listed below are the channel (input/output interface) assignments of interest for this flight. For more complete discussions of the channel bits, see section IIE of 3420.5-27. By convention, channel numbers are given in octal.

Channel	Bits	Function
03	14-1	Most significant 14 bits from a 33-stage binary counter driven by a 102.4 kc signal derived from the computer master oscillator (which keeps running when the computer is placed in standby). Scale factor B23, units centi-seconds.
04	14-1	Next most significant 14 bits of the counter used for channel 03. Scale factor B9, units centi-seconds.
05	8-1(SM)	X-axis RCS jets for Service Module.

	Jet	Quad	Rotation	Translation
8	6	B	- Yaw	+ X
7	7	B	+ Yaw	- X
6	8	D	- Yaw	- X
5	5	D	+ Yaw	+ X
4	2	A	- Pitch	+ X
3	3	A	+ Pitch	- X
2	4	C	- Pitch	- X
1	1	C	+ Pitch	+ X

05	8-1(CM)	Pitch and yaw RCS jets for Command Module (entry).

	Jet	Rotation
8	6	- Yaw
7	7	+ Yaw
6	8	- Yaw
5	5	+ Yaw
4	2	- Pitch
3	3	+ Pitch
2	4	- Pitch
1	1	+ Pitch

Channel	Bits	Function
06	8-1(SM)	Y-axis and Z-axis RCS jets for Service Module.

	Jet	Quad	Rotation	Translation
8	14	C	- Roll	+ Y
7	15	C	+ Roll	- Y
6	16	A	- Roll	- Y
5	13	A	+ Roll	+ Y
4	10	D	- Roll	+ Z
3	11	D	+ Roll	- Z
2	12	B	- Roll	- Z
1	9	B	+ Roll	+ Z

Channel	Bits	Function
06	4-1(CM)	Roll RCS jets for Command Module (entry).

	Jet	Rotation
4	10	- Roll
3	11	+ Roll
2	12	- Roll
1	9	+ Roll

Channel	Bits	Function
07	7-5	Bits used to select appropriate portion of fixed memory for FBANK register contents (see 3420.5-27) of 30_8 and above. Proper setting determined by program service routines, hence generally not shown in this document. Known as F EXT or SUPERBNK.
10	15-1	Channel used to drive latching relays associated with the display system (bits 15-12 = 0001_2 - 1100_2). Loaded based on entrance to "T4RUPT". Known as OUTO.
11	13	SPS Engine-on signal.
	10	Caution Reset signal (for display system lights).
	9	Test connector Outbit (set 1 in "READACCS" and 0 in "AVGEND", indicating when Average-G is running).
	7	Operator Error light.
	6	Flash of verb and noun lights.
	5	Key Release light.
	4	Temperature Caution light.
	3	Uplink Activity light. Also used for other purposes.
	2	Computer Activity light.

Channel	Bits	Function
11(cont)	1	ISS Warning light.
12	15	ISS Turn-on Delay Complete.
	14	S4B Cutoff (set in "SIVBOFF" and "SPSOFF"; reset in "R00").
	13	S4B Injection Sequence start (reset in "R00").
	11	Disengage optics DAC (digital to analog converter). Bit set in "S40.6" (and in "REDOTVC" for a restart) to avoid driving optics system with TVC commands if optics left in Computer Control mode.
	10	Zero optics (reset in "R00").
	9	S4B Takeover Enable.
	8	TVC Enable. Also enables optics rate command mode.
	6	Enable IMU CDU error counters.
	5	Zero IMU CDU's.
	4	Coarse Align IMU.
	3	Enable star tracker, not used (reset in "R00").
	2	Enable Optics CDU Error Counters (also used for TVC).
	1	Zero Optics CDU's.
13	15	Enable TIME6 countdown (at 1600 pps rate).
	14	Reset trap 32 (bits 10-1 of channel 32).
	13	Reset trap 31B (bits 12-7 of channel 31).
	12	Reset trap 31A (bits 6-1 of channel 31).
	11	Enable Standby.
	10	Test DSKY lights (used for alarms etc.).
	9-8	Not assigned (reset in "R00").
	7	Telemetry word order code bit.
	6	Block cell 0045_8 (uplink inputs etc.).
	5	Connect crosslink input to cell 0045_8.

Channel	Bits	Function
13(cont)	4	Bit set to 1 to initiate transmission of VHF data to cell 0046_8 (if bits 3-1 = 001_2).
	3-1	Bits set to 001_2 to specify that range information from VHF ranging system is to be provided (see bit 4).
14	15	Generate output pulses from CDUXCMD.
	14	Generate output pulses from CDUYCMD.
	13	Generate output pulses from CDUZCMD.
	12	Generate output pulses from TVCYAW/CDUTCMD.
	11	Generate output pulses from TVCPITCH/CDUSCMD.
	10	Generate output pulses from GYROCMD.
	9	Negative gyro torquing command.
	8-7	No gyro axis if 00_2 X gyro if 01_2 Y gyro if 10_2 Z gyro if 11_2
	6	Enable power supply associated with gyro torquing.
15	5-1	Main panel DSKY key code input.
16	7	Optics mark reject.
	6	Optics mark.
	5-1	Navigation panel DSKY key code input.
30	15	Bit sensed 0 if stable member temperature within design limits.
	14	Bit sensed 0 if ISS has been turned on or commanded to be turned on.
	13	Bit sensed 0 if IMU fail indication generated.
	12	Bit sensed 0 if CDU fail indication generated(IMU CDU's).
	11	Bit sensed 0 if IMU cage command generated (by crew).
	10	Bit sensed 0 if CMC given control of Saturn.
	9	Bit sensed as 0 if IMU turned on and operating with no malfunctions.
	7	Bit sensed as 0 if optics CDU fail indication generated.
	5	Bit sensed as 0 if liftoff signal generated by S4B Instrumentation Unit.

Channel	Bits	Function
31	15	Bit sensed as 0 if computer in control of spacecraft ("G&N Autopilot Control").
	14	Bit sensed as 0 if crew selects "Free" mode.
	13	Bit sensed as 0 if crew selects "Hold" mode. If bits 14-13 are 11_2, this indicates that "Automatic" mode selected.
	12	Bit sensed as 0 if translation in -Z direction commanded.
	11	Bit sensed as 0 if translation in +Z direction commanded.
	10	Bit sensed as 0 if translation in -Y direction commanded.
	9	Bit sensed as 0 if translation in +Y direction commanded.
	8	Bit sensed as 0 if translation in -X direction commanded.
	7	Bit sensed as 0 if translation in +X direction commanded.
	6	Bit sensed as 0 if rotation commanded in negative roll direction.
	5	Bit sensed as 0 if rotation commanded in positive roll direction.
	4	Bit sensed as 0 if rotation commanded in negative yaw direction.
	3	Bit sensed as 0 if rotation commanded in positive yaw direction.
	2	Bit sensed as 0 if rotation commanded in negative pitch direction.
	1	Bit sensed as 0 if rotation commanded in positive pitch direction.
32	14	Bit sensed as 0 if "proceed key" (formerly labeled Standby) is depressed.
	6	Bit sensed as 0 if negative roll commanded by minimum impulse controller.
	5	Bit sensed as 0 if positive roll commanded by minimum impulse controller.

Channel	Bits	Function
32(cont)	4	Bit sensed as 0 if negative yaw commanded by minimum impulse controller.
	3	Bit sensed as 0 if positive yaw commanded by minimum impulse controller.
	2	Bit sensed as 0 if negative pitch commanded by minimum impulse controller.
	1	Bit sensed as 0 if positive pitch commanded by minimum impulse controller.
33	15	Flip-flop bit sensed as 0 if computer oscillator has stopped.
	14	Flip-flop bit sensed as 0 if a computer warning indication produced.
	13	Flip-flop bit sensed as 0 if an accelerometer fail indication produced.
	12	Flip-flop bit sensed as 0 if a telemetry end-pulse rejected, indicating an excessive downlink rate.
	11	Flip-flop bit sensed as 0 if an uplink input bit is rejected, indicating an excessive uplink rate.
	5	Bit sensed as 0 if computer control of optics is set (by crew switch).
	4	Bit sensed as 0 if zero optics mode is set (by crew switch). If bits 5-4 = 1, manual mode selected.
	2	Bit sensed as 0 if VHF ranging data considered OK.
34	15-1	Output channel used to contain the first word of a pair telemetered periodically.
35	15-1	Output channel used to contain the second word of a pair telemetered periodically.
77	9-1	Bits indicating cause of restart (for telemetry; reset by "writing" into channel).

Meanings of MODREG Values

Decimal Value	Starting Location	Verb 37	Other Entrances	Title
-0	---		"DOFSTART" "SOMERR2"	Fresh start (display of MODREG is suppressed).
00	---	x	"VERB96" "AUTO37"	CMC Idling.
01	"GTSCPSS"	x		Prelaunch Initialization.
02	"SLEEPIE"		"ESTIMS" "GTSOPTCS" (after P03)	Prelaunch Gyro Compassing.
03	"GCOMPVER"		"CKOPTVB" (V65E)	Prelaunch Optical Verification of Gyrocompassing.
06	"P06"	x		CMC Power Down (power up "POSTAND").
07	---			System Test (start in erasable memory, then to "GEOIMUTT").
11	"P11"		"CHKCOMED"	Earth Orbit Insertion Monitor.
15	"P15JOB"	x		TLI Initiate/Cutoff.
20	"PROG20"	x	"P86" "REND3OS"	Universal Tracking and Rendezvous Navigation.
21	"PROG21"	x		Ground Track Determination.
22	"PROG22"	x		Orbital Navigation.
23	"P23"	x		Cislunar Midcourse Navigation.
24	"PROG24"	x		Rate-aided Optics Tracking.
27	"V73UPDAT"		"V7iUPDAT" (i = 0-3)	CMC Update.
29	"P29"	x		Time of Longitude.
30	"P30"	x		External Delta V.
31	"P31"	x	"P81"	Height Adjustment Maneuver (HAM).
32	"P32"	x	"P82"	Co-elliptic Sequence Initiation (CSI).
33	"P33"	x	"P83"	Constant Delta Altitude (CDH).
34	"P34"	x	"P84"	Transfer Phase Initiation (TPI).
35	"P35"	x	"P85"	Transfer Phase Midcourse (TPM).

Decimal Value	Starting Location	Verb 37	Other Entrances	Title
36	"P36"	x	"P86"	Plane Change (PC).
37	"P37"	x		Return to Earth.
40	"P40CSM"	x	"BURNHOW"	SPS Thrusting.
41	"P41CSM"	x	"BURNHOW" "P86"	RCS Thrusting.
47	"P47CSM"	x		Thrust Monitor.
51	"P51"	x		IMU Orientation Determination.
52	"PROG52"	x	"P86"	IMU Realign.
53	"P51"	x		Backup IMU Orientation Determination.
54	"PROG52"	x		Backup IMU Realign.
61	"P61"	x		Entry Preparation.
62	"P62"	x	"NEWRNVN"	Entry - CM/SM Separation and Pre-Entry Maneuver.
63	"P63"		"P62.1" "WAKEP62"	Entry Initialization.
64	"INITROLL" "KEP2" (if V below 27,000 fps at 0.05 g).			Entry - Post 0.05G.
65	"UPCONTRL"		"RANGER"	Entry - Upcontrol.
66	"KEP2"		"UPCONTRL"	Entry - Ballistic.
67	"PREFINAL"		"HUNTEST1" "KEP2" "UPCONTRL"	Entry - Final Phase.
72	"P72"	x		LM CSI Targeting.
73	"P73"	x		LM CDH Targeting.
74	"P74"	x		LM TPI Targeting.
75	"P75"	x		LM TPM Targeting.
76	"P76ER77"	x	"HARTBURN"	Target Delta V.
77	"P76ER77"	x		Impulsive Delta V.
79	"P79"	x	"P85"	Rendezvous Final.
81-86	"P81"-"P86"		"REND30S"	Driver Programs for Minimum Key Rendezvous.

Special Program Routines

Routine Number	Starting Location	Title
00	"GOTOPOOH"	Final Automatic Request Terminate.
	"V37"	Manual input of V37E xx E.
01	"ABCLOAD"	Erasable and Channel Modification Routine (V25 N07 E).
02	"R02BOTH"	IMU Status Check.
03	"DAPDISP"	Digital Autopilot Data Load (V48E).
05	"VB64"/ "SBANDANT"	S-Band Antenna (V64E).
21	---	Rendezvous Tracking Sighting Mark (enabled when bit 14 of FLAGWRD2 = 1; no special displays).
22	"R22"	Rendezvous Tracking Data Processing.
23	"GOTOR23"/ "R23CSM"	Backup Rendezvous Tracking Sighting Mark (V54E).
30	"V82PERF"/ "V82CALL"	Orbit Parameter Display (V82E).
31	"V83PERF"/ "R31CALL"	Rendezvous Parameter Display Routine #1 (V83E/P79).
33	---	CMC/LGC Clock Synchronization (DSKY procedure).
34	"V85PERF"/ "R31CALL"	Rendezvous Parameter Display Routine #2 (V85E).
36	"V90PERF"/ "R36"	Rendezvous Out-of-Plane Display (V90E).
40	"CLOCKJOB"	SPS Thrust Fail (NVWORD1 = -0).
41	"MIDTOAV1" "MIDTOAV2"	State Vector Integration.
50	"CAL53A"	Coarse Align.
52	"R52"	Automatic Optics Positioning.
53	"R53"	Sighting Mark.
54	"R54"	Sighting Data Display.
55	"R55"	Gyro Torquing.

Routine Number	Starting Location	Title
56	"R56"	Alternate LOS Sighting Mark.
57	"R57"	Optics Calibration.
60	"R60CSM"	Attitude Maneuver.
61	"R61CSM"	Tracking Attitude.
62	"CREWMANU"/ "R62DISP"	Crew-defined Maneuver (V49E).
63	"V89PERF"/ "V89CALL"	Rendezvous Final Attitude (V89E).
66	"R66CSM"	Universal Tracking options #1/#5.
67	"R67"	Automatic Rotation Routine (Universal Tracking option #2).

Checklist and Option Codes

Checklist (V50 N25)

Octal Code	Location	Significance
0013	"P52D"	Key in normal or gyro torque coarse align.
0014	"R51K"	Key in fine alignment option.
0015	"P51A" "P52C" "R57"	Perform celestial body acquisition.
0016	"MKVB50" "R56"	Key in terminate mark sequence.
0017	"MINKDISP"	Perform MINKEY Rendezvous selection.
0020	"PERF20"	Perform MINKEY plane change gyro pulse torquing.
0041	"P62"	Switch CM/SM separation to up.
0062	"P06"	Key CMC to standby.
0202	"V94ENTER"	Perform P23 three-axis maneuver.
0204	"P40SXTY"	Key in SPS gimbal trim test option.

Option Selection (V04 N06)

Octal Option Identification (in R1)	Octal Option Selection (in R2)	Location	Significance
0001		"P52B"	Specify P52/P54 IMU Orientation.
	0001		Preferred Orientation.
	0002		Nominal Orientation.
	0003		[REFSMMAT] Orientation.
	0004		Landing Site Orientation.
0002		"PROG21"	Specify P21/P29 Vehicle.
	0001		This vehicle (CSM).
	0002		Other vehicle (LM).

Octal Option Identification (in R1)	Octal Option Selection (in R2)	Location	Significance
0007		"P37E"	Specify P37 Propulsion System.
	0001		SPS.
	0002		RCS.
0024		"PROG20"	Specify P20 Mode.
	0000		Rendezvous, "VECPOINT".
	0001		Celestial body, "VECPOINT".
	0002		Body rotation.
	0004		Rendezvous, 3-axis.
	0005		Celestial body, 3-axis.
	Option Selection (V04 N12)		
0002		"V82CALL"	Specify R30 Vehicle.
	0001		This vehicle (CSM).
	0002		Other vehicle (LM).
0004		"V57CALL"	Specify FULTKFLG Setting.
	0000		VHF *and* optics.
	0001		VHF *or* optics.

Error Codes

The following error codes are included in the program. The "Use" column indicates the type of error code that is provided:

B signifies an entrance to the "BAILOUT" routine (to generate a "software restart" for a nominally recoverable problem).

D signifies that a display of the alarm code (V05 N09) is generated by the program after "ALARM" is entered.

L signifies an entrance to the "ALARM" routine without an automatic N09 display by program (the PROG light is energized).

P signifies an entrance to the "POODOO" routine to generate a "software restart". If Average-G is running or EXTVBACT (13-1) is non-zero (e.g. extended verb with displays), has same effect as a "B" type; otherwise, all restart groups are reset, giving entrance to "GOTOPOOH".

Error codes are stored in the FAILREG set of cells, and may be examined by manual input of V05 N09 E (see page GENP-31).

Octal Code	Use	Location	Significance
00110	L	"MKREJECT"	Mark reject has been entered but ignored (previous marks, if any, already accepted).
00113	L	"MARKRUPT"	No input information in channel 16 when program interrupt #6 processed.
00114	L	"MARKDIF"	More marks made than desired.
00115	L	"VBCOARK"	V41 N91 E keyed with optics mode switch not at computer position.
00116	L	"OPTMON"	Optics mode switch moved from zeroing position before expiration of minimum time period (about 15 seconds).
00117	L	"VBCOARK"	V41 N91 E keyed but CMC has reserved optics CDU's (from gimbal test in P40 to end of burn). Not expected since V41N91 is P00 only.
00120	L	"OPTTEST"	Optics torque request with optics not zeroed since last fresh start or hardware restart.
00121	L	"MARKDIF"	IMU CDU angles changed by more than 0.033° in short interval (e.g. 0.05 seconds) after mark. Bypassed in P24.

Octal Code	Use	Location	Significance
00205	L	"SERVICER"	Maximum accelerometer output (in excess of 6399 pulses in a 2-second period).
00206	L	"IMUZERO"	Request for zeroing of IMU CDU's rejected since in coarse align with gimbal lock.
00207	L	"IMUMON"	Inertial sub-system turn-on request not present for the required 90-second period.
00210	L	"IMUZERO" "P51" "RO2BOTH"	IMU not operating.
00211	L	"COARS2"	Excessive angle error (more than 2^o) after IMU coarse align.
00212	L	"C33TEST" "PIPFREE"	Accelerometer (PIPA) fail signal with failure inhibit set (to inhibit generation of ISS warning).
00213	L	"TNONTEST"	ISS turn-on request with IMU operate signal not present.
00214	L	"IMUMON"	Program using IMU when IMU turned off.
00217	L	"217ALARM"	Error return from "IMUSTALL" routine.
00220	L	"RO2BOTH"	IMU not aligned (bit 13, REFSMFLG, of FLAGWRD3 is zero).
00401	L	"CALCGA" "UTOPT45"	Desired middle gimbal angle exceeds 60^o.
00402	L	"PERF20"	Second MINKEY pulse torque request must be honored.
00404	D	"COM52"	Priority alarm. Optics not tracking LM and target outside of 90^o limit.
00405	D	"PICEND"	Acceptable star pair is not available.
00406	L	"GOTOR23"	V54E rejected since P20 rendezvous mode not running and/or bit 5(TRACKFLG) of FLAGWRD1 is zero.

Octal Code	Use	Location	Significance
00421	L	"DIFEQ+2"	Overflow in orbital integration package (probably due to W-matrix).
00600	D	"CIRCL"	No solution on first iteration for P31/P32/P72 (desired TPI LOS not intersect necessary circle).
00601	D	"CIRCL"	CSI periapsis too small (85 nmi/35000 ft).
00602	D	"CIRCL"	CDH periapsis too small (85 nmi/35000 ft).
00603	D	"CIRCL"	Time interval between CSI and CDH too small.
00604	D	"CIRCL"	Time interval between CDH and TPI too small.
00605	D	"CSI/B1" "GAMDV25" "PREC125" "PREC175" "RTE360"	Number of iterations exceeds loop maximum.
00606	D	"CSI/B1"	CSI velocity change exceeds maximum for a second time.
00611	D	"P33/P73B" "P34/P74C"	No T_{ig} for given elevation angle.
00612	D	"P37"	State vector in wrong sphere of influence (i.e. moon's) in P37 at ignition.
00613	D	"PREC175"	Reentry angle outside of limits in P37.
00777	L	"SETISSW"	A PIPA fail caused ISS warning.
01102	L	"ERRORS"	CMC self-test error.
01105	L	"C33TEST"	Downlink too fast.
01106	L	"C33TEST"	Uplink too fast.
01107	L	"GOPROG4"	Restart phase table entries disagree (hence a fresh start is done on the assumption that erasable memory contents not valid).

Octal Code	Use	Location	Significance
01301	L	"ARCCOM"	Arc cosine or arc sine argument too big.
01407	L	"S40.8"	V_g increasing.
01426	D	"S61.1A"	IMU orientation unsatisfactory for entry (program waits 10 seconds, then proceeds).
01427	D	"S61.1A"	IMU orientation reversed for entry, so ball reads 0° in roll for lift down. Program waits 10 seconds, then proceeds.
01520	L	"AUTO37" "ISITPOO"	V37 (change major mode) cannot be accepted at this time.
01600	L	"SOMEERRR"	Overflow in prelaunch gyro drift test (probably).
01601	L	"SOMERR2"	Error return from mode switching or gyro torquing in prelaunch program (P01, P02, or P07).
01703	L	"CKMID2" "MIDTOAV1"	Ignition time slipped since insufficient time available for integration.
03777	L	"SETISSW"	ICDU fail caused ISS warning.
04777	L	"SETISSW"	ICDU and PIPA fails caused ISS warning.
07777	L	"SETISSW"	IMU fail caused ISS warning.
10777	L	"SETISSW"	IMU and PIPA fails caused ISS warning.
13777	L	"SETISSW"	IMU and ICDU fails caused ISS warning.
14777	L	"SETISSW"	IMU, ICDU, and PIPA fails caused ISS warning.
20430	P	"GOBAQUE"	Faulty state vector used for precision integration (e.g. position magnitude too low), causing overflow that rectification would not eliminate.
20607	P	"COMMNOUT" "TIMERAD" "TIMETHET"	No solution from conic routine.
20610	P	"V2T100"	P37 position vector at ignition too small.

Octal Code	Use	Location	Significance
21204	P	"LONGCALL" "WAITLIST"	Task insertion attempted with 0 or negative delay (see Section VIIA of 3420.5-27).
21206	P	"ENDIDLE" "NVSUBUSY"	Display system address buffers full, so another job cannot be accommodated.
21210	P	"IMUSTALL"	Job already waiting in IMU "stall" routine, hence another job cannot be accommodated.
21302	P	Square root (page MATH-4)	Square root argument too negative.
21501	P	"DSPALARM"	Display system alarm from internal ("NVSUB") use (such as an illegal noun).
21502	P	"FLASHSUB"	Illegal request for a flashing display (one already in system, so second cannot be satisfactorily accommodated).
21521	P	"GTSCPSS"	New program request (for P01) cannot be made after P11 started.
31104	B	"DELAYJOB"	Too many jobs (more than 4) attempting to use job delay routine.
31201	B	"FINDVAC"	No VAC areas available (see Section VIIB of 3420.5-27).
31202	B	"NOVAC2"	No job register sets available (see Section VIIB of 3420.5-27).
31203	B	"WAIT2" "WTLST5"	New task cannot be inserted successfully in waitlist system queue (see Section VIIA of 3420.5-27).
31211	B	"TESTMARK"	Optics use not allowed with extended verb being performed.

Job Priorities

Given below are the priorities with which the various Jobs are executed. A Job of lower priority will be suspended in favor of one of higher priority, except where prohibited by other program logic (such as the "INTSTALL" routine). The sequence with which Jobs of the same priority are performed is somewhat indeterminate (VAC-area Jobs of a given priority take precedence over non-VAC-area ones, see 3420.5-27), but they will all be done before Jobs of lower priority.

Priority	Starting Address	Established By
35_8	"NBDONLY"	"SVCT3"
34_8	"AMBGUPDT"	"KMATRIX"; "REDAP"
33_8	"MAKEPLAY"	"MAKEPLAY" (change)
32_8	"JAMTERM"	"NVDSP"
	"NORMLIZE"	"PREREAD"
30_8	"CHARIN"	"KEYCOM"
	"DSPMMJB"	"GOPROG4"; "NEWMODEX"; "TSTLTS3"
	"MONDO"	"MONREQ"
	"PROCKEY"	"PROCEEDE"
	"P47BODY"	"TIGON"
	"UPJOB"	"UPSTORE"
27_8	"CLOCKJOB"	"CLOKTASK"
	"DYNDISP"	"CLOKTASK"
	"RENDISP2"	"RENDISP"
	"R23CSM"	"R23CSM" (change)
26_8	"LIGHTON"	"VHFREAD"
	"NEWDELHI"	"UPDTCALL"
	"RANGERD1"	"VHFREAD"
	"REDOR22"	"REDOR22" (change)
	"R22"	"NDUTINPT"; "R22" (change)
	"R67RSTRT"	"R67RSTRT" (change); "UPDTCALL"
	"R67START"	"R67"
25_8	"ENDEXT"	"TICKTEST"
24_8	"ENDEXT"	"ENTANSWR"
	"R53JOB"	"R52C"
22_8	"ALFLT"	"ALLOOP"
	"ALFLT$_d$"	"ALLOOP$_d$"
	"CHKCOMED"	"CHKCOMED" (change, ineffective)
	"MKVBDSP"	"MKREJECT"
	"MKVB5X"	"MARKDONE"

Priority	Starting Address	Established By
21_8	"LASTBIAS"	"PREREAD"
	"1/CHECK"	"1/PIPA"
20_8	"GTSCPSS"	"GTSCPSS" (change)
	"POSTAND"	"P06" (via restart)
	"S40.13"	"TIG-5"; "V97E"
	"SERVICER"	"READACCS"
	"TIGNOW"	"TTG/0"
16_8	"AZMTHCG1"	"CHAZFOGC"
	"GCOMPVER"	"CKOPTVB"
	"P67.1"	"PREDICT3"
	"RANGER"	"RANGER" (change)
	"R21END"	"R21END" (change)
	"R23CSM"	"GOTOR23"
15_8	"P47BODY"	"P47BODY" (change)
	"PLAYJUM1"	"NORMRET"
14_8	"ENDMANUV"	"ENDMANU"
	"INITDSP"	(restart logic)
	"PIKUP20"	"PIKUP20" (change)
	"P40BLNKR"	"TIGBLNK"
	"RELINUS"	"RELINUS" (change)
	"V94ENTER"	"VERB94" (via restart)
13_8	"ENTANSWR"	"ENTANSWR" (change)
	"GTSCPSS"	"V37XEQ"
	"PROG20"	"V37XEQ"
	"PROG21"	"V37XEQ"
	"PROG22"	"V37XEQ"
	"PROG24"	"V37XEQ"
	"PROG52"	"V37XEQ"
	"P06"	"V37XEQ"
	"P15JOB"	"V37XEQ"
	"P23"	"V37XEQ"
	"P29"	"V37XEQ"
	"P30"	"V37XEQ"
	"P31"	"V37XEQ"
	"P32"	"V37XEQ"
	"P33"	"V37XEQ"
	"P34"	"V37XEQ"
	"P35"	"V37XEQ"
	"P36"	"V37XEQ"
	"P37"	"V37XEQ"
	"P40CSM"	"V37XEQ"
	"P41CSM"	"V37XEQ"
	"P47CSM"	"V37XEQ"
	"P51"	"V37XEQ"
	"P61"	"V37XEQ"
	"P62"	"V37XEQ"

Priority	Starting Address	Established By
13_8 (cont)	"P63"	"WAKEP62"
	"P65.1"	"RANGER"
	"P72"	"V37XEQ"
	"P73"	"V37XEQ"
	"P74"	"V37XEQ"
	"P75"	"V37XEQ"
	"P76ER77"	"V37XEQ"
	"P79"	"V37XEQ"
	"P81"	"V37XEQ"
	"P82"	"V37XEQ"
	"P83"	"V37XEQ"
	"P84"	"V37XEQ"
	"P85"	"V37XEQ"
	"P86"	"V37XEQ"
	"S61.1A"	"S61.1C"
12_8	"POSTBURN"	"ENGINOFF"
	"POSTTLI"	"SIVBOFF"
	"P40S/SV"	"T6RESET"
10_8	"ATTACHIT"	"ATTACHED"
	"DONOUN46"	"DAPDISP"
	"LMTOCM"	"LMTOCMSV"
	"R62DISP"	"CREWMANU"
	"S40.9"	"SETUP.9"
	"V89CALL"	"V89PERF"
07_8	"HARTBURN"	"AFTERBRN"
	"R36"	"V90PERF"
	"V82GOFF1" (VAC area)	"V82GOFLP"
	"V82GON1" (VAC area)	"V82CALL"
	"V82PERF" (no VAC area)	"V82PERF" (change)
05_8	"MARKDISP"	"MARKDIF"
	"R31CALL"	"V83PERF"; "V85PERF"
	"SBANDANT"	"SBANDANT" (change)
	"STATINT1"	"NDUTINPT"; "R00" (via restart logic); "STATEINT"
	"V67CALL"	"V67"
	"V83PERF"	"P79A"
04_8	"SBANDANT"	"SBANDANT"(change); "VB64"
03_8	"V83CALL"	"R31CALL"
01_8	"ATERJOB"	"ATERTASK"
Same as user	"ENDRET2"	"ENDRET2" (change)
	"INITDSP"	"INITDSP" (change)
	"MAKEPLAY"	"GODSPRS+1"
	"NV50DSP"	"NV50DSP" (change)

Erasable Memory Prelaunch Load

Listed below are the quantities in erasable memory which should be (or, in a few cases, can be) loaded as part of the prelaunch erasable memory load. The "Address" information gives the erasable memory bank format of the address, followed by the erasable CADR format (if different) needed if the cell is loaded by specification of the machine address. An asterisk to the left of the address means that the quantity would have to be loaded by uplink or DSKY means (other than by use of a noun assigned specifically to that quantity) in the event that re-initialization of the complete erasable memory were necessary in flight (these items exclude such quantities as the state vector etc., for which an uplink capability must exist for the nominal mission).

The "Reference" column gives one routine where the quantity has a significant application (or is updated), followed by the page in this document where the quantity is defined (scale factor, units, etc.).

Address (Octal)	Memory Notation	Document Notation	Reference	Comments
0075	FLAGWRD1	FLAGWRD1	"DOFSTART" pg. 47	Bit 12(NODOP01) left alone until start P11, when bit set 1.
0077	FLAGWRD3	FLAGWRD3	"DOFSTART" pg. 51	Bit 13(REFSMFLG) left alone by fresh start.
0104	FLAGWRD8	FLAGWRD8	"DOFSTART" pg. 59	Bits 12(CMOONFLG), 11(LMOONFLG), and 8(SURFFLAG) left alone by fresh start.
0106	FLGWRD10	FLGWRD10	"DOFSTART" pg. 63	Bit 11(HDSUPFLG) left alone by fresh start.
*0374	C31FLWRD	C31FLWRD	"RCSATT" pg. DPRC-25	Backup for bits 15-13 of channel 31 and bits 5-4 of channel 33.
*0737	NO.PASS	C_{nopass}	"R52C" pg. OPTC-25	Frequency of R52 check in P24 to see if new landmark estimate should be computed.
1016	N26/PRI	N26dPRI	"VBRQEXEC" pg. DATA-38	R1 of N26. Set 0 to lock out V30 and V31.
1133-4	PGNCSALT	$C_{pgncsat}$	"P11" pg. BOOS-10	Altitude of IMU above 6,373,338 meters (used to compute initial conditions for state vector).
1135-6	PADLONG	C_{pdong}	"P11" pg. BOOS-10	Pad longitude used to compute initial conditions for state vector at liftoff.
*1341	CDUCHKWD	$C_{cduchkwd}$	"MARKRUPT" pg. OPTC-25	Delay before checking IMU CDU after a non-P24 optics mark.

Address (Octal)	Memory Notation	Document Notation	Reference	Comments
*1342-3	RTED1	C_{rted1}	"RTE360" pg. RTER-19	Constant term for desired flight path angle cotangent in P37.
*1344	DVTHRESH	$C_{dvthresh}$	"S40.8" pg. STER-9	Velocity increment gate for Delta-V monitor.
*1345-6	HORIZALT	$C_{horizalt}$	"HORIZ" pg. MEAS-29	Horizon altitude used in P23 (with HORISLP) above Fischer ellipse.
*1347	ALTVAR	C_{altvar}	"REND7" pg. MEAS-28	A priori measurement accuracy of back-up optics in R23.
*1350	EMDOT	C_{emdot}	"S40.8" pg. STER-9	Mass flow rate for SPS engine.
*3,1452 (1452)	PBIASX	$C_{pipabias_x}$	"1/PIPA" pg. IMUC-20	$PIPA_x$ bias correction.
*3,1453 (1453)	PIPASCFX	$C_{pipascf_x}$	"1/PIPA" pg. IMUC-21	$PIPA_x$ scale factor correction.
*3,1454 (1454)	PBIASY	$C_{pipabias_y}$	"1/PIPA" pg. IMUC-20	$PIPA_y$ bias correction.
*3,1455 (1455)	PIPASCFY	$C_{pipascf_y}$	"1/PIPA" pg. IMUC-21	$PIPA_y$ scale factor correction.
*3,1456 (1456)	PBIASZ	$C_{pipabias_z}$	"1/PIPA" pg. IMUC-20	$PIPA_z$ bias correction.
*3,1457 (1457)	PIPASCFZ	$C_{pipascf_z}$	"1/PIPA" pg. IMUC-21	$PIPA_z$ scale factor correction.
*3,1460 (1460)	NBDX	$C_{nbd,x}$	"1/PIPA" pg. IMUC-20	X Gyro bias correction.
*3,1461 (1461)	NBDY	$C_{nbd,y}$	"1/PIPA" pg. IMUC-20	Y Gyro bias correction.
*3,1462 (1462)	NBDZ	$C_{nbd,z}$	"1/PIPA" pg. IMUC-20	Z Gyro bias correction.
*3,1463 (1463)	ADIAX	$C_{ad,x}$	"1/PIPA" pg. IMUC-20	X Gyro input axis drift.
*3,1464 (1464)	ADIAY	$C_{ad,y}$	"1/PIPA" pg. IMUC-20	Y Gyro input axis drift.

Address (Octal)	Memory Notation	Document Notation	Reference	Comments
*3,1465 (1465)	ADIAZ	$C_{ad,z}$	"1/PIPA" pg. IMUC-20	Z Gyro input axis drift.
*3,1466 (1466)	ADSRAX	$C_{sr,x}$	"1/PIPA" pg. IMUC-21	X Gyro spin axis drift.
*3,1467 (1467)	ADSRAY	$C_{sr,y}$	"1/PIPA" pg. IMUC-21	Y Gyro spin axis drift.
*3,1470 (1470)	ADSRAZ	$C_{sr,z}$	"1/PIPA" pg. IMUC-21	Z Gyro spin axis drift.
*3,1706-10 (1706-10)	TEPHEM	T_{eph}	"P11" pg. BOOS-13	Triple precision time between ephemeris origin and clock origin (updated at liftoff).
*3,1711-6 (1711-6)	UNITW	$\underline{C}_{unitw}$	"CALCGRAV" pg. GENP-29	Earth polar axis unit vector (scale factor B0) in reference coordinates.
*3,1763 (1763)	EIMP1SEC	$C_{eimp1sec}$	"S40.13" pg. STER-9	Impulse from first second of SPS thrusting.
*3,1764 (1764)	EFIMP01	$C_{efimp01}$	"S40.13" pg. STER-9	Slope of minimum impulse curve for SPS 0-1 second.
*3,1765 (1765)	EFIMP16	$C_{efimp16}$	"S40.13" pg. STER-9	Slope of minimum impulse curve for SPS 1-6 seconds, i.e. thrust.
*3,1766 (1766)	E3J22R2M	$C_{e3j22r2m}$	"OBLATE" pg. ORBI-24	J_{22} term for lunar gravity model.
*3,1767 (1767)	E32C31RM	$C_{e32c31rm}$	"OBLATE" pg. ORBI-24	C_{31} term for lunar gravity model.
*3,1770 (1770)	TRUNSF	C_{trunsf}	"RATESUB" pg. OPTC-25	Scale factor for rate command to optics trunnion in P24.
*3,1771 (1771)	SHAFTSF	$C_{shaftsf}$	"RATESUB" pg. OPTC-25	Scale factor for rate command to optics shaft in P24.
*4,1400 (2000)	WRENDPOS	WRENDPOS	"REND5C" pg. MEAS-38	W matrix initialization for rendezvous. Can be changed by V67 processing, and "VN1645".
*4,1401 (2001)	WRENDVEL	WRENDVEL	"REND5C" pg. MEAS-38	W matrix initialization for rendezvous. Can be changed by V67 processing, and "VN1645".

Address (Octal)	Memory Notation	Document Notation	Reference	Comments
*4,1402 (2002)	RMAX	C_{rmax}	"REND7" pg. MEAS-29	Maximum automatic rendezvous position update.
*4,1403 (2003)	VMAX	C_{vmax}	"REND7" pg. MEAS-29	Maximum automatic rendezvous velocity update.
*4,1404 (2004)	WORBPOS	WORBPOS	"S22.1" pg. MEAS-38	W matrix initialization for P22. Can be changed by V67 processing.
*4,1405 (2005)	WORBVEL	WORBVEL	"S22.1" pg. MEAS-38	W matrix initialization for P22. Can be changed by V67 processing.
*4,1406 (2006)	S22WSUBL	$C_{s22wsubl}$	"S22NXTIN" pg. ORVN-24	W matrix initialization for P22 for known landmark.
*4,1407-10 (2007-10)	RPVAR	C_{rpvar}	"S22NXTIN" pg. ORVN-24	Variance of primary body radius vector, used to compute P22 W matrix initialization for unknown landmark.
*4,1411-6 (2011-6)	504LM	$\underline{C}_{504lm}$	"RP-TO-R" pg. COOR-13	Libration correction vector for moon (B0 radians)
*4,1417-20 (2017-20)	EMSALT	C_{emsalt}	"S61.2" pg. DISP-20	Entry monitoring system initialization altitude.
*4,1421-2 (2021-2)	ATIGINC	$C_{atiginc}$	"P35" pg. BURN-37	CSM-active time delay in P35 until midcourse maneuver
*4,1423-4 (2023-4)	PTIGINC	$C_{ptiginc}$	"P75" pg. BURN-37	CSM-passive (i.e. LM-active) time delay in P75 until midcourse maneuver.
*4,1425-32 (2025-32)	RLS	$RL\underline{S}$	"SETRE" pg. COOR-19	Landing site position vector. Can be updated (in "S22I=N") in P22.
*4,1433-5 (2033-5)	TIMEMO	C_{timemo}	"LSPOS" pg. COOR-13	Triple precision elapsed time between ephemeris origin and midpoint of interval for which lunar-position polynomial is valid.
*4,1436-43 (2036-43)	VECOEM	$\underline{C}_{vcem00}$	"LUNPOS" pg. COOR-14	Vector coefficient of highest-order term (t^9 for position) in lunar-position polynomial.
*4,1444-51 (2044-51)	VECOEM +6	$\underline{C}_{vcem06}$	"LUNPOS" pg. COOR-14	Vector coefficient of next highest-order term (t^8 for position) in lunar-position polynomial.

Address (Octal)	Memory Notation	Document Notation	Reference	Comments
*4,1452- 4,1523 (2052- 2123)	VECOEM +12	$\underline{C}_{vcem12}$ to $\underline{C}_{vcem48}$	"LUNPOS" pg. COOR-14	Vector coefficients of subsequent terms (t^7 through t^1 for position) in lunar-position polynomial.
*4,1524-31 (2124-31)	VECOEM +54	$\underline{C}_{vcem54}$	"LUNPOS" pg. COOR-14	Vector coefficient of constant (non-time dependent) term in lunar-position polynomial.
*4,1532-7 (2132-7)	RESO	$\underline{C}_{reso}$	"LSPOS" pg. COOR-13	Position vector of sun relative to earth at C_{timemo}.
*4,1540-5 (2140-5)	VESO	$\underline{C}_{veso}$	"LSPOS" pg. COOR-14	Velocity vector of sun relative to earth at C_{timemo}.
*4,1546-7 (2146-7)	OMEGAES	$C_{omegaes}$	"LSPOS" pg. COOR-13	Angular velocity of $\underline{C}_{reso}$ at C_{timemo}.
*4,1754 (2354)	DTF	C_{dtf}	"SIVBCOMP" pg. BOOS-10	P15 S4B cutoff time delay.
*4,1767-70 (2367-70)	HAMDELH	$C_{hamdelh}$	"P31RT" pg. BURN-37	P31 altitude difference.
*4,1771 (2371)	WRDTIME	$C_{wrdtime}$	"AUTOW" pg. MINK-11	Minimum time between automatic W matrix initializations.
*4,1772 (2372)	MINBLKTM	$C_{minblktm}$	"AUTOW" pg. MINK-10	Check for time since last mark in automatic W matrix initialization.
*4,1773 (2373)	TBEFCOMP	$C_{tbefcomp}$	"AUTOW" pg. MINK-10	Check of time until final target computation in auto W matrix logic.
*4,1774 (2374)	BRNBLKTM	$C_{brnblktm}$	"AUTOW" pg. MINK-10	Interval w/o marks before & after doing a maneuver.
*4,1775 (2375)	MAXWTIME	$C_{maxwtime}$	"AUTOW" pg. MINK-10	Maximum desired time between successive W matrix initializations.
*4,1776 (2376)	FINCMPTM	$C_{fincmptm}$	"AUTOW" pg. MINK-10	Time before maneuver of final mark.
*4,1777 (2377)	INTVAR	C_{intvar}	"REND7" pg. MEAS-29	Integration variance employed in R22.
5,1400-1 (2400-1)	AZIMUTH	C_{azmth}	"GTSCPSS" pg. PREL-10	Azimuth of vehicle Z-axis.

Address (Octal)	Memory Notation	Document Notation	Reference	Comments
5,1402-3 (2402-3)	LATITUDE	C_{atd}	"ESTIMS" pg. PREL-10	Launch pad latitude.
5,1432 (2432)	TAZEL1	TAZ_1	"GCOMPVER" pg. PREL-13	Azimuth of optics target #1.
5,1433 (2433)	TAZEL1 +1	TEL_1	"GCOMPVER" pg. PREL-13	Elevation of optics target #1.
5,1434 (2434)	TAZEL1 +2	TAZ_2	"GCOMPVER" pg. PREL-13	Azimuth of optics target #2.
5,1435 (2435)	TAZEL1 +3	TEL_2	"GCOMPVER" pg. PREL-13	Elevation of optics target #2.
5,1633-4 (2633-4)	LAUNCHAZ	LAUNCHAZ	"GTSCPSS" pg. PREL-12	Desired launch azimuth. Can be updated using V78E. Should be loaded with same bit pattern as would be produced by V78E, to avoid re-initializing of gyro compassing unnecessarily.
*6,1400 (3000)	WMIDPOS	WMIDPOS	"POINTAXS" pg. MEAS-38	W matrix initialization for midcourse navigation. Can be changed by V67 processing.
*6,1401 (3001)	WMIDVEL	WMIDVEL	"POINTAXS" pg. MEAS-38	W matrix initialization for midcourse navigation. Can be changed by V67 processing.
*6,1402-3 (3002-3)	RVAR	C_{rvar}	"REND7" pg. MEAS-29	Value of $(\text{percent error}/100)^2$ for VHF ranging measurement.
*6,1404-6 (3004-6)	RVARMIN	$C_{rvarmin}$	"REND7" pg. MEAS-29	Triple precision complement of minimum variance for VHF measurement (a negative number in memory for coding ease).
*6,1407 (3007)	LADPAD	C_{adpad}	"STARTENT" pg. ENRY-18	Nominal L/D.
*6,1410 (3010)	LODPAD	C_{odpad}	"STARTENT" pg. ENRY-18	Final phase L/D.
*6,1411 (3011)	ALFAPAD	$C_{alfapad}$	"P62.1" pg. ENTP-6	Nominal entry trim angle (expected to be negative number).

Address (Octal)	Memory Notation	Document Notation	Reference	Comments
*6,1412 (3012)	P37RANGE	$C_{p37range}$	"RTEVN" pg. RTER-19	Override on entry range (not time) in P37 if cell non-zero.
*6,1413 (3013)	ETDECAY	C_{tdecay}	"S40.8" pg. STER-10	Value of thrust decay (cell subtracted from time-to-go).
*6,1414 (3014)	EKPRIME	$C_{ekprime_0}$	"TVCINIT1" pg. DPTV-16	LM-off cross-product steering gain factor (for TVC DAP input).
*6,1415 (3015)	EKPRIME +1	$C_{ekprime_1}$	"TVCINIT1" pg. DPTV-16	LM-on high bandwidth cross-product steering gain factor.
*6,1416 (3016)	EKTLX/I	$C_{ektlxdi_0}$	"TVCINIT1" pg. DPTV-16	LM-off TVC DAP nominal gain.
*6,1417 (3017)	EKTLX/I +1	$C_{ektlxdi_1}$	"TVCINIT1" pg. DPTV-16	LM-on high bandwidth TVC DAP nominal gain.
*6,1420 (3020)	EKTLX/I +2	$C_{ektlxdi_2}$	"SWICHOVR" pg. DPTV-16	LM-on low bandwidth TVC DAP nominal gain.
*6,1421 (3021)	EREPFRAC	$C_{erepfrac_0}$	"TVCINIT1" pg. DPTV-17	LM-off TVC DAP repetitive c.g. correction fraction.
*6,1422 (3022)	EREPFRAC +1	$C_{erepfrac_1}$	"TVCINIT1" pg. DPTV-17	LM-on high b.w. TVC DAP repetitive c.g. correction fraction.
6,1423 (3023)	PACTOFF	PACTOFF	"TVCINIT1" pg. DPTV-24	TVC DAP pitch trim. Loaded by N48.
6,1424 (3024)	YACTOFF	YACTOFF	"TVCINIT1" pg. DPTV-27	TVC DAP yaw trim. Loaded by N48.
*6,1425 (3025)	HBN10	C_{hbn10_0}	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1426 (3026)	HBN11/2	C_{hbn10_1}	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1427 (3027)	HBN12	C_{hbn10_2}	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1430 (3030)	HBD11/2	C_{hbn10_3}	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1431 (3031)	HBD12	C_{hbn10_4}	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.

Address (Octal)	Memory Notation	Document Notation	Reference	Comments
*6,1432 (3032)	HBN20	C_{hbn10_5}	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1433 (3033)	HBN21/2	C_{hbn10_6}	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1434 (3034)	HBN22	C_{hbn10_7}	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1435 (3035)	HBD21/2	C_{hbn10_8}	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1436 (3036)	HBD22	C_{hbn10_9}	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1437 (3037)	HBN30	$C_{hbn10_{10}}$	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1440 (3040)	HBN31/2	$C_{hbn10_{11}}$	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1441 (3041)	HBN32	$C_{hbn10_{12}}$	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1442 (3042)	HBD31/2	$C_{hbn10_{13}}$	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
*6,1443 (3043)	HBD32	$C_{hbn10_{14}}$	"TVCINIT1" pg. DPTV-17	LM-on high bandwidth filter.
6,1465 (3065)	DAPDATR1	DAPDATR1	"S41.2" pg. DPIR-12	DAP control word. Loaded by R1 of N46.
6,1466 (3066)	DAPDATR2	DAPDATR2	"S41.2" pg. DPIR-13	DAP control word. Loaded by R2 of N46.
6,1472 (3072)	LEMMASS	LEMMASS	"FIXCW" pg. DPIR-21	LM mass. Loaded by R2 of N47.
6,1473 (3073)	CSMMASS	CSMMASS	"FIXCW" pg. DPIR-11	CSM mass. Loaded by R1 of N47, and updated (double precision) in "S40.8".
6,1661 (3261)	POLYNUM	C_{ponum}	"ATERJOB" pg. BOOS-11	Booster pitch polynomial degree information.
6,1662-77 (3262-77)	POLYNUM +1	$C_{pyc,1}$	"ATERJOB" pg. BOOS-11	Booster pitch polynomial (stored double precision, with lowest-powered term in 3262-3, etc.).
6,1700-1 (3300-1)	SATRLRT	SATRLRT	"ATERJOB" pg. BOOS-13	Booster roll rate (sign can be changed in "P11").

