

APOLLO GUIDANCE COMPUTER
INFORMATION SERIES

Issue 16

PROGRESS CONTROL

and

FRESH START AND RESTART

FR-2-116

15 December 1965

CONTENTS

Paragraph		Page
16-1	INTRODUCTION	16-1
16-4	PROGRESS CONTROL	16-3
16-6	Phases	16-3
16-8	Phase Tables.	16-4
16-10	Functional Description	16-4
16-12	Entry at PHASCHNG	16-5
16-18	Entry at GETPHASE	16-6
16-22	Entry at MODROUT	16-9
16-27	Entry at MAJEXIT	16-10
16-31	Entry at PHASWAIT	16-11
16-33	Detailed Description	16-12
16-35	Phase Modification	16-12
16-48	Phase Determination.	16-17
16-54	Manual Mission Function Modification	16-18
16-64	Mission Function Termination	16-20
16-73	Mission Function Deactivation	16-24
16-80	FRESH START AND RESTART	16-27
16-82	Functional Description	16-27
16-96	Detailed Description	16-32
16-98	Fresh Start.	16-32
16-107	Restart	16-35

ILLUSTRATIONS

Figure		Page
16-1	Phase Table	16-4
16-2	Progress Control Functional Flowchart	16-7
16-3	Fresh Start and Restart Functional Flowchart	16-29
16-4	Sample Comparison	16-38

TABLES

Table		Page
16-1	Phase Numbers	16-3
16-2	New Phase Code Positioning	16-15
16-3	Masking Quantities	16-15
16-4	Program Displays	16-23
16-5	Phase Word Comparisons	16-37

APPENDICES

Appendix

16-A

ATTACHMENTS

Number

- 16-1 Progress Control Detailed Flowchart
- 16-2 Fresh Start and Restart Detailed Flowchart

16.1 INTRODUCTION

16-2. This is the sixteenth issue of the AGCIS which is published to inform the technical staff at MIT/IL and Raytheon about the Apollo guidance computer (AGC) subsystem. This issue contains a description of program sections Progress Control and Fresh Start and Restart as they appear in SUNRSE 45 and SUNRSE 69. These two program sections are presented together because of the interaction between them. Progress Control supplies Fresh Start and Restart with the Phase at which a Mission Function should resume operation following a GO sequence (GOJAM). Fresh Start and Restart initializes the phase tables for Progress Control following a fresh start or following a restart in which the phase tables do not agree.

16-3. The descriptions contained in this issue are presented on two levels: functional and detailed. The functional level describes what the program section does for the overall Sunrise Program; the detailed level describes how these operations are accomplished. (Issue 15 contains a general description of all Block I AGC programs and program sections in paragraphs 15-98 through 15-203.)

16.4 PROGRESS CONTROL

16-5. Program section Progress Control is a group of routines that performs control operations for all Mission Functions except System Test. These control operations include starting, stopping, and changing the Mission Functions. Each Mission Function is divided into sections called Phases; Progress Control maintains a table of phase numbers in triplicate to indicate the status of the control operations.

16-6. PHASES

16-7. A Phase is a milestone used to identify a segment of a Mission Function. This identification enables program control to be returned to the beginning of a particular Phase within the Mission Function following a GOJAM Sequence, thus avoiding the necessity of returning to the beginning of the interrupted Mission Function. Phases are numbered from octal 0 to octal 37 (table 16-1); however, only those Phases numbered between 10 and 36 identify segments of a Mission Function. Phases numbered from 1 to 7 are defined as manual Phases and are used only at the DSKY's to provide the operator access to a particular internal Phase or sequence of internal Phases. Phases 0 and 37 have special significance.

TABLE 16-1
PHASE NUMBERS

Number	Meaning
0	Request to terminate an active Mission Function. Since this Phase is entered via a keyboard action, it is considered a manual Phase.
1 through 7	Manual Phases. The numbers used and the significance of each varies with the Mission Function.
10 through 36	Internal Phases. The numbers used and the significance of each varies with the Mission Function.
37	Mission Function has been terminated.

16-8. PHASE TABLES

16-9. The phase tables are three sets of registers, each containing the same information. Two of the tables (PHASETAB and BACKPHAS) contain duplicate phase information in true form, while the third table (PHASEBAR) contains the same phase information in complement form. Each phase table (figure 16-1) consists of four 15-bit registers or phase words. Each phase word is divided into three 5-bit cells; each of the cells in a table corresponds to a particular Mission Function. The content of each cell is the five-bit number denoting the Phase of the Mission Function associated with that cell.

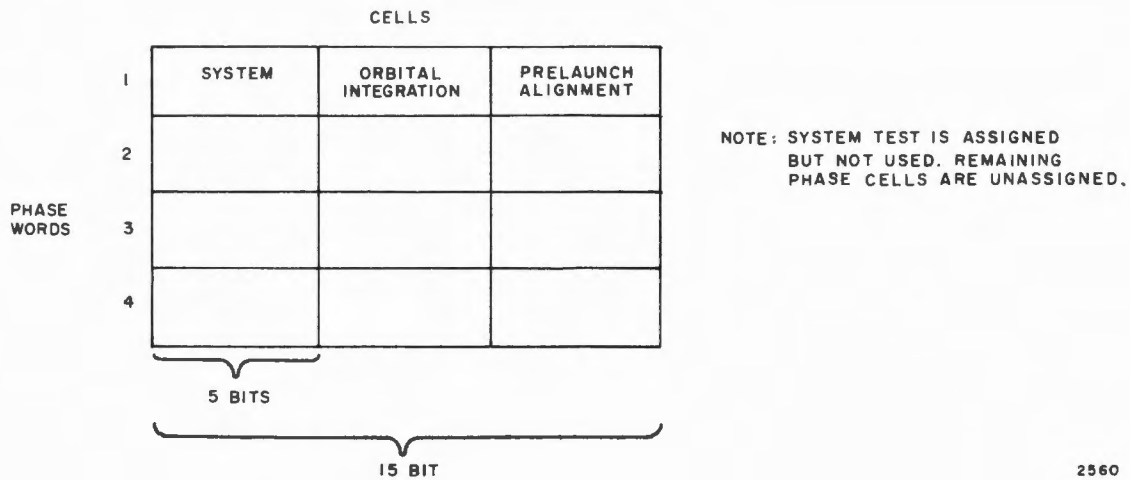


Figure 16-1. Phase Table

16-10. FUNCTIONAL DESCRIPTION

16-11. Each operation performed by Progress Control has its own entry point or entry routine. The following are the operations performed by Progress Control and their associated entry routines:

<u>Operation</u>	<u>Entry Routine</u>
Phase Modification	PHASCHNG
Phase Determination	GETPHASE
Manual Mission Function Modification	MODROUT (Keyboard Request)
Mission Function Termination	MAJEXIT
Mission Function Deactivation	PHASWAIT

All the routines listed are entered from a Mission Function except for Routine MODROUT. By convention in the Sunrise programs (except for operator intervention), the contents of a phase cell may only be modified by the Mission Function associated with that cell. Routine MODROUT is entered from program section Pinball as the result of a keyboard request to change a Mission Function and/or Phase.

16-12. ENTRY AT PHASCHNG

16-13. Progress Control is entered at Routine PHASCHNG (figure 16-2) from a requesting Mission Function when the Phase of a Mission Function (not necessarily the requesting Mission Function itself) requires changing and the phase tables must be updated to reflect this change. The requesting Mission Function requests the execution of Routine PHASCHNG, furnishing the new phase code and the mission function code of the Mission Function whose phase is to be changed via the calling sequence:

L	TC	PHASCHNG
L+1	OCT	N
L+2		(Return address following the detection of a request to terminate the Mission Function specified by D ₁)
L+3		(Normal return address)

where OCT N has the octal form 0D₃D₂0D₁ with digits D₃ and D₂ specifying the new phase code while D₁ specifies the mission function code. In the Sunrise programs, the mission function code specified is always that of the requesting Mission Function, but it must be remembered that this need not always be the case in future programs.