Address (Octal)	Memory Notation	Document Notation	Reference	Comments
6,1702 (3302)	RPSTART	$C_{rpstart}$	"ATERJOB" pg. BOOS-11	Time from CMC-deduced liftoff when roll and pitch drive of error needles (with e.g. pitch polynomial) should start.
6,1703 (3303)	POLYSTOP	$C_{polystop}$	"ATERJOB" pg. BOOS-10	Value of complement of (stop time minus $C_{rpstart}$), where stop time is when performance of pitch polynomial evaluation should halt (quantity a negative number in memory).
*6,1710 (3310)	SATRATE	C_{satrt_0}	"SATSTICK" pg. BOOS-11	Number of pulses (B12) loaded into error counter for no RHC deflection in Saturn DAP (e.g. 0).
*6,1711 (3311)	SATRATE +1	C_{satrt_1}	"SATSTICK" pg. BOOS-11	Number of pulses loaded into error counter for positive RHC deflection in Saturn DAP.
*6,1712 (3312)	SATRATE +2	C_{satrt_2}	"SATSTICK" pg. BOOS-11	Number of pulses loaded into error counter (with sign) for negative RHC deflection in Saturn DAP.
*6,1713 (3313)	SATRATE +3	C_{satrt_3}	"SATSTICK" pg. BOOS-11	Number of pulses loaded into error counter for simultaneous positive and negative RHC input in Saturn DAP (e.g. 0).
6,1720 (3320)	SATSCALE	$C_{satscale}$	"NOPOLYM" pg. BOOS-11	Gain factor for automatic Saturn steering in P11.
*6,1775-6 (3375-6)	HORISLP	$C_{horislp}$	"HORIZ" pg. MEAS-28	Slope of horizon altitude used in P23 (is subtracted from other terms).
7,1400-1 (3400-1)	LAT(SPL)	LATSPL	"DELRSPL" pg. DISP-27	Target latitude for R1 of N50 (R30 quantity in P00/P11). It is loaded by R1 of N61.
7,1402-3 (3402-3)	LNG(SPL)	LNGSPL	"DELRSPL" pg. DISP-27	Target longitude for R1 of N50 (R30 quantity in P00/P11). It is loaded by R2 of N61.

In addition to the quantities listed on the previous pages, the "LM" state vector should be satisfactory for use in the periodic P00 state vector integration, as well as for the integration done at the end of Average-G. If the LM vehicle is not of concern, the P00 integration can be bypassed, and the integration at the end of Average-G simplified, by setting bit 8(SURFFLAG) of FLAGWRD8 = 1 (can be set by V44E and reset by V45E). Otherwise, the LM state vector can be initialized by a P27 (V71) update, by V66E to move the CSM state vector into LM cells, or by the following erasable initialization:

Position vector into $\underline{R}_{rectlm}$ (1626-1633) and RC$\underline{V}_{lm}$ (1660-1665).

Velocity vector into $\underline{V}_{rectlm}$ (1634-1641) and VC$\underline{V}_{lm}$ (1666-1673).

Time tag into T_{etlm} (1642-1643).

Zeros into DELTA$\underline{V}_{lm}$ (1644-1651), NU$\underline{V}_{lm}$ (1652-1657), T_{clm} (1674-1675), and $XKEP_{lm}$ (1676-1677).

See Orbital Integration (and "MOVEPLEM") for these quantities.

In order to complete the initialization functions, R03 (V48E) should be done, with DAPDATR1 digit A = 1,2,5, or 6 (to allow the "MASSPROP" routine to be done, which initializes the MASS cell).

In the event that the memory must be initialized in flight, cell 1477 (GCOMPSW) should be checked for a non-negative value: it is unnecessary to set the cell as part of the prelaunch load, since the program sets it to 0 before entering P02 from P01.

See Uplink Processing for other memory addresses which may be of interest for initialization.

Attitude Maneuvers

```
R62DISP        Established by "CREWMANU" for a V49E

    TS = $0622_{vn}$

    Proceed to "GOFLASH": if terminate, proceed to "ENDEXT"
                          if proceed, proceed
                          otherwise, proceed to previous line

    Set bit 6(3AXISFLG) of FLAGWRD5 = 1

    Perform "R60CSM"

    Proceed to "ENDEXT"

V89CALL        Established by "V89PERF" for a V89E.  This is R63

    Perform "R02BOTH"

    Set bit 8(AZIMFLAG) of FLGWRD11 = 0

    UTPIT = 0

    UTYAW = 0

    TS = $0678_{vn}$

    Perform "GOFLASHR": if terminate, proceed to "ENDEXT"
                        if proceed, proceed to "V89RECL"
                        otherwise, proceed to previous line

    TS = $100_2$ and perform "BLANKET"        (R3BLNK)

    End of job

V89RECL

    R63TIME = $T_{now} + K_{dplmin}$

    Perform "R63"

    TS = $0618_{vn}$

    Proceed to "GOFLASH": if terminate, proceed to "ENDEXT"
                          if proceed, proceed
                          otherwise, proceed to "V89RECL"

    Set bit 6(3AXISFLG) of FLAGWRD5 = 0

    Perform "R60CSM"

    Proceed to "ENDEXT"
```

```
R60CSM     (Entered from "P40SXTY" (P40,P41), "R60CALL"(P23), "R61CSM",
                                         "R62DISP", and "V89RECL")
    TEMPR60 = Return address

    If bit 6(3AXISFLG) of FLAGWRD5 = 0:

        Perform "VECPOINT"

        THETAD = TS

    Proceed to "TOBALL"

TOBALL

    TS = $0618_{vn}$

    Perform "GOPERF2R": if terminate, proceed to "R61TEST"
                        if proceed, skip next 2 lines
                        otherwise, proceed to "ENDMANU1"

    Perform "CHKLINUS"

    End of job

    If bit 6(3AXISFLG) of FLAGWRD5 = 0:

        Perform "VECPOINT"

        THETAD = TS

    TS = channel 31

    If bits 14-13 of C31FLWRD $\neq 00_2$:

        TS = C31FLWRD

    If bits 15-13 (G&N Autopilot Control complement, Free Mode complement,
        and Hold Mode complement) of TS $\neq 011_2$:    ($11_2$ means Automatic)

        Proceed to "TOBALL"

    TS = $0618_{vn}$

    Perform "GODSPR"

    Perform "CHKLINUS"

    Proceed to "KALCMAN3"     ("ENDMANU" establishes "ENDMANUV" for exit)

ENDMANU1

    Proceed to address specified by TEMPR60
```

```
ENDMANUV          (Established by "ENDMANU")

     If bit 7(AUTOSEQ) of FLGWRD10 = 1:

          If bit 5(TRACKFLG) of FLAGWRD1 = 1:

               Proceed to "ENDMANU1"

     Proceed to "TOBALL"

CHKLINUS

     If bit 12(PDSPFLAG) of FLAGWRD4 = 0:

          Return

     TBASE1 = (Return address -3 for last display, i.e. the TS setting)

     Set restart group 1 to phase 7 (1.7, causing "RELINUS" to be established
          with priority $10_8$ if a restart)

     TS = $100_8$ and perform "BLANKET"        (PRIOREQ)

     Return

RELINUS          (Entered for restart group 1.7, due to "CHKLINUS")

     If bit 5(TRACKFLG) of FLAGWRD1 = 0:

          Proceed to "FIXDB"

     Set bit 12(PDSPFLAG) of FLAGWRD4 = 1

     If bit 9(UTFLAG) of FLAGWRD8 = 0:

          Set bit 10(LMTRG) of FLAGWRD1 = 1

          OPTIND = +0

          Set bit 14(R21MARK) of FLAGWRD2 = 1

     Change priority of present job to $14_8$

     Proceed to address specified by TBASE1  (regenerate last display
          due to setting of this cell in "CHKLINUS")

R61TEST

     If MODREG = 0, proceed to "ENDMANU1"

     If bit 12(PDSPFLAG) of FLAGWRD4 = 0, proceed to "GOTOPOOH"

     Proceed to "TRACKTRM"
```

VECPOINT

```
$T\underline{S}_3 = CD\underline{U}$                    (In push-down address 25D)
$T\underline{S} = T\underline{S}_3$
X2 = "MIS"
Perform "CDUTODCM"
$T\underline{S} = \text{unit}(POINTVS\underline{M}\ [MIS])$              (In push-down address 28D)
$T\underline{S}_1 = T\underline{S} * SCAXI\underline{S}$
If $|T\underline{S}_1| - K_{minmag} < 0$  (or if unit$T\underline{S}_1$ overflows):
    If $T\underline{S} \cdot SCAXI\underline{S} \geq 0$:
        $T\underline{S} = T\underline{S}_3$
        Return
    $T\underline{S} = \text{unit}(MI\underline{S}_3 * \text{unit}\underline{X}) * SCAXI\underline{S}$
    If $|T\underline{S}| - K_{minmag} < 0$  (or if unit$T\underline{S}$ overflows):
        $T\underline{S} = \text{unit}\underline{X}$
    $CO\underline{F} = \text{unit}T\underline{S}$        (unit operation omitted if $T\underline{S}$ set to unit$\underline{X}$)
    $TS = \frac{1}{2}$              (i.e. 180°)
    Proceed to "COMPMATX"
$CO\underline{F} = -\ \text{unit}T\underline{S}_1$
$TS = \cos^{-1}(SCAXI\underline{S} \cdot T\underline{S})$
Proceed to "COMPMATX"
```

NOTE: See Apollo Software Information Memo 5-68-9, "VECPOINT Automatic Maneuvers for Program Sundisk," dated 30 May 1968, by S. A. Gorman for discussion of large attitude maneuver equation design. The document is published by MSC's Flight Software Branch.

```
COMPMATX
    Perform "DELCOMP"
    X1 = "MIS"
    X2 = "DEL"
    Perform "MXM3"    (leaves $[TS] = [MFS]$)
    If $|TS_3| - K_{singim} < 0$, proceed to "FINDGIMB" ($TS_3$ is 4th element of
                                                            matrix $[TS]$,
    If $|SCAXIS_x| - K_{sinvec1} \geqslant 0$, proceed to "FINDGIMB"    sine middle
                                                            gimbal angle)
    $[MIS] = [TS]$
    $T\underline{S} = (MI\underline{S}_3 \text{ sgn } MIS_3) * SCAXI\underline{S}$
    $CO\underline{F} = - SCAXI\underline{S} \text{ sgn } TS_x$
    $TS = K_{vecang1}$
    If $|SCAXIS_x| - K_{sinvec2} < 0$:
        $TS = K_{vecang2}$
    Perform "DELCOMP"
    X1 = "MIS"
    X2 = "DEL"
    Perform "MXM3"
    Proceed to "FINDGIMB"
FINDGIMB
    X1 = "TS"
    Perform "DCMTOCDU"
    $T\underline{S} = T\underline{S}$, converted to twos complement single precision, B-1 revolutions
    Return (to routine calling "VECPOINT")
```

KALCMAN3 (Entered from "STKTEST" and "TOBALL")

```
    $BCD\underline{U} = CD\underline{U}$
    X2 = "MIS"
    $T\underline{S} = BCD\underline{U}$
    Perform "CDUTODCM"
    X2 = "MFS"
    $T\underline{S} = THETA\underline{D}$
    Perform "CDUTODCM"
    X1 = "MIS"
    Perform "TRANSPOS"
    $[TMIS] = [TS]$
    X1 = "TMIS"
    X2 = "MFS"
    Perform "MXM3"
    $[MFI] = [TS]$
    X1 = "MFI"
    Perform "TRANSPOS"
    $[TMFI] = [TS]$
    $COFSKE\underline{W} = (TMFI_5 - MFI_5, MFI_2 - TMFI_2, TMFI_1 - MFI_1)$
    $CAM = \frac{1}{2} (MFI_0 + MFI_8 - 1 + MFI_4)$
    $AM = \cos^{-1} CAM$
    If $AM - K_{mina} < 0$:
        Inhibit interrupts
        HOLDFLAG = -1
        $CDUX\underline{D} = THETA\underline{D}$
        Proceed to "NOGO"
```

```
If $AM - K_{maxa} < 0$:                                  (Tag here "CHECKMAX")
    $CO\underline{F} = unitCOFSKE\underline{W}$
    Proceed to "LOCSKIRT"
$[MFISYM] = \frac{1}{2}([MFI] + [TMFI])$
TS = 1 - CAM, with magnitude limited $< 2$
$CO\underline{F} = unit\left(\sqrt{\frac{MFISYM_0 - CAM}{\frac{1}{2} TS}}, \sqrt{\frac{MFISYM_4 - CAM}{\frac{1}{2} TS}}, \sqrt{\frac{MFISYM_8 - CAM}{\frac{1}{2} TS}}\right)$
If $COF_x \geq COF_y$ and $COF_x \geq COF_z$:   ($COF_x$ largest component)
    $CO\underline{F} = CO\underline{F}$ sgn $COFSKEW_x$          (Tag here "METHOD1")
    $COF_y = COF_y$ sgn $MFISYM_1$
    $COF_z = COF_z$ sgn $MFISYM_2$
    Proceed to "LOCSKIRT"
If $COF_y > COF_x$ and $COF_y \geq COF_z$:   ($COF_y$ largest component)
    $CO\underline{F} = CO\underline{F}$ sgn $COFSKEW_y$          (Tag here "METHOD2")
    $COF_x = COF_x$ sgn $MFISYM_1$
    $COF_z = COF_z$ sgn $MFISYM_5$
    Proceed to "LOCSKIRT"
If $COF_z > COF_x$ and $COF_z > COF_y$ (as it will):   ($COF_z$ largest)
    $CO\underline{F} = CO\underline{F}$ sgn $COFSKEW_z$          (Tag here "METHOD3")
    $COF_x = COF_x$ sgn $MFISYM_2$
    $COF_y = COF_y$ sgn $MFISYM_5$
    Proceed to "LOCSKIRT"
```

TRANSPOS

$$[TS] = \begin{bmatrix} {}^{E}X1_0 & {}^{E}X1_3 & {}^{E}X1_6 \\ {}^{E}X1_1 & {}^{E}X1_4 & {}^{E}X1_7 \\ {}^{E}X1_2 & {}^{E}X1_5 & {}^{E}X1_8 \end{bmatrix}$$

Return

CDUTODCM

Convert $\underline{TS}$ to double precision ones complement angles TS_x, TS_y, and TS_z. Matrix elements formed starting at address given by the contents of X2.

$E_{X2_0} = \cos TS_y \cos TS_z$

$E_{X2_1} = \sin TS_x \sin TS_y - \cos TS_x \cos TS_y \sin TS_z$

$E_{X2_2} = \cos TS_x \sin TS_y + \sin TS_x \cos TS_y \sin TS_z$

$E_{X2_3} = \sin TS_z$

$E_{X2_4} = \cos TS_x \cos TS_z$

$E_{X2_5} = - \sin TS_x \cos TS_z$

$E_{X2_6} = - \sin TS_y \cos TS_z$

$E_{X2_7} = \sin TS_x \cos TS_y + \cos TS_x \sin TS_y \sin TS_z$

$E_{X2_8} = \cos TS_x \cos TS_y - \sin TS_x \sin TS_y \sin TS_z$

Return

DELCOMP

Entered with angle TS. Matrix elements stored starting at DEL. All matrix element have magnitudes limited < 1.

$TS_1 = 1 - \cos TS$ (magnitude limited < 2)

$DEL_0 = COF_x^2 \, TS_1 + \cos TS$

$DEL_4 = COF_y^2 \, TS_1 + \cos TS$

$DEL_8 = COF_z^2 \, TS_1 + \cos TS$

$DEL_3 = COF_x \, COF_y \, TS_1 + COF_z \sin TS$

$DEL_1 = COF_x \, COF_y \, TS_1 - COF_z \sin TS$

$DEL_2 = COF_x \, COF_z \, TS_1 + COF_y \sin TS$

$DEL_6 = COF_x \, COF_z \, TS_1 - COF_y \sin TS$

$DEL_7 = COF_y \, COF_z \, TS_1 + COF_x \sin TS$

$DEL_5 = COF_y \, COF_z \, TS_1 - COF_x \sin TS$

Return

MXM3

$$T\underline{S}_1 = \left[E_{X1}\right] \left(E_{X2_0} \;,\; E_{X2_3} \;,\; E_{X2_6}\right)$$

$$T\underline{S}_2 = \left[E_{X1}\right] \left(E_{X2_1} \;,\; E_{X2_4} \;,\; E_{X2_7}\right)$$

$$T\underline{S}_3 = \left[E_{X1}\right] \left(E_{X2_2} \;,\; E_{X2_5} \;,\; E_{X2_8}\right)$$

$$\left[TS\right] = \begin{bmatrix} TS_{1x} & TS_{2x} & TS_{3x} \\ TS_{1y} & TS_{2y} & TS_{3y} \\ TS_{1z} & TS_{2z} & TS_{3z} \end{bmatrix}$$

Return

DCMTOCDU

Entered with matrix starting address in X1. Cf. "CDUTODCM" for matrix elements.

$TS_z = \sin^{-1} E_{X1_3}$

$TS = \cos TS_z$, with magnitude limited < 1

$TS_y = \sin^{-1} (- E_{X1_6} / TS)$

If $E_{X1_0} < 0$:

$TS_y = \frac{1}{2} \operatorname{sgn} TS_y - TS_y$ (the $\frac{1}{2}$ corresponds to 180°)

$TS_x = \sin^{-1} (- E_{X1_5} / TS)$

If $E_{X1_4} < 0$:

$TS_x = \frac{1}{2} \operatorname{sgn} TS_x - TS_x$

$T\underline{S} = (TS_x, TS_y, TS_z)$ (double precision, B0 revolutions)

Return

LOCSKIRT (Tag also "WCALC")

$TS = K_{arate_{RATEINDX}}$

Perform "DELCOMP"

$BRAT\underline{E} = \left[QUADROT\right] (K_{arate_{RATEINDX}} \quad CO\underline{F})$

```
    $T_{tm} = K_{angtm}\ \mathrm{AM}\ /\ K_{arate_{RATEINDX}}$
    If bit 4(CYC61FLG) of FLAGWRD0 = 1:    ("KALCMAN3" entered from
                                             "STKTEST")
        Proceed to "RCYCLR61"
    $\mathrm{BIASTEM}\underline{\mathrm{P}} = K_{biasc}\ \mathrm{BRAT}\underline{\mathrm{E}}$
    Set bit 2(CALCMAN2) of FLAGWRD2 = 1    (means start of KALCMANU
                                             steering generation)
    Proceed to "NEWANGL"
NOGO
    Inhibit interrupts
    Perform "STOPRATE"
    If bit 2(R67FLAG) of FLAGWRD8 = 0:
        Call "ENDMANU" in 0.02 second
        End of job
    Call "R67" in 1 second
    Set restart group 1 to phase 11 (1.11, causing "PIKUP20" to be
        established with priority $10_8$ if a restart)
    End of job
NEWDELHI        (Established by "UPDTCALL")
    If bit 2(R67FLAG) of FLAGWRD8 = 1:
        If bit 5(TRACKFLG) of FLAGWRD1 = 0:
            Proceed to "FIXDB"
    If  HOLDFLAG $\ngtr$ 0:
        Inhibit interrupts
        $\mathrm{CDUX}\underline{\mathrm{D}} = \mathrm{CD}\underline{\mathrm{U}}$
        Proceed to "NOGO"
    Proceed to "NEWANGL"
```

```
NEWANGL          (Entered from "LOCSKIRT", "NEWDELHI", and "R67RSTRT")
    X1 = "MIS"
    X2 = "DEL"
    Perform "MXM3"
    $[MIS] = [TS]$
    X1 = "MIS"
    Perform "DCMTOCDU"
    NCDU = TS, converted to twos complement single precision, B-1 rev.
    If bit 2(CALCMAN2) of FLAGWRD2 = 1:
        Set bit 2(CALCMAN2) of FLAGWRD2 = 0     (Tag here "MANUSTAT")
        If bit 2(R67FLAG) of FLAGWRD8 = 1:
            $T_{tm} = (2^{28} - 1)$ centi-seconds      $(37777_8\ 37777_8)$
        If bit 2(R67FLAG) of FLAGWRD8 = 0:
            $T_{tm} = T_{tm} + T_{now} - 100$ centi-seconds
        Inhibit interrupts   (Tag here "TMDON")
        HOLDFLAG = -1
        If RATEINDX $\geq$ 6:          (Note that this done for R67 also,
                                        regardless of the R67-specified rate)
            Set bit 15 of RCSFLAGS = 1   (means high rate)
        WBODY = BRATE
        BIAS = BIASTEMP  (least significant half only)
        NEXTIME = TIME1 + 100, modulo $2^{14}$ cs
    DELCDU = $QUADROT_{0_{sp}}$ (NCDU - BCDU)     (ones complement difference formed)
    CDUXD = BCDU     (most significant half of CDUXD loaded; least
                      significant half left alone)
    BCDU = NCDU
    Release interrupts
    TS = $T_{tm} - T_{now}$
```

```
    If TS $> 0$:
        Call "UPDTCALL" in (NEXTIME - TIME1) centi-seconds (if
            difference is negative, 16384 is added, i.e. $2^{14}$)
        NEXTIME = NEXTIME + 100, modulo $2^{14}$ cs
        End of job
    TS = TS + 101, limited $\ngtr$ 1 centi-second and $\leq$ 100 centi-seconds
    Call "MANUSTOP" in TS centi-seconds
    End of job
UPDTCALL
    If bit 2(R67FLAG) of FLAGWRD8 = 1:
        If R61CNTR = 0:
            Establish "R67RSTRT"  (priority $26_8$)
            End of task
        R61CNTR = R61CNTR - 1
    Establish "NEWDELHI" (priority $26_8$)
    End of task
MANUSTOP
    DELCDUY = 0
    DELCDUZ = 0
    $WBODY_1 = 0$
    $WBODY_2 = 0$
    $BIAS_1 = 0$
    $BIAS_2 = 0$
    CDUYD = $THETAD_y$        (least significant
    CDUZD = $THETAD_z$              half of
    CDUXD = $THETAD_x$         CDUXD left alone)
    Perform "STOPRATE"
    Proceed to "ENDMANU"
```

```
ENDMANU
    Establish "ENDMANUV" (priority $14_8$)
    End of task
STOPRATE
    DELCDUX = 0
    $WBODY_0$ = 0
    $BIAS_0$ = 0
    Set bit 15 of RCSFLAGS = 0      (means no longer have high rate)
    DELCDUY = 0
    DELCDUZ = 0
    $WBODY_1$ = 0
    $WBODY_2$ = 0
    $BIAS_1$ = 0
    $BIAS_2$ = 0
    Return
FIXDB
    Inhibit interrupts
    If bit 12(MAXDBFLG) of FLAGWRD9 = 1:
        Perform "SETMAXDB"
    If bit 12(MAXDBFLG) of FLAGWRD9 = 0:
        Perform "SETMINDB"
    Set bit 2(R67FLAG) of FLAGWRD8 = 0
    Release interrupts
    Set restart group 1 to phase 11 (1.11, causing "PIKUP20" to be
        established with priority $10_8$ if a restart)
    Inhibit interrupts
    Perform "STOPRATE"
    End of job  (where interrupts released)
```

```
R61CSM          (Entered from "PIKUP20", "R52H", and "R66CSM")
    GENRET = Return address
    If  $R61CNTR > 0$:
        R61CNTR = R61CNTR - 1, limited $\geq +0$
        Proceed to address specified by GENRET
    If  $R61CNTR < 0$:
        Proceed to address specified by GENRET
    Inhibit interrupts
    $ADB = K_{mindb}$
    $TS = |DBPTC|$        (absolute value converts input of e.g. $350^{\circ}$ to $\sim 10^{\circ}$)
    If  $TS \neq 0$:
        ADB = TS
    Release interrupts
    Perform "CRS61.1"
    Set bit 15(V50N18FL) of FLAGWRD3 = 0
    If  TS = 0:       (no R60 performance)
        R61CNTR = 3
        Proceed to address specified by GENRET
    Set bit 6(3AXISFLG) of FLAGWRD5 = 0       (Tag here "R61C1")
    If bit 8(AZIMFLAG) of FLGWRD11 = 1:
        Set bit 6(3AXISFLG) of FLAGWRD5 = 1
    R61CNTR = -1
    Set bit 12(PDSPFLAG) of FLAGWRD4 = 1
    Perform "R60CSM"
    Set bit 12(PDSPFLAG) of FLAGWRD4 = 0
    TS = 0
    Perform "PRIODSP"        (clears DSKY of V06N18 if in Minkey mode)
```

```
    Set restart group 1 to phase 11 (1.11, causing "PIKUP20" to be
         established with priority $10_8$ if a restart)

    R61CNTR = 0

    Proceed to address specified by GENRET

CRS61.1          (Entered only from "R61CSM")

    Q611 = Return address

    R63TIME = $T_{now}$

    Perform "R63"   (enter here from "RCYCLR61" to refine maneuver by R60)

    Set bit 9(RANGFLAG) of FLGWRD10 = 0      (note R22 could interrupt here,bit 0)

    If  $(R63TIME - K_{328nm}) < 0$:       (R63TIME has LOS value; constant
                                            is 327.67 nmi)
         Set bit 9(RANGFLAG) of FLGWRD10 = 1   (Note could be improper if
                                                refined maneuver not require
    If MODREG $\neq$ 27:                        R60)

         Set bit 3(Uplink Activity) of channel 11 = 0

    If  HOLDFLAG $> 0$:         (includes e.g. in Free or SCS control)

         Set bit 4(CYC61FLG) of FLAGWRD0 = 0

         TS = 0

         Proceed to address specified by Q611

    If bit 8(AZIMFLAG) of FLGWRD11 = 1:

         $T\underline{S} = CD\underline{U} - THETA\underline{D}$      (ones complement difference formed)

         If any( $|TS_i| - K_{degree10}) \not> 0$:       (i = x,y,z)

              Proceed to "STKTEST"

    If bit 8(AZIMFLAG) of FLGWRD11 = 0:

         $CDUSPO\underline{T} = (THETADX_y, THETADX_z, THETADX_x)$    (Tag here "NOAZFLG1")

         $T\underline{S} = SCAXI\underline{S}$

         Perform "TRG*NBSM"

         If  $(T\underline{S} \cdot POINTVS\underline{M}) - K_{costen} < 0$:      (Note this is check using
                                                DAP deadband center, not the
              Proceed to "STKTEST"               present attitude)
```

```
(If bit 8(AZIMFLAG) of FLGWRD11 = 0):
    $T\underline{S}_1 = K_{deltyme}\ (T\underline{S} * POINTVS\underline{M})$
    If$(|T\underline{S}_1| - K_{maxrate}) \geqslant 0$:
        $T\underline{S}_1 = K_{maxrate}\ unitT\underline{S}_1$
    $DTHETAS\underline{M} = K_{radtorev}\ T\underline{S}_1$
TS = channel 31              (Tag here "AUTOCK")
If bits 14-13 of C31FLWRD $\neq 00_2$:
    TS = C31FLWRD
If bits 15-13 (G&N Autopilot Control complement, Free Mode complement,
    and Hold Mode complement) of TS $\neq 011_2$: ($11_2$ means Automatic)
    Set bit 4(CYC61FLG) of FLAGWRD0 = 0
    TS = 0
    Proceed to address specified by Q611
If bit 14(STIKFLAG) of FLAGWRD1 = 1:          (Tag here "DAPCK")
    Set bit 4(CYC61FLG) of FLAGWRD0 = 0
    TS = 0
    Proceed to address specified by Q611
If bit 9(UTFLAG) of FLAGWRD8 = 1:        (Tag here "STEP3CK")
    If  R63TIME = 0:      (set 0 in "R63" computations, i.e. "UTAREA1",
                                    for body code $\leqq 45_8$)
        $T\underline{S}_3 = 0$         (14D)
        $T\underline{S} = 0$          (MPAC)
        Proceed to "CRS61.2A"
$T\underline{S} = DCD\underline{U} - SAVEVE\underline{L}$          (Tag here "CRS61.2"; $DCD\underline{U}$ has LM velocity)
$T\underline{S}_1 = K_{rvcsds}\ (-\ unitT\underline{S} * SAVEPO\underline{S})$        (B6)
$T\underline{S}_2 = [REFSMMAT]\ (|T\underline{S}|/R63TIME)\ T\underline{S}_1$   (in 20D; R63TIME has |LOS| value)
If bit 8(AZIMFLAG) of FLGWRD11 = 0:
    $T\underline{S}_2 = T\underline{S}_2 + DTHETAS\underline{M}$            (B0)
```

```
    $DTHETAS\underline{M} = K_{tnth}\ T\underline{S}_2$
    Perform "CDUTRIG"
    Perform "SMCDURES"
    $T\underline{S}_3$ = DCD$\underline{U}$, rescaled to B-1 revolutions (in 14D)
    Perform "CDUTRIG"
    $T\underline{S} = T\underline{S}_2$
    Perform "*SMNB*"
    $T\underline{S} = K_{pt8}\ [MBDYTCTL]\ T\underline{S}$        (B-3 rev/deci-second, in MPAC)
    Proceed to "CRS61.2A"
CRS61.2A
    Inhibit interrupts
    If bit 8(AZIMFLAG) of FLGWRD11 = 1:
        $T\underline{S}_3 = T\underline{S}_3 + K_{1d200}\ (THETA\underline{D} - THETAD\underline{X})$     (ones comp. difference
                                                                formed)
    $CDUX\underline{D}_{sp} = THETAD\underline{X}$
    $WBOD\underline{Y} = T\underline{S}$          (from MPAC)
    $DELCD\underline{U} = T\underline{S}_3$         (from 14D)
    HOLDFLAG = -1
    Release interrupts
    Set bit 4(CYC61FLG) of FLAGWRD0 = 0
    TS = 0
    Proceed to address specified by Q611
STKTEST     (Entered from "CRS61.1" if R60 maneuver appropriate)
    TS = bit 4(CYC61FLG) of FLAGWRD0
    Complement bit 4(CYC61FLG) of FLAGWRD0
    If TS = 0:
        Proceed to "KALCMAN3" (exits to "RCYCLR61" since CYC61FLG set,
                               using check in "LOCSKIRT")
```

```
    If bit 15(V50N18FL) of FLAGWRD3 = 1:
        TS = 1
        Proceed to address specified by Q611
    Set bit 3(Uplink Activity) of channel 11 = 1
    Set bit 4(CYC61FLG) of FLAGWRD0 = 0
    TS = 0
    Proceed to address specified by Q611
RCYCLR61        (Entered from "LOCSKIRT" if CYC61FLG set)
    R63TIME = $T_{now} + K_{20p48sec} + T_{tm}$
    Proceed to third line of "CRS61.1"
R63        (Entered from "CRS61.1" and "V89RECL")
    Q6111 = Return address
    $T_{decl}$ = R63TIME
    Perform "CSMCONIC"
    $\text{SAVEPO}\underline{\text{S}} = \underline{R}_{att}$
    $\text{SAVEVE}\underline{\text{L}} = \underline{V}_{att}$
    $\text{POINTVS}\underline{\text{M}} = (\text{SAVEVE}\underline{\text{L}} * \text{SAVEPO}\underline{\text{S}})$          (B36, for use in "UTOPT45")
    If bit 9(UTFLAG) of FLAGWRD8 = 1:
        Proceed to "UTAREA1"
    $T_{decl}$ = R63TIME
    Perform "LEMCONIC"
    $\text{DCD}\underline{\text{U}} = \underline{V}_{att}$      (here is LM velocity)          (B7 m/cs)
    $\text{T}\underline{\text{S}} = \underline{R}_{att} - \text{SAVEPO}\underline{\text{S}}$
    $\text{SAVEPO}\underline{\text{S}} = \text{unitT}\underline{\text{S}}$
    $\text{R63TIME} = |\text{T}\underline{\text{S}}|$      (here is LOS magnitude)      (B29 meters)
    $\text{T}\underline{\text{S}} = \text{SAVEPO}\underline{\text{S}}$
    Proceed to "R63COM1"
```

```
R63COM1
    If bit 8(AZIMFLAG) of FLGWRD11 = 1:
        Proceed to "UTOPT45"      (P20 option 4/5, 3-axis solution)
    $POINTVS\underline{M} = [REFSMMAT]\ T\underline{S}$
    $SCAXI\underline{S}$ = (cos UTYAW cos UTPIT, sin UTYAW cos UTPIT, - sin UTPIT)
    Perform "VECPOINT"
    $THETA\underline{D} = T\underline{S}$
    Proceed to address specified by Q6111
UTAREA1       (Entered from "R63" if UTFLAG set, i.e. option 1/5 of P20)
    TS = R63TIME
    Perform "LSPOS"
    If bit 12(CMOONFLG) of FLAGWRD8 = 1:
        $T\underline{S}_2 = -unit(T\underline{S} + SAVEPO\underline{S})$    (in 8D, earth vector)
        $SCAXI\underline{S} = -\ unitSAVEPO\underline{S}$    (moon vector)
        $T\underline{S}_1 = unit(T\underline{S}_1 - T\underline{S})$      (in 2D, sun vector)
    If bit 12(CMOONFLG) of FLAGWRD8 = 0:
        $SCAXI\underline{S} = unit(T\underline{S} - SAVEPO\underline{S})$     (moon vector)
        $T\underline{S}_2 = -unitSAVEPO\underline{S}$       (in 8D, earth vector)
        $T\underline{S}_1 = unitT\underline{S}_1$          (in 2D, sun vector)
    $VELd\underline{C} = K_{1dc}\ SAVEVE\underline{L} + (T\underline{S}_1 * \underline{K}_{eclipol})$     (Tag here "SAVESTAR")
    $DCD\underline{U} = 0$     (velocity cell)
    $R63TIME = |SAVEPO\underline{S}|$                    (B29 meters)
    $SAVEPO\underline{S} = -\ unitSAVEPO\underline{S}$
    If  UTSTARNO = $\pm$ 0:
        $SAVEVE\underline{L} = 0$
        R63TIME = 0     (zero value used in "CRS61.1" to bypass LOS rate)
        $T\underline{S} = unit\left(unit(K_{1Dsqrt3}\ PLANVCU\underline{T}) + VELd\underline{C}\right)$
        Proceed to "R63COM1"
```

```
If $(UTSTARNO - 46_8) < 0$:          (catalog star)
    $SAVEVE\underline{L} = 0$
    $R63TIME = 0$
    $TS = 6\ UTSTARNO$
    $T\underline{S} = unit(\underline{K}_{cat_{TS}} + VELd\underline{C})$
    Proceed to "R63COM1"
If $UTSTARNO = 46_8$:    (sun)
    $T\underline{S} = unit(T\underline{S}_1 + VELd\underline{C})$
    Proceed to "R63COM1"
If $UTSTARNO = 47_8$:     (earth)
    $T\underline{S} = unit(T\underline{S}_2 + VELd\underline{C})$
    Proceed to "R63COM1"
If $UTSTARNO \geq 50_8$:      (moon)
    $T\underline{S} = unit(SCAXI\underline{S} + VELd\underline{C})$
Proceed to "R63COM1"
UTOPT45        (Entered from "R63COM1" if AZIMFLAG = 1, i.e. option 4/5 of P20)
    $UTS\underline{B} = T\underline{S}$          (line-of-sight information)
    $UTSB\underline{P} = (\cos UTYAW \cos UTPIT,\ \sin UTYAW \cos UTPIT,\ -\sin UTPIT)$
    $UTSA\underline{P} = unit(UTSB\underline{P} * unit\underline{Y})$       (Note this same as unit of
                                                    (sin P, 0, cos Y cos P), so error
                                                   if P = 0 and Y = $90^o$)
    $POINTVS\underline{M} = unit(UTS\underline{B} * POINTVS\underline{M})$      (POINTVS$\underline{M}$ has $\underline{v} * \underline{r}$ from "R63")
    $UTS\underline{A} = POINTVS\underline{M} \cos AZIMANGL + unit(UTS\underline{B} * POINTVS\underline{M}) \sin AZIMANGL$
    $UTUY\underline{P} = unit(UTSA\underline{P} * UTSB\underline{P})$          (cf. ST$\underline{V}$' in "AXISGEN")
    $UTUZ\underline{P} = UTSA\underline{P} * UTUY\underline{P}$                (cf. ST$\underline{W}$' in "AXISGEN")
    $UTU\underline{Y} = unit(UTS\underline{A} * UTS\underline{B})$            (cf. ST$\underline{V}$ in "AXISGEN")
    $UTU\underline{Z} = UTS\underline{A} * UTU\underline{Y}$                  (cf. ST$\underline{W}$ in "AXISGEN")
```

```
    $UT\underline{X} = UTSAP_x \; UTS\underline{A} + UTUYP_x \; UTU\underline{Y} + UTUZP_x \; UTU\underline{Z}$
    $UT\underline{Y} = UTSAP_y \; UTS\underline{A} + UTUYP_y \; UTU\underline{Y} + UTUZP_y \; UTU\underline{Z}$
    $UT\underline{Z} = unit(UTSAP_z \; UTS\underline{A} + UTUYP_z \; UTU\underline{Y} + UTUZP_z \; UTU\underline{Z})$
    $UT\underline{Y} = unitUT\underline{Y}$
    $UT\underline{X} = unitUT\underline{X}$
    $UTAM\underline{G} = unit(UT\underline{X} * REFSMMA\underline{T}_3)$    (cf. $T\underline{S}$ in "CALCGA")
    $COSTH = UTAM\underline{G} \cdot UT\underline{Z}$
    $SINTH = UTAM\underline{G} \cdot UT\underline{Y}$
    Perform "ARCTRIG"
    UTOGA = THETA
    $COSTH = (UTAM\underline{G} * UT\underline{X}) \cdot REFSMMA\underline{T}_3$
    $SINTH = REFSMMA\underline{T}_3 \cdot UT\underline{X}$
    Perform "ARCTRIG"
    UTMGA = THETA
    If $|THETA| - K_{p166} \geq 0$:
        Perform "ALARM" (pattern $401_8$)
    $COSTH = REFSMMA\underline{T}_6 \cdot UTAM\underline{G}$        (Tag here "UTCGA1")
    $SINTH = REFSMMA\underline{T}_0 \cdot UTAM\underline{G}$
    Perform "ARCTRIG"
    UTIGA = THETA
    $THETA\underline{D}$ = (UTOGA, UTIGA, UTMGA), converted to twos complement B-1 rev, sp.
    Proceed to address specified by Q6111
CALLR6X      (Entered from "PIKUP20" if UTFLAG = 1, i.e. options 1, 2, 5 of P20)
    Set bit 14(R21MARK) of FLAGWRD2 = 0
    If OPTNTYPE = 0:      (original option selection 1 or 5)
        Proceed to "R66CSM"
    $TS = R67TIME - T_{now}$, with sign agreement forced  (Tag "TYPE2", option 2)
```

```
    If  $TS_{sp} > 0$:      (TS B28 cs, so this means 163.84 sec or more)
        Delay 163.83 seconds (by putting job to sleep via "DELAYJOB")
        Proceed to 4th line of "CALLR6X"
    If  $TS \leq 0$:
        $TS = 1$
    Call "R67" in TS centi-seconds
    End of job
R66CSM        (Entered from "CALLR6X")
    If bit 5(TRACKFLG) of FLAGWRD1 = 0:
        Proceed to "FIXDB"
    Perform "R61CSM"
    Delay 0.5 second (by putting job to sleep via "DELAYJOB")
    Proceed to "R66CSM"
R67           (Called by "CALLR6X" and "NOGO")
    Establish "R67START"  (priority $26_8$)
    Set bit 2(R67FLAG) of FLAGWRD8 = 1
    End of task
R67START
    If bit 14(STIKFLAG) of FLAGWRD1 = 1:      (note that HOLDFLAG not checked on startup, nor DAP auto/hold mode)
        Inhibit interrupts
        $CDUX\underline{D} = CD\underline{U}$
        Proceed to "NOGO"
    $BCD\underline{U} = CD\underline{U}$
    $CO\underline{F} = (\cos UTYAW \ \cos UTPIT, \ \sin UTYAW \cos UTPIT, \ - \sin UTPIT)$
    Inhibit interrupts
    $ADB = K_{mindb}$
```

```
        TS = |DBPTC|

        If  TS  ≠ 0:

             ADB = TS

        Release interrupts

        $\text{TS} = K_{\text{tufits}}\ \text{RATEPTC}$

        Perform "DELCOMP"

        $\text{BRAT}\underline{\text{E}} = \left[\text{QUADROT}\right]\ K_{\text{tufits}}\ \ \text{RATEPTC}\ \ \text{CO}\underline{\text{F}}$

        $\text{BIASTEM}\underline{\text{P}} = 0$

        Set restart group 1 to phase 13 (1.13, causing "R67RSTRT" to be
             established with priority $10_8$ if a restart)

        Proceed to "R67RSTRT"

    R67RSTRT    (Also entered from "UPDTCALL" establishing)

        If bit 5(TRACKFLG) of FLAGWRD1 = 0:

             Proceed to "FIXDB"

        R61CNTR = 2048      (used in "UPDTCALL" to cause "R67RSTRT" to be
                                               established when = 0)
        Change priority of present job to $26_8$

        X2 = "MIS"

        $\text{T}\underline{\text{S}} = \text{BCD}\underline{\text{U}}$

        Set bit 2(R67FLAG) of FLAGWRD8 = 1

        Perform "CDUTODCM"

        Set bit 2(CALCMAN2) of FLAGWRD2 = 1

        Proceed to "NEWANGL"        (Note that HOLDFLAG, even if enter
                                     "R67RSTRT" due to "UPDTCALL", not checked;
                                     STIKFLAG, though not HOLDFLAG, checked
                                     at start of "R67START", where DAP mode
                                     check missing)
```

Quantities in Computations

See also list of major variables and list of routines

ADB: See Digital Autopilot Interface Routines.

AM: Magnitude of required maneuver angle, scale factor B0, units revolutions. The $\cos^{-1}$ routine gives an answer in the range $0^{\circ} - 180^{\circ}$.

AZIMANGL: See Orbital and Rendezvous Navigation. In documentation is "OMICRON".

BCDU: Set of three single precision cells, scale factor B-1, units revolutions, used to contain the previously desired values of the CDU angles.

BIAS (components $BIAS_0$, $BIAS_1$, $BIAS_2$): See Digital Autopilot RCS Routines.

BIASTEMP: Temporary storage for the values of BIAS to be loaded into those cells in "NEWANGL", scale factor B13, units revolutions. Since only the least significant parts of BIASTEMP are used, this gives a loading for BIAS with the B-1 scaling required.

BRATE: Vector giving maneuver rates about the x, y, and z axes, scale factor B-3, units revolutions/deci-second (i.e. revolutions per 0.1 second). Could also be considered scaled B0 in "units" of 450°/second. The "axes" are "control" ones (see [QUADROT]).

C31FLWRD: See Digital Autopilot RCS Routines.

CAM: Cosine of AM, scale factor B1.

CDUSPOT: See Coordinate Transformations.

CDUXD (components CDUXD, CDUYD, CDUZD): Values of desired CDU angles, scale factor B-1, units revolutions, used in RCS DAP if HOLDFLAG negative. See Digital Autopilot RCS Routines.

COF: Unit vector, scale factor B1, giving the "eigenvector of rotation when rotating SCAXIS to the desired position." It is computed in "VECPOINT" as the cross-product of the desired (POINTVSM) orientation in navigation base (i.e. spacecraft) coordinates and the SCAXIS complement, and is recomputed in "KALCMAN3".

COFSKEW: Vector, scale factor B2, in direction of COF (before maneuver magnitude checks, and equal to 2 sin AM times maneuver vector). Its unit (giving a scaling of B1) is employed if AM is between K_{mina} and K_{maxa}.

COSTH: See Coordinate Transformations.

DBPTC: See Orbital and Rendezvous Navigation.

DCD$\underline{U}$: See Coordinate Transformations. In addition to being an output of "SMCDURES", also used in "R63" to contain LM velocity vector, scale factor B7, units meters/centi-second (this special meeting indicated by comment as necessary).

$[\mathrm{DEL}]$: Matrix computed in "DELCOMP", whose elements have scale factor B0. The matrix is a transformation representing the effect of a rotation about the input vector CO$\underline{F}$ by the angle TS (i.e. the matrix taking a vector in initial navigation base/spacecraft coordinates to the same vector in final navigation base coordinates).

DELCD$\underline{U}$ (DELCDUX, DELCDUY, DELCDUZ): Desired CDU change each 0.1 second, scale factor B-1, units revolutions (or revolutions/deci-second).

DTHETAS$\underline{M}$: See Coordinate Transformations. Used in earlier part of "CRS61.1" to contain a limited correction factor to null pointing error, scale factor B0, units revolutions/second.

GENRET: Single precision cell used to contain return address from "R61CSM".

HOLDFLAG: See Digital Autopilot Interface Routines.

K_{1d200}: Constant, program notation "1/200", scale factor B0, single precision. Value is 0.005, corresponding to (1/200), nulling error in 20 seconds.

K_{1dc}: See Inflight Alignment.

$K_{1Dsqrt3}$: Constant, program notation "1/SQRT3", scale factor B0, value 0.57735021. Value corresponds to $1/\sqrt{3}$.

$K_{20p48sec}$: Constant, program notation "20.48SEC", scale factor B28, units centi-seconds. Value is 1×2^{-17}, corresponding to $20.48 \times 100 \times 2^{-28}$, where first term is value in seconds, second converts to centi-seconds, and third is scale factor.

K_{328nm}: Constant, program notation "328NM", scale factor B29, units meters. Value is 606844×2^{-29} (as stored), corresponding approximately to $327.67 \times 1852 \times 2^{-29}$, where first term is value in nautical miles (77777_8 for VHF 15-bit input, whose least increment is 0.01 nmi), second converts from nautical miles to meters, and third is scale factor. Note check based on conic estimate of on-board vectors, however.

K_{angtm}: Constant, program notation "ANGLTIME", scale factor B19, used to rescale time from seconds to centi-seconds. Octal value is 00003_8 04000_8, corresponding to 100×2^{-19}.

K_{arate_I}: Set of four constants, program notation "ARATE", defined only for I = 0, 2, 4, 6. They give the allowable maneuver rates, scale factor B-4, units revolutions/second. Values are 0.0022222222, 0.0088888889, 0.0222222222, and 0.0888888889: these correspond to rates of 0.05, 0.2, 0.5, and 2.0 degrees/second.

K_{biasc}: Constant, program notation "BIASCALE", scale factor B16, value 0.0002543132. Value corresponds to $(1/0.6) \times 10 \times 2^{-16}$, where first term is reciprocal of "slope of decision line" (boundary of dead zone of plot of rate error vs. angle error has slope of $0.6^{o}/$second$/^{o}$), second converts BRAT$\underline{E}$ from deci-seconds to seconds, and third is scale factor.

$\underline{K}_{cat}$: See Inflight Alignment.

K_{costen}: Constant, program notation "COSTEN", scale factor B2. Used double precision but stored single precision with nominal value of 0.98481×2^{-2}, where first term is cos 10^{o} and second is scale factor. Double precision octal is 07702_8 00034_8, corresponding to about 0.984864×2^{-2}.

$K_{degree10}$: Constant, program notation "DEGREE10", scale factor B-1, units revolutions. Used double precision but stored single precision with nominal value of 0.05556, corresponding to 10^{o}. Double precision octal is 01616_8 37651_8, corresponding to about 10.008^{o}.

$K_{deltyme}$: Constant, program notation "DELTYME", scale factor B-2. Used double precision but stored single precision with nominal value of 0.05×2^{2}, where first term is desired nulling rate (i.e. zero error in 20 seconds, as for K_{ld200}) and second is scale factor. Double precision octal is 06315_8 01616_8, corresponding to about 0.050004×2^{2}.

K_{dp1min}: Constant, program notation "DP1MIN", scale factor B28, units centi-seconds. Value is 60000×2^{-28}, corresponding to 60 seconds or one minute.

$\underline{K}_{eclipol}$: See Inflight Alignment.

K_{maxa}: Constant, program notation "MAXANG", scale factor B0, units revolutions. Used double precision but stored single precision with nominal value of 0.472222, corresponding to 170^{o}. The octal is 17071_8 00004_8, giving about the same result.

$K_{maxrate}$: Constant, program notation "MAXRATE", scale factor B0, units of radians/second. Value is 0.00174, corresponding approximately to $0.09969 \times \pi/180$, where first term is value in degrees/second and second converts to radians.

K_{mina}: Constant, program notation "MINANG", scale factor B0, units revolutions. Used double precision but stored single precision with nominal value of 0.00069375, corresponding to 0.25^{o}. Octal value is 00013_8 17071_8, corresponding to about 0.252^{o}.

K_{mindb}: See Digital Autopilot Interface Routines.

K_{minmag}: Constant, program notation "DPB-14", scale factor B2, value 2^{-14}, corresponding to a true value of 2^{-12}. Since is used with unit vectors after forming cross product, corresponds to an angle between the vectors of about 0.244 mr. Equivalent function performed elsewhere in program without explicit use of constant.

K_{p166}: See Coordinate Transformations.

K_{pt8}: Constant, program notation "POINT8", scale factor B-3, value 0.8. Value corresponds to 0.1 x 2^3, to convert between units of revolutions/second scaled B0 and revolutions/deci-second scaled B-3 (equivalent to multiplication by $360^\circ/(45^\circ \times 10)$).

$K_{radtorev}$: Constant, program notation "RADTOREV", scale factor B0, value 0.15915494, to convert between radians and revolutions. Value corresponds to $(1/2\pi)$.

K_{rvcsds}: Constant, program notation "RVCS/RDS", scale factor B4, value 15.915494 x 2^{-4}, used to convert velocity from meters/centi-second to meters/second and angles from radians to revolutions. Value corresponds to 100 x $(1/2\pi)$ x 2^{-4}, where first term is time conversion, second converts radians, and third is scale factor.

K_{singim}: Constant, program notation "SINGIMLC", scale factor B1, value 0.4285836003. True value is 0.8571672006, corresponding to sin 59°(approximately: the "2" should be "3" in the true value).

$K_{sinvec1}$: Constant, program notation "SINVEC1", scale factor B1, value 0.3796356537. True value is 0.7592713074, corresponding to sin 49.4°. Program comments indicate this angle is used as a check that "are pointing the thrust axis".

$K_{sinvec2}$: Constant, program notation "SINVEC2", scale factor B1, value 0.24621178. True value is 0.49242356, corresponding to sin 29.5°. Program comments indicate this angle is used as a check that "are pointing the AOT" (LM hardware), or otherwise "must be pointing the transponder or some vector in the Y or Z plane."

K_{tnth}: Constant, program notation "TENTH", scale factor B-3, value 0.1 x 2^3, corresponding to a true value of 0.1 (to convert angle increments from revolutions/second to revolutions/deci-second).

K_{tufits}: Constant, program notation "TUFITS", scale factor B-2, value 0.4. Value corresponds to 0.1 x 2^2, where first term converts the normal XX.XXX°/sec scaling of RATEPTC to X.XXXX°/sec and second is scale factor.

$K_{vecang1}$: Constant, program notation "VECANG1", scale factor B0, units revolutions. Value is 0.1388888889, corresponding to 50°.

$K_{vecang2}$: Constant, program notation "VECANG2", scale factor B0, units revolutions. Value is 0.09722222222, corresponding to 35°.

[MBDYTCTL]: Matrix of constants, elements with scale factor B1. True values are:

$$\begin{bmatrix} 1.0 & 0 & 0 \\ 0 & 0.99200495 & -0.12619897 \\ 0 & 0.12619897 & 0.99200495 \end{bmatrix}$$

Values are the same as those used in conjunction with [QUADROT].

[MFI]: Matrix with elements having scale factor B2, "relating final spacecraft axes to initial spacecraft axes."

[MFISYM]: Matrix with elements having scale factor B2, "the symmetric part of [MFI]", computed if AM exceeds K_{maxa}.

[MFS]: Matrix with elements having scale factor B1, "a direction cosine matrix relating the final spacecraft orientation to stable member axes." If $\underline{A}$ is a vector in spacecraft coordinates, then the same vector in stable member axes ($\underline{B}$) is given by $\underline{B} = [MFS]\ \underline{A}$.

[MIS]: Matrix with elements having scale factor B1, "a direction cosine matrix relating the initial spacecraft orientation to stable member axes."

NCDU: Value of next-desired CDU angles, single precision, scale factor B-1, units revolutions.

NEXTIME: Single precision value of time, scale factor B14, units centi-seconds, when "NEWDELHI" is to be entered next. Used to compensate for delays in performing "NEWDELHI" due to jobs of higher priority.

OPTIND: See Optics Computations.

OPTNTYPE: See Orbital and Rendezvous Navigation.

PLANVCUT: See Orbital and Rendezvous Navigation.

POINTVSM: Unit vector, scale factor B1, giving the direction in which SCAXIS is to be pointed. The components of this vector should be in stable member coordinates. Also used in R63 computations for utility storage purposes.

Q611: Single precision cell used to contain return address from "CRS61.1".

Q6111: Single precision cell used to contain return address from "R63".

[QUADROT]: Matrix of constants, elements with scale factor B0. Values are:

$$\begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.09920 & -0.01262 \\ 0 & 0.01262 & 0.09920 \end{bmatrix}$$

Matrix serves to convert time units from rates/second to rates/decisecond (a factor of 0.1), and also "accounts for the roll displacement of the reaction jets with respect to navigation base coordinates" by rotating about X by -7.25°. Sin 7.25° = 0.1262 and cos 7.25° = 0.9920. $QUADROT_0$ used to convert from rev/sec to rev/deci-sec.

R61CNTR: See Orbital and Rendezvous Navigation. Also used in option 2 of P20 to control period of entrance to "R67RSTRT" (about once every 34 minutes) from "UPDTCALL" to restore [MIS] .

R63TIME: Value of time for which R63 computations are carried out, scale factor B28, units centi-seconds. After being used for that purpose, loaded in R63 computations with magnitude of the line-of-sight vector, scale factor B29, units meters. Set zero if option 1/5 of P20 with star code $\leq 45_8$ (i.e. not sun, earth, or moon); otherwise has CSM vector.

R67TIME: See Orbital and Rendezvous Navigation.

RATEINDX: See Digital Autopilot Interface Routines.

RATEPTC: Single precision cell, scale factor B-2, units revolutions, loaded by R1 of N79 (for option 2 of P20). Since the desired display scale is X.XXXX°/sec but the noun processing routine considers the input to be XX.XXX°, another factor of 0.1 is accomplished by K_{tufits}. Due to the noun routine, the maximum value cannot exceed about 8.9999°/sec.

RCSFLAGS: See Digital Autopilot Interface Routines.

SAVEPOS: Cells used in "R63" initially to contain CSM position vector, scale factor B29, units meters. For P20 options 0/4 subsequently loaded with the unit line-of-sight vector, scale factor B1, between the vehicles; for options 1/5 loaded with the negative unit vector.

SAVEVEL: Cells used in "R63" initially to contain CSM velocity vector, scale factor B7, units meters/centi-second. Zeroed for options 1/5 of P20 if star code $\leq 45_8$.

SCAXIS: Unit vector, scale factor B1, giving the "axis to be pointed" along the direction indicated by "POINTVSM". SCAXIS components should be in spacecraft (i.e. navigation base) coordinates. Also used in R63 computations for utility storage purposes.

SINTH: See Coordinate Transformations.

T_{tm}: Time required to perform maneuver of magnitude AM (computed in "LOCSKIRT"), scale factor B28, units centi-seconds. Subsequently changed to be the value of the computer clock when the maneuver should be completed (as part of initial-cycle computations). Set to +MAX to disable the exit if doing option 2 of P20.

TBASE1: Cell used for address storage purposes in "CHKLINUS" to permit restoration of program display after a restart (by "RELINUS"). Cell normally used to contain the waitlist restart information (see 3420.5-27), and is single precision.

TEMPR60: Single precision cell used to contain return address from "R60CSM".

THETADX: See Digital Autopilot RCS Routines.

[TMFI]: Transpose of matrix [MFI], scale factor B2.

[TMIS]: Transpose of matrix [MIS], scale factor B1.

UTAMG: Unit vector, scale factor B1, used in the dedicated "CALCGA" routine included in "UTOPT45". Stored in push-down list location 6D.

UTIGA, UTMGA, UTOGA: Values of inner, middle, and outer gimbal angles, scale factor B0, loaded from THETA (see Coordinate Transformations), units revolutions. Stored in 26D, 28D, and 24D respectively.

UTPIT: See Orbital and Rendezvous Navigation. Sometimes called "gamma".

UTSA: Information on desired orientation (based on AZIMANGL) about the LOS (UTSB) vector, scale factor B1, computed in "UTOPT45". Stored in 18D, with UTUX also used in program as notation.

UTSAP: Information on "primed coordinate system" orientation of UTSA, scale factor B1, computed in "UTOPT45". Stored in 0D, with UTUXP also used in program as notation.

UTSB: Information on desired pointing (LOS) vector, scale factor B1, loaded in "UTOPT45". Stored in 24D.

UTSBP: Information on "primed coordinate system" orientation of UTSB (i.e. spacecraft axis to be pointed in LOS direction), scale factor B1, loaded in "UTOPT45". Stored in 6D.

UTSTARNO: See Orbital and Rendezvous Navigation.

UTUY, UTUYP, UTUZ, UTUZP: Vectors, scale factor B1, used in "UTOPT45" computations within the dedicated "AXISGEN" routine.

UTX, UTY, UTZ: Output of the dedicated "AXISGEN" routine in "UTOPT45", formed before "exit" as unit vectors, scale factor B1, for use in the dedicated "CALCGA" routine.

UTYAW: See Orbital and Rendezvous Navigation. Sometimes called "rho".

VELdC: See Inflight Alignment.

WBODY (components $WBODY_0$, $WBODY_1$, $WBODY_2$): See Digital Autopilot RCS Routines.

Boost Computations

```
P11      (Entered from "CHKCOMED" after liftoff deduced)

    Set $T_{evt} = T_{now}$ and $T_{now} = 0$     (with special provisions for
                                                        restarts)
    $T_{eph} = T_{eph} + T_{evt}$   (special restart provisions)
    Inhibit interrupts

    Perform "PREREAD1"

    Release interrupts

    Call "ATERTASK" in 0.5 second
    TS = 11 and perform "NEWMODEX"

    Set bit 12(NODOPO1) of FLAGWRD1 = 1

    DNLSTCOD = 3

    Perform "CLEANDSP"      (gets rid of backup liftoff, V75)

    $\underline{ERCOMP} = \underline{ERCOMP} + [X_{sm}]\ \underline{THETAN}$

    TS = $T_{evt}$
    Perform "EARTHR*" (starting at second line)
    (OGC, IGC, MGC) = $\underline{CDU}$, rescaled to scale factor B0 revolutions,
                                  double precision

    RTX2 = 0      (means earth)

    ALT = $C_{pgncsat}$

    LONG = $C_{pdong}$

    LAT = $C_{atd}$

    Set bit 13(ERADCOMP) of FLAGWRD1 = 0  (means pad radius altitude
                                            base; bit 12(LUNLATLO) of
    TS = 0                                  FLAGWRD3 apparently presumed
                                            zero from "SLAP1")
    Perform "LALOTORV"

    $\underline{R}_{nl}$ = $\underline{ALPHAV}$
```

```
    $TS_2 = 0$    (time argument)
    TS = 1    (GAMRP value, scaled B1)
    Perform "LALOTORV"  (starting at 5th line)
    $\underline{Z}_r = -$ unitALPHA$\underline{V}$
    $\underline{V}_{nl} = K_{merthrt} (\underline{R}_{nl} * \underline{C}_{unitw})$
    $T\underline{S}_1 = unit(\underline{Z}_r * \underline{C}_{unitw})$        (east)
    $T\underline{S}_2 = unit(T\underline{S}_1 * \underline{Z}_r)$        (- south)
    $\underline{X}_r = unit( T\underline{S}_1 \sin LAUNCHAZ + T\underline{S}_2 \cos LAUNCHAZ)$
    $\underline{Y}_r = - unit(\underline{X}_r * \underline{Z}_r)$
    $[REFSMMAT] = \begin{bmatrix} \underline{X}_r \\ \underline{Y}_r \\ \underline{Z}_r \end{bmatrix}$    (cells also used for temporary storage
                                                during computation)
    $SATRLRT = SATRLRT \ sgn\ (\frac{1}{2} - LAUNCHAZ + C_{azmth})$
    Set bit 13(REFSMFLG) of FLAGWRD3 = 1
    AVEGEXIT = "VHHDOT"
    $1dPIPADT = K_{2secp}$
    TS = 200 - TIME1, limited $\geqslant$ 1
    Call "READACCS" in TS centi-seconds  (i.e. 2 seconds after liftoff
                                                      deduced)
    Proceed to "NORMLIZE"
VHHDOT
    $VMAGI = |\underline{V}|$
    $HDOT = unit\underline{R} \cdot \underline{V}$
    $TS = K_{rpad}$
    If bit 2(AMOONFLG) of FLAGWRDO = 1:    (apparently presumed 0 from
                                                  "SLAP1")
        $TS = |RL\underline{S}|$ , rescaled to scale factor B29
    $ALTI = |\underline{R}| - TS$
    $TS = 0662_{vn}$
    Proceed to "REGODSP"
```

```
ATERTASK        Called by "P11" 0.5 second after liftoff deduced, or if
                                                   restart (group 4.51)
    Establish "ATERJOB"  (priority $01_8$)
    Set bit 3 of RCSFLAGS = 1
    Perform "NEEDLER"
    SATSW = 1
    End of task
ATERJOB         (Job has low priority, hence interrupted for others)
    If bits 15-14(DAPBIT1, DAPBIT2) of FLAGWRD6 $\neq 00_2$:
        End of job   (restart group 4.51 remains set, however)
    If bit 10 (CMC Control of Saturn complement) of channel 30 = 0:
        If  $SATSW > 0$:
            Proceed to "ATRESET"    (control just given)
    If bit 10 (CMC Control of Saturn complement) of channel 30 = 1:
        If  $SATSW \leq 0$:
            Proceed to "ATRESET"    (control just removed)
    $TS_1 = TIME1 - C_{rpstart}$          (tag here "ATTDISP")
    If  $TS_1 \leq 0$, proceed to "NOPOLYM"
    If  $TS_1 + C_{polystop} > 0$:        ($C_{polystop}$ negative number)
        Proceed to "NOPOLYM"
    If  $T_{now} \geq 163.84$ seconds:       (i.e. $TIME2 > 0$)
        Proceed to "NOPOLYM"
    $I = C_{ponum} + 1$
    $POLYANS = \sum_{i=0}^{I} C_{pyc,i} \; (TS_1)^i$
    NMPH = POLYANS, rescaled to scale factor B0
    $ANGROLL = SATRLRT \; TS_1 - \frac{1}{2} + LAUNCHAZ - C_{azmth}$
    If $(ANGROLL_{sp} \;\; SATRLRT{+}1_{sp}) > 0$:      (only most significant half
                                                   of product checked)
        ANGROLL = 0               (rollout completed)
    $\underline{X}_{dc}$ = unit(sin NMPH,  0,  -cos NMPH)
```

```
    $\underline{Y}_{dc}$ = unit(- sin ANGROLL cos NMPH,  - cos ANGROLL,  - sin ANGROLL sin NMPH)
    $\underline{Z}_{dc}$ = unit(- cos ANGROLL cos NMPH,  sin ANGROLL,  - cos ANGROLL sin NMPH)
    Perform "CALCGTA"
    Proceed to "NOPOLYM"
NOPOLYM
    Perform "CDUTRIG"
    $T\underline{S}_1$ = (IGC, MGC, OGC), converted to single precision twos comp.,B-1 rev.
    $T\underline{S}_2$ = CDUSPO$\underline{T}$, converted to single precision twos comp., B-1 rev.
    $T\underline{S}$ = $T\underline{S}_1$ - $T\underline{S}_2$   (ones comp. difference formed)   (y,z,x for 0,1,2)
    $AK_0 = \frac{1}{4}$ ($TS_2$ + $TS_0$ SINCDUZ)
    $AK_1$ = $TS_0$ COSCDUX COSCDUZ + $TS_1$ SINCDUX
    $AK_2$ = - $TS_0$ SINCDUX COSCDUZ + $TS_1$ COSCDUX
    If SATSW > 0:   (no takeover in effect)
        Perform "NEEDLER"
        Delay 0.25 seconds (by putting job to sleep via "DELAYJOB")
        Proceed to "ATERJOB"   (program comments indicate loop rate of about 0.56 second)
    If SATSW = 0:   (first takeover cycle)
        BIASA$\underline{K}$ = - A$\underline{K}$
        SATSW = -1
    A$\underline{K}$ = BIASA$\underline{K}$ + A$\underline{K}$        (tag here "STEERSAT")
    A$\underline{K}$ = $C_{satscale}$ A$\underline{K}$
    Perform "NEEDLER"
    Delay 0.25 seconds (by putting job to sleep via "DELAYJOB")
    Proceed to "ATERJOB"
```

```
ATRESET          Entered from "ATERJOB" for a change in bit 10 of channel 30
    Set bit 3 of RCSFLAGS = 1       (causes "NEEDLER" re-initialization)
    Perform "NEEDLER"
    Delay 0.06 seconds (by putting job to sleep via "DELAYJOB")
    TS = SATSW
    If  $TS > 0$:          (control just given)
        Set bit 9(S4B Takeover Enable) of channel 12 = 1
        SATSW = 0      (first takeover cycle, causing BIASAK loading)
    If  $TS \leq 0$:          (control just removed)
        SATSW = 1
        Set bit 9(S4B Takeover Enable) of channel 12 = 0
    Perform "NEEDLER"
    Delay 0.25 seconds (by putting job to sleep via "DELAYJOB")
    Proceed to "ATERJOB"
SATSTKON         Entered from "STABLISH"
    T5LOC = "REDOSAT"
    Set TIME5 to cause program interrupt #2 in 0.01 seconds
    Set bits 15-14 (DAPBIT1, DAPBIT2) of FLAGWRD6 = $11_2$
    Perform "ZEROJET"
    Proceed to "PINBRNCH"
REDOSAT
    Set bit 3 of RCSFLAGS = 1
    Perform "NEEDLER"
    Set bit 9 (S4B Takeover Enable) of channel 12 = 1  (Reset e.g. by
                                                         "STABLISH")
    T5LOC = "SATSTICK"
    Set TIME5 to cause program interrupt #2 in 0.10 seconds
    Resume
```

```
SATSTICK

    T5LOC = "SATSTICK"
    Set TIME5 to cause program interrupt #2 in 0.10 seconds
    TS = bits 6-1 of (- channel 31)
    Perform "STICKCHK"
    $AK_0 = C_{satrt_{RMANNDX}}$
    $AK_1 = C_{satrt_{PMANNDX}}$
    $AK_2 = C_{satrt_{YMANNDX}}$
    Perform "NEEDLER"
    Resume

P15JOB

    TS = $0633_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    TLITMP = $T_{ig} + K_{tlidt}$
    P40TMP = $(T_{ig} - T_{now})$   (done in interpretive language, thus maximizing
                                     the error in entering "T6SET")
    Call "T6SET" in P40TMP centi-seconds (via "LONGCALL" double-precision
                                          waitlist time entrance)
    $T_{ig}$ = TLITMP
    TS = $0614_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    VMAGI = $|\underline{V}_{rectcm}|$
    Set bit 11(TIMRFLAG) of FLAGWRD7 = 1
    VGTLI = VCdO - VMAGI
    AVEGEXIT = "SIVBCOMP"
```

```
        NVWORD1 = $0695_{vn}$
        Call "CLOKTASK" in 0.01 seconds
        End of job
T6SET
        $T_{evt} = T_{now}$
        Call "T6RESET" in 10 seconds
        Set bit 13 (S4B Injection Sequence start) of channel 12 = 1
        Set bit 3(Uplink activity) of channel 11 = 1
        End of task
T6RESET
        Set bit 13(S4B Injection Sequence Start) of channel 12 = 0
        Set bit 3(Uplink activity) of channel 11 = 0
        AVEGDT = $K_{100sec}$
        Establish "P40S/SV"  (priority $12_8$)
        End of task
SIVBCOMP
        If  $(T_{togo} - K_{tensec}) \geqslant 0$:           ($T_{togo} = T_{now} - T_{ig}$; $T_{ig}$
             Set bit 11(STEERSW) of FLAGWRD2 = 1          not updated by P15
                                                          until STEERSW = 1)
        VMAGI =  $|\underline{V}|$
        HDOT = unit$\underline{R} \cdot \underline{V}$
        TS = $K_{rpad}$
        If bit 2(AMOONFLG) of FLAGWRD0 = 1:
             TS = $|RL\underline{S}|$ , rescaled to scale factor B29
        ALTI = $|\underline{R}|$ -  TS
        VGTLI =  VCd0 - VMAGI
        If bit 11(STEERSW) of FLAGWRD2 = 0:
             VPAST = VMAGI
             Proceed to "SERVXT1"
```

```
    If (VMAGI - VPAST) $<$ 0:
        VPAST = VMAGI
        Proceed to "SERVXT1"
    TS = VGTLI / (VMAGI - VPAST)
    If overflow has occurred since "SIVBCOMP" entered:    (e.g $|TS| \geq 2^9$)
        VPAST = VMAGI
        Proceed to "SERVXT1"
    $TS_1 = K_{200b19} \, TS - C_{dtf}$
    $T_{ig} = T_{pptm} + TS_1$    (predicted cutoff time)
    If $(TS_1 - K_{4sec}) \geq 0$:
        VPAST = VMAGI
        Proceed to "SERVXT1"
    Inhibit interrupts      (Tag here "KILLSIVB")
    (TS, AVEGDT+1) = $(T_{ig} - T_{now})$, with sign agreement forced
    If AVEGDT+1 $\leq$ 0:
        AVEGDT+1 = 1
    Call "SIVBOFF" in AVEGDT+1 centi-seconds
    Set bit 11(STEERSW) of FLAGWRD2 = 0
    Proceed to "SERVEXIT"
SIVBOFF
    Set bit 14(S4B Cutoff) of channel 12 = 1
    $T_{evt} = T_{now}$
    Delay 2.5 seconds
    NVWORD1 = +0
    Set bit 11(TIMRFLAG) of FLAGWRD7 = 0
    Establish "POSTTLI"  (priority $12_8$)
    End of task
```

POSTTLI

$TS = 1695_{vn}$

```
Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                      if proceed, proceed
                      otherwise, proceed to previous line

Proceed to "GOTOPOOH"
```

Quantities in Computations

See also list of major variables and list of routines

ldPIPADT: See IMU Computations.

$\underline{AK}$ (AK_0, AK_1, AK_2): See Digital Autopilot Interface Routines.

$ALPHA\underline{V}$: See Coordinate Transformations.

ALT: See Coordinate Transformations.

ALTI: Value of vehicle altitude above K_{rpad} (or $RL\underline{S}$ magnitude if state vector has lunar origin), scale factor B29, units meters. It is displayed in R3 of N62.

ANGROLL: Value of roll angle remaining to be performed, scale factor BO, units revolutions.

AVEGDT: See Burn Control. Also used to retain the required delay before cutoff (in least significant half, B14 centi-seconds) for use in restart protection computations.

AVEGEXIT: See General Program Control.

$BIASA\underline{K}$:Value of the complement of $A\underline{K}$, same scaling, saved on the pass through "ATERJOB" following the one which senses a logic 0 to 1 transition (binary 1 to 0) of bit 10 of channel 30. It effectively permits the errors to be "nulled" at this point, and is applied until Saturn control is removed. If get a restart, or if switch subsequently set to CMC position, a new set of $BIASA\underline{K}$ values is determined.

C_{atd}: See Prelaunch Alignment.

C_{azmth}: See Prelaunch Alignment.

C_{dtf}: Single precision erasable memory constant, program notation "DTF", scale factor B14, units centi-seconds, subtracted from the "raw" time-to-go in order to account for thrust decay effects in "SIVBCOMP".

C_{pdong}: Erasable memory constant, program notation "PADLONG", scale factor BO, units revolutions, giving longitude of pad.

$C_{pgncsat}$: Erasable memory constant, program notation "PGNCSALT", scale factor B29, units meters, giving "altitude of IMU" (measured from K_{erad}, see Coordinate Transformations). The magnitude of the initial-condition position vector is the sum of K_{erad} and $C_{pgncsat}$.

$C_{polystop}$: Single precision erasable memory constant, program notation "POLYSTOP", scale factor B14, units centi-seconds, giving minus (time from liftoff minus $C_{rpstart}$) when performance of pitch polynomial should be stopped. The pitch polynomial will be halted, regardless of the numbers in the erasable memory constants, when time from liftoff reaches 163.84 seconds.

C_{ponum}: Single precision erasable memory constant, program notation "POLYNUM", scale factor B14, giving "polynomial degree -1" of the polynomial used to compute POLYANS. Erasable memory cell allocation is such that C_{ponum} should not exceed 5 (giving a final term of $C_{pyc,6}$ TS_1^6).

$C_{pyc,i}$ (i = 0 - C_{ponum} +1): Erasable memory coefficients used to compute POLYANS, with highest order coefficient stored in("POLYLOC" +3),next highest in ("POLYLOC" +1), etc. Scale factor of $C_{pyc,0}$ is B5 revolutions; scale factor of $C_{pyc,1}$ is B-9 revolutions/centi-second; scale factor of ith is B(5 - 14i), with units of revolutions/(centi-second)^i. For C_{ponum} = 5, the first term, $C_{pyc,0}$, is in ("POLYNUM" +1).

$C_{rpstart}$: Single precision erasable memory constant, program notation "RPSTART", scale factor B14, units centi-seconds, giving the time (from CMC-deduced liftoff) when the evaluation of the roll and pitch computations should be started for generation of attitude error displays (as opposed to driving the displays from the difference between sampled attitude and present attitude).

C_{satrt_i} (i = 0 - 3): Set of single precision erasable memory constants, program notation (i = 0) "SATRATE", scale factor B12, units output pulses. Cells used in "SATSTICK" to specify the number of pulses generated by "NEEDLER" routine for deflections of the RHC in roll, pitch, and yaw (a positive input via $\underline{AK}$ gives a negative output). Values for i = 0 and i = 3 should be zero; for i = 1, the value for a positive maneuver is supplied; and for i = 2, the value for a negative maneuver is supplied. A saturated error counter would be produced for a setting of 384 x 4 = 1536 (the factor of 4 is due to the value of K_{dacsc} in "NEEDLES") least increments.

$C_{satscale}$: Single precision erasable memory constant, program notation "SATSCALE", scale factor B2, used as the gain on the $\underline{AK}$ outputs for Saturn steering based on polynomial output. A value of 1.00 for the constant would cause a saturated error counter for $67\frac{1}{2}^{\circ}$ of roll error and 16.875° of pitch/yaw error (the saturated error counter corresponds to 384 pulses). The "1.00" value is effectively the one employed to drive the needles if no Saturn steering is done. For RHC control, C_{satrt} is used to specify the output information.

$\underline{C}_{unitw}$: See General Program Control.

CDUSPO$\underline{T}$: See Coordinate Transformations.

COSCDUX, COSCDUY, COSCDUZ: See Coordinate Transformations.

DNLSTCOD: See Telemetry.

ERCOM$\underline{P}$: See Prelaunch Alignment.

HDOT: Value of altitude rate for display purposes, scale factor B7, units meters/centi-second.

IGC: See Coordinate Transformations. Used to retain sampled value of CDU_y near deduced liftoff for attitude error display during the first $C_{rpstart}$ centi-seconds after liftoff deduced.

K_{100sec}: Constant, program notation "100SEC", scale factor B28, units centi-seconds. Value is 10000 x 2^{-28}, corresponding to 100 seconds.

K_{200b19}: Constant, program notation "200B+19", scale factor B19, units of (centi-seconds/computing interval). Value is 200 x 2^{-19}, corresponding to the two-second computing interval.

K_{2secp}: See IMU Computations.

K_{4sec}: Constant, program notation "4SEC", scale factor B28, units centi-seconds. Value is 400 x 2^{-28}, corresponding to 4 seconds.

$K_{merthrt}$: Constant, program notation "-ERTHRAT", scale factor B-18, units radians/centi-second. Value is -7.292115138E-7 x 2^{18}, corresponding to an earth rotation period of about 86164.099 seconds. The octal value gives a decimal equivalent of about -7.2921150717E-7.

K_{rpad}: See Burn Control.

K_{tensec}: Constant, program notation "TENSEC", scale factor B28, units centi-seconds. Value is 1000 x 2^{-28}, corresponding to 10 seconds.

K_{tlidt}: Constant, program notation "TLIDT", scale factor B28, units centi-seconds. Value is 57760 x 2^{-28}, corresponding to 577.60 seconds (or 9 minutes 37.60 seconds).

LAT, LONG: See Coordinate Transformations.

LAUNCHAZ: See Prelaunch Alignment.

MGC: See Coordinate Transformations. Used to retain sampled value of CDU_z near liftoff (see IGC).

NMPH: Value of nominal pitch history, scale factor B0, units revolutions, obtained from POLYANS.

NVWORD1: See Burn Control.

OGC: See Coordinate Transformations. Used to retain sampled value of CDU_x near liftoff (see IGC).

P40TMP: Required time delay until when "T6SET" is to be performed, scale factor B28, units centi-seconds.

PMANNDX: See Digital Autopilot Interface Routines.

POLYANS: Output of polynomial used for determination of nominal pitch history, scale factor B5, units revolutions.

$\underline{R}_{nl}$: See General Program Control.

RCSFLAGS: See Digital Autopilot Interface Routines.

RL$\underline{S}$: See Coordinate Transformations.

RMANNDX: See Digital Autopilot Interface Routines.

RTX2: See Orbital Integration.

SATRLRT: Quantity used to specify nominal Saturn roll rate, scale factor B0, units revolutions/centi-second. It is set as part of the prelaunch erasable load, and subsequently has sign changed in the latter part of "P11" depending on the deduced polarity of the required Saturn roll. Roll rate should not exceed about 2.2 degrees/second (i.e. 2^{-14} rev/centi-second), or malfunction of the check in "ATERJOB" for zeroing of ANGROLL could result. If it does not, check there will set ANGROLL to 0 when the value reaches about(0.05/n) degrees beyond the nominal roll amount, where "n" is the Saturn roll rate in degrees/second.

SATSW: Single precision cell, scale factor B14, used to control the logic associated with automatic steering of Saturn (based on polynomial outputs). An initial condition of +1 is set in "ATERTASK" (entered also if a restart), meaning display only. A value of +0 is set in "ATRESET" just after control takeover (i.e. present value of SATSW = 1 and of channel 30 bit 10 = 0). In "NOPOLYM", the value of 0 causes BIASA$\underline{K}$ to be initialized, and the value of SATSW set to -1 (indicating a subsequent pass through Saturn automatic steering). When the value of channel 30 bit 10 is sensed as 1 (a logic 0, meaning no control), and SATSW is ≤ 0, then "ATRESET" entered again, this time causing a reset of SATSW to +1, the configuration for attitude error display only.

SINCDUX, SINCDUY, SINCDUZ: See Coordinate Transformations.

T_{eph}: Triple precision value of ephemeris time, program notation "TEPHEM", scale factor B42, units centi-seconds. Cell must be initialized as part of prelaunch erasable memory load to the elapsed time between ephemeris origin and the pre-launch origin setting for the computer clock. When liftoff is deduced, T_{evt} is added to T_{eph}. T_{eph} is changed by V70.

T_{togo}: See Burn Control.

T5LOC: See Digital Autopilot Interface Routines.

THETA$\underline{N}$: See Prelaunch Alignment.

TIME5: See Digital Autopilot Interface Routines.

TLITMP: Value of input T_{ig} (N33) plus K_{tlidt}, used to write over T_{ig} with ignition time (since the input is of the Injection Sequence Start time). Scale factor is B28, units centi-seconds.

$\underline{V}_{nl}$: See General Program Control.

$\underline{V}_{rectcm}$: See Orbital Integration.

VCdO: Value of desired cutoff velocity, scale factor B7, units meters/centi-second, displayed in R1 of N14. Program notation "VC/O".

VGTLI: Required velocity to be gained for TLI burn, scale factor B7, units meters/centi-second. It is displayed in R2 of N95, and occupies the same cells as V_{gtig_z} (for telemetry purposes).

VMAGI: Value of magnitude of vehicle inertial velocity, scale factor B7, units meters/centi-second. It is displayed by several nouns, and also has program notation "VNOW". For telemetry purposes, it occupies the same cells as V_{gtig_y}.

VPAST: Value of previous VMAGI, scale factor B7, units meters/centi-second, loaded in "SIVBCOMP" and used in the denominator of the time-to-go computation.

$\underline{X}_r$: Unit vector, scale factor B1, forming first row of [REFSMMAT] computed after liftoff deduced, and defined as "horizontal vector at launch azimuth".

$\underline{Y}_r$: Unit vector, scale factor B1, forming second row of [REFSMMAT], and formed as $\underline{Z}_r * \underline{X}_r$.

YMANNDX: See Digital Autopilot Interface Routines.

$\underline{Z}_r$: Unit vector, scale factor B1, forming third row of [REFSMMAT], and defined as "local vertical".

Burn Control

P30

```
Set bit 7(UPDATFLG) of FLAGWRD1 = 1     (See page 13 at front
                                         of document for verbal
Set bit 5(TRACKFLG) of FLAGWRD1 = 1      description of parts
                                         of this program)
$TS = 0633_{vn}$

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                      if proceed, proceed
                      otherwise, proceed to previous line

$TS = 0681_{vn}$

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                      if proceed, proceed
                      otherwise, proceed to previous line

Set bit 8(XDELVFLG) of FLAGWRD2 = 1

Set bit 7(UPDATFLG) of FLAGWRD1 = 0

$T_{decl} = T_{ig}$
Perform "CSMPREC"

RTX2 = X2        (0 for earth, 2 for moon)

$\underline{V}_{tig} = \underline{V}_{att}$
$\underline{R}_{tig} = \underline{R}_{att}$
$\underline{R}_{act3} = \underline{R}_{att}$
$UNR\underline{M} = unit(\underline{R}_{act3} * \underline{V}_{tig})$
$T\underline{S}_2 = - UNR\underline{M}$
$T\underline{S}_3 = - unit\underline{R}_{act3}$
$T\underline{S}_1 = T\underline{S}_3 * UNR\underline{M}$

$DELVSI\underline{N} = DELVLV\underline{C} \begin{bmatrix} T\underline{S}_1 \\ T\underline{S}_2 \\ T\underline{S}_3 \end{bmatrix}$      (DELVLVC notation also "DELVSLV")

RTX1 = X1        (set to -2 for earth, -10 for moon)
```

```
$VGDISP = |DELVSI\underline{N}|$
$X2 = RTX2$        (0 for earth, 2 for moon)
$VVE\underline{C} = (\underline{V}_{tig} + DELVSI\underline{N})$, shifted left X2 places (B7 earth, B5 moon)
$RVE\underline{C} = \underline{R}_{tig}$, shifted left X2 places (B29 earth, B27 moon)
$X1 = RTX1$
$TS = K_{rpad}$
If $X1 \neq -2$:      (i.e. not earth)
    $TS = |RL\underline{S}|$
XXXALT = TS
Perform "APSIDES"
$HPER = (TS_{rp} - XXXALT)$, shifted right RTX2 places (B29), limited $\leq K_{maxnm}$
$HAPO = (TS_{ra} - XXXALT)$, shifted right RTX2 places (B29), limited $\leq K_{maxnm}$
$TS = 0642_{vn}$            (Tag here "PARAM30")
Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                       if proceed, proceed
                       otherwise, proceed to previous line
If bit 13(REFSMFLG) of FLAGWRD3 = 1:
    $T\underline{S} = DELVSI\underline{N}$
    Perform "GET+MGA"
If bit 13(REFSMFLG) of FLAGWRD3 = 0:
    $pMGA = - K_{marsdp}$
Set bit 11(TIMRFLAG) of FLAGWRD7 = 1
NVWORD1 = +0
Call "CLOKTASK" in 0.01 seconds
$TS = 1645_{vn}$
Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                       if proceed, proceed
                       otherwise, proceed to previous line
Set bit 11(TIMRFLAG) of FLAGWRD7 = 0
Proceed to "GOTOPOOH"
```

P31

```
Set bit 5(LMACTFLG) of FLAGWRD2 = 0

ECSTEER = $K_{1b2}$

Set bits 7(UPDATFLG) and 5(TRACKFLG) of FLAGWRD1 = 1

Set bit 1(PCFLAG) of FLGWRD10 = 0

Set bit 6(CSISFLAG) of FLGWRD11 = 0   (tag here "P31ALMRT")

CENTANG = $K_{130deglo}$          (tag here "INPUTDSP")

ELEV = $K_{208deglo}$          (constant is $208.3^{o}$)

If bit 6(CSISFLAG) of FLGWRD11 = 0:          (as it will)

    NN1 = 1

TS = $0611_{vn}$

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                      if proceed, proceed
                      otherwise, proceed to previous line

TS = $0655_{vn}$

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                      if proceed, proceed
                      otherwise, proceed to previous line

TS = $0637_{vn}$

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                      if proceed, proceed
                      otherwise, proceed to previous line

If bit 12(CMOONFLG) of FLAGWRD8 = 1:        (if zero, X1 not initialized,
                                          so garbled results for earth-orbit)
    X1 = -10  (indicates moon)

$TS = (K_{2b6} - K_{mutab_{-X1}} |\underline{V}_{rectcm}|^{2} |\underline{R}_{rectcm}|) / |\underline{R}_{rectcm}|$     (B-22 earth,
                                                                                   B-20 moon;
                                                                                   value of 1/a)

$TS_1 = K_{mutab_{4-X1}} (K_{2pisc} / \sqrt{TS})$     (B0 earth, B2 moon)

$T_{ig} = T_{csi} - \frac{1}{2} TS_1 / TS$
```

```
    $TS = 0633_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    $X1 = -2$        (settings for
    $X2 = 0$              earth)
    If bit 12(CMOONFLG) of FLAGWRD8 = 1:
        $X1 = -10$       (settings for
        $X2 = 2$              moon)
    $RTX1 = X1$
    $RTSR1dMU = K_{mutab_{4-X1}}$
    $TS = K_{mutab_{-2-X1}}$        (subscript = 0 for earth, 8 for moon)
    If bit 12(CMOONFLG) of FLAGWRD8 = 1:
        Shift TS right 6 places
    $RTMU = TS$      (scale factor B36)
    $RTX2 = X2$
    Set bit 6(FINALFLG) of FLAGWRD2 = 0
    If bit 1(PCFLAG) of FLGWRD10 = 1:      (should be 0)
        Proceed to "P36A"
    Perform "VN1645"
    Proceed to "P31RECYC"
P31RECYC
    Set bit 7(HAFLAG) of FLGWRD11 = 1          (bit only reset by "P32/P72C")
    $TS = T_{csi}$
    Proceed to second line of "P32/P72B"  (returns to "P31RT" due to HAFLAG,
                                          unless get alarm)
```

```
P31RT
    $\underline{DELVLVC} = 0$
    $DELVLVC_x = K_{25thous}$ (DIFFALT + $C_{hamdelh}$)
    Perform "N90/N81"        (page BURN-23)
    Perform "VN1645"
    Proceed to "P31RECYC"
P32
    Set bit 5(LMACTFLG) of FLAGWRD2 = 0
    $ECSTEER = K_{1b2}$
    Proceed to second line of "P72"
P72
    Set bit 5(LMACTFLG) of FLAGWRD2 = 1
    Set bits 7(UPDATFLG) and 5(TRACKFLG) of FLAGWRD1 = 1  (tag here "P32/P72A")
    Set bit 1(PCFLAG) of FLGWRD10 = 0
    If bit 7(HAFLAG) of FLGWRD11 = 1:      (i.e. P31 alarm return)
        Proceed to 5th line of "P31"   (note HAFLAG remains set, locking
                                                  out P32/P72).
    $CENTANG = K_{130deglo}$              (tag here "INPUTDSP")
    $ELEV = K_{208deglo}$            (constant is 208.3$^o$)
    If bit 6(CSISFLAG) of FLGWRD11 = 0:       (as it will, see "P82")
        NN1 = 1
    $TS = 0611_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    $TS = 0655_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    $TS = 0637_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
```

```
Set bit 6(CSISFLAG) of FLGWRD11 = 0
$T_{ig} = T_{csi}$
X1 = -2          (settings for
X2 = 0               earth)
If bit 12(CMOONFLG) of FLAGWRD8 = 1:
    X1 = -10     (settings for
    X2 = 2           moon)
RTX1 = X1
RTSR1dMU = $K_{mutab_{4-X1}}$
TS = $K_{mutab_{-2-X1}}$          (subscript = 0 for earth, 8 for moon)
If bit 12(CMOONFLG) of FLAGWRD8 = 1:
    Shift TS right 6 places
RTMU = TS        (scale factor B36)
RTX2 = X2
Set bit 6(FINALFLG) of FLAGWRD2 = 0
If bit 1(PCFLAG) of FLGWRD10 = 1:     (should be 0)
    Proceed to "P36A"
Perform "VN1645"
Proceed to "P32/P72B"
```