16-14. Routine PHASCHNG initially transfers program control to Routine PHASECOM where the address of OCT N (L+1) is saved for later use. Program control is then transferred to Routine PHASCOM2 which executes an INHINT to prevent any program interrupts while the phase tables are being updated. This routine then stores OCT N which is later used to update the phase tables. Next, return address L+2 is stored and program control is transferred to Routine PHASESUB which copies and stores the old Phase identified with the Mission Function denoted by D₁ of OCT N and then transfers program control to Routine PHASCH3.

16-15. Routine PHASCH3 tests the old Phase to determine if there has been a keyboard request to terminate the Mission Function (phase code of 0) since the last examination of the Phase for this Mission Function. If there has been a request to terminate the Mission Function, program control is transferred directly to Routine PHASCH4 and the return address remains L+2.

If no such request has been made, program control goes to Routine INCET4 where the return address is incremented from L+2 (request to terminate detected) to L+3 (the normal return address). Program control is then transferred to Routine PHASCH4. In Routine PHASCH4, OCT N is processed such that the new phase code is separated from the mission function code and the Phase is entered into the accumulator, right-hand justified. Program control is then transferred to Routine SETPHASE.

16-16. Routine SETPHASE erases the old phase code and sets the new phase code into that cell (in all three phase tables) identified with the Mission Function specified by D1 of OCT N. Upon return to Routine PHASCH4 at location PHASCH4+4, the old Phase is stored for a second time to allow for further testing of this Phase. Program control is then transferred to Routine ONSKIP where the old phase is tested to determine whether or not the Mission Function was in the terminated state (phase code of 37) when entering Routine SETPHASE. Since it is impossible for a terminated Mission Function to require a Phase change, the phase code cannot be 37; therefore, program control is transferred to Routine PHASEREL. (The redundant use of Routine ONSKIP at this point is necessary because entry point MODROUT also shares Routine PHASCH4.)

16-17. Routine PHASEREL executes a RELINT, which offsets the INHINT executed in Routine PHASCOM2, to allow program interrupts to be accepted and processed. Program control is transferred to Routine PHASOUT which provides a return of program control to the requesting Mission Function at location L+2 or L+3 depending upon whether or not termination of the Mission Function has been requested. If there was a request to terminate the active Mission Function, the action to be taken by the Mission Function at location L+2 is at the option of the designer; it is often a transfer of program control to Progress Control at entry MAJEXIT for effecting the termination, sometimes preceded by housekeeping operations.

16-18. ENTRY AT GETPHASE

16-19. Progress Control is entered at Routine GETPHASE (figure 16-2) when an active (requesting) Mission Function needs to know the Phase it is currently in. (It also appears permissible to inquire about the Phase of some other Mission Function.) The Mission Function requests the execution of Routine GETPHASE and furnishes the mission function code via the calling sequence:

L	TC	GET PHASE
L+1	OCT	N
L+2		(Return address following a request to terminate)
L+3		(Normal return address)

where OCT N has the form 0000D₁ and digit D₁ specifies the mission function code.

16-20. Routine GETPHASE saves the address of OCT N (L+1) for later use and transfers program control to Routine PHASCOM2. Routine PHASCOM2 initially executes an INHINT to prevent any program interrupts while the phase tables are being interrogated. Then OCT N is stored to enable access to the proper cell in the phase tables. Next, return address L+2 is stored and program control is transferred to Routine PHASESUB via Routine REPHASE2. Routine PHASESUB copies and stores the current Phase associated with the Mission Function specified by D₁ and then transfers program control to Routine ONSKIP.

16-21. Routine ONSKIP tests the current Phase to determine if the Mission Function specified by D₁ has been terminated. Since it is impossible for a terminated Mission Function to request that its Phase be determined, program control is transferred to Routine GETPHASE at location GETPHASE+7 where the current Phase is tested to determine if there has been a request to terminate the Mission Function. If there has been such a request, the return address is L+2. If there has been no request to terminate the Mission Function the return address is incremented from L+2 to L+3. After it has been determined if program control is to be transferred to L+2 or L+3, the return address is formed, a RELINT is executed, and program control is transferred to the requesting Mission Function at the appropriate return address.

16-22. ENTRY AT MODROUT

16-23. Progress Control is entered at Routine MODROUT (figure 16-2) when the operator wishes to change the Phase of an active Mission Function or to initiate the execution of a currently inactive Mission Function. These operations are requested via the DSKY's by keying in VERB, 3, 7, (Change major mode to), ENTER, D₁, D₂, ENTER where digit D₁ specifies the Mission Function code and D₂ specifies the manual phase code. The keyed-in data is accepted and processed by program section Pinball (Issue 17, paragraphs 17-473 through 17-484) with program control ultimately being transferred to Routine MODROUT as part of Job CHARIN.

16-24. Routine MODROUT rearranges the keyed-in request from 000D₁D₂ to 00D₂0D₁ so that the mission function and phase codes are in a format compatible with the other Progress Control entries. Digit D₃ is zero in this case since only one digit is required to specify a manual Phase. Then an INHINT is executed to prevent any interrupts while the phase tables are being modified. Finally Routine MODROUT stores the complete address of Routine DOEJ to provide an ENDOFJOB exit for Job CHARIN and transfers program

control to Routine REPHASE2. (Recall that Routine MODROUT is executed under the auspices of Job CHARIN.) Routine REPHASE2 stores the return address of the routine that called Routine MODROUT and transfers program control to Routine PHASESUB. (Routine REPHASE2 is also entered from Routine GOJUMP, part of program section Fresh Start and Restart, as the result of a GOJAM situation. Refer to paragraph 16-114 for description of this operation.)

16-25. Routine PHASESUB copies and stores the old Phase and transfers program control to Routine PHASCH4 and subsequently to Routines SETPHASE and ONSKIP as described in paragraphs 16-15 and 16-16. In Routine ONSKIP, the old Phase is tested to determine if the Mission Function had been terminated when Routine SETPHASE was entered. If the Mission Function was not terminated, program control is transferred to Routine PHASOUT via Routine PHASREL as described in paragraph 16-17. If the Mission Function was terminated, program control is transferred to Routine DEMANDON, part of program section Fresh Start and Restart.

16-26. Routine DEMANDON updates (for subsequent display) the mission function activity status code to include the requested Mission Function in case this is a request to activate a previously inactive Mission Function. Program control is then returned to Routine PHASCH4 where a request is made to execute the requested Mission Function as a Job. Upon the completion of this request, program control is transferred to Routine STARTRET which executes a RELINT to enable program interrupts to be accepted and processed and transfers program control to Routine DSPMM. Routine DSPMM (paragraph 17-470), part of program section Pinball, causes a DSKY display of the newly updated Mission Function activity status code. This code, displayed as a two-digit octal number, indicates the active Mission Function(s). Program control is then returned to Progress Control at Routine PHASEOUT which provides the complete address of Routine DOEJ and transfers control to it. Routine DOEJ transfers program control to Routine ENDOFJOB (paragraph 12-47), thus terminating Job CHARIN which supervised the Change Major Mode activity.

16-27. ENTRY AT MAJEXIT

16-28. Progress Control is entered at Routine MAJEXIT (figure 16-2) either via a keyboard entry or directly from a Mission Function. The former occurs when the operator requests that a Mission Function terminate itself. Such a request (made by a previous CHARIN call to MODROUT) is discovered and honored during that execution of Routine PHASCHNG or GETPHASE following the keyboard action (refer to Routine MODROUT, paragraph 16-22). When the request is discovered, program control is returned to location L+2 of the

calling sequence for the requesting Mission Function. The latter occurs when a Mission Function has fulfilled its role and wishes to terminate itself (or when one Mission Function wishes to terminate another Mission Function). Both the manual and program entries result in the calling of Routine MAJEXIT by a Mission Function. The calling sequence is:

L	TC	MAJEXIT
L+1	OCT	N

where N has the form 0000D₁ and digit D₁ specifies the mission function code of the Mission Function to be terminated.

16-29. Routine MAJEXIT transfers program control to Routine PHASECOM and on to Routines PHASCOM2 and PHASESUB in a manner similar to that described in paragraph 16-12. Upon the completion of Routine PHASESUB, program control is transferred to Routine MAJEX3 which supplies octal 37 for insertion into the proper cell in the phase tables. Program control is then transferred to Routine SETPHASE which replaces the old Phase associated with the Mission Function to be terminated with the new Phase, which is 37 (terminated). When all three phase tables have been updated to reflect the termination request and deactivate the Mission Function, program control is transferred to Routine MAJEX4.