P32/P72B

```
TS = $T_{ig}$
Perform "ADVANCE" (page BURN-13)    (Tag "P31ENT", from "P31RECYC")
Perform "INTSTALL"
Set bit 4(CONICINT) of FLAGWRD3 = 0
Set bit 4(CONICINT) of FLAGWRD3 = 1    (superseding previous line)
```

```
          $T_{decl} = T_{tpi}$
          Set bit 12(MOONFLAG) of FLAGWRD0 = 1
          X2 = RTX2
          If bit 12(CMOONFLG) of FLAGWRD8 = 0:
               Set bit 12(MOONFLAG) of FLAGWRD0 = 0
          $T_{et} = T_{csi}$
          $RC\underline{V} = \underline{R}_{pass1}$, shifted left X2 places  (B29 earth, B27 moon)
          $VC\underline{V} = \underline{V}_{pass1}$, shifted left X2 places  (B7 earth, B5 moon)
          Perform "INTEGRVS"
          $\underline{R}_{pass3} = \underline{R}_{att}$
          $\underline{V}_{pass3} = \underline{V}_{att}$
          Proceed to "CSI/A"        (returns to "P32/P72C" for non-error exit)
     P32/P72C
          If bit 7(HAFLAG) of FLGWRD11 = 1:     (e.g. "P32/P72B" entered from
                                                             "P31RECYC")
               Set bit 7(HAFLAG) of FLGWRD11 = 0
               Proceed to "P31RT"
          $T2TOT3 = TS + K_{600sc}$          (TS left with time from end of "CIRCL")
          If bit 6(FINALFLG) of FLAGWRD2 = 0:
               Set bit 7(UPDATFLG) of FLAGWRD1 = 1
          TS = T1TOT2
          T1TOT2 = TS
          $TS = T1TOT2 - K_{60min}$
          If TS $\geq$ 0, proceed to 2nd previous line (load T1TOT2 with TS)
          TS = T2TOT3
          T2TOT3 = TS
          $TS = T2TOT3 - K_{60min}$
          If TS $\geq$ 0, proceed to 2nd previous line (load T2TOT3 with TS)
```

```
If bit 7(AUTOSEQ) of FLGWRD10 = 0:

    $TS = 0675_{vn}$

    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line

$T\underline{S}_2 = - UP\underline{1}$                (tag here "PASS75")

$T\underline{S}_3 = -unit\underline{R}_{act1}$

$T\underline{S}_1 = T\underline{S}_3 * UP\underline{1}$

$DELVLV\underline{C} = \begin{bmatrix} T\underline{S}_1 \\ T\underline{S}_2 \\ T\underline{S}_3 \end{bmatrix} \quad DELVEET\underline{1}$

(YCSM, YDOTC, YDOTL) = - (AUTOY, CMYDOT, LMYDOT)

$DELVLVC_y = CMYDOT$

If bit 1(PCFLAG) of FLGWRD10 = 1:          (should be 0)

    If $|DELVLVC_y| - K_{1d10fps} < 0$:

        $DELVLV\underline{C} = 0$

$TS = 0690_{vn}$

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                      if proceed, proceed
                      otherwise, proceed to previous line

$TS = 0681_{vn}$

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                      if proceed, proceed
                      otherwise, proceed to previous line

$DELVSI\underline{N} = DELVLV\underline{C} \begin{bmatrix} T\underline{S}_1 \\ T\underline{S}_2 \\ T\underline{S}_3 \end{bmatrix}$      (Matrix recomputed for restart reasons,
                                              but elements should have same values)

$\underline{R}_{act1} = \underline{R}_{act2}$        (to allow use of common subroutine)

$T\underline{S}_2 = - UP\underline{1}$

$T\underline{S}_3 = -unit\underline{R}_{act1}$          (cell has $\underline{R}_{act2}$ value here)
```

```
        $TS\underline{}_1 = TS\underline{}_3 * UP\underline{1}$
        $DELVO\underline{V} = \begin{bmatrix} TS\underline{}_1 \\ TS\underline{}_2 \\ TS\underline{}_3 \end{bmatrix} \quad DELVEET\underline{2}$
        If bit 7(AUTOSEQ) of FLGWRD10 = 0:
            $TS = 0682_{vn}$
            Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                                   if proceed, proceed
                                   otherwise, proceed to previous line
        $T_{tpi0} = T_{tpi}$
        $DELVEET\underline{3} = DELVSI\underline{N}$     (same cell)
        Perform "VN1645"
        Proceed to "P32/P72B"
P33
        Set bit 5(LMACTFLG) of FLAGWRD2 = 0
        $ECSTEER = K_{lb2}$
        Proceed to second line of "P73"
P73
        Set bit 5(LMACTFLG) of FLAGWRD2 = 1
        Set bits 7(UPDATFLG) and 5(TRACKFLG) of FLAGWRD1 = 1
        Set bit 1(PCFLAG) of FLGWRD10 = 0
        $TS = 0613_{vn}$
        Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                               if proceed, proceed
                               otherwise, proceed to previous line
        $T_{tpi} = T_{tpi0}$
        $T_{ig} = T_{cdh}$
```

```
    X1 = -2          (settings for
    X2 = 0                 earth)
    If bit 12(CMOONFLG) of FLAGWRD8 = 1:
        X1 = -10         (settings for
        X2 = 2                 moon)
    RTX1 = X1
    RTSR1dMU = $K_{mutab_{4-X1}}$
    TS = $K_{mutab_{-2-X1}}$            (subscript = 0 for earth, 8 for moon)
    If bit 12(CMOONFLG) of FLAGWRD8 = 1:
        Shift TS right 6 places
    RTMU = TS        (scale factor B36)
    RTX2 = X2
    Set bit 6(FINALFLG) of FLAGWRD2 = 0
    If bit 1(PCFLAG) of FLGWRD10 = 1:          (should be 0)
        Proceed to "P36A"
    Perform "VN1645"
    Proceed to "P33/P73B"
P33/P73B
    TS = $T_{ig}$
    Perform "ADVANCE"    (page BURN-13)
    Perform "CDHMVR"
    DELVEET1 = DELVEET2        (allows use of a P32/P72 subroutine later)
    Perform "INTSTALL"
    Set bit 4(CONICINT) of FLAGWRD3 = 0
    Set bit 4(CONICINT) of FLAGWRD3 = 1     (superseding previous line)
    $T_{decl} = T_{tpi}$
    Set bit 12(MOONFLAG) of FLAGWRD0 = 1
```

```
X2 = RTX2
If bit 12(CMOONFLG) of FLAGWRD8 = 0:
    Set bit 12(MOONFLAG) of FLAGWRD0 = 0
$T_{et} = T_{cdh}$
$RC\underline{V} = \underline{R}_{act2}$, shifted left X2 places (B29 earth, B27 moon)
$VC\underline{V} = \underline{V}_{act3}$, shifted left X2 places (B7 earth, B5 moon)
Perform "INTEGRVS"
$\underline{R}_{act3} = \underline{R}_{att}$
$\underline{V}_{act3} = \underline{V}_{att}$
Perform "INTSTALL"
Set bit 4(CONICINT) of FLAGWRD3 = 0
Set bit 4(CONICINT) of FLAGWRD3 = 1    (superseding previous line)
$T_{decl} = T_{tpi}$
Set bit 12(MOONFLAG) of FLAGWRD0 = 1
X2 = RTX2
If bit 12(CMOONFLG) of FLAGWRD8 = 0:
    Set bit 12(MOONFLAG) of FLAGWRD0 = 0
$T_{et} = T_{cdh}$
$RC\underline{V} = \underline{R}_{pass2}$, shifted left X2 places    (B29 earth, B27 moon)
$VC\underline{V} = \underline{V}_{pass2}$, shifted left X2 places    (B7 earth, B5 moon)
Perform "INTEGRVS"
$\underline{R}_{pass3} = \underline{R}_{att}$
$\underline{V}_{pass3} = \underline{V}_{att}$
Set bit 14(ITSWITCH) of FLAGWRD7 = 1
NOMTPI = 0
Perform "S33/34.1"
```

```
If $TS \neq 0$:        (non-zero for error return from "S33/34.1")
     Perform "ALARM"  (pattern $0611_8$)
     $TS = 0509_{vn}$
     Proceed to "GOFLASH": if terminate, proceed to "GOTOPOOH"
                           if proceed, proceed
                           otherwise, proceed to second line of "P73"
     NOMTPI = 0
If bit 6(FINALFLG) of FLAGWRD2 = 0:        (tag here is "P33/P73C")
     Set bit 7(UPDATFLG) of FLAGWRD1 = 1
$T_{tpi} = NOMTPI + T_{tpi}$
$TS = T_{tpi} - T_{cdh}$
$TS = TS - K_{60min}$
If $TS \geqslant 0$, proceed to previous line
$T1TOT2 = TS + K_{60min}$        (note that if negative, no modulo
                                    action occurs)
$TS = T_{tpi} - T_{tpi0}$
$TS = |TS| - K_{60min}$
If $TS \geqslant 0$, proceed to previous line
$T2TOT3 = (TS + K_{60min})\ sgn\ (T_{tpi} - T_{tpi0})$
If bit 7(AUTOSEQ) of FLGWRD10 = 0:
     $TS = 0675_{vn}$
     Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                           if proceed, proceed
                           otherwise, proceed to previous line
$T\underline{S}_2 = - UP\underline{1}$        (tag here "SKIP75")
$T\underline{S}_3 = -unit\underline{R}_{actl}$
$T\underline{S}_1 = T\underline{S}_3 * UP\underline{1}$
$DELVLV\underline{C} = \begin{bmatrix} T\underline{S}_1 \\ T\underline{S}_2 \\ T\underline{S}_3 \end{bmatrix} DELVEET\underline{1}$        (DELVEET1 has DELVEET2 from start of
                                                "P33/P73B")
```

```
    (YCSM, YDOTC, YDOTL) = - (AUTOY, CMYDOT, LMYDOT)
    $DELVLVC_y$ = CMYDOT
    If bit 1(PCFLAG) of FLGWRD10 = 1:                    (should be 0)
        If $|DELVLVC_y| - K_{1d10fps} < 0$:
            $DELVLV\underline{C}$ = 0
    TS = $0690_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    TS = $0681_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    $DELVSI\underline{N} = DELVLV\underline{C} \begin{bmatrix} T\underline{S}_1 \\ T\underline{S}_2 \\ T\underline{S}_3 \end{bmatrix}$   (Matrix recomputed for restart reasons,
                                                    but elements should have same values)
    $DELVEET\underline{3} = DELVSI\underline{N}$  (same cell)
    Perform "VN1645"
    Proceed to "P33/P73B"
ADVANCE     Entered from "HARTBURN", "P32/P72B", and "P33/P73B"
    $T_{decl}$ = TS          (TS set to required time before enter)
    Perform "PRECSET"        (which loads $UP\underline{1}$)
    Set bit 8(XDELVFLG) of FLAGWRD2 = 1
    $\underline{V}_{pass2} = \underline{V}_{pass3}$
    $\underline{V}_{pass1} = \underline{V}_{pass3}$
    $\underline{R}_{pass2} = \underline{R}_{pass3}$
    $\underline{R}_{pass1} = \underline{R}_{pass3}$
    $\underline{R}_{tig} = \underline{R}_{act3}$
    $\underline{R}_{act2} = |\underline{R}_{tig}| \text{ unit } \left( \underline{R}_{tig} - (\underline{R}_{tig} \cdot UP\underline{1})\, UP\underline{1} \right)$
    $\underline{R}_{act1} = \underline{R}_{act2}$
```

```
    $\underline{V}_{tig} = \underline{V}_{act3}$
    $\underline{V}_{act2} = |\underline{V}_{tig}| \text{ unit}\left( \underline{V}_{tig} - (\underline{V}_{tig} \cdot \underline{UP1})\, \underline{UP1}\right)$
    $\underline{V}_{act1} = \underline{V}_{act2}$
    Return

P34

    Set bit 5(LMACTFLG) of FLAGWRD2 = 0
    ECSTEER = $K_{1b2}$
    Proceed to second line of "P74"

P74

    Set bit 5(LMACTFLG) of FLAGWRD2 = 1
    Set bits 7(UPDATFLG) and 5(TRACKFLG) of FLAGWRD1 = 1
    Set bit 1(PCFLAG) of FLGWRD10 = 0
    Set bit 3(TPIMNFLG) of FLGWRD10 = 0
    CENTANG = $K_{130deg}$
    ELEV = 0
    NN1 = 0
    TS = $0637_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    TS = $0655_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    Set bit 7(HAVEELEV) of FLAGWRD2 = 0
    $T_{ig} = T_{tpi}$
    If ELEV $\neq$ 0:
        Set bit 7(HAVEELEV) of FLAGWRD2 = 1
```

```
    X1 = -2            (settings for
    X2 = 0                  earth)
    If bit 12(CMOONFLG) of FLAGWRD8 = 1:
        X1 = -10       (settings for
        X2 = 2                moon)
    RTX1 = X1
    RTSR1dMU = $K_{mutab_{4-X1}}$
    TS = $K_{mutab_{-2-X1}}$        ($K_{mutab_0}$ for earth, $K_{mutab_8}$ for moon)
    If bit 12(CMOONFLG) of FLAGWRD8 = 1:
        Shift TS right 6 places
    RTMU = TS      (scale factor B36)
    RTX2 = X2
    Set bit 6(FINALFLG) of FLAGWRD2 = 0
    If bit 1(PCFLAG) of FLGWRD10 = 1:        (should be 0)
        Proceed to "P36A"
    Perform "VN1645"
    Proceed to "P34/P74C"
P34/P74C
    Set bit 14(ITSWITCH) of FLAGWRD7 = 1      (means iterate "S33/34.1" below)
    If bit 7(HAVEELEV) of FLAGWRD2 = 0:
        Set bit 14(ITSWITCH) of FLAGWRD7 = 0
    NOMTPI = 0
    $T_{decl} = T_{tpi} + NOMTPI$      (tag here "INTLOOP", enter here from below)
    Perform "PRECSET"
    Perform "S33/34.1"
```

```
If TS $\neq$ 0:        (non-zero for error return from "S33/34.1")

    Perform "ALARM" (pattern $0611_8$)

    TS = $0509_{vn}$

    Proceed to "GOFLASH": if terminate, proceed to "GOTOPOOH"
                          if proceed, proceed
                          otherwise, proceed to 2nd previous line
                            (regenerate alarm again)

    Proceed to second line of "P74"

If bit 14(ITSWITCH) of FLAGWRD7 = 1:        (tag here "SWCHCLR")

    Set bit 14(ITSWITCH) of FLAGWRD7 = 0

    Proceed to 5th line of "P34/P74C" ($T_{decl}$ loading)

If bit 7(HAVEELEV) of FLAGWRD2 = 1:

    TS = $0637_{vn}$        (tag here "P34/P74D")

    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line

    If bit 6(FINALFLG) of FLAGWRD2 = 1:

        If bit 7(AUTOSEQ) of FLGWRD10 = 1:

            Set bit 7(HAVEELEV) of FLAGWRD2 = 0

            ELEV = 0

            $T_{ig} = T_{tpi}$
            Proceed to "P34/P74C"

If bit 7(HAVEELEV) of FLAGWRD2 = 0:

    TS = $0655_{vn}$

    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line

X1 = RTX1          (Tag here "P34/74E")

X2 = RTX2          (as "least significant half" of X1 and RTX1)

CSTH = cos CENTANG

SNTH = sin CENTANG
```

```
$RVE\underline{C} = \underline{R}_{pass3}$, shifted left X2 places (B29 earth, B27 moon)
$VVE\underline{C} = \underline{V}_{pass3}$, shifted left X2 places (B7 earth, B5 moon)
Set bit 9(RVSW) of FLAGWRD7 = 1      (means new $\underline{R}$, $\underline{V}$ not desired)
Perform "TIMETHET"        (leaves time in T)
$INTIME = T_{tpi}$
$T_{pass4} = T_{tpi} + T$
Perform "S34/35.2"
$DELVTPI = |DELVEET\underline{3}|$
$DELVTPF = |\underline{V}_{pass4} - \underline{V}_{tprime}|$
X2 = RTX2
$VVE\underline{C} = \underline{V}_{iprime}$, shifted left X2 places (B7 earth, B5 moon)
$RVE\underline{C} = \underline{R}_{act3}$, shifted left X2 places (B29 earth, B27 moon)
X1 = RTX1
$TS = K_{rpad}$
If $X1 \neq -2$:       (i.e. not earth)
    $TS = |RL\underline{S}|$
XXXALT = TS
Perform "APSIDES"
$POSTTPI = (TS_{rp} - XXXALT)$, shifted right RTX2 places (B29)
$T_{ig} = T_{tpi}$
$TS = 0658_{vn}$
Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                      if proceed, proceed
                      otherwise, proceed to previous line
Perform "S34/35.5"      (page BURN-19)
Perform "VN1645"
Proceed to "P34/P74C"
```

VN1645

```
SUBEXIT = Return address
pMGA = $K_{dpm01}$
If bit 6(FINALFLG) of FLAGWRD2 = 1:
    pMGA = 2 $K_{dpm01}$
    If bit 13(REFSMFLG) of FLAGWRD3 = 1:
        If MODREG $<$ 64:
            $T\underline{S}$ = DELVEET$\underline{3}$      (same cell as DELVSI$\underline{N}$)
            Perform "GET+MGA"          (If $T\underline{S}$ = 0, may get $1301_8$
                                        alarm: possible e.g. P38)
Set bit 11(TIMRFLAG) of FLAGWRD7 = 1
NVWORD1 = +0
Call "CLOKTASK" in 0.01 seconds
Delay 1 second (by putting job to sleep via "DELAYJOB")
TS = $1645_{vn}$
Proceed to "GOFLASH": if terminate, proceed to "GOTOPOOH"
                      if proceed, proceed
                      otherwise, proceed to "CLUPDATE"
If bit 6(FINALFLG) of FLAGWRD2 = 0:        (tag here "N45PROC")
    Set bit 6(FINALFLG) of FLAGWRD2 = 1
    Proceed to "CLUPDATE"
NOMTIG = $T_{ig}$                        (tag here "N45ENAJ")
If bit 1(PCFLAG) of FLGWRD10 = 0:
    Set bit 5(MANEUFLG) of FLGWRD10 = 1
If bit 7(AUTOSEQ) of FLGWRD10 = 1:
    WRENDPOS = $K_{posvel2}$
    WRENDVEL = $K_{posvel3}$
Proceed to second line of "GOTOPOOH"
```

```
CLUPDATE
    Set bit 11(TIMRFLAG) of FLAGWRD7 = 0
    CADRFLSH+2 = "VN1645R"  +  3
    Set restart group 4 to phase 1 (4.1, causing "INITDSP" to enter
        "VN1645R"if restart)
    Proceed to "VN1645R"
VN1645R
    Set bit 7(UPDATFLG) of FLAGWRD1 = 0
    Proceed to address specified by SUBEXIT
S34/35.5          Entered from "P34/P74C" and "P35/P75B"
    If bit 6(FINALFLG) of FLAGWRD2 = 0:
        Set bit 7(UPDATFLG) of FLAGWRD1 = 1
    If bit 6(FINALFLG) of FLAGWRD2 = 1:
        Set bit 3(TPIMNFLG) of FLGWRD10 = 1
    $\underline{V}_{tprime} = DELVLV\underline{C}$          (temporary storage)
    $TS = 0681_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    If $|DELVLV\underline{C} - \underline{V}_{tprime}| \neq 0$:     (at least one component change by
                                                 $\geq 2^{-14}$ meter/cs $\cong$ 0.02 fps)
        Perform "S34/35.3"
    $T\underline{S}_1 = ULO\underline{S}$
    $T\underline{S}_3 = -\ \mathrm{unit}(ULO\underline{S} * UNR\underline{M})$
    $T\underline{S}_2 = T\underline{S}_3 * ULO\underline{S}$
    $DVLO\underline{S} = \begin{bmatrix} T\underline{S}_1 \\ T\underline{S}_2 \\ T\underline{S}_3 \end{bmatrix} \quad DELVEET\underline{3}$
    Return
```

P35

```
Set bit 5(LMACTFLG) of FLAGWRD2 = 0
ECSTEER = $K_{lb2}$
KT = $C_{atiginc}$
Proceed to third line of "P75"
```

P75

```
Set bit 5(LMACTFLG) of FLAGWRD2 = 1
KT = $C_{ptiginc}$
Set bits 7(UPDATFLG) and 5(TRACKFLG) of FLAGWRD1 = 1
Set bit 1(PCFLAG) of FLGWRD10 = 0
X1 = -2          (settings for
X2 = 0                earth)
If bit 12(CMOONFLG) of FLAGWRD8 = 1:
    X1 = -10         (settings for
    X2 = 2                moon)
RTX1 = X1
RTSR1dMU = $K_{mutab_{4-X1}}$
TS = $K_{mutab_{-2-X1}}$                (subscript = 0 for earth, 8 for moon)
If bit 12(CMOONFLG) of FLAGWRD8 = 1:
    Shift TS right 6 places
RTMU = TS       (scale factor B36)
RTX2 = X2
Set bit 6(FINALFLG) of FLAGWRD2 = 0
If bit 1(PCFLAG) of FLGWRD10 = 1:        (should be 0)
    Proceed to "P36A"
Perform "VN1645"
Proceed to "P35/P75B"
```

P35/P75B

```
$T_{strt} = T_{now}$
$T_{ig} = T_{strt} + KT$
$INTIME = T_{ig}$
$T_{decl} = T_{ig}$
Perform "PRECSET"
$ULO\underline{S} = unit(\underline{R}_{pass3} - \underline{R}_{act3})$
$UNR\underline{M} = unit(\underline{R}_{act3} * \underline{V}_{act3})$
Perform "S34/35.2"
Perform "S34/35.5"  (page BURN-19)
If bit 6(FINALFLG) of FLAGWRD2 = 1:
    Set bit 8(P35FLAG) of FLGWRD10 = 1
Perform "VN1645"
Proceed to "P35/P75B"
```

P36

```
Set bit 5(LMACTFLG) of FLAGWRD2 = 0
$ECSTEER = K_{lb2}$
Set bits 7(UPDATFLG) and 5(TRACKFLG) of FLAGWRD1 = 1
Set bit 1(PCFLAG) of FLGWRD10 = 0
Set bit 1(PCFLAG) of FLGWRD10 = 1     (superseding previous line)
X1 = -2       (settings for
X2 = 0              earth)
If bit 12(CMOONFLG) of FLAGWRD8 = 1:
    X1 = -10      (settings for
    X2 = 2              moon)
RTX1 = X1
$RTSR1dMU = K_{mutab_{4-X1}}$
$TS = K_{mutab_{-2-X1}}$        (subscript = 0 for earth, 8 for moon)
```

```
    If bit 12(CMOONFLG) of FLAGWRD8 = 1:
        Shift TS right 6 places
    RTMU = TS      (scale factor B36)
    RTX2 = X2
    Set bit 6(FINALFLG) of FLAGWRD2 = 0
    If bit 1(PCFLAG) of FLGWRD10 = 1:     (should be)
        Proceed to "P36A"
    Perform "VN1645"
    Proceed to "P36A"
P36A
    Set bit 8(XDELVFLG) of FLAGWRD2 = 1
    $T_{decl} = T_{csi}$
    Perform "CSMCONIC"
    Set bit 9(RVSW) of FLAGWRD7 = 1     (means new $\underline{R}$, $\underline{V}$ not desired)
    CSTH = 0
    SNTH = $K_{cs359p}$       (i.e. angle of about $90^o$)
    X2 = RTX2
    $VVE\underline{C} = \underline{V}_{att}$, shifted left X2 places   (B7 earth, B5 moon)  (could use $\underline{V}_{attl}$
    X1 = RTX1
    $RVE\underline{C} = \underline{R}_{att}$, shifted left X2 places   (B29 earth, B27 moon) and $\underline{R}_{attl}$)
    Perform "TIMETHET"
                                    (although update flag set, note that P20
    $T_{ig} = T_{csi} + T$                restart logic such that P20 not able to run
                                    after the V37 logic until get the display)
    TS = $0633_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    Perform "VN1645"
    Proceed to "P36RECYC"
```

```
P36RECYC

    DELVLV$\underline{C}$ = 0
    Perform "N90/N81"
    Perform "VN1645"
    Proceed to "P36RECYC"

N90/N81                Entered from "P31RT" and "P36RECYC"

    $T_{decl} = T_{ig}$
    Perform "PRECSET"
    If bit 6(FINALFLG) of FLAGWRD2 = 0:
        Set bit 7(UPDATFLG) of FLAGWRD1 = 1
    $\underline{V}_{tig} = \underline{V}_{act3}$
    $\underline{R}_{tig} = \underline{R}_{act3}$
    (YCSM, YDOTC, YDOTL) = - (AUTOY, CMYDOT, LMYDOT)
    $\text{DELVLVC}_y$ = CMYDOT
    If bit 1(PCFLAG) of FLGWRD10 = 1:
        If $|\text{DELVLVC}_y| - K_{1d10fps} < 0$:
            DELVLV$\underline{C}$ = 0        (may give $1301_8$ alarm at N45 display)
    TS = $0690_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    TS = $0681_{vn}$
    Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
                          if proceed, proceed
                          otherwise, proceed to previous line
    $T\underline{S}_2$ = - UNR$\underline{M}$
    $T\underline{S}_3$ = - unit$\underline{R}_{act3}$
    $T\underline{S}_1 = T\underline{S}_3$ * UNR$\underline{M}$
```

```
    $DELVSI\underline{N} = DELVLV\underline{C} \begin{bmatrix} T\underline{S}_1 \\ T\underline{S}_2 \\ T\underline{S}_3 \end{bmatrix}$

    $DELVEET\underline{3} = DELVSI\underline{N}$          (same cell)

    Return

P40CSM

    Set bit 11(RCSBURN) of FLAGWRD1 = 0

    TS = (ECSTEER, 0)       (most significant half ECSTEER, least sig. 0)

    If bit 8(XDELVFLG) of FLAGWRD2 = 1:

        TS = 0

    CSTEER = TS

    CAPF = $K_{feng}$

    Proceed to "P40S/F"

P40S/F          Entered from "P40CSM" and "P41CSM"

    NOMTIG = $T_{ig}$

    AVEGDT = $K_{sec29p96}$

    Set bit 10(BURNFLAG) of FLGWRD10 = 0

    Set bit 11(TIMRFLAG) of FLAGWRD7 = 1

    NVWORD1 = +0

    Call "CLOKTASK" in 0.01 second

    Perform "R02BOTH"

    Proceed to "S40.1"      (which exits to "S40.2,3", in turn to "P40SXTY")

P40SXTY

    Set bit 4(PFRATFLG) of FLAGWRD2 = 1

    Inhibit interrupts

    Perform "SETMINDB"

    Set bit 6(3AXISFLG) of FLAGWRD5 = 0
```

```
Release interrupts

Perform "R60CSM"

NBRCYCLS = -1    (initialization for "UPDATEVG" for Lambert burn)

If bit 11(RCSBURN) of FLAGWRD1 = 1:

    $\underline{TS} = \underline{V}_{gtig}$                    (Tag here "P41/DSP")

    Perform "S41.1"

    $\underline{V}_{gbody} = \underline{TS}$        (also updated by "DYNDISP" in "CLOKTASK" logic)

    AVEGEXIT = "CALCN85"

    NVWORD1 = $0685_{vn}$

    Proceed to "P40S/SV"

VGDISP = $|\underline{V}_{gtig}|$

DVTOTAL = 0

AVEGEXIT = "S40.8"

TS = $0204_8$

Proceed to "GOPERF1": if terminate, proceed to "POST41"
                      if proceed, MRKRTMP = +1 and proceed (do test)
                      otherwise, MRKRTMP = -1 and proceed

CNTR = +0             (used as flag in "S40.6")

Call "S40.6" in 0.01 second

TS = 5 seconds

If MRKRTMP $>$ 0:

    TS = 18 seconds

Delay TS seconds (by putting job to sleep via "DELAYJOB")

NVWORD1 = $0640_{vn}$        (causes "CLOCKJOB" execution)

Set restart group 6 to phase 2 (i.e. 6.2, causing "PRE40.6" to be
                                 called if a restart)

Proceed to "P40S/SV"
```

```
P40S/SV        Entered from "P40SXTY"; established by "T6RESET" for P15

    $T_{decl} = T_{ig}$ - AVEGDT      (AVEGDT = 29.96 sec for P40/P41; 100 sec for P15)

    Perform "MIDTOAV1": if return to calling address +1, skip next line
                        if return to calling address +2 (time slipped),
                         proceed

    $T_{ig} = T_{pptm1} + K_{sec29p96}$      (not expected to be executed for P15)

    P40TMP = TS - $K_{5secdp}$      (TS set by "MIDTOAV1")

    Call "TIGBLNK" in P40TMP seconds

    End of job

TIGBLNK        Entered 5 seconds before start of Average-G

    Call "TIGAVEG" in 5 seconds

    NVWORD1 = +0

    Establish "P40BLNKR"  (priority $14_8$)

    End of task

P40BLNKR

    Perform "CLEANDSP"  (blanks DSKY except for program number)

    End of job

TIGAVEG        Entered when Average-G should start: 29.96 sec before $T_{ig}$
               for P40/P41; 100 sec before $T_{ig}$ for P15.

    If bit 6 of MODREG = 0:    (i.e. not in range 32-63, hence e.g. P15)

        NVWORD1 = $0695_{vn}$      (Tag here "P15AVEG")

        Proceed to "PREREAD"

    Set bit 4(PFRATFLG) of FLAGWRD2 = 0

    If bit 11(RCSBURN) of FLAGWRD1 = 1:

        NVWORD1 = $1685_{vn}$            (Tag here "P41REDSP")

        Call "TTG/0" in 29.96 seconds

        Proceed to "PREREAD"
```

```
    NVWORD1 = $0640_{vn}$      (P40)
    Call "TIG-5" in 24.96 seconds (i.e. at exactly 5 seconds before $T_{ig}$)
    Proceed to "PREREAD"
TIG-5
    Call "TIG-0" in 5 seconds
    NVWORD1 = $-400_8$      (causes "NV50DSP" to paste a V99)
    Establish "S40.13"  (priority $20_8$)
    End of task
TIG-0
    Set bit 13(IGNFLAG) of FLAGWRD7 = 1
    If bit 12(ASTNFLAG) of FLAGWRD7 = 0, End of task (no authorization
                                                       received yet)
    NVWORD1 = $0640_{vn}$
    Proceed to "IGNITION"
IGNITION      Entered from "TIG-0" and called by "V99P"
    OGAD = $CDU_x$
    $T_{evt} = T_{now}$
    Set bit 7(ENGONFLG) of FLAGWRD5 = 1    (engine commanded on)
    Set bit 13(SPS Engine On) of channel 11 = 1
    Set bit 10(BURNFLAG) of FLGWRD10 = 1    (also set 1 in "SERVXT1")
    If bit 9(IMPULSW) of FLAGWRD2 = 0:
        Set bit 13(STRULLSW) of FLAGWRD6 = 1    (enable steering below)
    If bit 9(IMPULSW) of FLAGWRD2 = 1:
        Set bit 13(STRULLSW) of FLAGWRD6 = 0     (Tag here "IMPLBURN")
        $T_{ig} = T_{now} + T_{go}$      (for "CLOKTASK")
        Call "ENGINOFF" in $(T_{go}+1)_{sp}$ centi-seconds (uses low half of $T_{go}$
                                                  loaded by "S40.13")
        Set bit 9(IMPULSW) of FLAGWRD2 = 0
    Set bits 15-14(DAPBIT1, DAPBIT2) of FLAGWRD6 = $00_2$  (Tag here "PREPTVC")
```

```
    T5LOC = "T5IDLOC"      (turn off RCS DAP except for jet timing)
    Delay 0.40 seconds (for thrust buildup)
    TVCPHASE = -1
    TVCEXPHS = +0
    Set bits 15-14(DAPBIT1, DAPBIT2) of FLAGWRD6 = $10_2$
    Set restart group 6 to phase 3 (6.3, removing "PRE40.6" protection)
    ERRBTMP = $-(\mathrm{ERROR}_1, \mathrm{ERROR}_2)$
    Set TIME5 to cause program interrupt #2 in 0.01 seconds
    T5LOC = "TVCDAPON"
    Delay 1.6 seconds (i.e. until 2 seconds from ignition)
    If bit 13(STRULLSW) of FLAGWRD6 = 1:
        Set bit 11(STEERSW) of FLAGWRD2 = 1
    Channel 5 = 0      (turn off ullage jets)
    End of task
CLOKTASK
    If bit 11(TIMRFLAG) of FLAGWRD7 = 0, End of task
    $T_{togo} = T_{now} - T_{ig}$
    Call "CLOKTASK" in 1 second
    If NVWORD1 = +0, End of task
    If  NVWORD1 $\neq 0685_{vn}$:      (i.e. not P41 before start of Average-G)
        Establish "CLOCKJOB" (priority $27_8$)
        End of task
    Establish "DYNDISP" (priority $27_8$)
    End of task
DYNDISP
    $\underline{TS} = \underline{V}_{gtig}$
    Perform "S41.1"
    $\underline{V}_{gbody} = \underline{TS}$
```

```
    Proceed to 3rd line of "CLOCKJOB"

CLOCKJOB

    CDUSPO$\underline{T}$ = CD$\underline{U}$    (most significant halves loaded, in y,z,x order)

    Perform "QUICTRIG"

    Inhibit interrupts (released e.g. as part of End of job)

    If NVWORD1 = +0, End of job

    If NVWORD1 > 0:

        TS = NVWORD1

        Proceed to "REGODSP"

    If  NVWORD1 < 0:        (e.g. $-400_8$, for V99 generation)

        TS = $0640_{vn}$

        Proceed to "CLOCPLAY": if terminate, proceed to "V99T"
                               if proceed, proceed to "V99P"
                               otherwise, proceed to "V99E"

    If  NVWORD1 = -0:       (as it will, for V97 generation)

        TS = $0640_{vn}$

        Proceed to "CLOCPLAY": if terminate, proceed to "V97T"
                               if proceed, proceed to "V97P"
                               otherwise, proceed to "V97E"

V97E        Entered for an ENTR response to V97, engine fail

    NVWORD1 = $0640_{vn}$

    $T_{ig_{sp}}$ = -3276.8 seconds  (most significant half set to -20, to give
                                    59 59 display in "CLOCKJOB" for $T_{now} > \sim 9$ min)

    Perform "SPSOFF"

    Delay 2.5 seconds (by putting job to sleep via "DELAYJOB")

    Call "PRE40.6" in 0.01 seconds

    Inhibit interrupts      (done in waitlist routine)

    Perform "RCSDAPON"

    Release interrupts
```

Set bit 12(ASTNFLAG) of FLAGWRD7 = 0

Set bit 13(IGNFLAG) of FLAGWRD7 = 1

NVWORD1 = - 400_8 (gives a V99 flash, cf. "TIG-5")

Establish "S40.13" (priority 20_8)

End of job

<u>V97P</u> Entered for a PRO response to V97, engine fail

NVWORD1 = 0640_{vn}

Set bit 6(IDLEFAIL) of FLAGWRD1 = 1 (keeps "S40.8" fail logic from being triggered)

Set bit 11(STEERSW) of FLAGWRD2 = 1

Delay 2 seconds (by putting job to sleep via "DELAYJOB")

Set bit 6(IDLEFAIL) of FLAGWRD1 = 0

End of job

<u>V97T</u> Entered for a V34E response to V97, engine fail

NVWORD1 = +0

Perform "SPSOFF"

Delay 2.5 seconds (by putting job to sleep via "DELAYJOB")

Inhibit interrupts

Perform "RCSDAPON"

Release interrupts

Proceed to "V99T"

<u>V99E</u> Entered for an ENTR response to V99, engine ignition

Inhibit interrupts

Perform "TVCZAP"

Proceed to "P40RCS"

<u>V99P</u> Entered for a PRO response to V99, engine ignition

Inhibit interrupts (released as part of End of job)

```
    If bit 12(ASTNFLAG) of FLAGWRD7 = 0:     (check for restart reasons)
        Set bit 12(ASTNFLAG) of FLAGWRD7 = 1
        If bit 13(IGNFLAG) of FLAGWRD7 = 1:     ("TIG-0" has been done)
            Call "IGNITION" in 0.01 second
    NVWORD1 = $0640_{vn}$
    End of job
V99T     Entered for a V34E response to V99 (and from "V97T")
    Inhibit interrupts
    Perform "TVCZAP"
    Proceed to "POST41"
S40.81   Entered from "S40.8" when time-to-go below 4 seconds
    OMEGA$\underline{C}$ = 0
    CNTR = +MAX
    Inhibit interrupts
    TS = $(T_{ig} - T_{now})$, with sign agreement forced
    $T_{go}+1 = TS+1_{sp}$, limited $\geq$ 1 centi-second (most significant half of $T_{go}$
                                                          not loaded)
    Call "ENGINOFF" in $(T_{go}+1)$ centi-seconds
    Set bit 11(STEERSW) of FLAGWRD2 = 0
    Release interrupts
    Proceed to "SERVEXIT"
ENGINOFF     Called by "IGNITION" and "S40.81"
    Perform "SPSOFF"
    Delay 2.5 seconds
    Perform "SETMAXDB"
    Perform "RCSDAPON"
    Perform "MASSPROP"
    Perform "TVCZAP"
    Establish "POSTBURN" (priority $12_8$)
    End of task
```

```
POSTBURN

    TS = $1640_{vn}$

    Proceed to "REFLASH": if terminate, proceed to "POST41"
                          if proceed, proceed
                          otherwise, proceed to previous line

    Proceed to "P40RCS"
P40RCS

    AVEGEXIT = "CALCN85"

    Delay 2 seconds (by putting job to sleep via "DELAYJOB")

    Inhibit interrupts

    Perform "SETMINDB"

    Release interrupts

    Proceed to "TIGNOW"

TIGNOW      (Entered from "P40RCS" and established by "TTG/0" for P41)

    TS = $1685_{vn}$

    Proceed to "REFLASH": if terminate, proceed to "POST41"
                          if proceed, proceed
                          otherwise, proceed to previous line

    AVEGEXIT = "SERVEXIT"

    Proceed to second line of "GOTOPOOH"

POST41

    AVEGEXIT = "SERVEXIT"

    Proceed to "GOTOPOOH"

P41CSM

    Set bit 11(RCSBURN) of FLAGWRD1 = 1

    CSTEER = 0

    TS = $K_{frcs2}$

    If bit 15(2JETSFLG) of FLAGWRD1 = 0:      (i.e. 4 jets)

        TS = 2 TS

    CAPF = TS

    Proceed to "P40S/F"
```

```
TTG/0     Called by "TIGAVEG" for P41
    Establish "TIGNOW"  (priority $20_8$)
    Set bit 11(TIMRFLAG) of FLAGWRD7 = 0
    End of task
P47CSM
    Perform "R02BOTH"
    Perform "MIDTOAV2"
    $P40TMP = TS+1_{sp}$      (TS set when exit from "MIDTOAV2")
    Call "TIGON" in P40TMP centi-seconds
    End of job
TIGON
    AVEGEXIT = "CALCN83"        (writes over R1 and R2 of N78)
    Establish "P47BODY"  (priority $30_8$)
    Proceed to "PREREAD"
P47BODY
    $\underline{DELVIMU} = 0$
    $\underline{V}_{gbody} = 0$
    Change priority of present job to $15_8$
    $TS = 1683_{vn}$
    Proceed to "GOFLASH": if terminate, proceed to "GOTOPOOH"
                          if proceed, proceed
                          otherwise, proceed to "P47BODY"
    Proceed to "GOTOPOOH"
P76ER77
    OPTFLAG = (bit 1 of MODREG)      (0 for P76, 1 for P77)
    $T_{ig} = NOMTIG$
    $TS = 0633_{vn}$
```

```
Proceed to "GOFLASH": if terminate, proceed to "ENDP76"
                      if proceed, proceed
                      otherwise, proceed to previous line

If OPTFLAG = 0:       (P76)

    $TS = 0684_{vn}$

If OPTFLAG = 1:       (P77)

    $TS = 0681_{vn}$

Proceed to "GOFLASH": if terminate, proceed to "ENDP76"
                      if proceed, proceed
                      otherwise, proceed to 4th previous line

Set bit 1(NODOV37) of FLAGWRD2 = 1

$T_{decl} = T_{ig}$

If OPTFLAG = 0:

    Perform "LEMPREC"

If OPTFLAG = 1:

    Perform "CSMPREC"

$T\underline{S}_3 = -\ \text{unit}\underline{R}_{att}$

$T\underline{S}_2 = \text{unit}(T\underline{S}_3 * \underline{V}_{att})$

$T\underline{S}_1 = \text{unit}(T\underline{S}_2 * T\underline{S}_3)$

If OPTFLAG = 0:

    $T\underline{S} = DELVO\underline{V}$

If OPTFLAG $\neq$ 0:

    $T\underline{S} = DELVLV\underline{C}$

$T\underline{S}_4 = \underline{V}_{att} + T\underline{S} \begin{bmatrix} T\underline{S}_1 \\ T\underline{S}_2 \\ T\underline{S}_3 \end{bmatrix}$          (in push-down list location 6D)

Perform "INTSTALL"

Set bit 12(MOONFLAG) of FLAGWRD0 = 0
```

```
If X2 $\neq$ 0:          (means integration output moon-centered)
    Set bit 12(MOONFLAG) of FLAGWRDO = 1
$VC\underline{V} = T\underline{S}_4$, shifted left X2 places  (B7 earth, B5 moon)
$RC\underline{V} = \underline{R}_{att}$, shifted left X2 places  (B29 earth, B27 moon)
$T_{et} = T_{ig}$
Set bit 4(CONICINT) of FLAGWRD3 = 0
$T_{decl} = T_{etcm}$
Perform "INTEGRVS"
Perform "INTSTALL"
Set bit 12(MOONFLAG) of FLAGWRDO = 0
If X2 $\neq$ 0:      (means "INTEGRVS" output was moon-centered)
    Set bit 12(MOONFLAG) of FLAGWRDO = 1
$\underline{R}_{rect} = \underline{R}_{attl}$          (scaled B29 earth, B27 moon)
$RC\underline{V} = \underline{R}_{rect}$
$T_{et} = T_{att}$
$\underline{V}_{rect} = \underline{V}_{attl}$        (scaled B7 earth, B5 moon)
$VC\underline{V} = \underline{V}_{rect}$
$TDELTA\underline{V} = 0$
$TNU\underline{V} = 0$
$T_c = 0$
XKEP = 0        (notation also "XPREV")
Set bit 13(INTGRAB) of FLGWRD10 = 1
If OPTFLAG = 0:          (i.e. P76)
    Perform "MOVEALEM"
    Set bit 11(LMOONFLG) of FLAGWRD8 = 1
    $\underline{R}_{other} = RC\underline{V} + TDELTA\underline{V}$        (X2 used to determine
    $\underline{V}_{other} = VC\underline{V} + TNU\underline{V}$                   necessary shifts)
```

```
  (If OPTFLAG = 0):
      If bit 12(MOONFLAG) of FLAGWRD0 = 0:
          Set bit 11(LMOONFLG) of FLAGWRD8 = 0
  If OPTFLAG = 1:      (i.e. P77)
      Perform "MOVEACSM"
      Set bit 12(CMOONFLG) of FLAGWRD8 = 1
      If bit 1(AVEMIDSW) of FLAGWRD9 = 0:
          $\underline{R} = RC\underline{V} + TDELTA\underline{V}$      (X2 used to determine
          $\underline{V} = VC\underline{V} + TNU\underline{V}$             necessary shifts)
          $T_{pptm} = T_{et}$
      If bit 12(MOONFLAG) of FLAGWRD0 = 0:
          Set bit 12(CMOONFLG) of FLAGWRD8 = 0
  QPRET = Return address (to line after next)
  Perform "INTWAKE" (starting at 3rd from last line, awaken jobs)
  Set bit 1(NODOV37) of FLAGWRD2 = 0      (Tag here "OUT")
  MRKBUF1 = -1
  Proceed to second line of "GOTOPOOH"
ENDP76
  MRKBUF1 = -1
  Proceed to "GOTOPOOH"
```

Quantities in Computations

See also list of major variables and list of routines

AUTOY: Value of out-of-plane position computed in "PRECSET", scale factor B29, units meters. It is computed as $\underline{R}_{act3} \cdot \text{UP}\underline{1}$.

AVEGDT: Required time before T_{ig} when Average-G is to be turned on, scale factor B28, units centi-seconds. It is 29.96 seconds for P40/P41 and 100 seconds for P15.

AVEGEXIT: See General Program Control.

$C_{atiginc}$: Erasable memory constant, program notation "ATIGINC", scale factor B28, units centi-seconds, used to set KT for P35 (i.e. CSM-active time delay until midcourse maneuver).

$C_{hamdelh}$: Erasable memory constant, program notation "HAMDELH", scale factor B29, units meters, giving the required altitude difference for the height-adjustment maneuver targeting in P31.

$C_{ptiginc}$: Erasable memory constant, program notation "PTIGINC", scale factor B28, units centi-seconds, used to set KT for P75 (i.e. LM-active time delay until midcourse maneuver).

CADRFLSH+2: See Display Interface Routines.

CAPF: Nominal value of engine thrust (SPS, 2-jet RCS, or 4-jet RCS), scale factor B7, units "M. Newtons", where the unit is equivalent to 10^4 Newtons (or the same as expressing force in kilogram-meters/centi-second^2). Program notation is "F".

CDUSPO$\underline{T}$: See Coordinate Transformations.

CENTANG: Value of desired orbital central angle of the passive vehicle from TPI time to intercept (input via R3 of N55), scale factor B0, units revolutions. Used in P31/P32/P72 as a program control quantity.

CMYDOT: Value of active vehicle out-of-plane velocity computed in "PRECSET", scale factor B7, units meters/centi-second. It is computed as $\underline{V}_{act3} \cdot \text{UP}\underline{1}$.

CNTR: See Digital Autopilot Interface Routines. A setting of +MAX (in "S40.81" for cutoff) corresponds to a time delay of about 2.27 hours, effectively disabling updates in "TVCEXEC".

CSTEER: See Steering Computations.

CSTH: See Conic Routines.

DELVEET$\underline{1}$, DELVEET$\underline{2}$, DELVEET$\underline{3}$: See Rendezvous Computations.

DELVIM$\underline{U}$: See Display Computations.

DELVLVC: Value of velocity increment for display in local vertical coordinates, scale factor B7, units meters/centi-second, displayed by N81 and N86. Program notation also "DELVSLV", and cells can also be used for temporary storage purposes before the display. The z-axis is along -r at ignition, the y-axis is along v * r, and the x-axis is along y * z.

DELVOV: Velocity change information for LM loaded by N84 in P76, scale factor B7, units meters/centi-second. Also used to contain CDH velocity change in local vertical coordinates for possible display via N82 in P32/P72.

DELVSIN: Value of velocity increment in reference inertial coordinates computed for external Delta-V burns (P30, P31, P32/P72, P33/P73, and P36), scale factor B7, units meters/centi-second. Same cell also has the tag DELVEET3: for compatibility with "VN1645", however, the "loading" of DELVEET3 with DELVSIN is shown where necessary. DELVSIN is used in "S40.1" to compute the required velocity-to-be-gained for the external Delta-V burn.

DELVTPF: Value of magnitude of velocity change required at intercept, scale factor B7, units meters/centi-second.

DELVTPI: Value of magnitude of velocity change required to perform the (non-midcourse) burn in P34/P74, scale factor B7, units meters/centi-second. It is the magnitude of DELVEET3.

DIFFALT: See Rendezvous Computations.

DVLOS: Value of required velocity increment in line-of-sight coordinates, scale factor B7, units meters/centi-second, computed for optional display by N59. The R1 component is in the direction of ULOS (i.e. "forward/aft", with + direction "forward"); the R2 component is along R3 * ULOS (i.e. "right/left", with + direction "right"); and the R3 component is along UNRM * ULOS (i.e. "up/down" with + direction "down"). These are all as seen by the active vehicle.

DVTOTAL: See General Program Control.

ECSTEER: Single precision cell used to initialize CSTEER in P40 if a Lambert burn is required, scale factor B2. It is set to 1 at the start of P31, P32, P33, P34, P35, and P36; it is set to 0.5 at the start of P37.

ELEV: See Rendezvous Computations.

ERRBTMP: See Digital Autopilot TVC Routines.

$ERROR_1$, $ERROR_2$: See Digital Autopilot RCS Routines.

HAPO, HPER: Apocenter and pericenter altitudes above K_{rpad} or $|RLS|$, scale factor B29, units meters, limited in P30 to K_{maxnm}.

INTIME: See Rendezvous Computations.

K_{130deg}: Constant, program notation "130DEG", scale factor B0, units revolutions. Value is 13434_8 16163_8, corresponding to about 0.36111112 revolutions or 130^o.

$K_{130deglo}$: Constant, program notation "130DEGLO", scale factor B0, units revolutions. Value is 13434_8 16163_8, the same as K_{130deg} (in the "low" half of memory for addressing considerations).

K_{1b2}: Single precision constant, program notation "BIT13", scale factor B2, value 1×2^{-2}, corresponding to an equation value of 1.0.

$K_{1d10fps}$: Constant, program notation "1/10FPS", scale factor B7, units meters/centi-second. It is equated by assembler techniques to $K_{epsilnl}$ (see Rendezvous Computations), a value of 0.1 fps.

$K_{208deglo}$: Constant, program notation "208DEGLO", scale factor B0, units revolutions. Value is 22407_8 36673_8, corresponding to about 0.57861112 revolutions (208.30^o).

$K_{25thous}$: Constant, program notation "25THOUS", scale factor B-17, units of (meters/centi-second) per meter. Value is -0.4114470842 E-5 $\times 2^{17}$, corresponding to $-2.5 \times 0.3048 \times 0.01 \times (1/1852) \times 2^{17}$, where first term is nominal value of the constant (fps/nmi), second converts to meters, third to centi-seconds, fourth to nautical miles, and fifth is scale factor.

K_{2b6}: See Conic Routines.

K_{2pisc}: See Conic Routines.

K_{5secdp}: Constant, program notation "5SECDP", scale factor B28, units centi-seconds. Value is 500×2^{-28}, corresponding to 5 seconds.

K_{600sc}: See Orbital Integration.

K_{60min}: Constant, program notation "60MIN", scale factor B28, units centi-seconds. Value is 360000×2^{-28}, corresponding to $60 \times 60 \times 10^2 \times 2^{-28}$, where first term is value in minutes, second converts to seconds, third converts to centi-seconds, and fourth is scale factor.

K_{cs359p}: See Rendezvous Computations.

K_{dpm01}: Constant, program notation "DP-.01", scale factor B0, units revolutions. Octal value is 77777_8 61337_8, corresponding to $-16440_8 \times 2^{-28}$ or -7456×2^{-28}. This is approximately equal to-0.01/360, i.e. -0.01^o.

K_{feng}: Constant, program notation "FENG", scale factor B7, same units as CAPF. Value is 9.1188544×2^{-7}, corresponding approximately to $20500 \times 9.80665 \times 0.45359237 \times 10^{-4} \times 2^{-7}$, where first term is SPS thrust in pounds, second is gravity in meters/second^2, third converts pounds to kilograms, fourth converts to centi-seconds^2, and fifth is scale factor.

K_{frcs2}: Constant, program notation "FRCS2", scale factor B7, same units as CAPF. Value is $0.087437837 \times 2^{-7}$, corresponding approximately to $2 \times 99.8 \times 0.98481 \times 9.80665 \times 0.45359237 \times 10^{-4} \times 2^{-7}$, where first term accounts for 2 RCS jets, second is RCS thrust in pounds, third is cosine 10^{o}(account for canting of nozzle), fourth is gravity in meters/second^2, fifth converts pounds to kilograms, sixth converts to centi-seconds^2, and seventh is scale factor. The product of the first three terms is about 196.568 pounds.

K_{marsdp}: Constant, program notation "MARSDP", scale factor B0, units revolutions. Value in octal is 00000_8 35100_8, corresponding to the complement of 2 K_{dpm01}, or 0.02 degrees (approximately).

K_{maxnm}: See Display Computations.

K_{mutab_i}: See Conic Routines.

$K_{posvel2}$: Single precision constant, program notation "POSVEL2", scale factor B19, units meters. Stored (and nominal) value is 608×2^{-19}, corresponding to about $1994.751 \times 0.3048 \times 2^{-19}$, where first term is value in feet, second converts to meters, and third is scale factor.

$K_{posvel3}$: Single precision constant, program notation "POSVEL3", scale factor B0, units meters/centi-second. Nominal value is 0.00608, corresponding to about $1.994751 \times 0.3048 \times 0.01$, where first term is value in fps, second converts to meters, and third to centi-seconds. The octal value is 00144_8, corresponding to 100×2^{-14} or about 2.0025 fps.

K_{rpad}: Constant, program notation "RPAD", scale factor B29, units meters. Value is 6373338×2^{-29}, corresponding to 6373338 meters (or 20,909,901.5747 feet or 3441.3272 nmi).

$K_{sec29p96}$: Constant, program notation "SEC29.96", scale factor B28, units centi-seconds. Value is 2996×2^{-28}, corresponding to 29.96 seconds.

KT: Value of time delay before performance of midcourse maneuver for P35/P75, scale factor B28, units centi-seconds (set to the appropriate erasable memory constant when these programs started).

LMYDOT: Value of passive vehicle out-of-plane velocity computed in "PRECSET", scale factor B7, units meters/centi-second. It is computed as $\underline{V}_{pass3} \cdot \text{UNR}\underline{\text{M}}$.

MRKBUF1: See Optics Computations.

MRKRTMP: Single precision cell, scale factor B14, used to indicate whether only the gimbal trim is to be performed (value of -1), or if a gimbal drive test in addition is to be performed (value of +1).

NBRCYCLS: See Steering Computations.

NN1: See Rendezvous Computations.

NOMTIG: Value of nominal ignition time sampled in "VN1645" and "P40S/F", scale factor B28, units centi-seconds. It is used in "S40.9" to correct Lambert for oblateness (for earth-centered computation), and in P76/P77 to initialize the N33 display.

NOMTPI: See Rendezvous Computations.

NVWORD1: Single precision communication cell with "CLOKTASK". If it is +0, T_{togo} is merely updated; if it is positive non-zero, "CLOCKJOB" is usually established (except in P41, where a value of 0685_{vn} causes "DYNDISP" to be established instead) to cause display of the verb-noun; if negative non-zero, "CLOCKJOB" is established to generate an 0640_{vn} display with option bits set to cause (for NVWORD1 = -400_8) the pasting of verb 99 by "NV50DSP"; and if negative 0 (set e.g. in "S40.8" for low thrust) then the same type of display is generated as for negative non-zero, except the verb is 97 instead of 99. The option bits are set by "CLOCPLAY".

OGAD: See Digital Autopilot TVC Routines.

OMEGA$\underline{C}$: See Steering Computations.

OPTFLAG: Single precision cell, scale factor B14, set to 1 if P77 is executed and to 0 if P76 is executed. It is used for indexing purposes in "P76ER77".

P40TMP: Required time delay until "TIGBLNK" is performed, scale factor B28, units centi-seconds (used for restart protection purposes). Used single precision in P47 for analogous purpose, B14 centi-seconds. It uses the same cells as the z component of DELVEET$\underline{2}$.

pMGA: See Display Computations.

POSTTPI: Value of pericenter altitude (above K_{rpad} or $|RL\underline{S}|$), scale factor B29, units meters, predicted to exist after the burn.

QPRET: See Orbital Integration.

$\underline{R}_{act1}$, $\underline{R}_{act2}$, $\underline{R}_{act3}$: See Rendezvous Computations.

$\underline{R}_{other}$: See Orbital Integration.

$\underline{R}_{pass1}$, $\underline{R}_{pass2}$, $\underline{R}_{pass3}$: See Rendezvous Computations.

$\underline{R}_{rect}$, $\underline{R}_{rectcm}$: See Orbital Integration.

$\underline{R}_{tig}$: Value of position vector at ignition time, scale factor B29, units meters.

RC$\underline{V}$: See Orbital Integration.

RL$\underline{S}$: See Coordinate Transformations.

RTMU: Value of "retained" mu of primary body, scale factor B36 (earth <u>or</u> moon), used e.g. in "CDHMVR".

RTSR1dMU: Value of "retained" square root of 1/mu for primary body, scale factor B-17(earth) or B-14(moon), program notation "RTSR1/MU", used e.g. in "CSI/B1".

RTX1, RTX2: See Orbital Integration.

RVE$\underline{C}$: See Conic Routines.

SNTH: See Conic Routines.

SUBEXIT: Single precision cell used to retain return address information.

T: See Conic Routines.

T_c: See Orbital Integration.

T_{cdh}, T_{csi}: See Rendezvous Computations.

T_{et}, T_{etcm}: See Orbital Integration.

T_{go}: See Steering Computations.

T_{pass4}: Time of intercept, scale factor B28, units centi-seconds.

T_{pptm1}: See IMU Computations.

T_{strt}: Time of start of P35/P75 targeting computation, scale factor B28, units centi-seconds (loaded at start of "P35/P75B" but not subsequently referenced).

T_{togo}: Value of time until T_{ig} (computed as $T_{now} - T_{ig}$ in "CLOKTASK", hence negative before T_{ig} has arrived), scale factor B28, units centi-seconds.

T_{tpi}: See Rendezvous Computations.

T_{tpi0}: Value of T_{tpi} sampled at the end of "P32/P72C" for use in initializing T_{tpi} in "P73", scale factor B28, units centi-seconds. It is also used in "P33/P73B" to derive T2TOT3, the change in T_{tpi} time.

T1TOT2, T2TOT3: See Rendezvous Computations.

T5LOC: See Digital Autopilot Interface Routines.

TDELTA$\underline{V}$: See Orbital Integration.

TIME5: See Digital Autopilot Interface Routines.

TNU$\underline{V}$: See Orbital Integration.

TVCEXPHS, TVCPHASE: See Digital Autopilot TVC Routines.

$\mathrm{ULO}\underline{\mathrm{S}}$: See Rendezvous Computations.

$\mathrm{UNR}\underline{\mathrm{M}}$: See Rendezvous Computations.

$\mathrm{UP}\underline{1}$: See Rendezvous Computations.

$\underline{V}_{act1}$, $\underline{V}_{act2}$, $\underline{V}_{act3}$: See Rendezvous Computations.

$\underline{V}_{gbody}$: Velocity-to-be-gained in (RCS DAP) "control" axes, scale factor B7, units meters/centi-second. See "S41.1" for the transformation used. In P41, it is updated every second prior to start of Average-G (after return from R60) due to "DYNDISP" computations. Program notation also "DELVCTL".

$\underline{V}_{gtig}$: Velocity-to-be-gained at ignition time in reference coordinates, scale factor B7, units meters/centi-second. Program notation also "VGPREV", reflecting the fact that the same cells used to store previous value of velocity-to-be-gained in "VGCOMP".

$\underline{V}_{iprime}$: See Rendezvous Computations.

$\underline{V}_{other}$: See Orbital Integration.

$\underline{V}_{pass1}$, $\underline{V}_{pass2}$, $\underline{V}_{pass3}$, $\underline{V}_{pass4}$: See Rendezvous Computations.

$\underline{V}_{rect}$, $\underline{V}_{rectcm}$: See Orbital Integration.

$\underline{V}_{tig}$: Value of velocity vector at ignition time, scale factor B7, units meters/centi-second.

$\underline{V}_{tprime}$: See Rendezvous Computations. Also used in "S34/35.5" for temporary storage purposes, to permit detection of a change in the specified velocity increment for the burn.

$\mathrm{VC}\underline{\mathrm{V}}$: See Orbital Integration.

VGDISP: Magnitude of velocity-to-be-gained vector, used for display purposes in N40, N42, and N80. Scale factor is B7, units meters/centi-second.

$\mathrm{VVE}\underline{\mathrm{C}}$: See Conic Routines.

WRENDPOS, WRENDVEL: See Measurement Incorporation.

XKEP: See Orbital Integration.

XXXALT: Value of base altitude for computing apogee and perigee information, scale factor B29 (earth) or B27 (moon), units meters.

YCSM: Value of active-vehicle (CSM for P3x program) out-of-plane position for display in R1 of N90, scale factor B29, units meters. It is the complement of AUTOY.

YDOTC: Value of active-vehicle (CSM for P3x program) out-of-plane velocity for display in R2 of N90, scale factor B7, units meters/centi-second. It is complement of CMYDOT.

YDOTL: Value of passive-vehicle (LM for P3x program) out-of-plane velocity for display in R3 of N90, scale factor B7, units meters/centi-second. It is complement of LMYDOT.

Conic Routines

```
KEPLERN          (Entered only from "KEPPREP")

    Reset overflow indicator          Start of "Kepler Subroutine"
    ITERCTR = 20
    ldMU = $K_{mutab_{-X1}}$          (X1 = -2 if earth, -10 if moon, as set near
                                          end of "KEPPREP")
    ROOTMU = $K_{mutab_{2-X1}}$
    ldROOTMU = $K_{mutab_{4-X1}}$
    $\text{URREC}\underline{\text{T}}$ = unit$\underline{R}_{rect}$
    R1 = $|\underline{R}_{rect}|$
    KEPC1 = ldROOTMU $\underline{R}_{rect} \cdot \underline{V}_{rect}$
    KEPC2 = ldMU R1 $|\underline{V}_{rect}|^2 - 1$
    ALPHA = (1 - KEPC2)/ R1
    If ALPHA $<$ 0:
        TS = $K_{m50sc}$ / ALPHA
        If overflow, reset indicator and set TS = +MAX
        If no overflow:
            TS = $\sqrt{TS}$
    If ALPHA $\geq$ 0:
        TS = $K_{2pisc} / \sqrt{ALPHA}$
        If overflow, reset indicator and set TS = +MAX
    XMAX = TS            (Tag here "STOREMAX")
```

```
$TS_1$ = ldROOTMU XMAX / ALPHA    (period, B28 c.s., in p.d. location 0)
Proceed to "PERIODCH"
PERIODCH
    If $TS_1 \geqslant 0$:
        If no overflow has taken place (e.g. $TS_1 < 2^{28}$ cs):
            $TS = |TAUORB| - TS_1$
            If $TS \geqslant 0$:
                TAUORB = TS sgn TAUORB
                Proceed to "PERIODCH" (starting at 3rd line, compute TS)
    X = XKEPNEW        (tag here "MODDONE")
    If X sgn TAUORB > 0:
        If $|X \text{ sgn } TAUORB| - XMAX < 0$:
            Skip next line
    X = $\frac{1}{2}$ XMAX sgn TAUORB
    If TAUORB $\geqslant 0$:        (tag here "STORBNDS")
        XMIN = 0
    If TAUORB < 0:
        XMIN = - XMAX
        XMAX = 0
    Proceed to "DXCOMP"
```

```
DXCOMP

    $EPSILONT = |K_{bee22} TAUORB|$     (rounded multiplication)
    $DELX = X - XKEP$     (program notation also XPREV)
    Proceed to "KEPLOOP"

KEPLOOP

    $PDXSQ = X^2$
    $XI = ALPHA PDXSQ$
    Perform "DELTIME"
    If overflow has taken place:
        Reset overflow indicator
        If $X \geqslant 0$:
            $XMAX = X$
        If $X < 0$:
            $XMIN = X$
        $DELX = \frac{1}{2} DELX$
        If $DELX = 0$:     (i.e. magnitude less than $K_{epsx}$)
            Proceed to address specified by KEPRTN
        $X = X - DELX$
        $T = T_c$
        Proceed to "BRNCHCTR"
    $DELT = TAUORB - T$
    If $EPSILONT - |DELT| \geqslant 0$:
        Proceed to "KEPCONVG"
    $TS = DELX DELT / (T - T_c)$     (numerator rounded before division,
                                      result shifted right 1 place by
                                      truncated shift for scaling)
```

```
    If TS < 0:
        XMAX = X
        $TS_1$ = TS
        If XMIN - (X + TS) ≥ 0: (or overflow taken place)
            $TS_1$ = 0.9 (XMIN - X)      (rounded multiplication)
    If TS ≥ 0:
        XMIN = X
        $TS_1$ = TS
        If XMAX - (X + TS) < 0:   (or overflow taken place)
            $TS_1$ = 0.9 (XMAX - X)      (rounded multiplication)
    DELX = $TS_1$
    If DELX = 0:    (i.e. magnitude less than $K_{epsx}$)
        Proceed to "KEPCONVG"
    X = X + DELX
    $T_c$ = T
    Proceed to "BRNCHCTR"
BRNCHCTR
    ITERCTR = ITERCTR - 1
    If ITERCTR = 0:
        Proceed to "KEPCONVG"
    Proceed to "KEPLOOP"
KEPCONVG
    $RC\underline{V} = (R1 - XSQCpXIp)\ URREC\underline{T} + (T - 1dROOTMU\ X^3\ SpXIp)\ \underline{V}_{rect}$
    $T\underline{S} = \frac{ROOTMU\ \ X}{|RC\underline{V}|}\ (XI\ SpXIp\ - 1)\ URREC\underline{T}$
    $VC\underline{V} = T\underline{S} + (1 - XSQCpXIp\ /|RC\underline{V}|)\ \underline{V}_{rect}$
    $T_c$ = T
    XKEP = X                    (XKEP notation also XPREV)
    Proceed to address specified by KEPRTN    (loaded by "KEPPREP")
```

DELTIME (Entered with PDXSQ loaded and XI available)

$$SpXIp = K_{s0} + K_{s1}\ XI + K_{s2}\ XI^2 + K_{s3}\ XI^3 + K_{s4}\ XI^4 + K_{s5}\ XI^5 + K_{s6}\ XI^6 + K_{s7}\ XI^7 + K_{s8}\ XI^8 + K_{s9}\ XI^9$$

$$CXI = K_{c0} + K_{c1}\ XI + K_{c2}\ XI^2 + K_{c3}\ XI^3 + K_{c4}\ XI^4 + K_{c5}\ XI^5 + K_{c6}\ XI^6 + K_{c7}\ XI^7 + K_{c8}\ XI^8 + K_{c9}\ XI^9$$

XSQCpXIp = PDXSQ CXI (This is "Kepler Equation Subroutine")

TS_1 = KEPC1 XSQCpXIp (triple precision)

TS_2 = KEPC2 PDXSQ SpXIp (triple precision)

$TS_3 = X\ (R1 + TS_2)$ (addition triple precision)

$T = 1dROOTMU\ (TS_3 + TS_1)$ (rounded shift and multiply, triple precision addition)

Return

LAMBERT Entered only from "INITVEL2"

RTNCONC = Return address Start of "Lambert Subroutine"

Reset overflow indicator

ITERCTR = 20

$1dMU = K_{mutab_{-X1}}$ (X1 = -2 if earth, -10 if moon)

$ROOTMU = K_{mutab_{2-X1}}$

$1dROOTMU = K_{mutab_{4-X1}}$

$EPSILONL = K_{bee19}$ TDESIRED (rounded multiplication)

Set bit 3(ITER1SW) of FLAGWRD1 = 1

$T\underline{S}_{r1} = R1VE\underline{C}$

$T\underline{S}_2 = R2VE\underline{C}$

Perform "GEOM"

SNTH = TS

```
PDLAMBDA = R1 / MAGVEC2
CSTH = $TS_{cos}$      (computed by "GEOM")
lmCSTH = 1 - CSTH     (scale factor B2, truncated shift)
TS = 1 - CSTH, using rounded right shift 1 (same scaling as lmCSTH)
If  TS = 0:    (angle $0^{o}$ or $360^{o}$)
    Set bit 3(NOSOLNSW) of FLAGWRD5 = 1
    Proceed to address specified by RTNCONC
MAX = $SNTH / lmCSTH + \sqrt{2\ PDLAMBDA/ TS}$    (MAX notation also COGAMAX)
If overflow has taken place:
    MAX = $K_{cogupm}$
    Skip next 3 lines
If  MAX $\geq$ 0:
    If  MAX - $K_{cogupm} \geq 0$:
        MAX = $K_{cogupm}$
CSTHmRHO = CSTH - PDLAMBDA
TS = $K_{coglom}$
If GEOMSGN $\geq$ 0:
    $TS_1$ = CSTHmRHO / SNTH
    If $TS_1$ does not overflow  (not $\geq 2^5$ in magnitude):
        TS = $TS_1$
MIN = TS        (notation also COGAMIN)
If bit 2(GUESSSW) of FLAGWRD1 = 0:
    TWEEKIT = $2^{-14}$
If bit 2(GUESSSW) of FLAGWRD1 = 1:
    TWEEKIT = $2^{-2}$  (i.e. 0.25)
    COGA = $\frac{1}{2}$ MIN + $\frac{1}{2}$ MAX
```

```
    (If bit 2(GUESSSW) of FLAGWRD1 = 1):
        DCOGA = COGA
    Proceed to "LAMBLOOP"
LAMBLOOP
    TS = 1mCSTH / (SNTH COGA - CSTHmRHO)
    If  $TS \leq 0$:
        If  $DCOGA \geq 0$:
            MAX = COGA
        If  $DCOGA < 0$:
            MIN = COGA
        Proceed to "COMMONLM"
    P = TS                                      (scaled B4)
    $RdA = K_{2b6} - P (K_{1b10} + COGA^2)$     (scaled B6)
    If overflow has taken place:     (e.g. $|P| \geq 2^4$  or  $|RdA| \geq 2^6$)
        MIN = COGA
        Proceed to "COMMONLM"
    TS = P
    Perform "GETX"
    TPREV = T
    If bit 7(INFINFLG) of FLAGWRD8 = 1:
        If  $DCOGA \geq 0$:
            MAX = COGA
        If  $DCOGA < 0$:
            MIN = COGA
        Proceed to "COMMONLM"
    Perform "DELTIME"
```

```
    If overflow has taken place:
        T = TPREV
        MAX = COGA
        Proceed to "COMMONLM"
    TERRLAMB = TDESIRED - T
    If $EPSILONL - |TERRLAMB| \geq 0$:
        Proceed to "INITV"
    ITERCTR = ITERCTR - 1
    If ITERCTR = 0:
        Proceed to "SUFFCHEK"
    If bit 3(ITER1SW) of FLAGWRD1 = 0:
        If (T - TPREV) = 0:
            Proceed to "SUFFCHEK"
    Perform "ITERATOR"
    If DCOGA = 0:    (i.e. magnitude less than $2^{-23}$)
        Proceed to "SUFFCHEK"
    COGA = COGA + DCOGA
    Proceed to "LAMBLOOP"
COMMONLM
    DCOGA = $\frac{1}{2}$ DCOGA
    If DCOGA = 0:        (i.e. magnitude less than $2^{-23}$)
        Proceed to "SUFFCHEK"
    COGA = COGA - DCOGA
    Proceed to "LAMBLOOP"
SUFFCHEK
    If $\frac{1}{4}\,TDESIRED + K_{lcs} - |TERRLAMB| < 0$:
        Set bit 3(NOSOLNSW) of FLAGWRD5 = 1
        Proceed to address specified by RTNCONC
    Proceed to "INITV"
```

```
INITV
    $TS = ROOTMU \sqrt{P / R1}$
    $\underline{VVEC} = TS\ COGA\ \underline{UR1} + TS\ (\underline{UN} * \underline{UR1})$
    Set bit 3(NOSOLNSW) of FLAGWRD5 = 0
    If VTARGTAG = 0:
        TS = MAGVEC2
        Perform "LAMENTER"
        $\underline{V}_{target} = \underline{TS}_{v}$
    Proceed to address specified by RTNCONC
ITERATOR      (Entered only from "LAMBLOOP")
    If bit 3(ITER1SW) of FLAGWRD1 = 0:        Start of "Iterator
                                                        Subroutine"
        $TS = TERRLAMB\ DCOGA / (T - TPREV)$
        If bit 6(ORDERSW) of FLAGWRD8 = 1:
            $TS = |TS|\ sgnTERRLAMB$
    If bit 3(ITER1SW) of FLAGWRD1 = 1:
        Set bit 3(ITER1SW) of FLAGWRD1 = 0
        $TS = (MAX\ TWEEKIT - MIN\ TWEEKIT)\ sgnTERRLAMB$
    If $TS < 0$:
        If bit 6(ORDERSW) of FLAGWRD8 = 0:
            MAX = COGA
        If $MIN - COGA - TS \geq 0$:  (or overflow taken place)
            $TS = 0.9\ (MIN - COGA)$
        DCOGA = TS
        Return
    If bit 6(ORDERSW) of FLAGWRD8 = 0:      ($TS \geq 0$)
        MIN = COGA
```

```
    If MAX - COGA - TS $<$ 0:   (or overflow taken place)
        TS = 0.9 (MAX - COGA)
    DCOGA = TS
    Return
TIMETHET    This is "Time-Theta Subroutine", entered from "CDHMVR",
                     "CSI/B1" (2), "HOPALONG", "P34/P74C", "P36A",
    RTNCONC = Return     and "P37E"
              address
    Reset overflow indicator
    $\underline{TS}_r$ = $\underline{RVEC}$
    $\underline{TS}_v$ = $\underline{VVEC}$
    Perform "PARAM"
    If overflow has taken place:
        Set bit 4(COGAFLAG) of FLAGWRD8 = 1
        Proceed to "POODOO" (pattern $20607_8$)
    TS = P
    Perform "GETX"
    Proceed to "COMMNOUT"
COMMNOUT
    If bit 7(INFINFLG) of FLAGWRD8 = 1:
        Proceed to "POODOO" (pattern $20607_8$)
    Set bit 4(COGAFLAG) of FLAGWRD8 = 0
    Perform "DELTIME"
    If bit 9(RVSW) of FLAGWRD7 = 0:    (means new $\underline{R}$, $\underline{V}$ desired)
        Perform "NEWSTATE"
    Proceed to address specified by RTNCONC
NEWSTATE   This is "State Vector Subroutine"
    $\underline{TS}_r$ = (R1 - XSQCpXIp) $\underline{UR1}$ + (T - 1dROOTMU $X^3$ SpXIp) $\underline{VVEC}$
    TS = $|\underline{TS}_r|$                    ($\underline{TS}_r$ at top of push-down list)
    Proceed to "LAMENTER"
```

```
LAMENTER    (Entered with TS set to radius magnitude)
    R2 = TS
    $T\underline{S} = \frac{\mathrm{ROOTMU\ X}}{\mathrm{R2}} (\mathrm{XI\ SpXIp} - 1)\ \mathrm{UR}\underline{1}$
    $T\underline{S}_v = T\underline{S} + (1 - \mathrm{XSQCpXIp} / \mathrm{R2})\ \mathrm{VVE}\underline{C}$    (same formulation as on page CONC-4, but different quantities enter calculation)
    Return    ($T\underline{S}_v$ in MPAC)
TIMERAD    This is "Time-Radius Subroutine", entered from "PREC125" and "TMRAD100"
    RTNCONC = Return address
    Reset overflow indicator
    $T\underline{S}_r = \mathrm{RVE}\underline{C}$
    $T\underline{S}_v = \mathrm{VVE}\underline{C}$
    Perform "PARAM"
    If overflow has taken place:
        Set bit 4(COGAFLAG) of FLAGWRD8 = 1
        Proceed to "POODOO" (pattern $20607_8$)
    $\mathrm{PDEC}\underline{C} = (1 - \mathrm{RdA})\ \mathrm{UR}\underline{1} - \mathrm{COGA} \sqrt{\mathrm{P}\ (\mathrm{K}_{2b6} - \mathrm{RdA})}\ \mathrm{U}\underline{2}$    (computed B7, then made B3)
    If all components of $\mathrm{PDEC}\underline{C} < 2^{-18}$, or if overflow has taken place:
        Set bit 3(NOSOLNSW) of FLAGWRD5 = 1
        Proceed to "POODOO" (pattern $20607_8$)
    $\mathrm{COSF} = \left((\mathrm{P\ R1} / \mathrm{RDESIRED}) - 1\right) / |\mathrm{PDEC}\underline{C}|$
    $\mathrm{TS} = 1 - \mathrm{COSF}^2$
    If $\mathrm{TS} < 0$, or if overflow has taken place:
        COSF = 1 sgn COSF
        Set bit 5(APSESW) of FLAGWRD8 = 1    (trajectory does not pass through RDESIRED)
        TS = 0
        Skip next 2 lines
    $\mathrm{TS} = \sqrt{\mathrm{TS}}$ sgn SGNRDOT
    Set bit 5(APSESW) of FLAGWRD8 = 0
```

```
    $\mathrm{PDUR\underline{2}} = \mathrm{COSF\ unitPDEC\underline{C}} + \mathrm{TS\ (U\underline{N} * unitPDEC\underline{C})}$
    $\mathrm{CSTH} = \mathrm{PDUR\underline{2}} \cdot \mathrm{UR\underline{1}}$, with magnitude limited less than 1
    $\mathrm{SNTH} = (\mathrm{UR\underline{1}} * \mathrm{PDUR\underline{2}}) \cdot \mathrm{U\underline{N}}$
    $\mathrm{TS} = \mathrm{P}$
    Perform "GETX"
    Set bit 3(NOSOLNSW) of FLAGWRD5 = 0
    Proceed to "COMMNOUT"
APSIDES            This is "Apsides Subroutine"
    RTNCONC = Return address
    Reset overflow indicator
    $\mathrm{T\underline{S}}_r = \mathrm{RVE\underline{C}}$
    $\mathrm{T\underline{S}}_v = \mathrm{VVE\underline{C}}$
    Perform "PARAM"
    Reset overflow indicator
    $\mathrm{ECC} = \sqrt{1 - \mathrm{P\ RdA}}$   (note that "SR30.1" formed absolute value
                                                before taking square root)
    $\mathrm{TS}_{rp} = \mathrm{R1\ P} / (1 + \mathrm{ECC})$     (at top of push-down list, in OD)
    $\mathrm{TS}_{ra} = 2(\mathrm{R1} / \mathrm{RdA}) - \mathrm{TS}_{rp}$      (left in MPAC)
    If $\mathrm{TS}_{ra} < 0$, or if overflow has taken place:
        $\mathrm{TS}_{ra} = +\mathrm{MAX}$
    Proceed to address specified by RTNCONC  (TS scaling same as $\mathrm{RVE\underline{C}}$)
PARAM     (Entered with $\mathrm{T\underline{S}}_r$ in OD and $\mathrm{T\underline{S}}_v$ in MPAC)
    RTNPRM = Return address      Start of "Conic Parameters Subroutine"
    Set bit 10(NORMSW) of FLAGWRD7 = 0
    Set bit 4(COGAFLAG) of FLAGWRD8 = 0
    $\mathrm{GEOMSGN} = 27777_8$     (a positive quantity)
    $\mathrm{T\underline{S}}_{r1} = \mathrm{T\underline{S}}_r$
    $\mathrm{T\underline{S}}_2 = \mathrm{T\underline{S}}_v$
```

```
          Perform "GEOM"
          $COGA = TS_{cos} / TS$      (TS left with sine of angle)   (overflow if $2^5$ or more)
          $ldMU = K_{mutab_{-X1}}$          ($X1 = -2$ if earth, $-10$ if moon)
          $ROOTMU = K_{mutab_{2-X1}}$
          $ldROOTMU = K_{mutab_{4-X1}}$
          $TS_1 = ldMU\ R1\ MAGVEC2^2$        ($MAGVEC2 = |T\underline{S}_v|$ from "GEOM")
          $RdA = K_{2b6} - TS_1$
          $P = TS_1\ TS^2$       (TS from "GEOM" with sine of angle)
          Proceed to address specified by RTNPRM
     GEOM      (Entered with $T\underline{S}_{r1}$ at top of pushdown list and $T\underline{S}_2$ in MPAC)
          $U\underline{2} = unitT\underline{S}_2$          Start of "Geometric Parameters Subroutine"
          $MAGVEC2 = |T\underline{S}_2|$
          $UR\underline{1} = unitT\underline{S}_{r1}$
          $TS_{cos} = UR\underline{1} \cdot U\underline{2}$     (left at top of push-down list)
          $R1 = |T\underline{S}_{r1}|$
          $T\underline{S} = UR\underline{1} * U\underline{2}$
          If bit 10(NORMSW) of FLAGWRD7 = 0:
               $U\underline{N} = unitT\underline{S}\ sgn\ GEOMSGN$
               If overflow has taken place: (e.g. $T\underline{S}$ components all $< 2^{-20}$)
                    $U\underline{N} = (\pm 1, \frac{1}{2} TS_y, \frac{1}{2} TS_z)$     (not meaningful)
          $TS = |T\underline{S}|\ sgn\ GEOMSGN$     (sine of angle, in MPAC)
          Return
     GETX      (Entered with TS set to P, in MPAC)
          $X2 = 3$       (to be used as counter)     Start of "Universal Variable Subroutine"
          Set bit 1(360SW) of FLAGWRD8 = 0
```

```
    $TS_1 = SNTH / (1 - CSTH)$        (scaled B5)
    If overflow has taken place (e.g. $TS_1$ magnitude $\geqslant 2^5$):
        If $TS_1 < 0$:
            Set bit 1(360SW) of FLAGWRD8 = 1
        Proceed to "INVRSEQN"
    $PDWVAL = \sqrt{TS} (TS_1 - COGA)$        (scaled B5)
    If overflow has taken place (e.g. PDWVAL magnitude $\geqslant 2^5$):
        If PDWVAL $< 0$:
            Set bit 1(360SW) of FLAGWRD8 = 1
        Proceed to "INVRSEQN"
    Proceed to "WLOOP"
WLOOP
    $TS_1 = PDWVAL^2$    (triple precision)
    $TS_2 = RdA + TS_1$
    If $TS_2 < 0$:
        Reset overflow indicator
        Set bit 7(INFINFLG) of FLAGWRD8 = 1
        Return
    $TS = PDWVAL + \sqrt{TS_2}$    (square root rounded to double precision)
    If overflow has taken place: (e.g. TS magnitude $\geqslant 2^5$)
        X2 = 3
        If TS $< 0$:
            Set bit 1(360SW) of FLAGWRD8 = 1
        Proceed to "INVRSEQN"
    If X2 $> 1$:
        X2 = X2 - 1
        PDWVAL = TS
        Proceed to "WLOOP"
```

```
          $TS = 1 / TS$          (scaled B2)
          If overflow (e.g. TS magnitude $\geq 2^2$):
               Reset overflow indicator
               Set bit 7(INFINFLG) of FLAGWRD8 = 1
               Return
          Proceed to "POLYCOEF"
     INVRSEQN
          $PDW2 = \left| \frac{SNTH}{\sqrt{P}\,(1 + CSTH - SNTH\ COGA)} \right|$
          $PDW3 = 1$
          Proceed to "1/WLOOP"
     1/WLOOP
          $TS_1 = PDW3^2$      (triple precision)
          $TS_2 = RdA\ \ PDW2^2$ (triple precision)
          $TS_3 = TS_1 + TS_2$
          If $TS_3 < 0$:
               Reset overflow indicator
               Set bit 7(INFINFLG) of FLAGWRD8 = 1
               Return
          $TS = PDW3 + \sqrt{TS_3}$
          If $X2 > 1$:
               $X2 = X2 - 1$
               $PDW3 = TS$

               Proceed to "1/WLOOP"
          $TS = PDW2 / TS$            (scaled B2)
          Proceed to "POLYCOEF"
```

POLYCOEF

```
If  $TS < 0$:
    Reset overflow indicator
    Set bit 7(INFINFLG) of FLAGWRD8 = 1
    Return
$PDAVAL = TS$
$PDB = RdA\ PDAVAL^2$
$FCTN = K_{db0} + K_{db1}\ PDB + K_{db2}\ PDB^2 + K_{db3}\ PDB^3 + K_{db4}\ PDB^4 + K_{db5}\ PDB^5 + K_{db6}\ PDB^6$
$PDXN = 16\ PDAVAL\ FCTN$
If bit 1(360SW) of FLAGWRD8 = 1:
    If $RdA < 0$:
        Reset overflow indicator
        Set bit 7(INFINFLG) of FLAGWRD8 = 1
        Return
    $PDXN = K_{2pisc} / \sqrt{RdA} - PDXN$
$XI = RdA\ PDXN^2$
$X = PDXN \sqrt{R1}$
$PDXSQ = X^2$
$KEPC1 = \sqrt{P\ R1}\ COGA$
$KEPC2 = 1 - RdA$
Set bit 7(INFINFLG) of FLAGWRD8 = 0
Return     (to routine calling "GETX", of course)
```

Quantities in Computations

See also list of major variables and list of routines

ldMU: Value of the reciprocal of μ of primary body, selected from K_{mutab_i} using X1 for indexing, scale factor B-34 (earth) or B-28 (moon), and stored in push-down list location 14D, program notation "1/MU".

ldROOTMU: Value of square root of ldMU of primary body, selected from K_{mutab_i} using X1 for indexing, scale factor B-17(earth) or B-14(moon), and stored in push-down list location 18D, program notation "1/ROOTMU".

lmCSTH: Value of (1 - CSTH), scale factor B2, computed in "LAMBERT".

ALPHA: Value of reciprocal of semi-major axis computed in "KEPLERN", scale factor B-22 (earth) or B-20 (moon), units of meters^{-1}, stored in push-down list location 8D.

COGA: Value of cotangent of flight path angle, scale factor B5 (measured from vertical). "PARAM" computes it as angle between incoming position and velocity arguments, with overflow indicator set if an overflow here (corresponding to an angle less than about 1° 47.5', or of course closer to 180° than about 178° 12.5'). Notation also "INDEP".

COSF: Value of cosine of the true anomaly at the desired radius in "TIMERAD", scale factor B1, stored in push-down list location 24D.

CSTH: Value of cosine of the true anomaly difference or of the angle between present and desired position vectors, scale factor B1.

CSTHmRHO: Value of difference between CSTH and PDLAMBDA computed in "LAMBERT", scale factor B7, program notation "CSTH-RHO".

CXI: Value of special transcendental function computed in "DELTIME" from a series expansion of XI, scale factor B4, not explicitly stored but shown for clarity.

DCOGA: Change in the independent variable COGA in Lambert iteration loop, scale factor B5. Program notation for the cell is also "DELINDEP", and it is stored in push-down list location 12D.

DELT: Difference between desired time and derived time in "KEPLOOP", scale factor B28, units centi-seconds.

DELX: Difference between new and old values of XKEP computed in "DXCOMP", scale factor B17 (earth) or B16 (moon). Updated in "KEPLOOP".

ECC: Value of eccentricity computed in "APSIDES", scale factor B3.

EPSILONL: Value of Lambert convergence limit computed in "LAMBERT", scale factor B28, units centi-seconds.

EPSILONT: Value of Kepler routine convergence limit computed in "DXCOMP", scale factor B28, units centi-seconds.

FCTN: Value of polynomial expansion function computed in "POLYCOEF", scale factor B1.

GEOMSGN: Single precision sign information for the sine of the true anomaly difference, used in "GEOM" (set to a positive value in "PARAM" before performing "GEOM"). It is a Lambert input. See SGNRDOT.

ITERCTR: Single precision iteration counter used to protect Kepler and Lambert iterations from excessive loops, scale factor B14. It is initialized to 20 in "KEPLERN" and "LAMBERT". In push-down cell 22D.

K_{1b10}: Constant, program notation "D1/1024", scale factor B10, value 1×2^{-10}, corresponding to an equation value of 1.

K_{1cs}: Constant, program notation "ONEBIT", scale factor B28, units centi-seconds. Value is 1×2^{-28}, corresponding to 1 centi-second.

K_{2b6}: Constant, program notation "D1/32", scale factor B6, value 1×2^{-5}. Value corresponds to 2×2^{-6}, where first term is equation value and second is scale factor.

K_{2pisc}: Constant, program notation "2PISC", scale factor B6, value 6.2831853×2^{-6}. Value corresponds to 2π, with $\pi = 3.14159265$.

K_{bee19}: Constant, program notation "BEE19", scale factor B0, value 1×2^{-19}, giving the Lambert time error ratio (used in "LAMBERT" to multiply TDESIRED to get EPSILONL).

K_{bee22}: Constant, program notation "BEE22", scale factor B0, value 1×2^{-22}, giving the Kepler time error ratio (used in "DXCOMP" to multiply TAUORB to get EPSILONT).

K_{ci} (i = 0-9): Set of constants used in "DELTIME" to evaluate CXI, scale factor B(4-6i), program notation of K_{c0} "DELTIME +29". Values are:

i	Stored Value	True Value	Infinite Series Value
0	0.031250001	0.5 (in memory)	(1/2!) = 0.5
1	-0.166666719	-0.041666680	-(1/4!) =-0.041666667
2	0.355555413	1.38888833E-3	(1/6!) = 1.38888889E-3
3	-0.406347410	-2.48014777E-5	-(1/8!) =-2.48015873E-5
4	0.288962094	2.75575727E-7	(1/10!)= 2.75573192E-7
5	-0.140117894	-2.08791932E-9	-(1/12!)=-2.08767570E-9
6	0.049247387	1.14663008E-11	(1/14!)= 1.14707456E-11
7	-0.013081923	-4.75917586E-14	-(1/16!)=-4.77947733E-14
8	0.002806389	1.59524745E-16	(1/18!)= 1.56192070E-16
9	-0.000529414	-4.70214090E-19	-(1/20!)=-4.11031762E-19

K_{coglom}: Constant, program notation "COGLOLIM", scale factor B5, value -0.999511597. Magnitude of true value corresponds to about 31.9843711, so chosen that $(1 + COGA^2)$, if COGA = K_{coglom}, will be less than 2^{10}, thus avoiding overflow in the computation of RdA in "LAMBLOOP", where this term appears.

K_{cogupm}: Constant, program notation "COGUPLIM", scale factor B5, value 0.999511597, the same decimal value (with opposite sign) as K_{coglom}.

K_{dbi} (i = 0-6): Set of constants used in "POLYCOEF" to evaluate FCTN, scale factor (of all) B1, program notation (of K_{db0}) "POLYCOEF +10". Values are:

i	Stored Value	True Value	Infinite Series Value
0	0.5	1.0	1.0
1	-0.166666770	-0.33333354	-(1/3) =-0.333333333
2	0.100000392	0.200000784	(1/5) = 0.2
3	-0.071401086	-0.142802172	-(1/7) =-0.142857143
4	0.055503292	0.111006584	(1/9) = 0.111111111
5	-0.047264098	-0.094528196	-(1/11)=-0.090909091
6	0.040694204	0.081388408	(1/13) = 0.076923076

K_{epsx}: Dummy constant used in "KEPLOOP" to indicate a value of one least increment for DELX, program value 2^{-28}, scale factor B17(earth) or B16(moon). Equation value corresponds to 2^{-11} (earth) or 2^{-12} (moon). Could also be considered to be a program value of 2^{-29} (depending on how rounding is interpreted), in which case requirement is for DELX to exceed the constant, and equation value would correspond to 2^{-12} (earth) or 2^{-13}(moon).

K_{m50sc}: Constant, program notation "-50SC", scale factor B12, value -50 x 2^{-12}, corresponding to an equation value of -50.

K_{mutab_i} (i = 0-14 in steps of 2): Set of constants providing functions of μ for earth (i = 0,2,4,6) and moon (i = 8,10,12,14). The second, third, and fourth constant in each set is used to initialize the push-down list cells 1dMU, ROOTMU, and 1dROOTMU for earth (X1 = -2) or moon (X1 = -10), as well as for references elsewhere.

i	Stored Value	Scaling	Significance
0	3.986032E10 x 2^{-36}	B36	Earth μ, m^3/cs^2
2	0.25087606E-10 x 2^{34}	B-34	Earth $1/\mu$, cs^2/m^3
4	1.99650495E5 x 2^{-18}	B18	Earth $\sqrt{\mu}$
6	0.50087529E-5 x 2^{17}	B-17	Earth $1/\sqrt{\mu}$
8	4.902778E8 x 2^{-30}	B30	Moon μ, m^3/cs^2
10	0.203966E-8 x 2^{28}	B-28	Moon $1/\mu$, cs^2/m^3
12	2.21422176E4 x 2^{-15}	B15	Moon $\sqrt{\mu}$
14	0.45162595E-4 x 2^{14}	B-14	Moon $1/\sqrt{\mu}$

K_{si} (i = 0-9): Set of constants used in "DELTIME" to evaluate SpXIp, scale factor B(1-6i), program notation of K_{s0} "DELTIME +3". Values are:

i	Stored Value	True Value	Infinite Series Value
0	0.083333334	0.166666668	(1/3!) = 0.166666667
1	-0.266666684	-8.33333387E-3	-(1/5!) =-8.33333333E-3
2	0.406349155	1.98412673E-4	(1/7!) = 1.98412698E-4
3	-0.361198675	-2.75572720E-6	-(1/9!) =-2.75573192E-6
4	0.210153242	2.50522187E-8	(1/11!)= 2.50521084E-8
5	-0.086221951	-1.60600899E-10	-(1/13!)=-1.60590438E-10
6	0.026268812	7.64523051E-13	(1/15!)= 7.64716373E-13
7	-0.006163316	-2.80275162E-15	-(1/17!)=-2.81145725E-15
8	0.001177342	8.36551806E-18	(1/19!)= 8.22063525E-18
9	-0.000199055	-2.20995444E-20	-(1/21!)=-1.95729411E-20

KEPC1: Value of coefficient in Kepler equation used in "DELTIME", scale factor B17(earth) or B16(moon), stored in push-down list location 34D and computed in "KEPLERN" and "POLYCOEF".

KEPC2: Value of coefficient in Kepler equation used in "DELTIME", scale factor B6, stored in push-down list location 36D and computed in "KEPLERN" and "POLYCOEF".

KEPRTN: Single precision return address from "KEPLERN" routine (preset in "KEPPREP").

MAGVEC2: Magnitude of "second" vector input to "GEOM": if entered from "PARAM", is velocity magnitude scaled B7 (earth) or B5 (moon), units meters/centi-second; if entered from "LAMBERT", is magnitude of R2VEC, scaled B29 (earth) or B27(moon), units meters.

MAX: Value of maximum COGA computed in "LAMBERT"and subsequently updated, scale factor B5. Stored in push-down list location 14D, with program notation also "COGAMAX".

MIN: Value of minimum allowable COGA computed in "LAMBERT" and subsequently updated, scale factor B5. Stored in push-down list location 8D, with program notation also "COGAMIN".

P: Value of ratio of semi-latus rectum to magnitude of present (R1) position, scale factor B4, computed in "PARAM" and "LAMBLOOP".

PDAVAL: Value of temporary quantity "a" computed in "GETX" computation and loaded in "POLYCOEF", scale factor B2, loaded in push-down list location 0D.

PDB: Value of temporary quantity "b" computed in "POLYCOEF", scale factor B0, used as argument for computing FCTN.

PDEC$\underline{C}$: "A vector in direction of pericenter whose magnitude is equal to eccentricity," computed in "TIMERAD" initially with scale factor B7 and then rescaled to have scale factor B3. Its unit is stored in push-down list location OD, scale factor B1, while its magnitude is in 36D, scale factor B3.

PDLAMBDA: Value of ratio of magnitudes of R1VE$\underline{C}$ and R2VE$\underline{C}$ computed in "LAMBERT", scale factor B7, stored in push-down list location OD.

PDUR$\underline{2}$: Unit vector in direction of final radius computed in "TIMERAD", scale factor B1, stored in push-down list location OD.

PDW2: Value of temporary quantity "w_2" computed in "INVRSEQN", scale factor B-1, stored in push-down list location OD.

PDW3: Value of temporary quantity "w_3" computed in "1/WLOOP", scale factor B4, stored in push-down list location 2D.

PDWVAL: Value of temporary quantity "w_1" computed in "WLOOP", scale factor B5, stored in push-down list location OD.

PDXN: Value of temporary quantity "x_N" computed in "POLYCOEF", scale factor B6, stored in push-down list location OD. The factor of "16" in the equation comes from this scale factor: without the "16", scale factor would be B2 (scale of B6 required for subsequent computations, of course).

PDXSQ: Value of the square of the quantity "X", stored in push-down list location OD by "KEPLOOP" and "POLYCOEF", and used in "DELTIME". If X already normalized, scale factor B34(earth) or B32(moon): scaling information also contained in X1, which is used in "DELTIME" to achieve proper scaling of end results (not shown in programmed equations).

$\underline{R}_{rect}$: See Orbital Integration.

R1: Magnitude of "present" position vector (of $\underline{R}_{rect}$ in "KEPLERN" and T$\underline{S}_{r1}$ in "GEOM"), scale factor B29(earth) or B27 (moon), units meters. Stored in push-down list location 32D.

R1VE$\underline{C}$: Initial position vector used as input to Lambert routine (for point of maneuver initiation), scale factor B29(earth) or B27 (moon), units meters.

R2: Magnitude of "desired" or final position vector, scale factor B29 (earth) or B27 (moon), units meters, loaded e.g. at start of "LAMENTER", where scaling information also in X1 (not shown).

R2VE$\underline{C}$: Desired terminal (i.e. "target") position vector used as input to Lambert routine, scale factor B29(earth) or B27(moon), units meters.

RC$\underline{V}$: See Orbital Integration. Low 5 bits 0 magnitude by "KEPCONVG".

RdA: Value of ratio of "present" (R1) position vector magnitude to semi-major axis, scale factor B6, computed in "LAMBLOOP" and "PARAM". Program notation is "R1A".

RDESIRED: Value of desired final radius used as input to "TIMERAD" routine, scale factor B29(earth) or B27(moon), units meters.

ROOTMU: Value of square root of μ of primary body, selected from K_{mutab_i} using X1 for indexing, scale factor B18(earth) or B15(moon), stored in push-down list location 16D.

RTNCONC: Single precision cell used to retain return address information from several conic routines. Program notations are "RTNAPSE", "RTNLAMB", "RTNTR", and "RTNTT".

RTNPRM: Single precision cell used to retain return address information from "PARAM".

RVE$\underline{C}$: Vector position input to several of the conic routines, scale factor B29(earth) or B27(moon), units meters.

SGNRDOT: Single precision cell used double precision in "TIMERAD" to determine required sign of radial velocity at desired radius: if most significant half is non-zero, then least significant half would have no effect.

SNTH: Value of sine of true anomaly difference or of the angle between present and desired position vectors, scale factor B1.

SpXIp: Value of special transcendental function computed in "DELTIME" from a series expansion of XI, scale factor B1, program notation "S(XI)", stored in push-down list location 26D.

T: Value of time computed in "DELTIME" to go from present position to "desired" position, scale factor B28, units centi-seconds, stored in push-down list location 30D. It contains the result from the performance of "TIMERAD" or "TIMETHET" routines. Is also "DEP".

T_c: See Orbital Integration. It is set in "KEPCONVG" to the value of T (less than one period).

TAUORB: See Orbital Integration. It is reduced in "PERIODCH" so as to be less than one period (for forward *or* backward integration).

TDESIRED: Value of desired transfer time (between R1VEC and R2VEC)used as input to "LAMBERT", scale factor B28, units centi-seconds.

TERRLAMB: Difference between desired and derived transfer time computed in "LAMBLOOP", scale factor B28, units centi-seconds; program notation also "DELDEP".

TPREV: Previous value of T in "LAMBLOOP", scale factor B28, units centi-seconds; program notation also "DEPREV".

TWEEKIT: Single precision quantity used to indicate required amount of modification on first iteration in "ITERATOR", scale factor B0, stored in push-down list location 40D. It is initialized to the appropriate value in "LAMBERT".

U2: Value of the unit of the vector whose magnitude is in MAGVEC2, scale factor B1, computed in "GEOM".

UN: Unit normal to the two vectors used as input to "GEOM", scale factor B1. If overflow is encountered in forming the unit vector, then the result is not meaningful. It can also be an input to "LAMBERT".

UR1: Value of the unit of the vector whose magnitude is in R1, scale factor B1, computed in "GEOM".

URRECT: Value of unit$\underline{R}_{rect}$ computed in "KEPLERN", scale factor B1.

$\underline{V}_{rect}$: See Orbital Integration.

$\underline{V}_{target}$: See Rendezvous Computations.

VCV: See Orbital Integration. Low 8 bits have zero magnitude when computed in "KEPCONVG".

VTARGTAG: See Rendezvous Computations.

VVEC: Vector velocity input to several of the conic routines, scale factor B7 (earth) or B5 (moon), units meters/centi-second. Lambert output.

X: Storage for updated values of XKEP information in "KEPLOOP", scale factor B17(earth) or B16(moon), stored in push-down list location 20D. Computed in "KEPLOOP" and "POLYCOEF".