16-30. Routine MAJEX4 tests to determine which Mission Functions requested the termination; however, this feature is not exercised in the Sunrise programs because it is used in conjunction with entry PHASWAIT, which is not used in Sunrise. Program control is transferred to Routine DEMOFF, which updates the mission function activity function code to show that the Mission Function being terminated is no longer active. Program control is then transferred to Routine DSPMM, part of program section Pinball, which causes the updated mission function activity status code to be displayed on the DSKY's. This display indicates to the operator that the Mission Function is no longer active. Program control is returned to Progress Control at Routine DOEJ which terminates the Mission Function Job by transferring program control to Routine ENDOFJOB in program section Executive.

16-31. ENTRY AT PHASWAIT

16-32. This entry is intended to be used by a Mission Function when it wishes to make itself or another Mission Function inactive until some other Mission Function has been accomplished. This feature of Progress Control is not used in the Sunrise programs; however, it appears to be designed to operate in close conjunction with Routine MAJEXIT. Any Mission Function put to sleep in Routine PHASWAIT will be awakened during execution of Routine MAJEXIT providing OCTAL N, as contained in the calling sequence to both

routines, is the same number. Presently the operation of making a Mission Function dormant and awakening can be accomplished using Routines JOBSLEEP and JOBWAKE (paragraph 12-20), part of program section Executive. The probable calling sequence to Routine PHASWAIT will be:

L	TC	PHASWAIT
L+1	OCT	N

16-33. DETAILED DESCRIPTION

16-34. Progress Control performs five major operations in support of a Mission Function. These operations include:

- a. Changing the Phase of a Mission Function
- b. Determining the current Phase of a Mission Function
- c. Manually modifying a Mission Function (changing the Phase or requesting the execution of a Mission Function at a particular Phase.)
- d. Terminating a Mission Function
- e. Deactivating a Mission Function as a Job

The following paragraphs contain a detailed description of each of the five operations. Since some of the routines discussed herein are used during the execution of one or more of these operations, such routines are discussed in detail once and in subsequent discussions only those details peculiar to the operation at hand are described.

16-35. PHASE MODIFICATION

16-36. Phase modification is performed under the auspices of Routine PHASCHNG (paragraphs 16-12 through 16-17) in response to a request by a Mission Function to change its own Phase or that of another Mission Function. Since, in the Sunrise programs, one Mission Function is not permitted to change the Phase of another Mission Function, this discussion assumes that the requesting Mission Function is changing its own Phase. The requesting Mission Function supplies both the new phase code and the mission function code via OCT N where N has the form $0D_3D_20D_1$. Routine PHASCHNG (attachment 16-1) initially places the address of OCT N (L+1 of the calling sequence) in the accumulator and transfers program control to Routine PHASECOM. Routine PHASECOM stores this address in register OVCTR and transfers program control to Routine PHASCOM2.

16-37. Initially Routine PHASCOM2 executes an INHINT to prevent the processing of program interrupts while the contents of the phase tables are being

changed. Next, the return address to Routine PHASCHNG is stored in register ITEMP2 and the content of location L+1 (OCT N) is stored in register PHASDATA for later use in determining the phase cell corresponding to the requesting Mission Function. OCT N contains both the mission function code and the phase code: the mission function code is contained in bit positions 5 through 1; the phase code in bit positions 11 through 7. Then the complete address (CADR) of register L+2 in the calling sequence is formed and program control is returned to Routine PHASECOM at location PHASCOM+3 where the CADR of L+2 is stored in register MPAC+1. Program control is transferred to Routine PHASESUB.

16-38. Routine PHASESUB determines the Phase of the requesting (current) Mission Function and stores it in register PHASE for later use. Initially, the content of register PHASDATA (OCT N) is masked against octal 00037 to extract the mission function code from OCT N. The code is then stored in register PROG and multiplied by octal 12540 (approximately 0.334 decimal). The integer part of the product (the 14 high-order bits as contained in the accumulator) is stored in register PHASEWD. The integer specifies which phase word (or row) in the phase table contains the Phase of the requesting Mission Function. The integers -0, 1, 2, and 3 specify phase words 1, 2, 3, and 4 respectively. Next, the integer is multiplied by three and complemented; then the content of register PROG (the mission function code) is added to the result. This operation produces a number which specifies the 5-bit group (cell) in the phase word that contains the current Phase of the requesting Mission Function. The number can be -0, 1, or 2 and it specifies the low-five, middle-five or high-five bit positions of the phase word respectively. The number is stored in register PHASEDIG and is also used for indexing register PHASMASK to obtain an octal number to mask out the unwanted bits in the phase word. The indexing and masking quantities are as follows:

<u>Indexing Quantity</u>	<u>Masking Quantity</u>
-0	c(PHASMASK) = 00037
1	c(PHASMASK+1) = 01740
2	c(PHASMASK+2) = 76000

Thus only the five-bit group representing the current Phase remains intact in the accumulator.

16-39. The current Phase is then stored in register PHASE. In addition, the Phase, as contained in the accumulator, is shifted to the low-five bit positions by indexing instruction TC+1 with the contents of PHASEDIG (-0, 1 or 2). If the content of PHASEDIG is -0, the Phase is already in the low-five bit positions and therefore needs no shifting. Program control is then

transferred to Routine PHASCH3 via instruction TC ITEMP2 where the content of ITEMP2 is the address (PHASCHNG+2) of instruction TC PHASCH3. If the c(PHASEDIG) is 1 or 2, the Phase is in the middle-five or high-five bit positions and must be shifted ten or five bit positions to the left respectively. Following the shifting operation, the content of the accumulator is masked to extract the low-five bit positions which now contain the current Phase. The Phase is then stored in register PHASE and program control is transferred to Routine PHASCH3 as previously described.

16-40. Routine PHASCH3 tests the content of register PHASE to determine if there has been a request to terminate the active Mission Function. If the content of PHASE is greater than plus zero, no termination request has been made and program control is transferred to Routine INCET4. In Routine INCET4, the CADR of L+2 of the calling sequence, contained in MPAC+1, is incremented to L+3 which is the normal return address. Program control is then transferred to Routine PHASCH4. If the content of PHASE is equal to plus zero, there has been a request to terminate the Mission Function. Routine INCET4 is bypassed and program control is transferred directly to Routine PHASCH4.

16-41. Routine PHASCH4 extracts the new phase code from OCT N and places this code in the accumulator. The octal quantity 400 is multiplied by the contents of register PHASDATA which results in placing the new phase code in bit position 5 through 1 of the accumulator and the Mission Function code in bit positions 13 through 9 of register LP. (Register PHASDATA contains the mission function code in bit positions 5 through 1 and the new phase code in bit positions 11 through 7.) Program control is then transferred to Routine SETPHASE with the contents of the accumulator intact.

16-42. Routine SETPHASE, as the name implies, sets the new Phase into the proper cell in the phase tables in place of the old Phase. Initially, Routine SETPHASE stores the new phase code in register OVCTR and places this code in register LP. Register LP is treated as having three 5-bit groups and the new phase code is placed into one of these groups dependent on the content of register PHASEDIG (-0, 1, or 2). Since there are only 14 value bits associated with register LP, the most significant bit of the high-five bits is placed into bit position one of register A. This is accomplished by indexing register PHASESHL with the content of register PHASEDIG and then multiplying the content of the appropriate PHASESHL register by the new phase code contained in register OVCTR. The indexing quantity, the contents of the appropriate PHASESHL register, and the result of the multiplication are shown in table 16-2. Thus, the new phase code is placed into the correct five-bit group of register LP in preparation for insertion into the phase tables.

TABLE 16-2
NEW PHASE CODE POSITIONING

Indexing Quantity c(PHASEDIG)	c(PHASESHL+n)	Result
-0	00001	Phase code in bit positions 5 through 1 of register LP.
+1	00040	Phase code in bit positions 10 through 6 of register LP.
+2	02000	Phase code in bit position 1 of register A and 14 through 11 of register LP.

16-43. The content of the accumulator (00000 or 00001) is stored in register OVCTR until later when it is recombined with the contents of register LP to update the phase tables. Register PHASMASK is then indexed with the contents of register PHASEDIG and the complemented result is placed in the accumulator to mask out that cell corresponding with the requesting Mission Function. Table 16-3 lists the masking quantities contained in the accumulator and the cell in the phase word affected.