XI: Value of square of eccentric anomaly difference for ellipse, or negative of square of its hyperbolic analog for hyperbola, scaled B6. It is used as the independent variable in the "DELTIME" expansions, and is computed in "KEPLOOP" and "POLYCOEF", and stored in push-down list location 24D.

XKEP: See Orbital Integration. Program notation also "XPREV".

XKEPNEW: See Orbital Integration.

XMAX: Value of upper bound on X computed in "KEPLERN", scale factor B17(earth) or B16(moon), stored in push-down list location 10D.

XMIN: Value of lower bound on X computed in "PERIODCH", scale factor B17(earth) or B16(moon), stored in push-down list location 12D.

XSQCpXIp: Value of X^2 times CXI computed in "DELTIME", scale factor B33(earth) or B31(moon), program notation "XSQC(XI)", stored in push-down list location 28D.

Coordinate Transformations

CALCGTA

```
    $\underline{TS} = \text{unit}\ (-X_{dc,z}\ ,\ 0,\ X_{dc,x})$     Equivalent to $\underline{X}_{dc} * \text{unit}\underline{Y}$
    $SINTH = TS_x$
    $COSTH = TS_z$
    Perform "ARCTRIG"
    $IGC = THETA$
    $SINTH = X_{dc,y}$
    $COSTH = X_{dc,x}\ TS_z\ -\ X_{dc,z}\ TS_x$
    Perform "ARCTRIG"
    $MGC = THETA$
    $COSTH = \underline{TS} \cdot \underline{Z}_{dc}$
    $SINTH = \underline{TS} \cdot \underline{Y}_{dc}$
    Perform "ARCTRIG"
    $OGC = THETA$
    Return
```

ARCTRIG Routine entered with SINTH and COSTH, scale factor B2. Return is with THETA, scale factor B0, units revolutions, in range $\pm \frac{1}{2}$ (i.e. ± 180°).

```
    If $|SINTH| - K_{qts4} \geq 0$:
        $THETA = \left(\cos^{-1} COSTH\right)\ \text{sgn}\ SINTH$
        Return
    $THETA = \sin^{-1} SINTH$
    If $COSTH \geq 0$, Return
    $THETA = \frac{1}{2}\ \text{sgn}\ SINTH\ -\ THETA$     (the $\frac{1}{2}$ corresponds to 180°)
    Return
```

CALCGA

```
$T\underline{S} = \text{unit}(\underline{X}_{dc} * \underline{Y}_{sm})$
$COSTH = T\underline{S} \cdot \underline{Z}_{dc}$
$SINTH = T\underline{S} \cdot \underline{Y}_{dc}$
Perform "ARCTRIG"
$OGC = THETA$
$COSTH = (T\underline{S} * \underline{X}_{dc}) \cdot \underline{Y}_{sm}$
$SINTH = \underline{X}_{dc} \cdot \underline{Y}_{sm}$
Perform "ARCTRIG"
$MGC = THETA$
If $|MGC| - K_{p166} \geq 0$:
    Perform "ALARM"  (pattern $0401_8$)
    Set bit 14(GLOKFAIL) of FLAGWRD3 = 1
$COSTH = \underline{Z}_{sm} \cdot T\underline{S}$
$SINTH = \underline{X}_{sm} \cdot T\underline{S}$
Perform "ARCTRIG"
$IGC = THETA$
$THETA\underline{D} = (OGC, IGC, MGC)$  converted to twos complement single
                                        precision, scale factor B-1 revolutions
Return
```

AXISGEN

Entered with stars "A" and "B" in the "C" and "D" systems

```
$ST\underline{V} = \text{unit}(STARA\underline{C} * STARB\underline{C})$
$ST\underline{W} = STARA\underline{C} * ST\underline{V}$
$ST\underline{V}' = \text{unit}(STARA\underline{D} * STARB\underline{D})$
$ST\underline{W}' = STARA\underline{D} * ST\underline{V}'$
```

$$\underline{X}_{dc} = \text{unit}\ (\text{STARAD}_x\ \text{STARA}\underline{C} + \text{STV}'_x\ \text{ST}\underline{V} + \text{STW}'_x\ \text{ST}\underline{W})$$

$$\underline{Y}_{dc} = \text{unit}\ (\text{STARAD}_y\ \text{STARA}\underline{C} + \text{STV}'_y\ \text{ST}\underline{V} + \text{STW}'_y\ \text{ST}\underline{W})$$

$$\underline{Z}_{dc} = \text{unit}\ (\text{STARAD}_z\ \text{STARA}\underline{C} + \text{STV}'_z\ \text{ST}\underline{V} + \text{STW}'_z\ \text{ST}\underline{W})$$

$$[\text{STARAD}] = \begin{bmatrix} \underline{X}_{dc} \\ \underline{Y}_{dc} \\ \underline{Z}_{dc} \end{bmatrix}$$

(for "C" system in reference coordinates and "D" system in IMU coordinates, $\underline{X}_{dc}$ is $\underline{X}_{sm}$ in reference coordinates)

Return

<u>CDUTRIG</u>

```
    CDUSPOT = CDU    (most significant halves, in y,z,x order)
    CDUSPOT = CDUSPOT, converted to double precision ones complement,
                       B0 revolutions
    (COSCDUY, COSCDUZ, COSCDUX) = cos CDUSPOT
    (SINCDUY, SINCDUZ, SINCDUX) = sin CDUSPOT
    Return
```

<u>QUICTRIG</u> (Entered from "CLOCKJOB" with CDUSPO$\underline{T}$ set)

```
    Inhibit interrupts
    (SINCDUY, SINCDUZ, SINCDUX) = $\sin_{sp}$ $\text{CDUSPO}\underline{T}_{sp}$, rescaled to scale
                                                           factor B1
    (COSCDUY, COSCDUZ, COSCDUX) = $\cos_{sp}$ $\text{CDUSPO}\underline{T}_{sp}$, rescaled to scale
                                                           factor B1
    Release interrupts
    Return
```

TRG*NBSM Entered with $T\underline{S}$ the vector to be rotated thru o,m,i

Perform "CDUTRIG" (starting at second line)

$TS_2 = TS_y \text{ COSCDUX} - TS_z \text{ SINCDUX}$ (tag to start here is "*NBSM*")

$TS_1 = TS_z \text{ COSCDUX} + TS_y \text{ SINCDUX}$

$T\underline{S} = (TS_x, TS_2, TS_1)$

$TS_2 = TS_x \text{ COSCDUZ} - TS_y \text{ SINCDUZ}$

$TS_1 = TS_y \text{ COSCDUZ} + TS_x \text{ SINCDUZ}$

$T\underline{S} = (TS_2, TS_1, TS_z)$

$TS_2 = TS_z \text{ COSCDUY} - TS_x \text{ SINCDUY}$

$TS_1 = TS_x \text{ COSCDUY} + TS_z \text{ SINCDUY}$

$T\underline{S} = (TS_1, TS_y, TS_2)$

Return

SMNB Entered with $T\underline{S}$ the vector to be rotated thru i,m,o

$TS_2 = TS_z \text{ COSCDUY} + TS_x \text{ SINCDUY}$

$TS_1 = TS_x \text{ COSCDUY} - TS_z \text{ SINCDUY}$

$T\underline{S} = (TS_1, TS_y, TS_2)$

$TS_2 = TS_x \text{ COSCDUZ} + TS_y \text{ SINCDUZ}$

$TS_1 = TS_y \text{ COSCDUZ} - TS_x \text{ SINCDUZ}$

$T\underline{S} = (TS_2, TS_1, TS_z)$

$TS_2 = TS_y \text{ COSCDUX} + TS_z \text{ SINCDUX}$

$TS_1 = TS_z \text{ COSCDUX} - TS_y \text{ SINCDUX}$

$T\underline{S} = (TS_x, TS_2, TS_1)$

Return

SMCDURES

$$\text{DCDU}_x = \frac{\text{DTHETASM}_x \text{ COSCDUY} - \text{DTHETASM}_z \text{ SINCDUY}}{\text{COSCDUZ}}$$

```
    $DCDU_y = DTHETASM_y - DCDU_x \; SINCDUZ$
    $DCDU_z = DTHETASM_x \; SINCDUY + DTHETASM_z \; COSCDUY$
    Return
CALCSMSC
    $\underline{X}_{dc} = (COSCDUY \; COSCDUZ, \;\; SINCDUZ, \;\; - SINCDUY \; COSCDUZ)$
    $\underline{Z}_{dc} = (COSCDUY \; SINCDUZ \; SINCDUX + SINCDUY \; COSCDUX,$
                            $- COSCDUZ \; SINCDUX,$
            $COSCDUY \; COSCDUX - SINCDUY \; SINCDUZ \; SINCDUX)$
    $\underline{Y}_{dc} = \underline{Z}_{dc} * \underline{X}_{dc}$
    Return
SXTNB
    $TS_1 = (E_{5-X1})_{sp}$, converted to ones complement double precision,
                     scale factor B-2 revolutions (i.e. CDUT)
    $TS_1 = TS_1 + K_{10dgm}$, rescaled to B0 revolutions
    $TS_2 = (E_{3-X1})_{sp}$, converted to ones complement double precision,
                     scale factor B0 revolutions (i.e. CDUS)
    $T\underline{S} = (\sin TS_1 \cos TS_2, \;\; \sin TS_1 \sin TS_2, \;\; \cos TS_1)$
    $T\underline{S} = \left[NB1NB2\right] \; T\underline{S}$
    Return                  ($T\underline{S}$ in 32D)
CALCSXA
    $T\underline{S} = STA\underline{R}$
    Perform "*SMNB*"
    $STA\underline{R} = \left[NB2NB1\right] \; T\underline{S}$
    $\underline{X}_{nb1} = unit\underline{X}$
    $\underline{Y}_{nb1} = unit\underline{Y}$
    $\underline{Z}_{nb1} = unit\underline{Z}$
    Proceed to "SXTANG1"
```

```
SXTANG
    $\underline{TS}_1 = [NB2NB1] \; (X_{dc,x}, Y_{dc,x}, Z_{dc,x})$
    $\underline{TS}_2 = [NB2NB1] \; (X_{dc,y}, Y_{dc,y}, Z_{dc,y})$
    $\underline{TS}_3 = [NB2NB1] \; (X_{dc,z}, Y_{dc,z}, Z_{dc,z})$
    $\underline{X}_{nbl} = (TS_{1,x}, TS_{2,x}, TS_{3,x})$
    $\underline{Y}_{nbl} = (TS_{1,y}, TS_{2,y}, TS_{3,y})$
    $\underline{Z}_{nbl} = (TS_{1,z}, TS_{2,z}, TS_{3,z})$
    Proceed to "SXTANG1"
SXTANG1
    $\underline{TS}_1 = unit(\underline{Z}_{nbl} * \underline{STAR})$
    If overflow has taken place (i.e. unit vector poorly defined):
        $SAC = K_{270dg}$
        $PAC = K_{20dgs}$
        Set bit 7(CULTFLAG) of FLAGWRD3 = 0
        Return
    $SINTH = - \underline{TS}_1 \cdot \underline{X}_{nbl}$
    $COSTH = \underline{TS}_1 \cdot \underline{Y}_{nbl}$
    Perform "ARCTRIG"
    SAC = THETA, rescaled to B-1 revolutions single precision twos
                 complement
    $TS = \cos^{-1} (\underline{STAR} \cdot \underline{Z}_{nbl})$
    If $TS \geqslant 0$:            (as it should be, i.e. in range 0 - 180°)
        If TS < 90°:
            TS = TS, rescaled to B-2 revolutions
            $TS = TS - K_{20dgsm}$
            PAC = TS, rescaled to B-3 revolutions single precision
                      twos complement
            Set bit 7(CULTFLAG) of FLAGWRD3 = 0
            Return
```

```
        Set bit 7(CULTFLAG) of FLAGWRD3 = 1

        Return

EARROT1

        $\underline{R}_{tes} = \underline{C}_{unitw} * \underline{R}_{ti}$

        $\underline{R}_{tnm} = \underline{R}_{tes} * \underline{C}_{unitw}$

        Proceed to "EARROT2"

EARROT2

        $TS = DTEAROT/K_{1dwie}$

        If $|TS| \geq 1$:

                $DTEAROT = DTEAROT - K_{1dwie}$ sgn DTEAROT

                Proceed to "EARROT2"

        $\underline{R}_{t} = \text{unit}\left(\underline{R}_{ti} + (\cos TS - 1)\,\underline{R}_{tnm} + (\sin TS)\,\underline{R}_{tes}\right)$

        Return

LSPOS           Entered with TS = GET of information, B28 centi-seconds, from
                                        "ACCOMP", "S50", and "UTAREA1"
        Use X1 for program control purposes

        $TIMEARG = T_{eph} + TS - C_{timemo}$, modulo $2^{26}$ centi-seconds (7 days, 18 hr,
                                                                24 min, 48.64 sec)

        $TS = C_{omegaes}\ TIMEARG$

        $\underline{TS}_{1} = \left(\text{unit}(\text{unit}\underline{C}_{reso} * \underline{C}_{veso}) * \underline{C}_{reso}\right) \sin TS + \underline{C}_{reso} \cos TS$

                                                ($\underline{TS}_{1}$ B38 meters, in pd 2D)

        Proceed to third line of "LUNPOS"

LUNPOS          Entered with TS = GET of information, B28 centi-seconds,
                                        from "CHKSWTCH", "POINTAXS",
        Use X1 for program control purposes     and "SBANDANT"

        $TIMEARG = T_{eph} + TS - C_{timemo}$, modulo $2^{26}$ centi-seconds

        $\underline{TS} = \underline{C}_{vcem54} + \underline{C}_{vcem48}\ TIMEARG + \underline{C}_{vcem42}\ TIMEARG^{2} + \underline{C}_{vcem36}\ TIMEARG^{3}$
                $+ \underline{C}_{vcem30}\ TIMEARG^{4} + \underline{C}_{vcem24}\ TIMEARG^{5} + \underline{C}_{vcem18}\ TIMEARG^{6}$
                $+ \underline{C}_{vcem12}\ TIMEARG^{7} + \underline{C}_{vcem06}\ TIMEARG^{8} + \underline{C}_{vcem00}\ TIMEARG^{9}$
```

Return (T$\underline{S}$ in MPAC, B29 meters, moon's position)

LUNVEL Entered with TS = GET of information, B28 centi-seconds, from "ORIGCHNG"

Use X1 for program control purposes

$\text{TIMEARG} = T_{eph} + \text{TS} - C_{timemo}$, modulo 2^{26} centi-seconds

$$T\underline{S} = 1\, \underline{C}_{vcem48} + 2\, \underline{C}_{vcem42}\, \text{TIMEARG} + 3\, \underline{C}_{vcem36}\, \text{TIMEARG}^2 +$$
$$4\, \underline{C}_{vcem30}\, \text{TIMEARG}^3 + 5\, \underline{C}_{vcem24}\, \text{TIMEARG}^4 + 6\, \underline{C}_{vcem18}\, \text{TIMEARG}^5 +$$
$$7\, \underline{C}_{vcem12}\, \text{TIMEARG}^6 + 8\, \underline{C}_{vcem06}\, \text{TIMEARG}^7 + 9\, \underline{C}_{vcem00}\, \text{TIMEARG}^8$$

Return (T$\underline{S}$ in MPAC, B7 meters/centi-second, moon's velocity)

LAT-LONG

$\text{ALPHAM} = |\text{ALPHA}\underline{V}|$

$T\underline{S}_1 = \text{ALPHA}\underline{V}$

$\text{TS}_2 = \text{TS}$ (time argument when enter)

TS = 0

If bit 12(LUNLATLO) of FLAGWRD3 = 1:

TS = ½ (i.e. non-zero quantity, formed as cos 0°, with scale factor B1)

Perform "R-TO-RP"

$\text{ALPHA}\underline{V} = \text{unit}T\underline{S}$

$\text{TS} = K_{b2a2}$

If bit 12(LUNLATLO) of FLAGWRD3 = 1:

$\text{TS} = K_{1b1}$

GAMRP = TS

Perform "SETRE"

$\text{COSTH} = \text{GAMRP} \sqrt{\text{ALPHAV}_x^{\ 2} + \text{ALPHAV}_y^{\ 2}}$

$\text{SINTH} = \text{ALPHAV}_z$

Perform "ARCTAN"

LAT = THETA

```
$COSTH = ALPHAV_x$
$SINTH = ALPHAV_y$
Perform "ARCTAN"
$LONG = THETA$
$ALT = ALPHAM - ERADM$
Return
```

LALOTORV

```
$TS_2 = TS$   (time argument when enter)
$TS = K_{b2a2}$
If bit 12(LUNLATLO) of FLAGWRD3 = 1:
    $TS = K_{1b1}$
$GAMRP = TS$          (enter here from "P11" for [REFSMMAT])
$ALPHA\underline{V} = unit(\cos LONG \cos LAT,\ \sin LONG \cos LAT,\ GAMRP \sin LAT)$
$T\underline{S}_1 = ALPHA\underline{V}$
Perform "SETRE"
$TS = 0$
If bit 12(LUNLATLO) of FLAGWRD3 = 1:
    $TS = \frac{1}{2}$  (i.e. non-zero quantity, formed as cos $0^{\circ}$, with scale
                         factor B1)
Perform "RP-TO-R" ($TS_2$ still set from entrance to "LALOTORV")
$ALPHA\underline{V} = T\underline{S}$
$ALPHA\underline{V} = (ERADM + ALT)\ ALPHA\underline{V}$
Return
```

GETERAD

```
$ERADM = \sqrt{\frac{K_{b2xsc}}{1 - K_{ee}(1 - ALPHAV_z^{2})}}$
Return
```

SETRE

```
If bit 12(LUNLATLO) of FLAGWRD3 = 0:
    ERADM = $K_{erad}$
    If bit 13(ERADCOMP) of FLAGWRD1 = 1:
        Perform "GETERAD"   (writes over ERADM)
    Return
ERADM = $K_{504rm}$        (i.e. LUNLATLO = 1, meaning lunar)
If bit 13(ERADCOMP) of FLAGWRD1 = 0:
    ERADM = $|\underline{RLS}|$
Return
```

ARCTAN

```
TS = $SINTH^2 + COSTH^2$
If TS = 0:
    THETA = 0
    Return
TS = SINTH / $\sqrt{TS}$
If $|TS| \geq 1$:
    THETA = $\frac{1}{4}$ sgn SINTH        (the $\frac{1}{4}$ is $90^\circ$)
    Return
THETA = $\sin^{-1}$ TS
If  COSTH $\geq$ 0,  Return
THETA = $\frac{1}{2}$ sgn THETA  -  THETA  (the $\frac{1}{2}$ is $180^\circ$)      (0 considered
                                                                                    negative)
Return
```

```
RP-TO-R        Entered with $T\underline{S}_1$ in OD to be rotated, $TS_2$ in 6D for the
               time, and TS in MPAC for earth (0) or moon ($\neq 0$)

    If $TS_{sp} = 0$:        (means earth)
        Perform "EARTHMX"
        $T\underline{S} = [MMATRIX]\ 504LP\underline{L}$
        Skip next 2 lines
    Perform "MOONMX"         (i.e. original $TS_{sp} \neq 0$)
    $T\underline{S} = \underline{C}_{5041m}$
    $T\underline{S} = (T\underline{S}_1 + T\underline{S} * T\underline{S}_1)\ [MMATRIX]$
    Return

R-TO-RP        Entered with same calling arguments as "RP-TO-R"

    If $TS_{sp} = 0$:
        Perform "EARTHMX"
        $T\underline{S} = 504LP\underline{L}$
        Skip next 2 lines
    Perform "MOONMX"         (i.e. original $TS_{sp} \neq 0$)
    $T\underline{S} = \underline{C}_{5041m}\ [MMATRIX]$
    $T\underline{S} = [MMATRIX]\ (T\underline{S}_1 - T\underline{S} * T\underline{S}_1)$
    Return

EARTHMX

    $AZ = K_{wearth}(T_{eph} + TS_2) + K_{azo}$, modulo $360^{\circ}$   (time argument modulo
                                                   $2^{33}$ cs, or about 994.2 days)
    $[MMATRIX] = \begin{bmatrix} \cos AZ & \sin AZ & 0 \\ -\sin AZ & \cos AZ & 0 \\ 0 & 0 & 1 \end{bmatrix}$
    $504LP\underline{L} = (-C_{unitw_y},\ C_{unitw_x},\ 0)$
    Return
```

MOONMX

$\mathrm{BVAL} = K_{bdot}\ (T_{eph} + TS_2)\ + K_{bsubo}$

$\mathrm{FVAL} = K_{fdot}\ (T_{eph} + TS_2)\ + K_{fsubo}$

(all angles modulo 360°, but time modulo 2^{33} cs, or ~ 994.2 days)

$\mathrm{OMGIVAL} = K_{noddot}\ (T_{eph} + TS_2)\ +\ K_{nodio}$

$\mathrm{AVECT}\underline{\mathrm{R}} = (\cos \mathrm{OMGIVAL},\ \sin \mathrm{OMGIVAL} \cos \mathrm{BVAL},\ \sin \mathrm{OMGIVAL} \sin \mathrm{BVAL})$

$\mathrm{BVECT}\underline{\mathrm{R}} = (-\sin \mathrm{OMGIVAL},\ \cos \mathrm{OMGIVAL} \cos \mathrm{BVAL},\ \cos \mathrm{OMGIVAL} \sin \mathrm{BVAL})$

$\mathrm{CVECT}\underline{\mathrm{R}} = (0,\ -\sin \mathrm{BVAL},\ \cos \mathrm{BVAL})$

$\mathrm{MMATR}\underline{\mathrm{IX}}_6 = K_{sini}\ \mathrm{BVECT}\underline{\mathrm{R}}\ +\ K_{cosi}\ \mathrm{CVECT}\underline{\mathrm{R}}$

$\mathrm{DVECT}\underline{\mathrm{R}} = K_{cosi}\ \mathrm{BVECT}\underline{\mathrm{R}}\ -\ K_{sini}\ \mathrm{CVECT}\underline{\mathrm{R}}$

$\mathrm{MMATR}\underline{\mathrm{IX}}_3 = \mathrm{AVECT}\underline{\mathrm{R}} \sin \mathrm{FVAL}\ -\ \mathrm{DVECT}\underline{\mathrm{R}} \cos \mathrm{FVAL}$

$\mathrm{MMATR}\underline{\mathrm{IX}}_0 = -\mathrm{AVECT}\underline{\mathrm{R}} \cos \mathrm{FVAL}\ -\ \mathrm{DVECT}\underline{\mathrm{R}} \sin \mathrm{FVAL}$

$$\left[\mathrm{MMATRIX}\right] = \begin{bmatrix} \mathrm{MMATR}\underline{\mathrm{IX}}_0 \\ \mathrm{MMATR}\underline{\mathrm{IX}}_3 \\ \mathrm{MMATR}\underline{\mathrm{IX}}_6 \end{bmatrix}$$

Return

Quantities in Computations

See also list of major variables and list of routines

504LPL: Vector "accounting for the precession and nutation of the polar axis (the deviation of the true pole from the mean pole)," scale factor B0, units radians.

ALPHAM: See Orbital Integration.

ALPHAV: Communication cell with coordinate transformation routines. If used with "GETERAD", contains a unit vector of position, scale factor B1; for latitude-longitude routines, contains position vector with scale factor B29, units meters, either when enter or when exit.

ALT: Value of altitude above appropriate base value, scale factor B29, units meters.

AVECTR: Temporary vector used in "MOONMX" routine, scale factor B1, stored in push-down list location 20D.

AZ: Value of rotation angle about Z axis, modulo 1 revolution, computed in "EARTHMX". Scale factor B0, units revolutions, stored in push-down list location 18D (with notation "504AZ").

BVAL: Value of the obliquity (angle between mean earth equatorial plane and the plane of the ecliptic) computed in "MOONMX" routine, scale factor B0, units revolutions, stored temporarily in push-down list. Its sine is stored in 34D and its cosine in 32D, both with scale factor B1, notations "SOB" and "COB" respectively.

BVECTR: Temporary vector used in "MOONMX" routine, scale factor B1, stored in push-down list location 26D.

$\underline{C}_{504lm}$: Erasable memory vector constant, scale factor B0, used to "correct for deviations because of physical libration" of moon. Program notation is "504LM", and units radians.

$C_{omegaes}$: Erasable memory constant, program notation "OMEGAES", scale factor B-26, units revolutions/csec:gives the "angular velocity of the vector $\underline{C}_{reso}$ at time C_{timemo}".

$\underline{C}_{reso}$: Erasable memory vector constant, scale factor B38, units meters, program notation "RESO". It gives the position vector of the sun relative to the earth at time C_{timemo}.

C_{timemo}: Erasable memory triple precision scalar constant, scalc factor B42, units centi-seconds, program notation "TIMEMO". It gives the elapsed time between the beginning of the AGC year (cf. K_{azo}) and when the lunar and solar location-determining erasable memory constants are valid.

$\underline{C}_{unitw}$: See General Program Control.

$\underline{C}_{vcem00}$: Erasable memory vector constant, program notation "VECOEM", scale factor B-203, units meters/centi-second^9, used as the coefficient of the highest order term in computing lunar position or velocity. Could alternatively be considered to be in units of "quasi-weeks" (of 2^{26} centi-seconds each, see TIMEARG), in which case scale factor is B31, units meters/quweeks^9.

$\underline{C}_{vcem06}$: Erasable memory vector constant, program notation "VECOEM +6", scale factor B-177, units meters/centi-second^8, used as the coefficient of the next highest order term in computing lunar position or velocity. Could also be considered B31 in units of meters/quweeks^8 (see $\underline{C}_{vcem00}$).

$\underline{C}_{vcem6i}$ (i = 2-7): Erasable memory vector constants, program notation "VECOEM +6i", used as coefficients for subsequent terms of lunar position or velocity computations. Note that the coefficients are stored in memory with the highest-order one <u>first</u>. Scale factors B-(203 - 26i), units meters/centi-second^{9-i} (or scale factor B31, units meters/quweeks^{9-i}).

$\underline{C}_{vcem48}$: Erasable memory vector constant, program notation "VECOEM +48", scale factor B5, units meters/centi-second, used as the first-order ("bias") term for lunar velocity. Exponent factors are scaled B4 when used for velocity multiplications, giving a "raw" velocity scaling of B9, which at the end is shifted left 2 places to give B7. Could also be considered scale factor B31, units meters/quweek.

$\underline{C}_{vcem54}$: Erasable memory vector constant, program notation "VECOEM +54", scale factor B31, units meters, used as the first-order ("bias") term for lunar position. The "raw" position scaling is B31, which at the end is shifted left 2 places to give B29 for output.

$\underline{C}_{veso}$: Erasable memory vector constant, program notation "VESO", scale factor B9, units meters/centi-second. It gives the velocity vector of the sun relative to the earth at time C_{timemo}. Used also in P23.

CDUSPO$\underline{T}$: Communication cells with coordinate transformation routines to specify (generally) the CDU angle information to be employed. When enter the routine, only the most significant half of each double precision word is used, considered to have a scale factor B-1, units revolutions in twos complement. Conventional order of storage of CDU angle information in CDUSPO$\underline{T}$ is y,z,x (i.e. inner, middle, outer respectively).

COSCDUX, COSCDUY, COSCDUZ: Value of cosines of information in CDUSPO$\underline{T}$, scale factor B1 (in the same y,z,x order as CDUSPO$\underline{T}$). If perform "QUICTRIG", a single-precision cosine routine is used, while if perform "CDUTRIG" the double precision interpretive routine is employed (taking about ten times more execution time). Updated once a second while "CLOCKJOB" is running (via "QUICTRIG" entrance), which includes the times that the TVC DAP is on.

COSTH: Value of argument cosine for "ARCTRIG", stored in push-down list location 16D, scale factor B2. Also used as communication cell with "ARCTAN", in which case only requirement is that scaling be the same as for SINTH to that routine.

CVECTR: Temporary vector used in "MOONMX" routine, scale factor B1, stored in push-down list location 8D.

DCDU: Incremental change in CDU angle due to DTHETASM (computed in "SMCDURES"), same scaling and units as DTHETASM.

DTEAROT: Value of earth rotation time for use in "EARROT2", scale factor B28, units centi-seconds.

DTHETASM: Communication cell with "SMCDURES" containing incremental angle changes measured about stable member axes. Scaling of DCDU (and units) will be the same as the scaling of this quantity: when entered from "CRS61.1", scale factor is B-3, with units of revolutions/deci-second.

DVECTR: Temporary vector used in "MOONMX" routine, scale factor B1, stored in push-down list location 8D.

ERADM: Value of radius of earth (from Fischer model or pad base) or moon (of average value or landing site base), scale factor B29, units meters.

FVAL: Value of angle from the mean ascending node of the moon's orbit to the mean moon computed in "MOONMX" routine, scale factor B0, units revolutions, stored temporarily originally in push-down list location 30D, then moved to 6D (where has notation "504F").

GAMRP: Value of square of ratio of polar to equatorial model radius (1.0 for moon), scale factor B1, stored in push-down list location 8D.

IGC: Value of inner gimbal angle, scale factor B0, units revolutions. The inner gimbal angle is identified with the Y axis.

K_{1b1}: Constant, program notation "1B1", scale factor B1, value 1×2^{-1}, corresponding to an equation value of 1.0 (the ratio of polar to equatorial model radius information for the moon).

K_{1dwie}: Constant, program notation "1/WIE", scale factor B28, units centi-seconds. Value is 8616410×2^{-28}, corresponding to 86164.10 seconds, the earth's rotation period (rounded to the nearest centi-second). Corresponding angular velocity to this time is about 7.292115054E-5 rad/sec (cf. K_{wearth}).

K_{10dgm}: Single precision constant, program notation "10DEGS-", scale factor B-2, units revolutions. Value is 3600×2^{-14}, corresponding to about 19.7754°. Notation arises from the fact that would be about 10° for the standard trunnion angle scaling of B-3 revolutions.

K_{20dgs}: Constant, program notation "20DEGS-", scale factor B-3, units revolutions. Value is -7199×2^{-14}, corresponding to -19.7754° (since used as if it was twos complement, equivalent to -7200×2^{-14} in ones complement).

K_{20dgsm}: Constant, program notation "20DEG-", scale factor B-2, units revolutions. Value is 3600 x 2^{-14}, corresponding to about 19.7754°.

K_{270dg}: Constant, program notation "270DEG", scale factor B-1, units revolutions. Value is 60000_8 00000_8, corresponding to -90° in twos complement (or + 270°).

K_{504rm}: Constant, program notation "504RM", scale factor B29, units meters. Value is 1738090 x 2^{-29}, corresponding to the mean lunar radius of 1738.09 km.

K_{azo}: Constant, program notation "AZO", scale factor B0, units revolutions. Value is 7.733314844E-1 revolutions, corresponding to 4.85898502016 x $(1/2\pi)$, where first term is value in radians and second converts to revolutions. It is "the angle between the X-axis of the Basic Reference Coordinate System and the X-axis of the Earth-fixed Coordinate System (intersection of Greenwich meridian and equatorial plane of the earth)" at the beginning of the AGC year (i.e. when T_{eph} + T_{now} equals zero).

K_{b2a2}: Constant, program notation "B2/A2", scale factor B1, value 0.9933064884 x 2^{-1}. Value corresponds to $(6356.784/6378.166)^2$ x 2^{-1}, where terms in parentheses are the Fischer polar and equatorial earth radius values expressed in kilometers, to nearest meter.

K_{b2xsc}: Constant, program notation "B2XSC", scale factor B51, value 0.0179450689. Value corresponds to $(6356.784E3)^2$ x 2^{-51}, where first term is Fischer polar radius in meters (squared), and second is scale factor.

K_{bdot}: Constant, program notation "BDOT", scale factor B-28, units revolutions/centi-second. Nominal value is -1.145531390E-16 x 2^{28}, corresponding to -7.19758599677E-14 x $(1/2\pi)$ x 10^{-2} x 2^{28}, where first term is rate of change of BVAL in radians/sec, second converts to revolutions, third converts to centi-seconds, and fourth is scale factor. Memory value is - 2^{-25} (-2.98023223877E-8).

K_{bsubo}: Constant, program notation "BSUBO", scale factor B0, units revolutions. Value is 6.511941688E-2, corresponding to 4.09157363336E-1 x $(1/2\pi)$, where first term is value in radians and second converts to revolutions.

K_{cosi}: Constant, program notation "COSI", scale factor B1, value 9.996417320E-1 x 2^{-1}. Value corresponds approximately to cosine 1° 32' 1.5" (5521.5 arc seconds), the "angle between the mean lunar equatorial plane and the plane of the ecliptic."

K_{ee}: Constant, program notation "EE", scale factor B0, value 6.6935116E-3. Value corresponds to (1 - K_{b2a2}), i.e. the square of Fischer model eccentricity (for K_{b2a2} values).

K_{erad}: Constant, program notation "ERAD", scale factor B29, units meters. Value is 6,373,338 x 2^{-29}, corresponding to 6,373,338 meters (same as K_{rpad}, see Burn Control).

K_{fdot}: Constant, program notation "FDOT", scale factor B-27, units revolutions/centi-second. Value is 4.253263471E-9 x 2^{27}, corresponding to 2.6724042548E-6 x (1/2π) x 10^{-2} x 2^{27}, where first term is rate of change of FVAL in radians/sec, second converts to revolutions, third converts to centi-seconds, and fourth is scale factor.

K_{fsubo}: Constant, program notation "FSUBO", scale factor B0, units revolutions. Value is 6.55273775E-1, corresponding to 4.11720655556 x (1/2π), where first term is value in radians and second converts to revolutions.

K_{noddot}: Constant, program notation "NODDOT", scale factor B-28, units revolutions/centi-second. Value is -1.703706128E-11 x 2^{28}, corresponding to -1.070470131E-8 x (1/2π) x 10^{-2} x 2^{28}, where first term is rate of change of OMGIVAL in radians/second, second converts to revolutions, third converts to centi-seconds, and fourth is scale factor.

K_{nodio}: Constant, program notation "NODIO", scale factor B0, units revolutions. Value is 8.7883086E-1, corresponding to 5.521857147 x (1/2π), where first term is value in radians and second converts to revolutions.

K_{p166}: Constant, program notation ".166...", scale factor B0, units revolutions. Value is 0.1666666667, corresponding to 60 degrees.

K_{qts4}: Constant, program notation "QTSN45", scale factor B2. Value is 0.1768, corresponding to 0.7072 x 2^{-2}, where first term is sin 45° and second is scale factor.

K_{sini}: Constant, program notation "SINI", scale factor B1, value 2.676579050E-2 x 2^{-1}, corresponding to the sine of the angle whose cosine is K_{cosi}.

K_{wearth}: Constant, program notation "WEARTH", scale factor B-23, units revolutions/centi-second. Value is 1.160576171E-7 x 2^{23}, corresponding to 7.29211514667E-5 x (1/2π) x 10^{-2} x 2^{23}, where first term is rotation rate of earth in radians/second, second converts to revolutions, third converts to centi-seconds, and fourth is scale factor. Octal value is 37116_8 32631_8, corresponding to about 0.9735618569, or a rate of about 7.2921151577E-5 rad/sec (a period of about 86164.0988 seconds).

LAT: Value of vehicle latitude, scale factor B0, units revolutions, positive for northerly latitudes. Notation also "LANDLAT". If earth, geodetic.

LONG: Value of vehicle longitude, scale factor B0, units revolutions, positive for easterly longitudes.

MGC: Value of middle gimbal angle, scale factor B0, units revolutions. The middle gimbal angle is identified with the Z axis.

[MMATRIX] : Value of "modification matrix" for earth (computed in "EARTHMX") or moon (computed in "MOONMX"), scale factor of elements B1, stored in push-down list location 20D.

[NB1NB2] : Matrix of constants, program notation "NB1NB2", scale factor B1. True values of elements (which are multiplied by 2^{-1} for scale factor purposes) are:

$$\begin{bmatrix} 0.843175692 & 0 & 0.5376381241 \\ 0 & 1 & 0 \\ -0.5376381241 & 0 & 0.843175692 \end{bmatrix}$$

Numerical values in first row correspond to the cosine and sine of 32° 31' 23.19".

[NB2NB1] : Matrix of constants, program notation "NB2NB1", scale factor B1. True values of elements (which are multiplied by 2^{-1} for scale factor purposes) are:

$$\begin{bmatrix} 0.843175692 & 0 & -0.5376381241 \\ 0 & 1 & 0 \\ 0.5376381241 & 0 & 0.843175692 \end{bmatrix}$$

Matrix is the transpose of [NB1NB2] .

OGC: Value of outer gimbal angle, scale factor B0, units revolutions. The outer gimbal angle is identified with the X axis. The quantity is stored such that $\underline{\text{OGC}}$ = (OGC, IGC, MGC).

OMGIVAL: Value of the "longitude of the node of the moon's orbit", scale factor B0, units revolutions, computed in "MOONMX". Value is stored temporarily in push-down list location 8D.

PAC: Required trunnion angle to have the star line-of-sight lie along $\underline{\text{STAR}}$, scale factor B-3, units revolutions (double precision).

$\underline{R}_t$: Target vector output of "EARROT2", scale factor B1 (a unit vector).

$\underline{R}_{tes}$: Value of easterly component of $\underline{R}_{ti}$, scale factor B1 (dimensionless), computed in "EARROT1".

$\underline{R}_{ti}$: Communication cell with earth rotation routine, containing initial value of target vector (at DTEAROT = 0), scale factor B1 (a unit vector).

$\underline{R}_{tnm}$: Value of normal component of $\underline{R}_{ti}$, scale factor B1 (dimensionless), computed in "EARROT1".

$\mathrm{RL}\underline{\mathrm{S}}$: Value of landing site vector (in moon-fixed coordinates), scale factor B27, units meters. Can be updated by e.g. P22 (or P27, of course), and forms part of the pre-launch erasable load.

SAC: Required optics shaft angle to have the star line-of-sight lie along $\mathrm{STA}\underline{\mathrm{R}}$, scale factor B-1, units revolutions (double precision).

SINCDUX, SINCDUY, SINCDUZ: Values of sines of information in $\mathrm{CDUSPO}\underline{\mathrm{T}}$, scale factor B1 (in the same y,z,x order as $\mathrm{CDUSPO}\underline{\mathrm{T}}$). See COSCDUX.

SINTH: Value of argument sine for "ARCTRIG", stored in push-down list location 18D, scale factor B2. Also used as communication cell with "ARCTAN", in which case only requirement is that scaling be the same as for COSTH to that routine.

$\mathrm{STA}\underline{\mathrm{R}}$: Unit vector, scale factor B1, specifying the (unit) vector direction to the appropriate star. It is in present stable member coordinates if "CALCSXA" is entered, and the coordinate system of $\left[X_{dc}\right]$ if "SXTANG" is entered.

$\mathrm{STARA}\underline{\mathrm{C}}$: Unit vector, scale factor B1, used as communication cell with "AXISGEN", giving direction to "star A" in the "C" coordinate system. Stored in push-down list location 6D.

$\mathrm{STARA}\underline{\mathrm{D}}$: Unit vector, scale factor B1, used as communication cell with "AXISGEN", giving direction to "star A" in the "D" coordinate system. Stored in cells starting at "STARAD".

[STARAD] : Matrix, elements with scale factor B1, the output of "AXISGEN". It defines "coordinate system D referred to coordinate system C".

$\mathrm{STARB}\underline{\mathrm{C}}$: Unit vector, scale factor B1, used as communication cell with "AXISGEN", giving direction to "star B" in the "C" coordinate system. Stored in push-down list location 12D.

$\mathrm{STARB}\underline{\mathrm{D}}$: Unit vector, scale factor B1, used as communication cell with "AXISGEN", giving direction to "star B" in the "D" coordinate system. Stored in cells starting at "STARAD +6".

$\mathrm{ST}\underline{\mathrm{V}}$, $\mathrm{ST}\underline{\mathrm{V}}'$, $\mathrm{ST}\underline{\mathrm{W}}$, $\mathrm{ST}\underline{\mathrm{W}}'$: Vectors derived in "AXISGEN", scale factor B1 (the primed set pertain to the "D" coordinate system). $\mathrm{ST}\underline{\mathrm{V}}$ is stored in push-down list location 12D and $\mathrm{ST}\underline{\mathrm{W}}$ in location 18D; $\mathrm{ST}\underline{\mathrm{V}}'$ in the cell originally used for $\mathrm{STARB}\underline{\mathrm{D}}$; and $\mathrm{ST}\underline{\mathrm{W}}'$ in STARAD +12.

T_{eph}: See Boost Computations.

THETA: Output angle from "ARCTRIG" or "ARCTAN", scale factor B0, units revolutions, stored in push-down list location 20D. It is in the range $\pm \frac{1}{2}$ (i.e. ± 180 degrees).

TIMEARG: Value of time argument used in computing position and velocity of moon from polynomials, scale factor B26, units centi-seconds. stored in push-down list location OD (also used for sun position computation). For convenience in defining the scaling of the $\underline{C}_{vcemi}$ constants, could also be considered scaled B0 in units of quasi-weeks, where 1 quweek = 2^{26} centi-seconds, i.e. 7 days 18 hours 24 minutes 48.64 seconds. Total time range that can be accommodated is 15½ days (plus 49 minutes 37.27 seconds) between -MAX and +MAX for TIMEARG.

$\underline{X}_{nbl}$, $\underline{Y}_{nbl}$, $\underline{Z}_{nbl}$: Unit vectors, scale factor B1, used as input to "SXTANG1" to provide information on navigation base orientation in same coordinate system as STA$\underline{R}$ (when enter "SXTANG1").