TABLE 16-3
MASKING QUANTITIES

c(PHASEDIG)	c(PHASMASK+n)	c(A)	Phase Cell Affected
-0	00037	177740	Right (low-five bits)
+1	01740	176037	Middle (middle-five bits)
+2	76000	101777	Left (high-five bits)

16-44. The phase word that contains the cell to be masked is selected next. This is accomplished by indexing register PHASETAB with the contents of register PHASEWD (0, 1, 2, or 3) which corresponds to the first, second,

third, or fourth word of each phase table. Then the phase word is masked against the contents of the accumulator, which contains zeros in those bit positions that form the cell identified with the requesting Mission Function. After the masking has been accomplished, the accumulator contains the selected phase word with the desired phase cell set to zero. The content of register LP, which contains the new phase code in the proper five-bit group, is added to the accumulator. Then the accumulator contains the phase word including the new phase code. However, if the new phase code is contained in the high-five bits, the most significant bit is missing since this bit is contained in register OVCTR (paragraphs 16-42 and 16-43). This bit is entered into the accumulator by indexing register B15FIX with the content of register OVCTR and adding the content of the appropriate B15FIX register to the accumulator as follows:

<u>c(OVCTR)</u>	<u>Quantity Added</u>	<u>Result</u>
0	c(B15FIX) = 00000	No change to the contents of the accumulator
1	c(B15FIX+1) = 40000	A ONE added to bit 15 of the accumulator

16-45. The updated phase word is next stored in the proper row in each of the three phase tables. Again the content of register PHASEWD is used for indexing to find the desired row and the content of the accumulator is stored in registers (phase tables) PHASETAB, BACKPHAS, and PHASEBAR. (The content of the accumulator must be complemented prior to storing it in register PHASEBAR.) Thus, the phase tables have been updated to reflect the new Phase. Program control is then transferred to Routine PHASCH4 at location PHASCH4+4 where the old phase code is stored in register MPAC prior to transferring program control to Routine ONSKIP.

16-46. Initially, Routine ONSKIP places the complement of the octal quantity 00037 (77740) into the accumulator. Then the content of register PHASE, the old phase code, is added to the accumulator and the sum is tested. If the sum is minus zero, then the old phase code is 37, signifying that the Mission Function was terminated. However, since it is impossible for a terminated Mission Function to request a phase change, the sum must be less than minus zero when the entry point is Routine PHASCHNG. Therefore, program control is transferred to location ONSKIP+5 where instruction TC Q is indexed by the contents of register Q. Since the address of register Q is 1 and c(Q) is PHASCH4+8, program control is transferred to location PHASCH4+9 which contains instruction TC PHASEREL.

16-47. Routine PHASEREL executes instruction RELINT (to allow program interrupts to be accepted and processed) and then transfers program control to Routine PHASEOUT. There the CADR (L+2 or L+3 of the calling sequence as contained in register MPAC+1) is loaded into the accumulator and program control is transferred to Routine BANKJUMP (paragraph 19-16), part of program section Interbank Communication.

16-48. PHASE DETERMINATION

16-49. Phase table interrogation is performed under the auspices of Routine GETPHASE (paragraphs 16-18 through 16-21) in response to a request by a Mission Function to determine its own Phase or the Phase of another Mission Function. The requesting Mission Function supplies the mission function code via OCT N where N has the form 0000D₁. Routine GETPHASE (attachment 16-1) initially stores the address of OCT N (L+1 of the calling sequence) in register OVCTR and transfers program control to Routine PHASCOM2. Routine PHASCOM2 stores the mission function code (not the phase code in this case) in register PHASDATA as described in paragraph 16-37; however, program control is returned to Routine GETPHASE at location GETPHASE+3 rather than to Routine PHASCHNG at location PHASCHNG+3. Upon returning to Routine GETPHASE, the complete address of register L+2 in the calling sequence is stored in register EXECTEM5. Program control is then transferred to Routine REPHASE2 which stores the return address to Routine GETPHASE in register ITEMP2 and transfers program control to Routine PHASESUB.

16-50. Routine PHASESUB determines the Phase of the requesting Mission Function as described in paragraphs 16-38 and 16-39 except that register ITEMP2 contains the address of location GETPHASE+5. Therefore, upon exiting from the routine, program control is transferred to location GETPHASE+5 which contains instruction TC ONSKIP. Thus program control passes on to Routine ONSKIP.

16-51. In Routine ONSKIP, the current Phase is tested to determine if the requesting Mission Function has been terminated (phase code equals 37). Since it is impossible for a terminated Mission Function to request that its Phase be determined, the phase code must be less than 37; therefore, program control is transferred to location ONSKIP+5 where the return address to Routine GETPHASE is determined as described in paragraph 16-46. In this case, the return address is GETPHASE+7 and program control is transferred to it.

16-52. At location GETPHASE+7, the content of register PHASE is tested to determine if there has been a request to terminate the active Mission Function. If a termination has been requested, the content of register

EXECTEM5 (the CADR of L+2 from the calling sequence) is added to the accumulator. Since the accumulator contains 00000 at this time, the c(A) equals the CADR of L+2 following the addition. If there has been no termination request, the octal quantity 00001 is entered into the accumulator and the content of register EXECTEM5 is added to this quantity to produce the CADR of L+3, the normal return address.

16-53. Bits 15 through 11 of the newly formed return address (L+2 or L+3) are stored in register BNK and then these same five bits are deleted from the accumulator by masking the c(A) against the octal quantity 01777. The remaining ten bits form the address (L+2 or L+3) - 6000 since bits 12 and 11 were lost by the masking operation. This address is stored in register Q temporarily to be used later in providing the return address. The current phase code is then taken from register PHASE and placed in the accumulator where it is available to the requesting Mission Function when program control is returned to it. A RELINT is executed to allow program interrupts to be processed and program control is transferred to location L+2 and L+3 in the calling sequence depending on whether or not there has been a request to terminate the Mission Function.

16-54. MANUAL MISSION FUNCTION MODIFICATION

16-55. Manual control of a Mission Function is accomplished under the auspices of Routine MODROUT (paragraphs 16-22 through 16-26) in response to a keyboard request to change the Phase of an active Mission Function or to initiate the execution of an additional Mission Function. The mission function and phase codes are keyed in as part of the request and have the form 000D₁D₂ where D₁ specifies the mission function code and D₂ the phase code. Program section Pinball (Job CHARIN) processes the keyed in data and program control is ultimately transferred to Progress Control at Routine MODROUT.

16-56. Routine MODROUT (attachment 16-1) changes the form of the keyed in request to 00D₂0D₁ to make the request compatible with subsequent routines in Progress Control. Initially the content of the accumulator (000D₁D₂) is stored in register SR which causes the data to be shifted one bit position to the right such that the low order bit of the phase code is lost. The c(A) is then multiplied by the octal quantity 00100 which causes the mission function code to be placed in bit positions 12 through 10 and the phase code to be placed in bit positions 9 through 7 of register LP. Thus the c(A) is 00000 and c(LP) is 0D₁D₂00. Next the content of register SR is shifted two bit positions to the right which moves the mission function code into bit positions 3 through 1. The phase code is lost and the c(SR) is 0000D₁.

16-57. The content of register LP is then masked by the octal quantity 700 which deletes D_1 , the mission function code, from the register leaving the content equal to $00D_200$. The content of register SR($0000D_1$) is added to the c(LP) to place $00D_20D_1$ in the accumulator which is the format necessary for processing by Progress Control. An INHINT is executed to prevent any program interrupts and the contents of the accumulator is stored in register PHASDATA. Finally, the complete address of Routine DOEJ is stored in register MPAC+1 and program control is transferred to Routine REPHASE2. (The only instruction in Routine DOEJ is TC ENDOFJOB).

16-58. Routine REPHASE2 stores the return address MODROUT+13 in register ITEMP2 and transfers program control to Routine PHASESUB. Routine PHASESUB determines the Phase of the requesting Mission Function, as described in paragraphs 16-38 and 16-39, and returns program control to Routine MODROUT at location MODROUT+3. This location contains instruction TC PHASCH4 which causes program control to be transferred to Routine PHASCH4.

16-59. Routine PHASCH4 isolates the new phase code, places it in the accumulator, and then transfers program control to Routine SETPHASE which updates the three phase tables as described in paragraphs 16-41 through 16-45. Program control is returned to Routine PHASCH4 at location PHASCH4+4 where the old phase code is stored in register MPAC prior to transferring program control to Routine ONSKIP.

16-60. Routine ONSKIP determines whether or not the requesting Mission Function has been terminated. If the old phase code is not octal 37, the Mission Function has not been terminated and program control is transferred to location ONSKIP+5 where the return address to Routine PHASCH4 is formed. Program control is transferred to Routine PHASREL via location PHASCH4+9 which contains instruction TC PHASREL. Routine PHASREL executes a RELINT to allow program interrupts to be processed and then transfers program control to Routine PHASOUT.

16-61. If the Mission Function has been terminated, program control is transferred to Routine DEMANDON via Routine PHASCH4. Routine DEMANDON, part of program section Fresh Start and Restart, updates the content of register MODREG to indicate that the requesting Mission Function is active. This indication is later displayed on the DSKY's under the control of Routine DSPMM. (Refer to paragraphs 16-117 and 16-118 for further details on Routine DEMANDON). Returning to Routine PHASCH4 at location PHASCH4+11, the accumulator is loaded with the complete address of the Mission Function to be executed depending upon the contents of register

PROG. (Register PROG was set previously in Routine PHASESUB with the keyed in Mission Function code.) The relationship is as follows:

<u>c(PROG)</u>	<u>c(A)</u>	<u>Requested Mission Function</u>
0	CADR STARTPL	Prelaunch Alignment
1	CADR STARTFF	Orbital Integration

16-62. Program control is transferred to the requested Mission Function via program section Interbank Communication. The Mission Function requests its own execution as a Job via the Executive according to priority. Then Progress Control is reentered at Routine STARTRET which executes a RELINT (to allow program interrupts to be processed) and transfers program control to Routine DSPMM, part of program section Pinball. Routine DSPMM (paragraph 17-470) displays on the DSKY's the mission function code to indicate which Mission Function(s) is (are) active. (This code was previously set into register MODREG by Routine DEMANDON). The code having been displayed, program control returns to Routine STARTRET at location STARTRET+3 and is immediately passed on to Routine PHASOUT.

16-63. Routine PHASOUT is entered from either Routine STARTRET (Mission Function terminated) or Routine PHASREL (Mission Function not terminated). This routine loads the CADR of Routine DOEJ, as contained in register MPAC+1, into the accumulator and transfers program control to Routine DOEJ via interbank Communication. Routine DOEJ has only one instruction, TC ENDOFJOB, which causes program control to be transferred to Routine ENDOFJOB, part of the Executive. Routine ENDOFJOB removes the active Job (Job CHARIN) from its Job Area and searches for the Job having the highest priority as described in paragraphs 12-47 through 12-52.

16-64. MISSION FUNCTION TERMINATION

16-65. Mission Function termination is performed under the auspices of Routine MAJEXIT (paragraphs 16-27 through 16-30) when a Mission Function detects a request to terminate itself or when a Mission Function wishes to terminate itself (or another Mission Function). The former is manually initiated at the DSKY's and is detected during the next execution of Routine PHASCHNG or GETPHASE. The latter is a direct entry into Progress Control at Routine MAJEXIT from a Mission Function. The calling sequence for all entries is the same and provides for a return to L+2 of the original

calling sequence when a termination request is detected. The instructions at location L+2 for the two current Mission Functions are:

<u>Prelaunch Alignment</u>		<u>Orbital Integration</u>	
L+2	TC PRELEXIT	L+2	TC FFEXIT
PRELEXIT	CS OCT 0	FFEXIT	TC MAJEXIT
	TS CDUIND		OCT 00001
	TC MAJEXIT		
	OCT 00000		

where OCT N (00000 for Prelaunch Alignment and 00001 for Orbital Integration) specifies the Mission Function to be terminated. Except for making the CDU's available in Prelaunch Alignment, both sets of instructions are the same and cause program control to be transferred to Routine MAJEXIT.

16-66. Routine MAJEXIT (attachment 16-1) places the address of OCT N (L+1, where L now refers to the location from which the TC MAJEXIT is executed) into the accumulator and transfers program control to Routine PHASECOM. Routine PHASECOM stores this address in register OVCTR and in turn transfers program control to Routine PHASCOM2. As described in paragraph 16-37, Routine PHASCOM2 executes and INHINT, stores OCT N in register PHASDATA, and forms the complete address (CADR) of location L+2. (The CADR of location L+2 has no significance in Routine MAJEXIT because program control can never return to L+2 since an entry at MAJEXIT is terminated by a TC ENDOFJOB). Program control is returned to Routine PHASCOM which stores the CADR of L+2 in register MPAC+1 and transfers program control to Routine PHASESUB.

16-67. Routine PHASESUB stores the mission function code in register PROG, determines which cell in the phase table contains the phase code of the requesting Mission Function, and stores the phase code in register PHASE. (Refer to paragraphs 16-38 and 16-39 for the details of these operations.) Program control is then returned to Routine MAJEXIT at location MAJEXIT+2 by instruction TC ITEMP2. (Register ITEMP2 was loaded with the address of location MAJEXIT+2 during the execution of Routine PHASCOM2.) At location MAJEXIT+2, program control is transferred to Routine MAJEX3 which enters the octal quantity 37 into the accumulator so that it may be placed in

the cell associated with the requested Mission Function specified by OCT N. Program control is then transferred to Routine SETPHASE.

16-68. Routine SETPHASE replaces the old phase code with the new phase code (octal 37 as contained in the accumulator) and updates the three phase tables to reflect that the active Mission Function has been terminated. (Refer to paragraphs 16-42 through 16-45 for the details of these operations.) Program control is then transferred to Routine MAJEX4 via location MAJEX3+2 where the octal quantity 00001 is entered into the accumulator.

16-69. Initially, Routine MAJEX4 stores the content of the accumulator in register OVCTR and enters the complemented content of register PROG (77777 for Prelaunch Alignment and 77776 for Orbital Integration) into the accumulator. Then register PWTPROG is indexed with the content of register OVCTR (initially octal 00001) and the content of register PWTPROG+1 is added to the content of the accumulator. (The content of registers PWTPROG and PWTPROG+1 was previously set to 00037 by program section Fresh Start and Restart during a fresh start; refer to paragraph 16-101). The resulting quantity contained in the accumulator is tested, and if it is greater than plus zero or less than minus zero program control is transferred to Routine MAJEX5. (The other alternative, the c(A) equal to minus zero, does not occur since the only other routine that can load the PWTPROG registers is Routine PHASWT4 which is never executed by the Sunrise programs.)

16-70. Program control is transferred twice to Routine MAJEX5. The first time the content of register OVCTR is 00001 and program control is transferred back to Routine MAJEX4 with the c(OVCTR) decremented by one. The second time the c(OVCTR) is 00000 and program control is transferred to Routine DEMOFF. In this way, the two existing Mission Functions are tested to determine which one requested the termination.

16-71. Routine DEMOFF places the complemented content of register PROG (77777 or 77776) into the accumulator. Then register BIT1 is indexed with the content of the accumulator and the complemented content of register BIT1 (77776) or register BIT1-1 (77775) is entered into the accumulator. The relationship is as follows:

<u>c(PROG)</u>	<u>c[BIT1 - c(PROG)]</u>	<u>c(A)</u>
0	00001	77776
1	00002	77775

The contents of register MODREG is masked against the content of register A such that the content of bit position 1 (Prelaunch Alignment) or bit position 2 (Orbital Integration) is set to zero. The result is then stored in register MODREG to be displayed later. Thus the content of MODREG indicates that the terminated Mission Function is no longer active. The content of bits 6 through 1 of register MODREG specifies which Mission Function(s) is active and controls the PROGRAM displays on the DSKY's. (Refer to table 16-4 for the relationship between the active Mission Function(s), the c(MODREG) and the PROGRAM displays.)