Data Input/Output

```
KEYRUPT1    (Entered after receipt of program interrupt #5, main DSKY)

        $T_{st} = T_{now}$
        $TS_1$ = bits 5-1 (key code) of channel 15
        Proceed to "KEYCOM"
KEYCOM

        Set bit 15(DSKYFLAG) of FLAGWRD5 = 1    (means DSKY display output to
                                                  be generated in "T4RUPT")
        Establish "CHARIN" (priority $30_8$)

        $MPAC+0_{LOCCTR} = TS_1$      (LOCCTR indexes MPAC+0 cell associated with
                                         the "CHARIN" job)

        Resume

UPRUPT      (Entered after receipt of program interrupt #7, uplink input)

        $T_{st} = T_{now}$
        Set TS = INLINK and INLINK = 0

        If bit 12(NODOP01) of FLAGWRD1 = 0:

            UPSUM = UPSUM + TS     (modulo $2^{14}$)

            UPSUM+1 = UPSUM+1   + 1   (modulo $2^{14}$)

        Set bit 3(Uplink Activity) of channel 11 = 1

        $TS_1$ = bits 5-1 of TS

        If bits 10-6 of TS are not the complement of $TS_1$ (i.e. if these bits,
            shifted right 5 places, do not add with $TS_1$ to equal $11111_2$):

            Set bit 4(UPLOCKFL) of FLAGWRD7 = 1

            Resume

        If bits 15-11 of TS are not equal to $TS_1$ (i.e. if complement of these
            bits, shifted right 10 places, do not add with $TS_1$ to equal $11111_2$):

            Set bit 4(UPLOCKFL) of FLAGWRD7 = 1

            Resume

        If $TS_1 = 22_8$ (Error Reset):

            Set bit 4(UPLOCKFL) of FLAGWRD7 = 0

            Skip next line

        If bit 4(UPLOCKFL) of FLAGWRD7 = 1, Resume

        Proceed to second line of "KEYCOM"
```

CHARIN

```
    21d22REG = DSPLOCK
    DSPLOCK = 1
    If CADRSTOR $\neq$ 0:        (i.e. internal display waiting for response)
        If MPAC+0 $\neq 22_8$:     (input character not Error Reset)
            Set bit 5(Key Release) of channel 11 = 1
    CHAR = MPAC+0   (loaded in "KEYCOM" with keyboard or uplink data)
    If CHAR = $00_8$, proceed to "CHARALRM"
    If CHAR = $01_8 - 07_8$, proceed to "NUM"
    If CHAR = $10_8 - 11_8$:
        If DSPCOUNT $\leq$ -0, End of job
        If bits 2-1 of DECBRNCH = $00_2$, proceed to "CHARALRM"
        Proceed to "NUM"
    If CHAR = $12_8 - 17_8$, proceed to "CHARALRM"
    If CHAR = $20_8$:
        CHAR = $00_8$
        Proceed to "NUM"
    If CHAR = $21_8$, proceed to "VERB"
    If CHAR = $22_8$, proceed to "ERROR"
    If CHAR = $23_8 - 30_8$, proceed to "CHARALRM"
    If CHAR = $31_8$, proceed to "VBRELDSP"
    If CHAR = $32_8$, proceed to "POSGN"
    If CHAR = $33_8$, proceed to "NEGSGN"
    If CHAR = $34_8$, proceed to "ENTER"
    If CHAR = $35_8$, proceed to "CHARALRM"
    If CHAR = $36_8$, proceed to "CLEAR"
    If CHAR = $37_8$, proceed to "NOUN"
    (Since CHAR is limited to 5 bits, this table includes all values)
```

```
CHARALRM

    Set bit 7(Operator Error) of channel 11 = 1
    End of job

NUM

    If DSPCOUNT $\leq$ -0, End of job
    Perform "GETINREL"
    If CLPASS $>$ 0, set CLPASS = +0
    CODE = (bits 5-1 of $K_{rtb_{CHAR}}$)
    COUNT = DSPCOUNT
    Perform "DSPIN"
    If bits 2-1 of DECBRNCH = $00_2$:
        TS = $WDREG_{INREL}$, cycled left three places (bit 12 into bit 15 etc.)
        $WDREG_{INREL}$ = TS + CHAR
        If DSPCOUNT $\neq$ $K_{crt_{INREL}}$:
            DSPCOUNT = DSPCOUNT - 1, limited $\geq$ +0
            End of job
        DSPCOUNT = - DSPCOUNT
        End of job
    $WDREG_{INREL}$ = (10) ($WDREG_{INREL}$)  +  CHAR
    If $|WDREG_{INREL}| < 2^{14}$ (i.e. 16384):
        If DSPCOUNT $\neq$ $K_{crt_{INREL}}$:
            DSPCOUNT = DSPCOUNT - 1, limited $\geq$ +0
            End of job
    Perform "DECEND"
    DSPCOUNT = - DSPCOUNT
    End of job
```

DECEND

```
If  INREL - 1 $\leq$ 0:     (i.e. verb or noun)
    DSPCOUNT = - DSPCOUNT
    End of job
Perform "GETINREL"       (Enter here from "ENTER")
$(WDREG_{INREL}, LPREG_{INREL}) = K_{dsc}\ WDREG_{INREL}$
If bits 2-1 of DECBRNCH = $10_2$:
    $(WDREG_{INREL}, LPREG_{INREL}) = -(WDREG_{INREL}, LPREG_{INREL})$
Return
```

GETINREL

Set INREL in accordance with the following table:

DSPCOUNT	INREL	Function
0-4	4	Register #3, Digits 5 - 1
5-9	3	Register #2, Digits 5 - 1
10-14	2	Register #1, Digits 5 - 1
15	244_8	Not assigned
16-17	1	Noun Register, Digits 2 - 1
18-19	0	Verb Register, Digits 2 - 1

Note that for the display registers, Digit 1 is the most significant and Digit 5 (or 2) is the least significant.

```
Return
```

VERB

```
VERBREG = 0
DSPCOUNT = 19
Perform "2BLANK"
DECBRNCH = $01_2$
REQRET = 0
ENTRET = "End of job"
End of job
```

```
NOUN
    NOUNREG = 0
    DSPCOUNT = 17
    Proceed to third line of "VERB"
NEGSGN
    If bits 2-1 of DECBRNCH $\neq 00_2$, End of job
    If DSPCOUNT not equal to 4, 9, or 14, End of job
    Perform "-ON"
    $DECBRNCH = DECBRNCH + K_{bt7_{INREL}} + 10_2$
    If CLPASS $> 0$, set CLPASS = +0
    End of job
-ON
    Perform "GETINREL"
    CODE = 0
    $TS = K_{sgt_{INREL}} + 1$
    Perform "11DSPIN"  (turn off appropriate plus sign)
    CODE = $2^{10}$  (i.e. bit 11 = 1)
    $TS = K_{sgt_{INREL}}$
    Perform "11DSPIN"  (turn on appropriate minus sign)
    Return
POSGN
    If bits 2-1 of DECBRNCH $\neq 00_2$, End of job
    If DSPCOUNT not equal to 4, 9, or 14, End of job
    Perform "+ON"
    $DECBRNCH = DECBRNCH + K_{bt7_{INREL}} + 01_2$
```

```
        If CLPASS > 0, set CLPASS = +0

        End of job

+ON

        Perform "GETINREL"

        CODE = 0

        TS = $K_{sgt_{INREL}}$

        Perform "11DSPIN"   (turn off appropriate minus sign)

        CODE = $2^{10}$     (i.e. bit 11 = 1)

        TS = $K_{sgt_{INREL}}$  +  1

        Perform "11DSPIN"       (turn on appropriate plus sign)

        Return

ERROR

        DSPLOCK = 21d22REG

        Inhibit interrupts

        Set bit 10(Caution Reset) of channel 11 = 1

        DSPTAB+11 = bits 6(Gimbal Lock) and 4(No Attitude) of DSPTAB+11, and
                        flag for output at next opportunity

        Set bits 13-11 (PIP2FLBT, DNLKFAIL, UPLKFAIL) of IMODES33 = 1

        Set bit 10(PIPAFLBT) of IMODES30 = 1

        Set bit 7(OCDUFBIT) of OPTMODES = 1

        Perform "C13STALL"

        Set bit 10(Test DSKY Lights) of channel 13 = 0

        Set bits 7(Operator Error) and 3(Uplink Activity) of channel 11 = 0

        Set bit 12 of |DSPTAB| - |DSPTAB+10| = 1

        Release interrupts

        FAILREG+i = 0     (i = 0-1)       (can zero FAILREG+2 by V23N9E E)

        SFAIL = 0

        End of job
```

CLEAR

```
TS = $|DSPCOUNT|$
Set INREL (using TS as argument) per table in "GETINREL"
If CLPASS $\leqslant +0$:
    If INREL $< 2$, End of job
    Perform "5BLANK" (starting at second line)
    CLPASS = CLPASS + 1
    End of job
INREL = INREL - 1, limited $\geqslant 0$
If INREL $< 2$, End of job
REQRET = REQRET + 3  ("backs up" data requests via "ABLOAD" or
                      "ABCLOAD")
$TS_2$ = INREL
VERBREG = VERBREG - 1
Perform "UPDATVB"
INREL = $TS_2$
Perform "5BLANK" (starting at second line)
CLPASS = CLPASS + 1
End of job
```

5BLANK

```
Perform "GETINREL"
$WDREG_{INREL} = 0$
$LPREG_{INREL} = 0$
CODE = 0
Set bits 1, 2, and $K_{bt7_{INREL}}$ of DECBRNCH = 0
COUNT = $K_{snb_{INREL-2}}$
Perform "DSPIN"
DSPCOUNT = $K_{dbk_{INREL-2}}$
```

```
    Perform "2BLANK"
    DSPCOUNT = DSPCOUNT - 2
    Perform "2BLANK"
    DSPCOUNT = $K_{rd_{INREL-2}}$
    Return
2BLANK
    TS = integral part of ($\frac{1}{2}$ DSPCOUNT)
    $DSPTAB_{TS} = 2^{11}$ (i.e. bit 12 = 1), and flag for output at next
        opportunity.  If not already flagged, NOUT = NOUT + 1.
    Return
ENTER
    CLPASS = +0
    ENTRET = "End of job"
    If  REQRET $\geqslant$ +0, proceed to "ENTPASO"
    If  REQRET $\neq$ - ("MMCHANG" + 1):
        If bits 2-1 of DECBRNCH $\neq 00_2$:
            If DSPCOUNT $\geqslant$ +0:      (i.e. 5 characters not yet input)
                Perform "DECEND" (starting at 4th line)
    REQRET = - REQRET
    Set bit 6(Flash) of channel 11 = 0
    Proceed to address specified by REQRET
ENTPASO
    DECBRNCH = 0
    DSPCOUNT = -19
    VERBSAVE = - VERBREG
    If  VERBREG - $K_{owvb} \geqslant$ 0, proceed to "VERBFAN"
    Proceed to "TESTNN"
```

TESTNN

```
Perform "LODNNTAB"
If MIXBR = 2:
    If NNADTEM = +0, proceed to "DSPALARM"    (Tag here "MIXNOUN")
    If VERBREG - 6 > 0, proceed to "VERBFAN"
    Perform the following for $\underline{I}$ = 3, 2, 1:
        $TS_2 = IDAD\underline{I}TEM$
        $DECOUNT = \underline{I} - 1$
        Perform "SFRUTMIX"
        If $TS_1$ indicates (via "DPTEST") double precision:
            $TS_2 = TS_2 + 1$
        $TS_2$ = bits 11-1 of $TS_2$
        $MIXTEMP_{\underline{I}-1} = E_{TS_2}$        ($TS_2$ in erasable CADR form)
    NOUNADD = "MIXTEMP"
    Proceed to "VERBFAN"
If NNADTEM > 0:
    NOUNCADR = NNADTEM
    EBANK = bits 11-9 of NNADTEM
    NOUNADD = $1400_8$ + bits 8-1 of NNADTEM
    Proceed to "VERBFAN"
If NNADTEM = +0, proceed to "DSPALARM"
If NNADTEM = -0:
    NOUNCADR = NOUNCADR + 1
    EBANK = bits 11-9 of NOUNCADR
    NOUNADD = $1400_8$ + bits 8-1 of NOUNCADR
```

```
(If NNADTEM = -0):
    If VERBREG = 5, proceed to "VERBFAN"
    DSPCOUNT = 4
    TS = NOUNCADR
    Perform "DSPOCTWD"
    Proceed to "VERBFAN"
If NNADTEM $<$ -0 (as it will):
    CLPASS = -16383                    (Tag here "REQADD")
    If ENTRET = "End of job":          (i.e. not from internal use)
        Perform "REQDATZ" (upon return, proceed)
        If DECBRNCH $>$ 0, proceed to "ALMCYCLE"
        DSPCOUNT = -19
        If CADRSTOR $\neq$ +0, set bit 6(Flash) of channel 11 = 1
        NOUNCADR = ZREG
        EBANK = bits 11-9 of NOUNCADR
        NOUNADD = $1400_8$ + bits 8-1 of NOUNCADR
        Perform "LODNNTAB"
        Proceed to "VERBFAN"
    NOUNCADR = MPAC+2                  (Tag here "INTMCTBS")
    EBANK = bits 11-9 of NOUNCADR
    NOUNADD = $1400_8$ + bits 8-1 of NOUNCADR
    If VERBREG = 5, proceed to "VERBFAN"
    DSPCOUNT = 4
    TS = NOUNCADR
    Perform "DSPOCTWD"
    Proceed to "VERBFAN"
```

```
REQDATX
    DSPCOUNT = 14
    Proceed to second line of "REQDATZ"
REQDATY
    DSPCOUNT = 9
    Proceed to second line of "REQDATZ"
REQDATZ
    DSPCOUNT = 4
    REQRET = - (Calling address +1)  Calling address is that of "REQDATi",
                                     i = X, Y, or Z.
    Perform "5BLANK"
    Set bit 6(Flash) of channel 11 = 1
    Proceed to address specified by ENTRET
UPDATNN
    Perform "LODNNTAB"
    If NNADTEM $\geq$ +0:
        NOUNCADR = NNADTEM
        EBANK = bits 11-9 of NNADTEM
        NOUNADD = $1400_8$ + bits 8-1 of NNADTEM
    DSPCOUNT = 17
    TS = NOUNREG
    Perform "DSPDECVN"
    Return
UPDATVB
    DSPCOUNT = 19
    TS = VERBREG
    Perform "DSPDECVN"
    Return
```

```
DSPABC      (verb 05)
    TS = -2
    Perform "COMPTEST"
    $BUF+2 = - E_{NOUNADD+2}$
    Proceed to "DSPAB"
DSPAB       (verb 04)
    TS = -1
    Perform "COMPTEST"
    $BUF+1 = - E_{NOUNADD+1}$
    Proceed to "DSPA"
DSPA        (verb 01)
    TS = -0
    Perform "DCOMPTST"
    Perform "TSTFORDP"
    $BUF+0 = - E_{NOUNADD}$
    Proceed to "DSPCOM2"
DSPB        (verb 02)
    TS = -1
    Perform "DCOMPTST"
    $BUF+0 = - E_{NOUNADD+1}$
    Proceed to "DSPCOM2"
DSPC        (verb 03)
    TS = -2
    Perform "DCOMPTST"
    $BUF+0 = - E_{NOUNADD+2}$
    Proceed to "DSPCOM2"
```

```
DSPCOM2
    I = VERBREG - 3, limited $\geq 0$
    DSPCOUNT = $K_{rd_I}$
    TS = - BUF+I
    Perform "DSPOCTWD"
    If I = 0, proceed to address specified by ENTRET
    I = I - 1
    Proceed to second line of "DSPCOM2"
COMPTEST
    If MIXBR = 1:
        $TS_1$ = NNTYPTEM
    If MIXBR = 2:
        $TS_1$ = NNADTEM
    $TS_1$ = bits 12-11 of $TS_1$, cycled left 5 places (placing in bits 2-1)
    If $TS_1$ + TS $< 0$, proceed to "DSPALARM"  (noun not have enough components)
    Return
DCOMPTST
    If MIXBR = 1:
        $TS_1$ = NNTYPTEM
    If MIXBR = 2:
        $TS_1$ = NNADTEM
    If bit 14 of $TS_1$ = 1:
        Proceed to "DSPALARM"   (noun is decimal-only)
    Proceed to "COMPTEST"
```

TSTFORDP

```
    If NNADTEM = -1:
        TS = (bits 9-1 of NOUNCADR)
        BUF+0 = - (contents of channel whose number is in TS)
        Proceed to "DSPCOM2"
    If MIXBR = 2, Return
    $TS_1$ = bits 10-6 of NNTYPTEM, cycled right 5 places (placing in bits 5-1)
    If $TS_1$ indicates (via "DPTEST") double precision:
        NOUNADD = NOUNADD + 1
    Return
```

DECDSP (verb 06)

```
    If MIXBR = 1:
        $TS_1$ = NNTYPTEM
    If MIXBR = 2:
        $TS_1$ = NNADTEM
    $TS_1$ = bits 12-11 of $TS_1$, cycled left 5 places (placing in bits 2-1)
    DECOUNT = $TS_1$
    $WDREG_{TS_1+2} = - E_{NOUNADD+TS_1}$
    If $TS_1 > 0$, $TS_1 = TS_1 - 1$ and proceed to previous line
    Proceed to "DSPDCPUT"
```

DSPDCPUT

```
    MPAC+i = 0     (i = 1,2)
    DSPCOUNT = $K_{rd_{DECOUNT}}$
    $MPAC_{sp} = - WDREG_{DECOUNT+2}$
    If MIXBR = 1:
        TS = bits 5-1 of NNTYPTEM
```

```
        If MIXBR = 2:
            TS = bits 5-1 of NNTYPTEM if DECOUNT = 0
            TS = bits 10-6 of NNTYPTEM cycled right 5 places if DECOUNT = 1
            TS = bits 15-11 of NNTYPTEM cycled left 5 places if DECOUNT = 2
        Perform "GTSFOUT"
        If MIXBR = 1:
            $TS_1$ = bits 10-6 of NNTYPTEM, cycled right 5 places
        If MIXBR = 2:
            Perform "SFRUTMIX"
        Proceed to "DECDSP3"
    DSPDCEND
        Perform "DSPDECWD"
        If DECOUNT = 0, proceed to address specified by ENTRET
        DECOUNT = DECOUNT - 1
        Proceed to "DSPDCPUT"
    DPOUT
        If MIXBR = 1:
            TS = NOUNADD
        If MIXBR = 2:
            EBANK = bits 11-9 of $IDAD\underline{I}TEM_{DECOUNT+1}$   (e.g. IDAD2TEM for DECOUNT = 1)
            TS = $1400_8$ + bits 8-1 of $IDAD\underline{I}TEM_{DECOUNT+1}$
        $MPAC_{dp} = E_{TS_{dp}}$ , with sign agreement forced
        $MPAC_{tp} = (MPAC_{dp})(SFTEMP1_{dp})$
        Return
    DSPDPDEC  (verb 07)
        If MIXBR = 2, proceed to "DSPALARM"
        $MPAC_{dp} = E_{NOUNADD_{dp}}$
        DSPCOUNT = 14
```

```
        MPAC+2 = 0
        Force sign agreement of $MPAC_{tp}$
        Perform "DSP2DEC"
        Proceed to address specified by ENTRET
```

ABCLOAD (verb 25)

```
        TS = -2
        Perform "COMPTEST"
        Perform "NOUNTEST"
        VERBREG = 21
        Perform "UPDATVB"
        Perform "REQDATX" ; upon return, proceed
        VERBREG = 22
        Perform "UPDATVB"
        Perform "REQDATY" ; upon return, proceed
        VERBREG = 23
        Perform "UPDATVB"
        Perform "REQDATZ" ; upon return, proceed
        TS = -6
        Perform "ALLDC/OC"
        Perform "LODNNTAB"
        DECOUNT = 0
        Perform "PUTCOM"
        $E_{NOUNADD}$ = TS
        DECOUNT = 1
        Perform "PUTCOM"
        $E_{NOUNADD+1}$ = TS
        DECOUNT = 2
        Perform "PUTCOM"
```

```
    $E_{NOUNADD+2}$ = TS
    If NOUNREG $\neq$ 07, proceed to "LOADLV"   (bit change not specified)
    Inhibit interrupts
    TS = XREG - $30_8$
    If TS > 0:
        EBANK = (bits 11-9 of XREG)
        NOUNADD = $1400_8$ + (bits 8-1 of XREG)
        Channel 1 = $E_{NOUNADD}$     (channel 1 is computer L register, see
                                                3420.5-27)
        XREG = $0001_8$
        TS = 1
    If  TS + $21_8$ = 0:          (i.e. original input was 07, channel 7)
        Release interrupts
        Proceed to "LOADLV"
    If  ZREG $\leq$ 0:
        Set those bits of $channel_{XREG}$ = 0 that are 1 in YREG
    If  ZREG > 0:
        Set those bits of $channel_{XREG}$ = 1 that are 1 in YREG
    If  XREG = 1:      (assumed to be due to setting above: if load
      (or if < 0)      XREG = 1 (or negative), no effect, since the
                       contents of NOUNADD = "XREG")
        $E_{NOUNADD}$ = Channel 1   (computer L register)
    Release interrupts
    Proceed to "LOADLV"
ABLOAD    (verb 24)
    TS = -1
    Perform "COMPTEST"
    Perform "NOUNTEST"
    VERBREG = 21
    Perform "UPDATVB"
    Perform "REQDATX" ; upon return, proceed
    VERBREG = 22
    Perform "UPDATVB"
```

```
    Perform "REQDATY" ; upon return, proceed
    TS = -5
    Perform "ALLDC/OC"
    Perform "LODNNTAB"
    DECOUNT = 0
    Perform "PUTCOM"
    $E_{NOUNADD}$ = TS
    DECOUNT = 1
    Perform "PUTCOM"
    $E_{NOUNADD+1}$ = TS
    Proceed to "LOADLV"
ALOAD    (verb 21)
    Perform "REQDATX" ; upon return, proceed
    Perform "LODNNTAB"
    DECOUNT = 0
    Perform "PUTCOM"
    $E_{NOUNADD}$ = TS
    Proceed to "LOADLV"
BLOAD    (verb 22)
    TS = -1
    Perform "COMPTEST"
    CLPASS = -16383
    Perform "REQDATY" ; upon return, proceed
    Perform "LODNNTAB"
    DECOUNT = 1
    Perform "PUTCOM"
    $E_{NOUNADD+1}$ = TS
    Proceed to "LOADLV"
CLOAD    (verb 23)
    TS = -2
    Perform "COMPTEST"
    CLPASS = -16383
```

```
    Perform "REQDATZ" ; upon return, proceed
    Perform "LODNNTAB"
    DECOUNT = 2
    Perform "PUTCOM"
    $E_{NOUNADD+2} = TS$
    Proceed to "LOADLV"
LOADLV
    DECBRNCH = 0
    LOADSTAT = -0
    Perform "RELDSP"
    DSPCOUNT = -19
    Proceed to "RECALTST"
ALLDC/OC
    $TS_1$ = DECBRNCH, shifted right 2 places (7 if 3 decimal words,
                                               6 if 2 decimal words)
    If $TS_1 = 0$, Return (all words octal)
    If $TS_1 + TS \neq 1$, proceed to "ALMCYCLE"
    Return
NOUNTEST    (Entered from "ABCLOAD" and "ABLOAD" only)
    If MIXBR = 1:
        $TS_1$ = NNTYPTEM
    If MIXBR = 2:
        $TS_1$ = NNADTEM
    If bit 15 of $TS_1 = 1$, proceed to "DSPALARM" ("no-load" noun)
    Return
SFRUTMIX
    If DECOUNT = 0:
        $TS_1$ = bits 5-1 of RUTMXTEM
```

```
If DECOUNT = 1:
    $TS_1$ = bits 10-6 of RUTMXTEM, cycled right 5 places (to bits 5-1)
If DECOUNT = 2:
    $TS_1$ = bits 15-11 of RUTMXTEM, cycled left 5 places (to bits 5-1)
Return
```

PUTCOM

```
DECRET = (calling address +1)
OVFINP = 0
$MPAC_{dp}$ = ($WDREG_{DECOUNT+2}$ , $LPREG_{DECOUNT+2}$)
If MIXBR = 1:
    EBANK = bits 11-9 of NOUNCADR                    (Tag here "PUTNORM")
    NOUNADD = $1400_8$ + bits 8-1 of NOUNCADR
    If DECBRNCH > 0, proceed to "PUTDECSF"
    If bit 14 of NNTYPTEM = 1, proceed to "ALMCYCLE" (decimal only)
    $TS_1$ = bits 10-6 of NNTYPTEM, cycled right 5 places
    If $TS_1$ indicates (via "DPTEST") double precision:
        DECOUNT = 0
        NOUNADD = NOUNADD + 1
        DECOUNT = NOUNADD + DECOUNT
        $E_{DECOUNT-1}$ = 0   (most significant half of word)
        TS = MPAC+0
        Proceed to address specified by DECRET
    If NNADTEM = -1:
        If NOUNCADR ≠ 7:   (note that no operator indication
                            of ignored input is provided)
            TS = (bits 9-1 of NOUNCADR)      (note that ought not to = 7)
            Set (channel whose number is in TS) = MPAC+0
        Proceed to "LOADLV"
```

```
    (If MIXBR = 1):
        TS = MPAC+0
        Proceed to address specified by DECRET
    NOUNCADR = bits 11-1 of $\text{IDADITEM}_{\text{DECOUNT+1}}$
    EBANK = bits 11-9 of NOUNCADR
    NOUNADD = $1400_8$ + bits 8-1 of NOUNCADR  -  DECOUNT
    If DECBRNCH > 0, proceed to "PUTDECSF"
    If bit 14 of NNADTEM = 1, proceed to "ALMCYCLE" (decimal only)
    Perform "SFRUTMIX"
    If $\text{TS}_1$ indicates (via "DPTEST") double precision:
        NOUNADD = NOUNADD + 1
        DECOUNT = DECOUNT + NOUNADD
        $\text{E}_{\text{DECOUNT-1}} = 0$   (most significant half of word)
        TS = MPAC+0
        Proceed to address specified by DECRET
    TS = MPAC+0
    Proceed to address specified by DECRET
PUTDECSF
    If MIXBR = 1:
        TS = bits 5-1 of NNTYPTEM
    If MIXBR = 2:
        TS = bits 5-1 of NNTYPTEM if DECOUNT = 0
        TS = bits 10-6 of NNTYPTEM cycled right 5 places if DECOUNT = 1
        TS = bits 15-11 of NNTYPTEM cycled left 5 places if DECOUNT = 2
    Perform "GTSFIN"
```

```
    If MIXBR = 1:
        $TS_1$ = bits 10-6 of NNTYPTEM, cycled right 5 places
    If MIXBR = 2:
        Perform "SFRUTMIX"
    Proceed to "PUTDCSF2"
BINROUND
    $MPAC_{dp} = MPAC_{dp} + MPAC+1$   set OVFINP to 1 sgn MPAC if overflow
    If OVFINP non-zero, proceed to "ALMCYCLE"
    TS = MPAC+0
    Proceed to address specified by DECRET
DPINSF
    $MPAC_{tp} = (MPAC_{dp})(SFTEMP1_{dp})$
    $MPAC_{tp} = MPAC_{tp} + MPAC+2$   set OVFINP to 1 sgn MPAC if overflow
    If OVFINP non-zero, proceed to "ALMCYCLE"
    If MIXBR = 1:
        $E_{NOUNADD+1}$ = MPAC+1
    If MIXBR = 2:
        $E_{NOUNADD+DECOUNT+1}$ = MPAC+1
    TS = MPAC+0
    Proceed to address specified by DECRET
MONITOR   (verbs 11-17)
    $TS_1$ = bits 13-1 of NOUNCADR
    If ENTRET = "End of job" (i.e. externally initiated monitor):
        Set bit 14 of $TS_1$ = 1
        MONSAVE2 = 0
    $TS_2$ = bits 7-1 of VERBREG (cycled left 7 places) + NOUNREG
    DSPLOCK = 0
```

```
    If CADRSTOR = 0:
        If DSPLIST = 0:
            Set bit 5(Key Release) of channel 11 = 0  (would have been
                                                       turned on if new monitor
    Inhibit interrupts                                 keyed in over old one)
    If MONSAVE = 0:
        Call "MONREQ" in 0.01 second
    MONSAVE = $TS_2$
    MONSAVE1 = $TS_1$
    Release interrupts
    Proceed to address specified by ENTRET
MONREQ
    $T_{st} = T_{now}$
    If MONSAVE1 $\leq$ -0:
        MONSAVE = 0
        MONSAVE1 = 0
        End of task
    Call "MONREQ" in $K_{mond}$ seconds
    Establish "MONDO" (priority $30_8$)
    End of task
MONDO
    If MONSAVE1 $\leq$ -0, End of job
    If DSPLOCK $>$ 0:
        Set bit 5(Key Release) of channel 11 = 1
        End of job
    NOUNREG = bits 7-1 of MONSAVE
    Perform "UPDATNN"
    VERBREG = (bits 14-8 of MONSAVE, shifted right 7 places) - 10
    ENTRET = "PASTEVB"
    MPAC+2 = bits 13-1 of MONSAVE1
```

```
    Proceed to "TESTNN"   (will return via ENTRET to "PASTEVB")

PASTEVB

    TS = bits 14-8 of MONSAVE2

    If TS = 0:      (non-zero if "MARKMONR" display interface routine used)

        TS = MONSAVE

    VERBREG = (bits 14-8 of TS, shifted right 7 places)

    Perform "UPDATVB"

    REQRET = +0

    TS = MONSAVE2

    Perform "BLANKSUB": if return to calling address +1, proceed
                        otherwise, proceed

    End of job

DSPFMEM   (verb 27)

    DSPCOUNT = 14

    If bits 15-11 of NOUNCADR $\leq 27_8$:     (fixed memory address in bank
                                                $27_8$ or below; NOUNCADR loaded in
        TS = $E_{NOUNCADR}$                          FCADR format by using V27 N01)

    If bits 15-11 of NOUNCADR $> 27_8$:      (fixed memory address in bank $30_8$
                                                or above; DSPTEM1+2 loaded by
        Channel 7 = bits 7-5 of                 using V23 N25 before V27 N01 that
                         DSPTEM1+2              is used to load FCADR NOUNCADR)

        TS = $E_{NOUNCADR}$    (address information also determined by Channel
                            7 contents)

        Restore Channel 7 to previous value

    Perform "DSPOCTWD"

    End of job

DSPSIGN

    If MPAC+0 $\geq$ +0:

        Perform "+ON"

    If MPAC+0 $\leq$ -0:

        $MPAC_{dp} = - MPAC_{dp}$

        Perform "-ON"
```

```
    Return
DSPDECWD
    Perform "DSPSIGN"
    $MPAC_{dp} = MPAC_{dp} + K_{dcrd}$        If overflow, $MPAC_{dp} = +MAX$
    WDCNT = 4
    Proceed to "DSPDCWD1"
DSPDCWD1
    $MPAC_{tp} = K_{bncn} MPAC_{tp}$
    CODE = (bits 5-1 of $K_{rtb_{MPAC+0}}$)
    COUNT = DSPCOUNT
    $MPAC_{tp} = 2^{14} MPAC_{tp}$      (overflow ignored: this preserves the fractional
                                        part of the multiplication in the first line)
    DSPCOUNT = COUNT - 1
    Perform "DSPIN"
    If WDCNT > 0:
        WDCNT = WDCNT - 1
        Proceed to "DSPDCWD1"
    DSPCOUNT = -19
    Return
DSPDC2NR
    Perform "DSPSIGN"
    WDCNT = 1
    Proceed to "DSPDCWD1"
DSP2DEC
    CODE = 0
    TS = 3
    Perform "11DSPIN"   (turn off R2 minus sign)
```

```
    TS = 4
    Perform "11DSPIN"   (turn off R2 plus sign)
    Perform "DSPSIGN"
    WDCNT = 9
    Proceed to "DSPDCWD1"
DSPDECVN
    MPAC+0 = $K_{vncon}$ TS
    MPAC+1 = 0
    WDCNT = 1
    Proceed to "DSPDCWD1"
DSPOCTWD
    $TS_1$ = TS
    DSPCOUNT = DSPCOUNT + $2^{13}$   (sets bit 14 = 1)
    WDCNT = 4
    Proceed to "WDAGAIN"
WDAGAIN
    Cycle $TS_1$ left 3 places and store in $TS_1$ (bits 15-13 become 3-1)
    TS = bits 3-1 of $TS_1$
    CODE = (bits 5-1 of $K_{rtb_{TS}}$)
    COUNT = DSPCOUNT
    DSPCOUNT = DSPCOUNT - 1
    Perform "DSPIN"
    If WDCNT > 0:
        WDCNT = WDCNT - 1
        Proceed to "WDAGAIN"
    DSPCOUNT = -19
    Return
```

```
DSPIN
    DSREL = bits 5-2 of COUNT, shifted right 1 place (i.e. $\frac{1}{2}$ COUNT)
    If bit 1 of COUNT = 0:
        COUNT = 0
        Proceed to "DSPIN1"
    Shift  CODE  left 5 places (bits 5-1 become bits 10-6)
    If bit 14 of COUNT = 1:
        COUNT = 3
        Proceed to "DSPIN1"
    COUNT = 1
    Proceed to "DSPIN1"
DSPIN1
    Inhibit interrupts
    TS = $|\text{DSPTAB}_{\text{DSREL}}|$
    If COUNT = 0, set bits 5-1 of TS = bits 5-1 of CODE
    If COUNT = 1, set bits 10-6 of TS = bits 10-6 of CODE
    If COUNT = 2, set bit 11 of TS = bit 11 of CODE
    If COUNT = 3, set bits 11-6 of TS = bits 11-6 of CODE
    If TS $\neq |\text{DSPTAB}_{\text{DSREL}}|$ :
        $\text{DSPTAB}_{\text{DSREL}}$ = TS, flagged for output at next opportunity
        If not already flagged, NOUT = NOUT + 1
    Release interrupts
    Return
11DSPIN
    DSREL = TS
    COUNT = 2
    Proceed to "DSPIN1"
```

```
DSPALARM
    If ENTRET = "NVSUBEND":     (i.e. internal request)
        Proceed to "POODOO" (pattern $21501_8$)
    If ENTRET = "PASTEVB":     (i.e. from "MONDO")
        MONSAVE1 = $40000_8$    (sets bit 15 to 1)
        Set bit 7(Operator Error) of channel 11 = 1
        Proceed to "PASTEVB"    (to put monitor verb back in VERBREG)
    Set bit 7(Operator Error) of channel 11 = 1
    End of job
ALMCYCLE
    Set bit 7(Operator Error) of channel 11 = 1
    REQRET = - VERBSAVE    (sets to a positive number)
    VERBREG = - VERBSAVE
    Perform "UPDATVB"
    Proceed to "ENTER"
MMCHANG    (verb 37)
    Perform "REQMM"
    If DSPCOUNT $\neq$ -16, proceed to "ALMCYCLE" (2 digits not entered)
    MPAC+0 = NOUNREG
    NOUNREG = 0
    DSPCOUNT = 17
    Perform "2BLANK"
    DSPCOUNT = -19
    TS = MPAC+0
    Proceed to "V37"
REQMM
    REQRET = - (Calling address + 1)
    DSPCOUNT = 17
    NOUNREG = 0
```

```
    Perform "2BLANK"

    Set bit 6(Flash) of channel 11 = 1

    DECBRNCH = $01_2$

    Proceed to address specified by ENTRET

VBRQEXEC  (verb 30)

    Perform "RELDSP"

    If  N26dPRI = 0:

        Proceed to "DSPALARM"

    If bit 1 of N26dPRI = 1:

        Establish a job with 2CADR information in $N26d2CAD_{dp}$, with
        a VAC area, and with priority given by bits 14-10 of N26dPRI

    If bit 1 of N26dPRI = 0:

        Establish a job with 2CADR information in $N26d2CAD_{dp}$, with
        no VAC area, and with priority given by bits 14-10 of N26dPRI

    End of job

VBRQWAIT  (verb 31)

    Perform "RELDSP"

    If N26dPRI = 0:

        Proceed to "DSPALARM"

    Call task in N26dPRI centi-seconds with starting address given
        by $N26d2CAD_{dp}$ (in 2CADR format)

    End of job

VBRESEQ   (verb 32)

    LOADSTAT = -0

    Proceed to second line of "VBPROC"

VBPROC    (verb 33; also entered from "PROCKEY")

    LOADSTAT = -1

    MONSAVE1 = $40000_8$        (sets bit 15 to 1 and bit 14 to 0)

    Perform "RELDSP"

    Set bit 6(Flash) of channel 11 = 0

    Proceed to "RECALTST"
```

```
VBTERM    (verb 34)
    LOADSTAT = -2
    Proceed to second line of "VBPROC"
PROCKEY    (Established by "PROCEEDE")
    If $|22 - VERBREG| \leq 1$:                (includes case of e.g. V21 PRO,
        Proceed to "CHARALRM"                   but not V25 PRO)
    REQRET = 0
    DSPCOUNT = -19
    Proceed to "VBPROC"
VBRELDSP    (Entered from "CHARIN")
    Set bit 3(Uplink Activity) of channel 11 = 0
    If 21d22REG $>$ 0:
        If bit 14 of MONSAVE1 = 1:        (external monitor)
            DSPLOCK = 0    (causes monitor to be started again
                                 due to "MONDO" logic)
            If CADRSTOR = 0:
                If DSPLIST = 0:
                    Set bit 5 (Key Release) of channel 11 = 0
            End of job
    Perform "RELDSP"
    If CADRSTOR = +0, End of job
    Proceed to "PINBRNCH"
NVSUB
    $TS_1$ = 0          (L-register, MONSAVE2 information)
    Proceed to "NVMONOPT"
NVMONOPT    (Entered with TS, verb-noun pattern or blanking code, in A)
    If bit 14 of MONSAVE1 = 1, or if DSPLOCK $>$ 0:
        Return (to calling address +1, indicating system "busy")
    MONSAVE2 = $TS_1$
    MONSAVE1 = $40000_8$    (sets bit 15 to 1)
    ENTRET = "NVSUBEND"
    If $|TS|$ = 0, proceed to "DSPALARM"
```

```
If TS $<0$:          (error if -5 or less)
    $i = |TS| + 6$
    Set DSPTAB+0 - DSPTAB+i = $2^{11}$ (i.e. blank) and flag for output
        at next opportunity.  If individual DSPTAB cell not already
        flagged, NOUT = NOUT + 1
    If TS $\leq -3$:
        VERBREG = 0
    If TS $\leq -2$:
        NOUNREG = 0
    CLPASS = 0
    DSPCOUNT = -19
    Set bit 6(Flash) of channel 11 = 0
    REQRET = 0
    Proceed to "NVSUBEND"
$TS_1$ = bits 7-1 of TS  (original input to routine, giving noun)
$TS_2$ = bits 14-8 of TS, shifted right 7 places (giving verb)
If $TS_1 = 0$:
    VERBREG = $TS_2$
    Perform "UPDATVB"
    REQRET = 0
    Proceed to "NVSUBEND"
If $TS_2 = 0$:
    NOUNREG = $TS_1$
    Perform "UPDATNN"
    Proceed to "NVSUBEND"
$TS_3$ = MPAC+2     (MPAC+2 contains machine address for address-to-
                     be specified nouns)
VERBREG = $TS_2$
Perform "UPDATVB"
NOUNREG = $TS_1$
Perform "UPDATNN"
```

```
        LOADSTAT = +0

        CLPASS = 0

        REQRET = 0

        MPAC+2 = $TS_3$

        Proceed to "ENTPASO"

NVSUBEND

        Proceed to ("NVMONOPT" calling address +2)

JAMTERM

        REQRET = 34

        VERBREG = 34

        DSPCOUNT = -19

        Proceed to "VBTERM"

BLANKSUB        (Entered with TS, bits 1-3, set to blank R1 - R3 respectively)

        If bit 14 of MONSAVE1 = 1, or if  DSPLOCK > 0:

                Return (to calling address +1, indicating system "busy")

        $TS_4$ = (bits 3-1 of TS)        ($TS_4$ in NVTEMP cell)

        If  $TS_4$ = 0:

                Return  (to calling address +2)

        $TS_3$ = DSPCOUNT       (in BUF+2 cell)

        If bit 1 of $TS_4$ = 1:

                DSPCOUNT = 14

                Perform "5BLANK"

        If bit 2 of $TS_4$ = 1:

                DSPCOUNT = 9

                Perform "5BLANK"
```

```
        If bit 3 of $TS_4$ = 1:
            DSPCOUNT = 4
            Perform "5BLANK"
        DSPCOUNT = $TS_3$
        Return (to "BLANKSUB" calling address +2)
    NEWMODEX        (Entered with TS = new major mode pattern)
        MODREG = TS
        Establish "DSPMMJB"  (priority $30_8$)
        Return
    DSPMMJB
        $TS_2$ = DSPCOUNT
        DSPCOUNT = 21
        If  MODREG = -0, perform "2BLANK"
        If  MODREG $\geq$ +0:
            TS = MODREG
            Perform "DSPDECVN"
        DSPCOUNT = $TS_2$
        End of job
    RECALTST
        If CADRSTOR = 0, End of job
        Inhibit interrupts
        Awaken job with starting address id = CADRSTOR
        CADRSTOR = 0
        $MPAC_{LOCCTR_{dp}}$ = (VERBREG, NOUNREG)
        Release interrupts
        Perform "RELDSP"
        End of job
```

RELDSP

```
Set bit 14 of MONSAVE1 = 0
If DSPLIST $\neq$ 0:
    Awaken job with starting address id = DSPLIST
    DSPLIST = 0
Set bit 5(Key Release) of channel 11 = 0
DSPLOCK = 0
Return
```

Quantities in Computations

See also list of major variables and list of routines

21d22REG: Temporary storage cell for previous value of DSPLOCK, used to avoid changing DSPLOCK if an Error Reset code is received, etc.

BUF: Temporary storage cell.

CADRSTOR: Single precision cell indicating, if non-zero, that a job has been put to sleep in "ENDIDLE". The contents of the cell are the address to which return is made: the cell is loaded with "IDLERET1" in "ENDIDLE" when putting the job to sleep.

CHAR: Input character (from DSKY or uplink), scale factor B14 (located in bits 5-1).

CLPASS: Control cell for "CLEAR": if negative, a single-component load verb (22 or 23) is being used, so Clear inputs must not "back up" (successive clears should not reset previously loaded registers). If CLPASS = +0, this indicates that the register indicated by DSPCOUNT has had a character (number or sign) loaded in it, so that register would be cleared by an input of the Clear code. Scaled B14.

CODE: Cell used to contain the required relay pattern code for loading $DSPTAB_I$ and causing the proper character to be displayed by DSKY, B14.

COUNT: Cell used in "DSPIN" to contain identification of the output character to be changed. Scale factor B14.

DECBRNCH: Cell used to control the processing of data inputs: the least significant two bits are 00_2 for an octal number, 01_2 for a positive decimal number (including verb, noun, and program number inputs), and 10_2 for a negative number. Bits 3-5 respectively are one if registers 3-1 are loaded with decimal information, and are used to check that multiple-component load verbs employ either all-decimal or all-octal input.

DECOUNT: Cell used for indexing purposes to control processing of multiple-component verbs, scale factor B14. Has values 0-2 for first, second, and third component respectively.

DECRET: Exit address after performance of "PUTCOM".

DSPCOUNT: Display counter, scale factor B14, identifying digit of data input expected next, as defined by table in "GETINREL". If is negative, no numerical inputs are accepted. It is set to -19 to inhibit all numerical inputs (and the Clear code) after the completion of certain functions. Values of 20-21 are for display of program number (MODREG).

DSPLIST: Single precision erasable memory cell indicating, if non-zero, that a job has been put to sleep in "NVSUBUSY" while waiting for the display system to be released by the crew. The contents of the cell are the starting address of the job.

DSPLOCK: Word used to indicate, if non-zero, that some external use (DSKY or uplink) of the display system has been made, and therefore internal attempts (via "NVMONOPT") to use the display system are delayed. When the display system is "released" (via "RELDSP"), DSPLOCK is set 0. Scale factor is B14.

DSREL: Indexing cell used to specify the DSPTAB cell to be modified in "DSPIN", scale factor B14.

EBANK: Hardware erasable memory bank register, required together with another address (see NOUNADD) to specify completely the memory location of an arbitrary erasable memory cell. See 3420.5-27 for more details.

ENTRET: Return address information from display program. Set to "End of job" for keyboard or uplink input, or to the address of additional computations if from "NVMONOPT" or "MONDO". Tags ENTRET and ENTEXIT both used in the program for this cell.

FAILREG: See General Program Control.

IDADITEM: See Noun Definitions.

IMODES30, IMODES33: See IMU Computations.

INLINK: Computer special erasable memory cell (address 0045_8) containing complete word (received one bit at a time) from the uplink receiver. Sixteen bits are received by the computer (additional bits are used by the receiver for spacecraft and system identification), with the first bit a binary one. When this first bit is shifted out of bit 15 (the most significant end) of the INLINK register, program interrupt #7 is generated, causing "UPRUPT" to be entered.

INREL: Indexing cell used to select internal computer word for the display system, set in accordance with table in "GETINREL", B14.

K_{bncn}: Single precision constant, program notation "BINCON", scale factor B14, value 10. Leaves integral part of product (in "DSPDCWD1") in MPAC+0.

K_{bt7_I}: Table of individual bits, giving values equal to bit 5, bit 4, and bit 3 for I = 2-4 respectively.

K_{crt_I}: Table of constants, program notation "CRITCON", giving value of DSPCOUNT for last character in verb, noun, and R1-R3 registers. Values (for I = 0-4 respectively) are 18, 16, 10, 5, and 0, B14.

K_{dbk_I}: Table of constants, program notation "DOUBLK", values 13, 9, and 3 for I = 0-2 respectively, scale factor B14.

K_{dcrd}: Constant, program notation "DECROUND -1", scale factor B0, octal value 00000_8 02476_8. Considering quantity involved to have a maximum value (for display purposes) of 0.99999, the constant has an equivalent value of $2476_8 \times 2^{-28}$, or about $\frac{1}{2} \times 10^{-5}$.

K_{dsc}: Constant, program notation "DECON", scale factor B-14, value $10^{-5} \times 2^{14}$. Constant performs the function of converting integer decimal input to a fraction (it is not used for verb, noun, or program number inputs).

K_{mond}: Single precision constant, program notation "MONDEL", scale factor B14, units centi-seconds. Value is 00144_8, corresponding to a decimal value of 100 or a monitor period of one second.

K_{owvb}: Single precision constant, program notation "LOWVERB", scale factor B14, value 28. VERBREG contents of this value or above bypass "TESTNN".

K_{rd_I}: Table of constants, program notation "R1D1", values 14, 9, and 4 for I = 0-2 respectively, scale factor B14.

K_{rtb_I}: Table of constants information, program notation "RELTAB", containing in least significant 5 bits the required value of CODE for the digit value specified by I:

I	Value	I	Value
0	25_8	5	36_8
1	03_8	6	34_8
2	31_8	7	23_8
3	33_8	8	35_8
4	17_8	9	37_8

See 3420.5-27 for more details.

K_{sgt_I}: Table of constants, program notation "SGNTAB", values 5, 3, and 0 for I = 2-4 respectively, scale factor B14.

K_{snb_I}: Table of constants, program notation "SINBLANK", values 14, 5, and 4 for I = 0-2 respectively, scale factor B14.

K_{vncon}: Single precision constant, program notation "VNDSPCON", scale factor B0, octal value 00244_8, corresponding to 164×2^{-14}, or about 0.01. Used to convert two-digit integer information to a fraction for subsequent processing by "DSPDCWD1".

LOADSTAT: Cell used in "ENDRET" to determine the proper response if a job was put to sleep in "ENDIDLE". It is set to +0 if waiting for an input; -1 if a proceed input is received; -2 if a terminate input is received; -0 if an "otherwise" (data load, V32E, etc.) is received; and +1 in "ENDEXT" to indicate use for terminating extended verbs. Scale factor is B14.

LOCCTR: Indexing parameter used to locate cells within the particular job register set assigned to a job which has been established (or a job which has been awakened). See 3420.5-27 for more details.

$LPREG_I$: Set of cells, defined only for I = 2-4 (XREGLP, YREGLP, and ZREGLP respectively), containing the least significant half of product for decimal input information.

MIXBR: See Noun Definitions.

MIXTEMP: Set of cells loaded if a display verb is used with the values of the memory cells specified by a "mixed" noun.

MONSAVE: Cell used to contain verb (bits 14-8) and noun (bits 7-1) information for monitor-type verbs. Cell is zero if no monitor function is being performed.

MONSAVE1: Cell used to contain in bits 13-1 the value of NOUNCADR for monitor verbs. Bit 15 (the sign) is set 1 if it is desired to terminate the monitor function, and bit 14 is set 1 if the monitor function was externally (DSKY or uplink) initiated.

MONSAVE2: Cell which can be set non-zero if internally initiated monitor started via "NVMONOPT", and used in "PASTEVB" to provide the capability for "pasting" a verb other than that used to generate display information, and to provide the capability for blanking a monitor display. Verb information in bits 14-8; blanking bits in bits 3-1 for R3, R2, and R1 respectively.

N26d2CAD: Pair of cells loaded by R2 and R3 of N26, used to specify in 2CADR format (see 3420.5-27) the starting address for the job/task to be initiated by V30E/V31E. Program notation "N26/2CAD".

N26dPRI: Single precision cell used to specify the priority for a V30 or the required delay time for a V31 before N26d2CAD step is entered. It is loaded by R1 of N26. A value of zero is used to lock out both V30 and V31, to avoid erroneous actions if keyed in inadvertently. Program notation "N26/PRI". (B14 cs)

NNADTEM, NNTYPTEM: See Noun Definitions.

NOUNADD: Cell containing erasable memory address information derived from noun data. If bits 12 and 11 are zero and bits 10 and 9 are one (i.e. in range 1400_8 - 1777_8), then the contents of EBANK determine which "bank" of 256 erasable memory cells is selected, while the contents of NOUNADD determine which cell within the bank is referenced. If NOUNADD is less than 1400_8, a unique cell is selected and EBANK is ignored. See 3420.5-27 for more details.

NOUNCADR: Cell containing the complete noun addressing information, obtained either from the noun tables or from an additional piece of input information. If the quantity is known to be in erasable memory, bits 11-9 are loaded into EBANK and bits 8-1 are added to 1400_8 to form NOUNADD.

NOUNREG: Cell containing (in bits 7-1) the decimal value of the noun, which specifies "to what". The verb indicates "do what". Scaled B14.

NOUT: See General Program Control.

OPTMODES: See Optics Computations.

OVFINP: Single precision cell, program notation "MPAC+6", used to retain the fact that overflow has taken place in the "Pinball" DSKY processing routines (mainly inputs). It is set 0 in "PUTCOM", B14.

REQRET: Cell controlling the interpretation of an "enter" command. If it is zero or any positive value, an initial pass for processing of the verb and noun information is performed; if negative, the cell is considered to contain the complement of the address to which return should be made when a word of data input has been completed.

RUTMXTEM: See Noun Definitions.

SFAIL: Cell used in computer self-check routine to contain return address from error subroutine (hence can be used to identify reason for failure of self-check). See Testing Routines. Scale factor B14.

SFTEMP1: See Noun Definitions.

T_{st}: Value of T_{now} sampled shortly after program interrupt acted upon (uplink, waitlist for monitor, or either keyboard). It can be displayed by noun 65. Scale factor B28, units centi-seconds.

UPSUM: Single precision cell, scale factor B14, used for uplink summing purposes (in "UPRUPT") if bit 12(NODOP01) of FLAGWRD1 = 0. It can be used to check the validity of e.g. erasable memory programs loaded via uplink.

UPSUM+1: Single precision cell, scale factor B14, used to count the number of uplink interrupts processed if bit 12(NODOP01) of FLAGWRD1 = 0. It is the cell after UPSUM in memory.

VERBREG: Cell containing (in bits 7-1) the value of the verb (c.f. NOUNREG). Scale factor is B14.

VERBSAVE: Cell loaded in "ENTPASO" with complement of VERBREG, used in "ALMCYCLE" to permit another attempt to be made to provide the required information. Loaded in "HMSIN" with - 25. Scaled B14.

WDCNT: Counter used in digit display routines to cause the proper number of digits to be displayed, scale factor B14.

$WDREG_I$: Set of registers, identified as VERBREG, NOUNREG, XREG, YREG, and ZREG for I = 0-4 respectively.

XREG, YREG, ZREG: Cells containing information (with $LPREG_I$ if decimal) associated with display registers R1, R2, and R3 respectively. Noun table information for noun 07 is such as to assign the data loaded for that noun to these same cells, for processing at the end of "ABCLOAD".

Format for DSPTAB+0 - DSPTAB+10

Cell	Bit 11	Bits 10-6	Bits 5-1
DSPTAB+0	-R3S	R3D4	R3D5
DSPTAB+1	+R3S	R3D2	R3D3
DSPTAB+2		R2D5	R3D1
DSPTAB+3	-R2S	R2D3	R2D4
DSPTAB+4	+R2S	R2D1	R2D2
DSPTAB+5	-R1S	R1D4	R1D5
DSPTAB+6	+R1S	R1D2	R1D3
DSPTAB+7			R1D1
DSPTAB+8		ND1	ND2
DSPTAB+9		VD1	VD2
DSPTAB+10		MD1	MD2

For a positive DSPTAB cell, bit 12 is always 1. To flag a change, the complete word is complemented.

R1, R2, and R3 are the three digital display registers, with D5 the least significant digit and D1 the most significant. Each register has an associated sign bit, indicated by -RiS or +RiS. If no sign is to be indicated, neither sign bit is set. The codes for the individual numbers (blank and 0-9) appear on page DATA-37 (under K_{rtb}).

The two-digit noun, verb, and mode (or program) registers are indicated by "N", "V", and "M" respectively. D2 is the least significant digit and D1 the most significant. The codes for the individual numbers are the same as for the digital display registers.

Display Interface Routines

BLANKET (TS set when enter with bits to be set 1 in MPAC+4 of the display job: bits 3-1 = 1 to blank R3, R2, and R1 respectively. Tag is also "LINUS")

```
    $TS = (- MPAC+4) \cap TS$      (sets those bits of TS = 1 that are 0
                                    in MPAC+4 of present job and 1 in TS)

    $MPAC+4_{MPAC+5} = MPAC+4_{MPAC+5} + TS$    ("GODSPRS+1" set MPAC+5 of
                                                 current job to LOCCTR of
                                                 job established there)

    Return
```

CLEANDSP

```
    MPAC+1 = 0   (causes "NVDSP" to establish "JAMTERM", returning to
                  calling address +1 with blanking in "ENDRET")
    $MPAC+4 = 20010_8$     (RESETREQ, FLREQ)    (bit 14 means repeat of
                                                 display flash requested)
    Proceed to "GOFLASH2+1"
```

CLEARMRK

```
    EXTVBACT = 0

    Set bit 1(XDSPFLAG) of FLAGWRD4 = 0

    Return
```

CLOCPLAY Entered from "CLOCKJOB" for V97/V99 generation

```
    MPAC+1 = TS

    $MPAC+4 = 24030_8$    (RESETREQ, BURNREQ, PERFREQ, FLREQ)

    Proceed to "GOFLASH2+1"
```

ENDEXT (Tags also "ENDEXTVB", "ENDMARK", and "TERMEXTV")

```
    Perform "CLEARMRK"

    LOADSTAT = 1

    If bits 14(PRIOIDLE) and 13(NORMIDLE) of FLAGWRD4 = $00_2$:

        Proceed to "NORMRET"    (neither priority or normal display
                                 waiting for response)
    Set bit 6(PINBRFLG) of FLAGWRD4 = 1

    If bit 14(PRIOIDLE) of FLAGWRD4 = 1:     (priority display waiting
                                                          for response)
        Proceed to second line of "REDOPRIO"

    Proceed to "PLAYJUM1"
```

EXDSPRET (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 40040_8 (MKEXREQ, DOTHNRET) (bit 6 means return request after display generated)

Proceed to "GOFLASH2+1"

GODSP (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 0

Proceed to "GOFLASH2+1"

GODSPR (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 0

TS_4 = 0 (Causes return to calling address +1)

Proceed to second line of "GODSPRS+1"

GODSPRET (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 00040_8 (DOTHNRET) (means return request after display generated)

Proceed to "GOFLASH2+1"

GOFLASH (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 00010_8 (FLREQ)

Proceed to "GOFLASH2+1"

GOFLASHR (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 00010_8 (FLREQ)

Proceed to "GODSPRS+1"

<u>GOMARK2</u> (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 40030_8 (MKEXREQ, PERFREQ, FLREQ)

Proceed to "GOFLASH2+1"

<u>GOMARK4</u> (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 40036_8 (MKEXREQ, PERFREQ, FLREQ, R3BLNK, R2BLNK)

Proceed to "GOFLASH2+1"

<u>GOMARKFR</u> (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 40010_8 (MKEXREQ, FLREQ)

Proceed to "GODSPRS+1"

<u>GOPERF1</u> (TS set to octal checklist)

DSPTEM1 = TS

MPAC+1 = 0125_{vn}

MPAC+4 = 00036_8 (PERFREQ, FLREQ, R3BLNK, R2BLNK)

Proceed to "GOFLASH2+1"

<u>GOPERF1R</u> (TS set to octal checklist)

DSPTEM1 = TS

MPAC+1 = 0125_{vn}

MPAC+4 = 00036_8 (PERFREQ, FLREQ, R3BLNK, R2BLNK)

Proceed to "GODSPRS+1"

GOPERF2R (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 00030_8 (PERFREQ, FLREQ)

Proceed to "GODSPRS+1"

GOPERF4 (TS set to octal OPTION1; OPTION2 already loaded)

OPTION1 = TS

MPAC+1 = 0406_{vn}

MPAC+4 = 00014_8 (FLREQ, R3BLNK)

Proceed to "GOFLASH2+1"

GOXDSPF (TS set to verb-noun pattern) Tag also "GOMARKF"

MPAC+1 = TS

MPAC+4 = 40010_8 (MKEXREQ, FLREQ)

Proceed to "GOFLASH2+1"

INITDSP (Entered if restart logic finds an x.1 restart phase setting, indicating a restart-protected normal display)

EBANK = (bits 11-9 of DSPFLG+2)

Channel 7 = (bits 7-5 of RESTREG)

TS = (bits 14-10 of RESTREG)

Change priority of present job to TS ("INITDSP" established with 14_8)

TS = CADRFLSH+2 - 3 (normal calling sequence for display routines such that this returns to step loading TS with information before transfer to display routine)

Proceed to address specified by TS

KLEENEX

MPAC+1 = 0 (see note with "CLEANDSP")

MPAC+4 = 40010_8 (MKEXREQ, FLREQ)

Proceed to "GOFLASH2+1"

```
MARKMONR        (TS set to verb-noun pattern)

     MPAC+1 = TS

     MPAC+4 = $40210_8$  (MKEXREQ, DECMON, FLREQ)

     Proceed to "GOFLASH2+1"

PINBRNCH        Entered from "IDLERET1" if noun used with load verb not
                what was requested, and from "VBRELDSP" if key release
                button depressed with no external monitor suspended and
                with CADRSTOR $\neq$ 0.  Also from other places (e.g. "ALM/END").

     Release interrupts  (may have been inhibited before entrance)

     MPAC+2 = MARK2PAC

     If bits 15(MARKIDLE), 14(PRIOIDLE), and 13(NORMIDLE) of FLAGWRD4
          all = 0:        (no display waiting for response)

          TS = -3   (blank all display registers except program)

          Perform "NVSUB": if busy, proceed; otherwise, proceed

          End of job

     If bit 15(MARKIDLE) of FLAGWRD4 = 1:

          Proceed to "MARKPLAY"

     Set bit 6(PINBRFLG) of FLAGWRD4 = 1

     If bit 14(PRIOIDLE) of FLAGWRD4 = 1:     (priority display waiting
                                                        for response)
          Proceed to second line of "REDOPRIO"

     Proceed to "PLAYJUM1"

PRIODSP   (TS set to verb-noun pattern)

     MPAC+1 = TS

     MPAC+4 = $00110_8$   (PRIOREQ, FLREQ)

     Proceed to "GOFLASH2+1"

PRIODSPR  (TS set to verb-noun pattern)

     MPAC+1 = TS

     MPAC+4 = $00110_8$   (PRIOREQ, FLREQ)

     Proceed to "GODSPRS+1"
```

REFLASH (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 20010_8 (RESETREQ, FLREQ) (bit 14 means repeat of display flash requested)

Proceed to "GOFLASH2+1"

REGODSP (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 20000_8 (RESETREQ)

Proceed to "GOFLASH2+1"

REGODSPR (TS set to verb-noun pattern)

MPAC+1 = TS

MPAC+4 = 20000_8 (RESETREQ)

$TS_4 = 0$ (Causes return to calling address +1)

Proceed to second line of "GODSPRS+1"

VNFLASH (TS set to verb-noun pattern)

MPAC+1 = TS

Increment calling address (i.e. QREG) by 1

Set bit 2(VNFLAG) of FLAGWRD4 = 1

Proceed to second line of "GOFLASH"

VNFLASHR (TS set to verb-noun pattern)

MPAC+1 = TS

Increment calling address (i.e. QREG) by 1

Set bit 2(VNFLAG) of FLAGWRD4 = 1

MPAC+4 = 00010_8 (FLREQ)

$TS_4 = 0$ (Causes return to calling address +"1" (actually 2 due to QREG increment))

Proceed to second line of "GODSPRS+1"

GODSPRS+1 (Entered for those display interface routines, except "MARKMONR", whose tags end in "R", to cause "MAKEPLAY" to be established and then return to calling routine)

$TS_4 = 3$ (Causes return to calling address +4; bypassed if from "GODSPR", "REGODSPR", or "VNFLASHR")

If bit 4 of MPAC+4 = 1: (FLREQ)

Establish "MAKEPLAY" (with VAC area, and with priority and channel 7 setting same as current job)

If bit 4 of MPAC+4 = 0: (FLREQ)

Establish "MAKEPLAY" (with no VAC area, with priority same as current job, and channel 7 setting to 3)

Set bits 11-9 of MPAC+4 = EBANK

MPAC+3 = (calling address +1, in FCADR format, of display interface routine, where "calling address" is that of the routine's CADR, i.e. cell after TC BANKCALL, except for "VNFLASHR")

Set MPAC+1 through MPAC+4 of the "MAKEPLAY" job (determined by using LOCCTR for indexing) equal to MPAC+1 through MPAC+4 of the present job

MPAC+5 = LOCCTR (for use in "BLANKET": appropriate interrupt inhibits done)

Proceed to address specified by ($MPAC+3 + TS_4$) (return to caller)

GOFLASH2+1 (Entered for those display interface routines whose tags do not end in "R", but which require display activity. "MARKMONR" an exception).

Set bits 11-9 of MPAC+4 = EBANK

MPAC+3 = (calling address +1, in FCADR format, of display interface routine: see explanation with "GODSPRS+1")

Proceed to "MAKEPLAY" (after releasing interrupts)

MAKEPLAY (Established by "GODSPRS+1", and entered directly from "GOFLASH2+1", as common starting point for all processing of those display interface routines requiring display activity)

MPAC+7 = (bits 14-10 of PRIORITY) (priority of present job)

Change priority of present job to 33_8

```
If bit 15(MKEXREQ) of MPAC+4 = 1:   (i.e. mark/extended verb)
    COPINDEX = 1        (Tag here "MAKEMARK")
    DSPFLG+1 = MPAC+4
    If bits 6(DOTHNRET) and 4(FLREQ) of MPAC+4 $\neq 00_2$:
        CADRFLSH+1 = MPAC+3    (calling address information
                                needed for flash or return)
    NVWORD+1 = MPAC+1    (verb-noun information)
    If bits 14(PRIOIDLE), 13(NORMIDLE), 12(PDSPFLAG), 8(NRMWTKEY),
        and 7(PROWTKEY) of FLAGWRD4 all = 0:  (no priority or
                                        normal display waiting for
                                        response or busy DSKY to be
                                        released)
        If bit 9(MRKWTKEY) of FLAGWRD4 = 1, End of job  (mark
                                          display waiting for
                                          busy DSKY to be released)
        Proceed to "MARKPLAY"
    If bits 14(PRIOIDLE), 12(PDSPFLAG), and 7(PROWTKEY) of
        FLAGWRD4 all = 0:    (i.e. bits 13 and/or 8 are 1 only)
        Set bit 3(MKOVNORM) of FLAGWRD4 = 1
        MPAC+0 = 1
        Proceed to "JOBXCHS+1"   (remove present normal display
                                  from DSKY interface registers,
                                  then go to "MARKPLAY")
    If bits 11 (MARKWAIT) and 5(RUPTMARK) of FLAGWRD4 $\neq 00_2$:
        End of job    (mark display already in system awaiting
                       generation or restoration)
    Set bit 11 (MARKWAIT) of FLAGWRD4 = 1
    MPAC+0 = 0
    Awaken job (if any) with starting address id = "MARKPLAY"
    If LOCCTR $>$ 0:    (indicates a job found)
        Set starting address of awakened job to "End of job"
    Put present job to sleep (starting address id = "MARKPLAY")
```

```
If bit 7(PRIOREQ) of MPAC+4 = 1:      (i.e. priority display)
    COPINDEX = 0        (Tag here "MAKEPRIO")
    If bit 14(RESETREQ) of MPAC+4 = 0:      (no repeat display)
        If bits 14(PRIOIDLE) and 7(PROWTKEY) of FLAGWRD4 $\neq 00_2$:
            Proceed to "POODOO"  (pattern $21502_8$)   (display in
                                                      system already)
    If bit 14(RESETREQ) of MPAC+4 = 1:
        If CADRFLSH+0 = MPAC+3:
            If DSPLOCK $>$ 0, End of job
    If bits 15(MARKIDLE) and 9(MRKWTKEY) of FLAGWRD4 $\neq 00_2$:
        MPAC+0 = 0
        Proceed to "JOBXCHS+1"  (remove present mark display from
                                 DSKY interface registers, then
                                 go to "OKTOCOPY")
    If bits 13(NORMIDLE) and 8(NRMWTKEY) of FLAGWRD4 $\neq 00_2$:
        MPAC+0 = 1
        Proceed to "JOBXCHS+1"  (remove present normal display from
                                 DSKY interface registers, then
                                 go to "OKTOCOPY")
    Proceed to "OKTOCOPY"
If bits 15(MKEXREQ) and 7(PRIOREQ) of MPAC+4 = $00_2$:  (as they will,
                                                          normal display)
    COPINDEX = 2        (Tag here "IFLEGAL")
    If bit 14(RESETREQ) of MPAC+4 = 0:
        If bit 4(FLREQ) of DSPFLG+2 = 1:
            If bits 13(NORMIDLE), 10(NORMWAIT), 8(NRMWTKEY), and
                4(RUPTNORM) of FLAGWRD4 $\neq 0000_2$:
                Proceed to "POODOO" (pattern $21502_8$) (normal display
                                                          already in system)
    If bit 14(RESETREQ) of MPAC+4 = 1:
        If CADRFLSH+2 = MPAC+3:
            If DSPLOCK $>$ 0, End of job
```

```
  (If bits 15(MKEXREQ) and 7(PRIOREQ) of MPAC+4 = $00_2$):
      DSPFLG+2 = MPAC+4        (Tag here "OKTOPLAY")
      If bits 6(DOTHNRET) and 4(FLREQ) of MPAC+4 $\neq 00_2$:
          CADRFLSH+2 = MPAC+3     (calling address information)
      NVWORD+2 = MPAC+1     (verb-noun information)
      RESTREG = MPAC+7 + Channel 7 (in bits 7-5, superbank value)
      If any of bits 15(MARKIDLE), 14(PRIOIDLE), 12(PDSPFLAG),
          11(MARKWAIT), 9(MRKWTKEY), 7(PROWTKEY), 5(RUPTMARK),
          or 1(XDSPFLAG) of FLAGWRD4 = 1:
          Set bit 10(NORMWAIT) of FLAGWRD4 = 1
          MPAC+0 = 1
          Awaken job (if any) with starting address id = "PLAYJUM1"
          If LOCCTR > 0:    (indicates a job found)
              Set starting address of awakened job to "End of job"
          Put present job to sleep (starting address id = "PLAYJUM1")
      Set bit 5(Key Release) of channel 11 = 0
      If bit 8(NRMWTKEY) of FLAGWRD4 = 1:  (bits 9,7  known 0)
          Set TS = DSPLIST and DSPLIST = 0     (normal display wait-
                                                ing for DSKY to be
      If bit 8(NRMWTKEY) of FLAGWRD4 = 0:       released)
          Set TS = CADRSTOR and CADRSTOR = 0
      Awaken job (if any) with starting address id = TS
      If LOCCTR > 0:    (indicates a job found)
          Set starting address of awakened job to "End of job"
      Proceed to "PLAYJUM1"
OKTOCOPY    (Entered after completion of DSKY interface registers
             resets, if needed, to allow for a priority display)
  COPINDEX = 0
  DSPFLG+0 = MPAC+4
  If bits 6(DOTHNRET) and 4(FLREQ) of MPAC+4 $\neq 00_2$:
      CADRFLSH+0 = MPAC+3     (calling address information)
```

```
NVWORD+0 = MPAC+1     (verb-noun information)
Set bit 5(Key Release) of channel 11 = 0
If bits 9(MRKWTKEY), 8(NRMWTKEY), and 7(PROWTKEY) of FLAGWRD4 $\neq 000_2$:
    Set TS = DSPLIST and DSPLIST = 0   (only bit 7 of interest)
If bits 9(MRKWTKEY), 8(NRMWTKEY), and 7(PROWTKEY) of FLAGWRD4 $= 000_2$:
    Set TS = CADRSTOR and CADRSTOR = 0     (only bit 7 of interest)
Awaken job (if any) with starting address id = TS
If LOCCTR > 0:   (indicates a job found)
    Set starting address of awakened job to "End of job"
Proceed to "REDOPRIO"
```

<u>JOBXCHS+1</u> Entered to replace the present contents of the DSKY interface registers (CADRSTOR and DSPLIST) with 0 for use by a display pre-empting the display system: a priority display replaces a mark/extended verb or normal display; a mark/extended verb display replaces a normal display.

```
Set bit 5(Key Release) of channel 11 = 0
If bits 9(MRKWTKEY), 8(NRMWTKEY), and 7(PROWTKEY) of FLAGWRD4 $\neq 000_2$:
    Set TS = DSPLIST and DSPLIST = 0   (only bits 9,8 of interest)
If bits 9(MRKWTKEY), 8(NRMWTKEY), and 7(PROWTKEY) of FLAGWRD4 $= 000_2$:
    Set TS = CADRSTOR and CADRSTOR = 0 (only bits 9,8 of interest)
Awaken job (if any) with starting address id = TS
If LOCCTR > 0:    (indicates a job found)
    Set starting address of awakened job to "XCHSLEEP"
    Set MPAC+0 of awakened job (using LOCCTR for indexing) equal
        to MPAC+0 of the present job
If MPAC+0 = 0:
    Set bits 15(MARKIDLE) and 9(MRKWTKEY) of FLAGWRD4 = 0
    Set bit 5(RUPTMARK) of FLAGWRD4 = 1     (mark display interrupted)
If MPAC+0 = 1:
    Set bits 13(NORMIDLE) and 8(NRMWTKEY) of FLAGWRD4 = 0
    Set bit 4(RUPTNORM) of FLAGWRD4 = 1     (normal display interrupted)
```

```
    If bit 3(MKOVNORM) of FLAGWRD4 = 1, proceed to "MARKPLAY"

    Proceed to "OKTOCOPY"

XCHSLEEP

    If MPAC+O = 0:

        TS = "MARKPLAY"

    If MPAC+O = 1:

        TS = "PLAYJUM1"

    Awaken job (if any) with starting address id = TS

    If LOCCTR > 0:     (indicates a job found)

        Set starting address of awakened job to "End of job"

    Put present job to sleep with starting address id = TS  (has MPAC+i
        registers set for functions of present job rather than
        previously sleeping job, which are lost)

REDOPRIO  Entered when it is concluded that a priority display on the
          DSKY should take place (provided DSKY not busy).

    PRIOTIME = TIME1

    COPINDEX = 0          (Tag here "KEEPPRIO")

    Proceed to "NVDSP"

MARKPLAY  Entered when it is concluded that a mark/extended verb display
          on the DSKY should take place (provided DSKY not busy).

    Set bit 3(MKOVNORM) of FLAGWRD4 = 0

    Set bit 1(XDSPFLAG) of FLAGWRD4 = 1

    $TS_1$ = 0      (L register, for "NVMONOPT" use)

    If bit 8(DECMON) of DSPFLG+1 = 1:     ("MARKMONR" entrance)

        $TS_1$ = (bits 14-8 of NVWORD+1)

        Skip next line        (i.e. always complement NVWORD+1)

    If bit 5(PERFREQ) of DSPFLG+1 = 1:

        NVWORD+1 = - NVWORD+1

    COPINDEX = 1

    Proceed to second line of "NVDSP"
```

PLAYJUM1 Entered when it is concluded that a normal display on the DSKY should take place (provided system not busy).

```
    COPINDEX = 2

    Proceed to "NVDSP"
```

NVDSP Entered (with COPINDEX set to indicate type of display) to initiate display on DSKY (provided system not busy due to keyboard/uplink use). Tag also "GOPLAY".

```
    $TS_1 = 0$          (L register, for "NVMONOPT" use)

    $MPAC+6 = K_{octmsk_{COPINDEX}}$          (enter here from "MARKPLAY")

    $MPAC+4 = DSPFLG_{COPINDEX}$

    EBANK = (bits 11-9 of MPAC+4)

    $TS_1 = TS_1 +$ (bits 3-1 of MPAC+4)        (bits 3-1 for blanking)

    Set bit 13(2NDPERF) of $DSPFLG_{COPINDEX} = 0$

    MPAC2SAV = MPAC+2      (contains machine address for address-to-

    MARK2PAC = MPAC+2              be-specified nouns)

    If $NVWORD_{COPINDEX} > 0$:

        $TS = NVWORD_{COPINDEX}$

    If $NVWORD_{COPINDEX} = 0$:    (e.g. "CLEANDSP" or "KLEENEX" entered)

        Establish "JAMTERM" (priority $32_8$)    (note that DSPLOCK not checked)

        Proceed to second line of "FLASHSUB"

    If $NVWORD_{COPINDEX} < 0$:      (set in "MARKPLAY")

        NVWORD+1 = - NVWORD+1

        TS = (bits 7-1 of NVWORD+1)  +  $01200_8$    (verb 05 scaled)

        If bit 8(DECMON) of $DSPFLG_{COPINDEX} = 1$: (i.e. from "MARKMONR")

            $TS = TS + 2600_8$      (changes verb to 16)

    Proceed to "NV50DSP"
```

NV50DSP

```
    Perform "NVMONOPT": if return to calling address +1(busy), proceed
                         to "REST"
                        otherwise, proceed        (TS and $TS_1$ set)
```

```
Set bit 6(Flash) of channel 11 = 0

MPAC+2 = MPAC2SAV

MPAC+6 = $K_{octmsk_{COPINDEX}}$

MPAC+4 = $DSPFLG_{COPINDEX}$

EBANK = bits 11-9 of MPAC+4

Set bits 9(MRKWTKEY), 8(NRMWTKEY), and 7(PROWTKEY) of FLAGWRD4 = 0

TS = MPAC+4     (bits 3-1 are R3BLNK, R2BLNK, and R1BLNK)

Perform "BLANKSUB": if return to calling address +1 (busy, as should
                     not be), proceed to "NVDSP"
                    otherwise, proceed

If bits 13(2NDPERF) and 5(PERFREQ) of MPAC+4 = $01_2$:     (Tag here "PERFCHEK")

    Set bit 13 of $DSPFLG_{COPINDEX}$ = 1

    $TS_1$ = 0     (L register, for "NVMONOPT" use)

    If bit 15(MKEXREQ) of $DSPFLG_{COPINDEX}$ = 1:

        TS = bits 14-8 of NVWORD+1  (i.e. $xx00_{vn}$)   (Tag here "MARKPERF")

        Proceed to "NV50DSP"

    If bit 12(BURNREQ) of $DSPFLG_{COPINDEX}$ = 0:  (i.e. not from
                                                      "CLOCKJOB")
        TS = $5000_{vn}$

        Proceed to "NV50DSP"

    TS = $9700_{vn}$ - NVWORD1     ($9700_{vn}$ for NVWORD1 = 0, $9900_{vn}$
                                   for NVWORD1 = $-400_8$: NVWORD1 should
                                   not be confused with NVWORD+i).
    Proceed to "NV50DSP"

If bit 4(FLREQ) of MPAC+4 = 1, proceed to "FLASHSUB"  (Tag here "GOANIDLE")

If bit 6(DOTHNRET) of MPAC+4 = 1:

    TS = $CADRFLSH_{COPINDEX}$

    Change priority of this job to that in bits 14-10 of MPAC+7
                                                    (restoring original
    Proceed to address specified by TS                        value)
```

```
          If bits 7-1 of $NVWORD_{COPINDEX} = 0$:      (i.e. $xx00_{vn}$)
               Proceed to "FLASHSUB"
          End of job
     REST      Entered from beginning of "NV50DSP" for a busy return from
               "NVMONOPT"  (DSKY or uplink being used externally)
          If  CADRSTOR $\neq$ 0, End of job
          If COPINDEX = 0:
               Set bit 7(PROWTKEY) of FLAGWRD4 = 1   (priority display)
          If COPINDEX = 1:
               Set bit 9(MRKWTKEY) of FLAGWRD4 = 1   (mark display)
          If COPINDEX = 2:
               Set bit 8(NRMWTKEY) of FLAGWRD4 = 1   (normal display)
          Proceed to "NVSUBUSY"
     NVSUBUSY
          If  CADRSTOR $\neq$ 0, proceed to "POODOO"  (pattern $21206_8$)
          If  DSPLIST $\neq$ 0, proceed to "POODOO" (pattern $21206_8$)
          Set bit 5(Key Release) of channel 11 = 1
          If COPINDEX = 0:
               DSPLIST = "REDOPRIO"
          If COPINDEX = 1:
               DSPLIST = "MARKPLAY"
          If COPINDEX = 2:
               DSPLIST = "PLAYJUM1"
          Put present job to sleep (starting address id = DSPLIST)
     FLASHSUB
          Set bit 6(Flash) of channel 11 = 1
          MPAC+3 = COPINDEX
```

```
    If COPINDEX = 0:
        Set bit 14(PRIOIDLE) of FLAGWRD4 = 1
    If COPINDEX = 1:
        Set bit 15(MARKIDLE) of FLAGWRD4 = 1
    If COPINDEX = 2:
        Set bit 13(NORMIDLE) of FLAGWRD4 = 1
    If CADRSTOR $\neq$ 0:
        If bits 15(MARKIDLE) and 6(PINBRFLG) of FLAGWRD4 = $00_2$:
            Proceed to "POODOO"  (pattern $21502_8$)
        End of job
    Proceed to "ENDIDLE"
ENDIDLE
    If CADRSTOR $\neq$ 0, proceed to "POODOO"  (pattern $21206_8$)
    If DSPLIST $\neq$ 0, proceed to "POODOO"  (pattern $21206_8$)
    CADRSTOR = "IDLERET1"
    Put present job to sleep (starting address id = CADRSTOR)
IDLERET1
    If  MPAC+0 $\neq$ 21, 22, or 23:   (loaded in "RECALTST" with verb)
        Proceed to "OKTOENT"
    TS = (bits 7-1 of $\text{NVWORD}_{\text{MPAC+3}}$)    (MPAC+3 has COPINDEX)
    If  MPAC+1 = TS:             (loaded in "RECALTST" with noun)
        Proceed to "OKTOENT"
    Proceed to "PINBRNCH"  (noun loaded not what display routine asking)
```

```
OKTOENT   Entered after response to flash
    If bit 15(MARKIDLE) of FLAGWRD4 = 1:      (mark display waiting)
        Set bit 3(MKOVNORM) of FLAGWRD4 = 0
        Proceed to "ENDRET"    ("NORMRET" processing done e.g. via "ENDEXT")
    If bit 14(PRIOIDLE) of FLAGWRD4 = 0:
        Proceed to "NORMRET"
    If  NVWORD+0 = 0:          (priority "JAMTERM"-type action, e.g. from "R61CSM")
        Proceed to "NORMRET"
    TS = PRIOTIME - TIME1      (priority display, enforce minimum delay)
    If  $TS > 0$:
        TS = 163.84 - TS  seconds  (correct for TIME1 overflow)
    If  $(|TS| + K_{m2sec}) \leq 0$:
        Proceed to second line of "REDOPRIO"  (response too near initiation)
    Proceed to "NORMRET"
NORMRET   Entered from "OKTOENT" and "ENDEXT"
    If bits 11(MARKWAIT) and 5(RUPTMARK) of FLAGWRD4 $\neq 00_2$:
        Set bits 11(MARKWAIT) and 5(RUPTMARK) of FLAGWRD4 = 0
        TS = "MARKPLAY"
        Awaken job with starting address id = TS
        Proceed to "ENDRET"
    If bits 10(NORMWAIT) and 4(RUPTNORM) of FLAGWRD4 $\neq 00_2$:
        Set bits 10(NORMWAIT) and 4(RUPTNORM) of FLAGWRD4 = 0
        TS = "PLAYJUM1"
        Awaken job with starting address id = TS
        Proceed to "ENDRET"
    If bits 6(DOTHNRET) and 4(FLREQ) of DSPFLG+2 $\neq 00_2$:
        Proceed to "ENDRET"      (both bits set by "R00")
    If NVWORD+2 = 0, proceed to "ENDRET"
```

```
    Establish "PLAYJUM1"    (priority $15_8$)
    Proceed to "ENDRET"

ENDRET
    If  LOADSTAT > 0, End of job      (e.g. from "ENDEXT")
    If  LOADSTAT = +0, Error
    If bit 2(VNFLAG) of FLAGWRD4 = 1:
        If  MPAC+3 = 2:      (i.e. normal display)
            Set bit 2(VNFLAG) of FLAGWRD4 = 0    (Tag here "VNRET")
            TS = LOADSTAT + 1
            If  TS > 0:    (LOADSTAT was -0, "otherwise" response of an
                                                  e.g. data load)
                MPAC+3 =  CADRFLSH+2  - 3
            If  TS = 0:    (LOADSTAT was -1, a PRO response)
                MPAC+3 = CADRFLSH+2  - 1
            If  TS < 0:    (LOADSTAT was -2, a V34E response)
                MPAC+3 = "GOTOPOOH"
            Proceed to "ENDRET2"
    MPAC+3 =  $CADRFLSH_{MPAC+3}$  +  LOADSTAT  +  2
    Proceed to "ENDRET2"

ENDRET2
    Set those bits of 15-12 and 6 in FLAGWRD4 to 0 that are 1 in MPAC+6
        (If MPAC+6 indicates priority, bits 14(PRIOIDLE) and 6(PINBRFLG)
         reset; if indicates mark, bit 15(MARKIDLE) reset; if normal,
         bits 13(NORMIDLE) and 6(PINBRFLG) reset.  MPAC+6 loaded near
         start of "NV50DSP").
    TS = -3          (blank all display registers except program)
    Perform "NVSUB": if return to calling address +1(busy), proceed
                     otherwise, proceed
    Change priority of present job to that specified in bits 14-10
         of MPAC+7
    Proceed to address specified by MPAC+3
```

Quantities in Computations

See also list of major variables and list of routines

CADRFLSH$_i$ (i = 0,1,2): Set of three single precision erasable memory cells used to retain return address information (in FCADR format, with bits 15-11 giving the FBANK and bits 10-1, plus 2000$_8$, giving the S-register contents) from the display interface routine (for bits 6(DOTHNRET) or 4(FLREQ) of DSPFLG$_i$ being 1). CADRFLSH+0 is the address for a priority display, CADRFLSH+1 for a mark/extended verb display, and CADRFLSH+2 is for a normal display. For most flashing three-response displays (all except "VNFLASH(R)"), the cells contain the information for the terminate return address, and are incremented in "ENDRET" for proceed or "otherwise" (e.g. data enter or recycle) responses. For "VNFLASH(R)", the cells contain one greater than the PRO return address (for "INITDSP" reasons): a terminate response exits directly to "GOTOPOOH", while an "otherwise" response causes the return to be 3 less than the present contents, i.e. the acquisition of the verb-noun pattern. CADRFLSH+2 also has the program notation "TEMPFLSH". "INITDSP", entered if restart logic invoked, causes transfer to CADRFLSH+2 - 3 (the "setting" of TS for the standard display routine calling sequence): consequently, if CADRFLSH+2 is set to "XXXX" +3, this logic will cause "XXXX" to be entered (and is so used at the end of "ROO").

CADRSTOR: See Data Input/Output. It is 0 if no job in "ENDIDLE", and otherwise is set to "IDLERET1".

COPINDEX: Single precision cell, scale factor B14, used to retain indexing information on the type of display being processed (0, 1, and 2 for priority, mark/extended verb, and normal respectively). It is assigned to the LOC cell of the job register set (which is written over with job starting information if the job is put to sleep).

DSPFLG$_i$ (i = 0,1,2): Set of three single precision erasable memory cells used to retain values of control-bit information for individual displays (priority, mark/extended verb, and normal respectively). FLAGWRD4 is used to contain control and status information for the display routines as a whole. When a display interface routine is initially entered, cell MPAC+4 is used to store temporarily the required settings for the pertinent DSPFLG cell. Individual bits in the DSPFLG cells have the following significance(words also "EBANKSAV", "MARKFLAG", and "EBANKTEM"):

Bit	Symbol	Meaning
15	MKEXREQ	Mark or extended verb request. Loaded only in DSPFLG+1 (and, of course, MPAC+4): bit in other DSPFLG cells expected to be 0.
14	RESETREQ	Repeat request (causes checks for already existing priority or normal displays to be bypassed: otherwise, an abort (pattern 21502$_8$) would be caused if a display of the same type already in system). Bit can also be set via the "BLANKET" routine (if a proper "R" display is used), as well as by the "RExxxx" interface routines.

Bit	Symbol	Meaning
13	2NDPERF	Second part of "perform" display sequence has been started (to "paste" a code verb, e.g. 50, after completion of a display loading). Bit set 0 at the start of "NVDSP" and to 1 in "NV50DSP" if bit 5(PERFREQ) is 1 (and this bit, of course, is 0).
12	BURNREQ	Special display verb to request ignition authorization should be generated. Verb 99 is used if NVWORD1 = -400_8; if NVWORD1 = 0, then verb 97 is "pasted" instead. See "CLOCPLAY".
11-9		Value of EBANK when display routine entered (restored in "NVDSP" to ensure proper value when job calling routine, if appropriate, is started again).
8	DECMON	Decimal monitor verb required for perform-type display (verb pasted over with one in calling sequence information), set for entrance to "MARKMONR". No requirement for resetting bit exists ($DSPFLG_i$ loaded completely for new display).
7	PRIOREQ	Priority display requested (interface routine "PRIODSP(R)" used). Loaded only in DSPFLG+0.
6	DOTHNRET	Return requested after performance of the DSKY loading (interface routine "EXDSPRET" or "GODSPRET" used), to allow, for example, a verb "paste" without a flash.
5	PERFREQ	Perform-type display requested (see bit 13).
4	FLREQ	Flashing-type display (with associated requirement for processing of crew response) requested.
3	R3BLNK	Blanking of register R3 requested (done by transfer to "BLANKSUB" in "NV50DSP"). Bit can be set (usually) by use of an "R" display interface routine, followed by transfer to "BLANKET" with TS = $1xx_2$.
2	R2BLNK	Blanking of register R2 requested (can be set by transfer to "BLANKET" with TS = $x1x_2$). See bit 3 discussion.
1	R1BLNK	Blanking of register R1 requested (can be set by transfer to "BLANKET" with TS = $xx1_2$). See bit 3 discussion.

DSPLIST: See Data Input/Output. It is a DSKY interface cell used to retain return address information if the display system is busy due to crew or uplink use (loaded in "NVSUBUSY" which is entered from "REST"). For index convenience, the cell is stored to make it the same as CADRSTOR+1.

DSPLOCK: See Data Input/Output (set 1 at start of "CHARIN" to indicate that the display system is busy with DSKY or uplink inputs, and hence should not be used by internal routines, not even for a priority display). It is checked automatically at the start of "NVSUB" and "BLANKSUB", causing return to calling address +1 for a non-zero value. Same check at start of "NVMONOPT" also.

EBANK: See Data Input/Output (only bits 11-9 are defined).

EXTVBACT: See Verb Definitions.

FLAGWRD4: Single precision flagword cell whose individual bits are used to contain control and status information for the display routines as a whole (DSPFLG is used for individual displays). It is set zero in "STARTSB2"[1]. Individual bits have the following significance:

Bit	Symbol	Meaning
15	MARKIDLE	A mark/extended verb display is waiting for a response (is the job asleep in "ENDIDLE"). Bit set in "FLASHSUB" prior to entering "ENDIDLE", and used in "OKTOENT" to identify the type of response received, whereupon it is reset (in "ENDRET2") prior to returning to caller. Could also be reset in "JOBXCHS+1" if display pre-empted by a priority display, whereupon bit 5(RUPTMARK) would be set 1 to record this fact.
14	PRIOIDLE	A priority display is waiting for a response (is the job asleep in "ENDIDLE"). Bit set in "FLASHSUB" prior to entering "ENDIDLE", and used in "OKTOENT" to identify the type of response received, whereupon it is reset (in "ENDRET2") prior to returning to caller.
13	NORMIDLE	A normal display is waiting for a response (is the job asleep in "ENDIDLE"). Bit set in "FLASHSUB" prior to entering "ENDIDLE", and used in "OKTOENT" to identify the type of response received, whereupon it is reset (in "ENDRET2") prior to returning to caller. Could also be reset in "JOBXCHS+1" if display pre-empted by a priority or mark/extended verb display, whereupon bit 4(RUPTNORM) would be set 1 to record this fact.

Bit	Symbol	Meaning
12	PDSPFLAG	Bit set 1 to indicate that R60 display should be of the priority type (checked in "CHKLINUS", and set in "R61CSM" if a maneuver must be done by R60 means, then reset).
11	MARKWAIT	A mark/extended verb display is waiting to be initiated. Bit is set in "MAKEPLAY" if a priority display presently on DSKY (assuming bit 5 (RUPTMARK) is 0), and job put to sleep with starting address id = "MARKPLAY". Job awakened, and bit reset, in "NORMRET" after completion of present display effort. Helps give display sequence for DSKY of: crew/uplink use, priority display, interrupted mark/extended verb display, waiting mark/extended verb display, waiting normal display, interrupted normal display. An interrupted and a waiting display of the same type, however, cannot be in system at same time. In addition, waiting or interrupted normal displays will <u>not</u> be restarted by an end to a mark/extended verb display until e.g. "ENDEXT" is entered, and bit 1 of FLAGWRD4 serves to keep out normal displays once a mark/extended verb display attempt initiated (until "CLEARMRK" done).
10	NORMWAIT	A normal display is waiting to be initiated. The bit is set 1 in "MAKEPLAY" if a priority or mark/extended verb activity in the display system is indicated (including bit 1(XDSPFLAG)), and the job is put to sleep with starting address id = "PLAYJUM1". Job awakened, and bit reset, in "NORMRET" after completion of present display effort, provided bits 11 and 5 are 0.
9	MRKWTKEY	A mark/extended verb display attempt found the display system busy (so must wait for a key release). Bit is set in "REST" to indicate that a display of this type in DSPLIST (via "NVSUBUSY"). When job is awakened (e.g. by "RELDSP"), it starts at "MARKPLAY", and the bit would be reset in "NV50DSP" after successful initiation of the display. Bit could also be reset in "JOBXCHS+1" if a mark attempt was pre-empted by a priority display (cf. bit 15, since it would be reset at same time).
8	NRMWTKEY	A normal display attempt found the display system busy (see bit 9 discussion). When awakened, starts at "PLAYJUM1"; could also be reset if normal display pre-empted (cf. bit 13, since it would be reset at the same time).
7	PROWTKEY	A priority display attempt found the display system busy (see bit 9 discussion). When awakened, starts at "REDOPRIO".

Bit	Symbol	Meaning
6	PINBRFLG	Bit set to 1 in "PINBRNCH" if it is concluded that interference with the existing display has taken place (e.g. computations at the end of "FLASHSUB" conclude that a data enter verb was received but the associated noun was <u>not</u> that originally requested by the display). Bit also set 1 in "ENDEXT" if a priority or normal display waiting for response (bits 14 and/or 13 $\neq$ 0). Bit reset 0 in "ENDRET2" upon successful conclusion of a priority or normal display (bit not set in "PINBRNCH" if bit 15 (MARKIDLE) = 1). Used in "FLASHSUB" to bypass abort otherwise caused if CADRSTOR $\neq$ 0.
5	RUPTMARK	A mark/extended verb display (or display attempt, e.g. if system busy) has been interrupted by a priority display. Bit is set in "JOBXCHS+1" (cf. bits 15 and 9, which would be reset) if the mark/extended verb activity is pre-empted by a priority display request. Job awakened, and bit reset, in "NORMRET" (same effect there as bit 11).
4	RUPTNORM	A normal display (or display attempt, e.g. if system busy) has been interrupted by a priority display or by a mark/extended verb display. Bit is set in "JOBXCHS+1" (cf. bits 13 and 8, which would be reset then), and used in "NORMRET" to awaken job, provided bits 11 and 5 are 0 (same effect there as bit 10), whereupon it is set 0.
3	MKOVNORM	Mark display is to pre-empt a normal display: set as described for bit 4 setting for mark/extended verbs. Bit set 1 in "MAKEPLAY" to control proper exit from "JOBXCHS+1", namely to "MARKPLAY", where bit is set 0 again. Also reset in "OKTOENT".
2	VNFLAG	Bit set 1 by "VNFLASH(R)" to indicate a special use of display interface (see MPAC+3), and reset in "ENDRET" at the conclusion of processing for it.
1	XDSPFLAG	Bit set 1 in "MARKPLAY" to signify that a mark/extended verb display has been initiated (or an attempt of that type). Used in "MAKEPLAY" to keep normal displays from using the DSKY if bit is 1. Bit reset by "CLEARMRK" (entered e.g. from "ENDEXT").

K_{m2sec}: Single precision constant, program notation "-2SEC", scale factor B14, units centi-seconds. Value is -200 x 2^{-14}, corresponding to -2 seconds.

K_{octmsk_i}: Set of three single precision octal constants used as bit masks, program notation (i = 0) "PRIOOCT", values 20144_8, 42424_8, and 11254_8 respectively. The setting of bit 3 = 1 is a holdover from a former flagbit routine, and is no longer employed.

LOADSTAT: See Data Input/Output. Values of -0, -1, and -2 are set for "otherwise" (e.g. V32E or data enter), PRO, and V34E responses respectively.

LOCCTR: Single precision cell set in executive system to -1 if the "job wake" routine found no job with the indicated starting address id; otherwise, it is set to an index parameter (B14) for locating the core register set of the awakened job (should not be zero if the "job wake" routine itself was entered from a job). See 3420.5-27 for details. It is set similarly when a job established.

MARK2PAC: Single precision cell used to retain the value of MPAC+2 when "NVDSP" was last entered, and used to load the proper value when "PINBRNCH" is entered. MPAC+2 is used to contain the machine address for address-to-be-specified nouns used internally.

MPAC$_i$ (i = 0-7): Set of cells associated with each job register (a collection of 12 erasable memory cells assigned to each job) set, used in the interpretive language for vector and scalar accumulator functions, and used in machine language as temporary storage cells. Assignments of significance for the display interface routines are listed below (other quantities also use these cells, but are listed separately). Note that each job has its own job register set (hence there could be distinct MPAC+7 values, for example, for several different display interface functions simultaneously, each waiting for the necessary action to be completed).

MPAC+0: Index cell used by "XCHSLEEP" and "JOBXCHS+1", set to 0 to change mark/extended verb displays and 1 for normal displays. Program notation is "FACEREG". Cell loaded in "RECALTST" with the verb pattern (VERBREG) when "RECALTST" entered.

MPAC+1: Cell loaded with verb/noun pattern when display interface routine entered (frequently the TS setting, mechanized as the hardware accumulator). Program notation "PLAYTEM1". Cell loaded in "RECALTST" with the noun pattern (NOUNREG) when "RECALTST" entered.

MPAC+2: Cell loaded with machine address for address-to-be-specified nouns.

MPAC+3: Cell loaded with calling address information (i.e. the "immediate return" for non-flash or the "terminate return" for flash). It is loaded with address of step "L" in the sequence (program notation "PLAYTEM3"):

```
L-3   CA   (TS information, e.g. verb/noun)
L-2   TC   BANKCALL    (routine to change program banks)
L-1   CADR XXXX        (address of interface routine)
L               (terminate return)
L+1             (proceed return)
L+2             (enter/recycle return)
L+3             (return for "R" flash-type displays, usually)
```

This sequence does not apply to "VNFLASH(R)" sequences. For these two calling routines, the entrance point is in fixed-fixed memory, meaning that the "BANKCALL" interface routine is not necessary. Hence MPAC+3 is loaded with address of step "L" in the following sequence:

```
L-3      CA    (TS information, e.g. verb/noun) (enter/V32
L-2      TC    "VNFLASH(R)"                           return)
L-1            (proceed return)
L              (return for "VNFLASHR")
```

The setting of "L" as indicated is accomplished by incrementing the Q register at the entrance to the routine: this allows "INITDSP" restart logic to transfer to L-3 to obtain the TS information. The return to L-1 for a PRO and L-3 for an "otherwise" is accomplished by the logic in "ENDRET"; a V34E causes direct exit to "GOTOPOOH".

The same cell is loaded with the value of COPINDEX at the start of "FLASHSUB", where has notation COPMPAC, to retain COPINDEX information when job put to sleep.

MPAC+4: Cell located with appropriate DSPFLG pattern in display interface routine. Program notation "PLAYTEM4" and "TEMPOR2".

MPAC+5: Cell loaded in "GODSPRS+1" with the value of LOCCTR for the display job established (for "R" type displays, except "MARKMONR"), to permit "BLANKET" to be entered to set additional bits of MPAC+4, such as bits 3-1 (RiBLNK).

MPAC+6: Cell used to contain the masking bit pattern (K_{octmsk}) for the display, employed in "ENDRET2" to reset the proper bits of FLAGWRD4. Program notation "GENMASK".

MPAC+7: Cell loaded in "MAKEPLAY" with the priority of the using routine, for use in "ENDRET2" to restore the same priority. This works for "R" type displays also, since "GODSPRS+1" establishes "MAKEPLAY" with the same priority as that of the calling job. Program notation is "USERPRIO": cell strictly speaking is not part of the "MPAC" set, but instead is used in interpretive language for indicating type of computation ("MODE" value) being performed.

MPAC2SAV: Cell (part of job register set, corresponding to MPAC+9) used to retain value of MPAC+2 at the start of "NVDSP", for restoration after "NVMONOPT" (which uses the cell for temporary storage purposes) has been performed.

NVWORD$_i$ (i = 0,1,2): Set of three single precision erasable memory cells used to retain values of the noun-verb pattern (bits 14-8 for verb and 7-1 for noun) for priority, mark/extended verb, and normal displays respectively. Program notations "NVWORD", "MARKNV", and "NVSAVE" respectively (NVWORD+2 and DSPFLG+2 set 0 in "DOFSTART").

NVWORD1: See Burn Control.

OPTION1, OPTION2: Pair of single precision cells displayed in octal by using noun 06 (implemented in some cases by loading OPTION2 before entering the display interface routine "GOPERF4", and setting TS to OPTION1). OPTION1 is used to contain the "option code identification" (for the purpose of the display, such as specification of IMU orientation) and OPTION2 is used to contain the "option selection" consistent with OPTION1 (such as "IMU orientation to preferred alignment"). A manual input changing only OPTION1 (leaving OPTION2 unchanged) will be interpreted as a response, and can not be used to change the subsequent program logic.

PRIORITY: See General Program Control.

PRIOTIME: Single precision value of least significant half of computer clock (TIME1), scale factor B14, units centi-seconds, used to retain the time when "REDOPRIO" last entered. Cell is used in "OKTOENT" to check that the present magnitude of the difference between PRIOTIME and TIME1 (after allowing for TIME1 overflow) is greater than - K_{m2sec}: if not, it is suspected that the response was associated with the interrupted display rather than the priority display, and consequently the priority display is repeated (without, however, resetting PRIOTIME). Since only single precision times are used, a time lapse of 163.84n +2 - 0 seconds (n an integer) is interpreted as being in the "too quick" zone.

RESTREG: Single precision cell used to retain restart information for normal displays. It is initialized to 30000_8 in "DOFSTART", meaning priority 30_8, and is used in "INITDSP" (entered if a restart encountered with a x.1 restart phase setting) to properly restore value of PRIORITY and Channel 7 (FEXT or SUPERBNK). The cell is also loaded in "MAKEPLAY" (for a normal display only).