16-72. After the content of register MODREG has been updated, a RELINT is executed to allow program interrupts to be accepted. Program control is then transferred to Routine DSPMM (paragraph 17-470) via program section Interbank Communication to display the updated contents of register MODREG. Thus, it is displayed that the requesting Mission Function has been terminated. Progress Control is reentered at Routine DOEJ which transfers program control to Routine ENDOFJOB in the Executive to remove the Mission Function from the status of an active Job. The Executive then searches for the Job having the highest priority as described in paragraphs 12-47 through 12-51.

TABLE 16-4
PROGRAM DISPLAYS

Active Mission Function	c(MODREG)	Program Display
None	00000	00
Prelaunch alignment	00001	01
Orbital Integration	00010	02
Prelaunch and Orbital	00011	03
System Test	00100	04
Prelaunch and System Test	00101 (1)	05
Orbital and System Test	00110	06
Prelaunch, Orbital and System Test	00111 (1)	07

- (1) If System Test is active prior to the request for Prelaunch Alignment, the two Mission Functions cannot be active simultaneously due to an interlocking scheme which prevents this occurrence. However, if Prelaunch Alignment is active prior to System Test, the two can be active simultaneously.

16-73. MISSION FUNCTION DEACTIVATION

16-74. Mission Function deactivation is performed under the auspices of Routine PHASWAIT (paragraph 16-31 and 16-32) when a Mission Function wishes to make itself or another Mission Function inactive. A Mission Function put to sleep by Routine PHASWAIT will be awakened by Routine MAJEXIT. This operation is not utilized in the Sunrise programs; however, a brief description of Routine PHASWAIT is included herein in anticipation of its subsequent use.

16-75. Many of the operations performed under the auspices of Routine PHASWAIT (attachment 16-1) are similar to those operations performed for Routine MAJEXIT. These similarities occur in Routines PHASECOM, PHASCOM2 and PHASESUB; therefore, the operation of these routines is not repeated, only the dissimilarities are pointed out. (Refer to paragraphs 16-66 and 16-67 for a description of the operations performed by these routines.) During the execution of Routine PHASCOM2, register ITEMP2 is loaded with the return address to PHASWAIT and this address is later used in Routine PHASESUB to return program control to Routine PHASWAIT at location PHASWAIT+2. At this location, program control is transferred to Routine PHASWT3 which immediately passes control on to Routine ONSKIP.

16-76. Routine ONSKIP tests the current Phase to determine whether or not the Mission Function, as specified by OCT N of the calling sequence, has been terminated. If the Phase is 37, indicating termination, program control is transferred to Routine PHASOUT2 which restores the contents of register LP and transfers program control to Routine PHASEREL and on to Routine PHASEOUT which returns program control to L+2 of the requesting Mission Function. If the Phase is less than 37, program control is transferred to Routine SETPHASE with the contents of the accumulator equal to 00000.

16-77. In Routine SETPHASE, the phase code 0 is entered into the cell associated with the requesting Mission Function in each of the three phase tables. Thus, a request to terminate the requesting Mission Function is entered into the phase tables. (Refer to paragraphs 16-42 through 16-45 for a detailed discussion of Routine SETPHASE.) Upon completing the execution of Routine SETPHASE, program control is transferred to Routine PHASWT5 with the contents of the accumulator equal to 00001.

16-78. Routine PHASWT5 initially stores the content of the accumulator in register OVCTR. Then the c(A) is used to index register PWTCADR and the content of register PWTCADR+1 is tested to determine if it is available. If the c(PWTCADR+1) is less than minus zero or greater than plus zero, the contents of register OVCTR is tested to determine if there is a PWTCADR

register available. If there is no register available, c(OVCTR) equal to zero, program control is transferred to Routine ABORT (paragraph 19-41), part of program section Alarm and Display Processor, to display the alarm code associated with the failure. If there is a PWTCADR register available, c(OVCTR) less than zero, Routine PHASWT5 is repeated. On this second pass through the routine, register PWTCADR is tested to determine if it is available since the content of register OVCTR has been decremented.

16-79. If the content of register PWTCADR or PWTCADR+1 equals zero (indicating availability), program control is transferred to Routine PHASWT4. Routine PHASWT4 stores the mission function code in register PWTPROG or PWTPROG+1 and the return address of the requesting Mission Function in register PWTCADR or PWTCADR+1. (The PWTPROG and PWTCADR registers used are determined by the content of register OVCTR.) Program control is then transferred to Routine JOBSLEEP (paragraph 12-53), part of program section Executive, to deactivate the Mission Function.

16.80 FRESH START RESTART

16-81. Program section Fresh Start and Restart, as the name implies, is executed whenever the AGC needs a complete fresh start or a restart as the result of some abnormal operational conditions. A fresh start is executed when the computer is initially turned on, when repeated alarms occur, or when there is disagreement among the phase tables following a GO sequence (resulting from a hardware failure or an abort condition). A restart is executed when the phase tables agree following a GO sequence. Program control is returned to the beginning of that Phase being executed when the GO sequence occurred. Both a fresh start and a restart results in program control being transferred to the dummy Job and remaining there until a Job of higher priority is detected.

16-82. FUNCTIONAL DESCRIPTION

16-83. A fresh start is manually initiated by an operator keying in VERB, 3, 6, ENTER. (When the computer is initially turned on, the fresh start is executed automatically.) This keyboard entry causes the current routine to be interrupted and results in program control being transferred to Routine SLAP1 of program section Fresh Start and Restart. (Refer to paragraphs 17-464 through 17-469 for a description of the operation.) Briefly, when the keyboard entry is made, program control is transferred to program section Pinball which digests the keyboard entry under the auspices of Job CHARIN. Program control is then transferred to Routine SLAP1 via Routine SLAPB of program section Executive.

16-84. Routine SLAP1 (figure 16-3) executes an INHINT to prevent the processing of program interrupts; clears register FAILREG, blanks display registers R1, R2, and R3 on the DSKY's; and transfers program control to Routine STARTSUB. During the execution of Routine STARTSUB, many registers in Erasable Memory are initialized to condition the AGC to start anew. This initialization results in the following major operations:

- a. The dummy Task is entered into all six Waitlist registers.
- b. The dummy Job is made the highest priority Job.
- c. All Work Areas are made available.
- d. All display relays are deactivated which blanks all DSKY displays.

Appendix A to this issue contains a list of all the registers that are initialized and the significance of each. Following this initialization, program control is transferred to Routine READIN0 (paragraph 14-142), part of program section Mode Switching and Mark. This routine saves the latest Mark and key code information and transfers program control to Routine NOGO.

16-85. Routine NOGO resets all three phase tables to indicate that all Mission Functions have been terminated. Since all the phase tables are reset to the same value, they now agree. Next the self check function of program section AGC Self-Check is disabled. Finally, Routine NOGO causes those DSKY relays associated with equipment external to the AGC Subsystem to be deenergized. Program control is then transferred to Routine ENDFRESH.

16-86. Routine ENDFRESH executes a RELINT to allow the processing of program interrupts and then transfers program control to Routine DSPMM, part of program section Pinball. Routine DSPMM (paragraph 17-470) causes the display of the mission function activity status code on the PROGRAM indicators on the DSKY's. Since all the Mission Functions were terminated by Routine STARTSUB, the PROGRAM indicators will display octal 00. Program control is returned to Routine ENDFRESH and is then transferred to Routine DUMMYJOB, part of program section AGC Self-Check.

16-87. Routine DUMMYJOB keeps the computer idling since there are no real Jobs awaiting execution. The program continues operating in a loop until an operator keys in a request for the execution of a Mission Function via the change major mode (Verb 37) entry.

16-88. A restart is initiated involuntarily as the result of a GO sequence caused by either a hardware failure or a program abort. These abnormal conditions cause program control to arrive at the restart entry of Fresh Start and Restart via hardware action. Briefly, the failure or abort (an abort causes a TC trap condition, refer to paragraph 19-41) is detected by the Alarm Processor (paragraph 9-29) which generates alarm signal ALGA when the abnormal condition is detected. The alarm signal causes the Timer to generate signal GOJAM (paragraph 5-13) which causes the execution of involuntary instruction GO (paragraph 5-56). This action results in program control arriving at the Interrupt Transfer Routines which execute an INHINT and transfer program control to the restart entry of Fresh Start and Restart (figure 16-3).

16-89. Program control arrives at Routine STARTSUB which initializes certain registers in Erasable Memory and transfers program control to Routine READIN0 as described in paragraph 16-84. After the latest Mark and keycode information has been saved by Routine READIN0, program

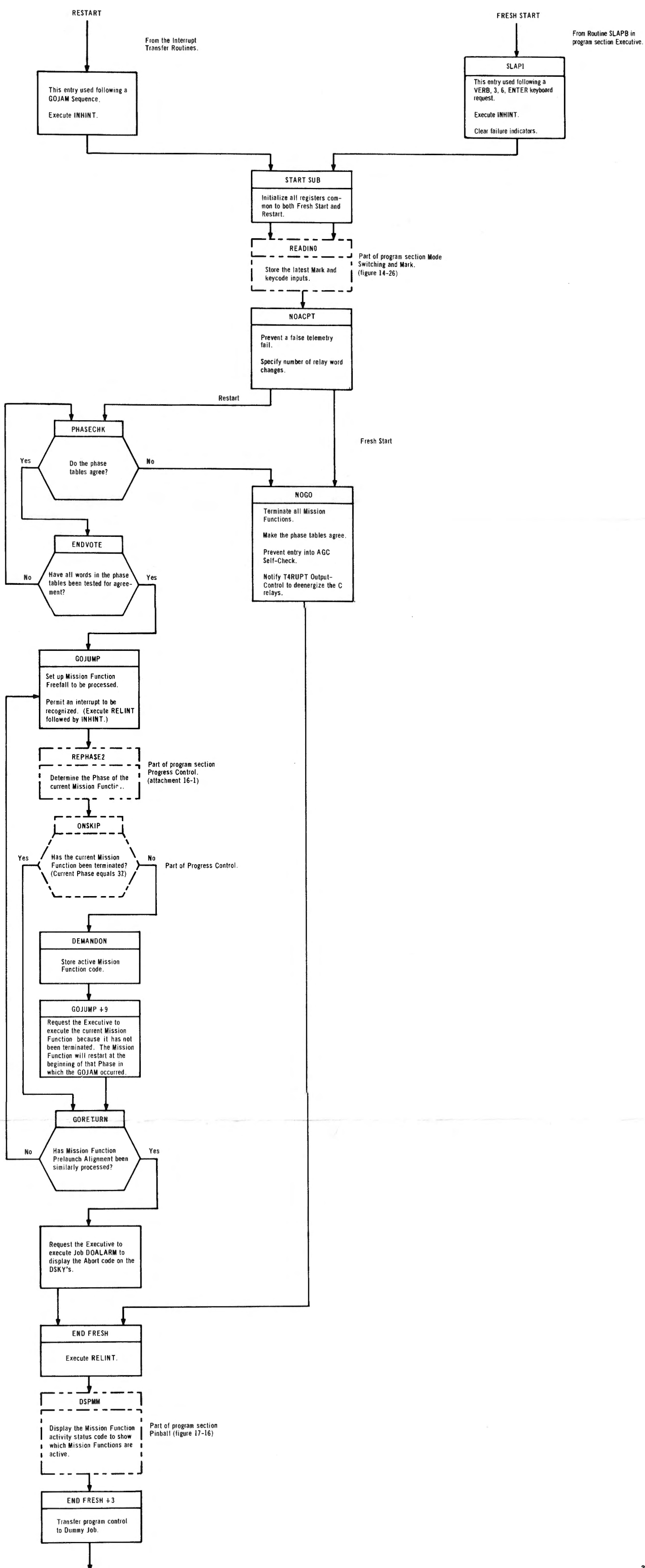


Figure 16-3. Fresh Start and Restart Functional Flowchart

control is transferred to Routine PHASECHK. This routine tests the content of the three phase tables for agreement. If the content of the tables do not agree, program control is transferred to Routine NOGO and the remainder of the restart operation is similar to a fresh start as described in paragraphs 16-85 through 16-87. If the content of the tables do agree, program control is transferred to Routine ENDVOTE.

16-90. Routine ENDVOTE determines if all four registers in the phase tables have been tested for agreement. If not all registers have been tested, program control is transferred back to Routine PHASECHK; otherwise, control is transferred to Routine GOJUMP. Thus each of the four words in a phase table is tested against the same word in the other two phase tables.

16-91. Initially, Routine GOJUMP designates Mission Function Orbital Integration to be processed during the first pass through this routine and subsequent routines (REPHAS2, ONSKIP, DEMANDON and GORETURN). Then a RELINT is executed followed immediately by an INHINT to allow program interrupts to be recognized. This prevents a hardware failure from occurring as the result of preventing interrupts for too long a time. Program control is transferred to Routine REPHASE2, part of program section Progress Control. Routine REPHASE2, in conjunction with Routine PHASESUB also of Progress Control, determines and stores the Phase of the current Mission Function (in this case Orbital Integration). Program control is then transferred to Routine ONSKIP, also part of Progress Control.

16-92. Routine ONSKIP determines whether or not the current Mission Function has been terminated. If it has been terminated, program control is transferred to Routine GORETURN; if it has not been terminated, program control is transferred to Routine DEMANDON.

16-93. Routine DEMANDON updates the mission function code to indicate that the current Mission Function has not been terminated. Program control is then returned to Routine GOJUMP at location GOJUMP+9, where the Executive is requested to execute the current Mission Function since it has not been terminated. The Mission Function will be initiated at the beginning of that Phase in which the GO sequence (GOJAM) occurred. Finally Routine DEMANDON transfers program control to Routine GORETURN.

16-94. There are two entries into Routine GORETURN: one from Routine ONSKIP when the current Mission Function has been terminated; the other from Routine GOJUMP when the current Mission Function has not been terminated. Routine GORETURN determines if both Mission Functions have been processed by the restart function of Fresh Start and Restart. (Orbital Integration is processed on the first pass, Prelaunch Alignment on the second pass.)

On the first pass, Routine GORETURN finds that Prelaunch Alignment has not been processed and program control is returned to Routine GOJUMP. On the second pass, Routine GORETURN finds that both Mission Functions have been processed and requests the Executive to execute Job DOALARM to display the failure or abort code on the DSKY's. Program control is then transferred to Routines ENDFRESH and DSPMM to execute a RELINT, to display the current Mission Function code on the DSKY's, and to transfer program control to Routine DUMMYJOB as described in paragraph 16-86.

16-95. Routine DUMMYJOB, as a matter of course, searches for the Job having the highest priority which, at this time, is Job DOALARM. Therefore, Job DOALARM is executed to display the code of the failure or abort on the DSKY's as described in paragraph 19-46. Program control is then returned to Routine DUMMYJOB to again search for the highest priority Job which is now the current Mission Function. Program control is transferred to this Mission Function at the beginning of the Phase it was in when the failure or abort occurred.

16-96. DETAILED DESCRIPTION

16-97. Program section Fresh Start and Restart performs two major functions as the result of abnormal operating conditions within the AGC. These functions are the initiation of either a fresh start or a restart depending upon the nature and severity of the abnormal condition. The following paragraphs contain a detailed description of both the fresh start and the restart functions. Since some routines are common to both fresh start and restart, such routines are discussed in detail once and in subsequent discussions only those details peculiar to the operation at hand are discussed.

16-98. FRESH START

16-99. The keyboard entry VERB, 3, 6, ENTER initiates the fresh start function of program section Fresh Start and Restart. This entry interrupts the current routine via Routine KEYRUPT (paragraph 15-164 and 17-6) which requests the Executive to initiate the execution of program section Pinball. Pinball accepts and processes the keyboard entry; displays the keyed in data on the DSKY's via T4RUPT Output Control; and transfers program control to Routine SLAPB in the Executive which passes program control on to Routine SLAP1 in Fresh Start and Restart.

16-100. Routine SLAP1 (attachment 16-2) initially executes an INHINT to prevent program interrupts while various registers in Erasable Memory are being initialized. The content of register FAILREG is set to zero to indicate no failures have previously occurred and program control is then transferred to Routine STARTSUB with the return address to Routine NOGO in register Q.

16-101. Routine STARTSUB stores the content of register Q in register BUF to provide for a return to Routine NOGO. Then Routine STARTSUB, which includes Subroutine DSPOFF, performs a series of operations to condition the AGC for a restart. These operations include:

- a. Setting the TIME3 and TIME4 counters to overflow as soon as possible thus preventing either a T3RUPT or a T4RUPT from occurring while the computer is being initialized. (Refer to paragraphs 15-159 and 15-161 for the significance of these interrupts.)
- b. Initializing the timelist of program section Waitlist to enter the timing for the dummy Task into each of the timelist registers (refer to paragraphs 12-72 through 12-74).
- c. Setting the address of the dummy Task into each of the address-list registers in the Waitlist. (Refer to paragraphs 12-72 through 12-77.)
- d. Setting the PRIORITY registers in Executive Job Areas 2 through 8 to minus zero which specifies that all Job Areas are available except Job Area 1. (Refer to locations 0114 through 0123 of Appendix B to Issue 15 and paragraphs 12-13 through 12-17.)
- e. Setting the content of register DSRUPTSW, CDUIND and OPTIND to minus zero to schedule the execution of T4RUPT Output Control in 40 msec and to make the IMU and Optics available to be driven by the AGC.
- f. Making all vector accumulator (VAC) areas within the Work Areas available for use.
- g. Placing the priority number of the dummy Job, which is the lowest priority Job, (1000) into register PRIORITY of Job Area 1.
- h. Resetting bit 7 of register OUT1 to clear an ERRUPT. (ERRUPTS are not used in the Sunrise programs.)
- i. Deenergizing all DSKY display relays to blank all the DSKY displays.
- j. Setting the content of 32 registers in Erasable Memory to zero. (Refer to Appendix A to this issue for a list of these registers and the significance of each.)
- k. Setting the content of registers PWTPROG and PWTPROG+1 to octal 37 which indicates to Routine MAJEX4 of program section Progress Control that neither Mission Function was put to sleep by Routine PHASWAIT (paragraphs 16-64 and 16-73).

1. Storing the IMU and Optic mode bits in registers WASKSET and WASOPSET respectively. In the process, old mode data is filtered out and new mode data is entered into register IN3.

Program control is then transferred to Routine READIN0 (paragraph 14-142) which protects the content of bit positions 15 (Mark) and 5 through 1 (key code) of registers IN0 and IN0WORD. This operation prevents the cause of a KEYRUPT from being lost when the content of register IN0 is read out.

16-102. Program control is returned to Subroutine DSPOFF at location DSPOFF+49 where the content of the accumulator, which was loaded by Routine READIN0, is tested to determine if there is any Mark information present. If the content of the accumulator is equal to or greater than positive zero (no Mark information in bit 15), program control is transferred to Routine NOACPT. If the content of the accumulator is less than or equal to negative zero, the Zero Optics CDU bit (bit 12) is masked out of register WASOPSET to indicate the Optics mode bit is to be ignored. Following this operation, program control is transferred to Routine NOACPT.

16-103. Routine NOACPT sets the content of register TELCOUNT to octal 6 to prevent a false indication that the downlink transmission rate is too slow (paragraph 18-93). Program section T4RUPT Output Control tests the content of register TELCOUNT every 120 μ sec and, if the test reveals that the register contains octal 7, an alarm is generated to indicate the transmission rate is too slow. Routine NOACPT next stores the address of the first location in the telemetry list (location DNLST1) in register DNLSTADR. The content of register NOUT is set to octal 13 (decimal 11) to indicate the number of relay words that have been changed since the last check by T4RUPT Output Control (refer to paragraphs 18-14 through 18-21). Register NOUT is loaded with the decimal quantity 11 because all eleven DSPTAB registers (DSPTAB through DSPTAB+12 octal) were just set to zero by Routine STARTSUB to turn off all display relays on the DSKY's. Finally the octal quantity 00023 is entered into register DSPCOUNT to indicate that the first key to be depressed during the next keyboard entry should be the VERB key (paragraph 17-102 and table 17-8). Program control is transferred to register BUF which was loaded with the address of Routine NOGO at the beginning of Routine STARTSUB.

16-104. Routine NOGO enters the octal quantity 77777 into all four registers (words) of phase tables PHASETAB and BACKPHAS. Since each word contains three five-bit cells (paragraph 16-8), each cell in both tables is set to octal 37 which signifies that every Mission Function has been terminated (table 16-1). The four words in the third phase table, PHASEBAR, are loaded with the octal quantity 00000 to specify Mission Function termination

because this table contains the complement of the other two tables. All three tables are now set to indicate all Mission Functions have been terminated. The content of register SMODE is also set to 00000 to indicate the self-check function of program section AGC Self-Check is not needed at this time. Bit 15 of registers DSPTAB+11D, DSPTAB+12D and DSPTAB+13D is set to ONE to indicate to T4RUPT Output Control that a change has taken place in those DSKY relays associated with equipment external to the AGC Subsystem. (Refer to table 18-1, note 2.) Bit positions 11 through 1 of the three DSPTAB registers contain all zeros; therefore, when T4RUPT Output Control transfers the content of these registers to registers OUT0 and DSPBUF, the associated relays will be deenergized. Program control leaves Routine NOGO with a transfer of control to Routine ENDFRESH.

16-105. Routine ENDFRESH executes a RELINT to allow program interrupts to be accepted and processed and then transfers program control to Routine DSPMM (paragraph 17-470), part of program section Pinball. This routine displays on the DSKY's the mission function code number which indicates which Mission Function(s) is (are) active. Since this code was set to zero in Routine STARTSUB, the PROGRAM indicators display the octal digits 00 to indicate that all Mission Functions have been terminated. The code having been displayed, program control returns to Routine ENDFRESH which transfers program control to Routine DUMMYJOB, part of program section AGC Self-Check, via Routine POSTJUMP, part of program section Interbank Communications (paragraph 19-16).

16-106. Routine DUMMYJOB is executed over and over again while searching for a Job of higher priority which is awaiting execution. This process continues until an operator keys in a request for a Mission Function via the keyboard sequence VERB, 3, 7, ENTER, D1, D2, ENTER, where D1 is a single digit identifying the Mission Function and D2 is a single digit identifying the manual Phase of the Mission Function (paragraph 17-473). After the appropriate Pinball operations have been performed, program control arrives at Routine MODROUT of program section Progress Control (paragraphs 16-22 and 16-54).

16-107. RESTART

16-108. The restart function of Fresh Start and Restart is initiated in response to a GO sequence as the result of a hardware failure or an abort. The hardware failures are:

- a. An interrupt which lasts too long or no interrupts within a given period of time (RUPT LOCK)
- b. Program control in a trap (TC TRAP)

- c. A parity failure
- d. A power failure
- e. A counter failure
- f. A scaler failure

An abort is caused by an abnormal program condition such as no Work Areas available (refer to table 19-2 for other causes). Routine ABORT, which is executed as a result of an abort condition, terminates in a TC trap which causes a GO sequence.

16-109. The GO sequence causes program control to be transferred to an Interrupt Transfer Routine as described in paragraph 16-88. This routine (attachment 16-2) execute an INHINT to prevent program interrupts while various registers in Erasable Memory are being initialized. Then program control is transferred to Routine GOPROG, the restart entry of Fresh Start and Restart. Routine GOPROG immediately transfers program control to Routine STARTSUB to initialize those registers concerned with starting or restarting program operation. (Refer to paragraphs 16-101 through 16-103 for a description of this operation.) Since the address of location GOPROG+1 was loaded into register BUF at the beginning of Routine STARTSUB, program control is transferred back to Routine GOPROG following the completion of the initializing operations. At location GOPROG+1, the octal quantity 00003 is entered into the accumulator and program control passes on to Routine PHASECHK.

16-110. Routine PHASECHK compares the content of the three phase tables for complete agreement. The octal quantity 00003 obtained at location GOPROG+1 is entered into register MPAC which causes the phase tables to be compared four times for thoroughness during four sequential passes through Routine PHASECHK. In addition, the quantity 00003 is also used for indexing to obtain the desired register (word) in each phase table. Thus

$$\begin{aligned}
 c(A) = c(MPAC) &= 00003 \text{ on the first pass} \\
 &= 00002 \text{ on the second pass} \\
 &= 00001 \text{ on the third pass} \\
 &= 00000 \text{ on the fourth pass.}
 \end{aligned}$$

Register PHASETAB is indexed by the content of the accumulator (n) and the complemented content of the appropriate PHASETAB register is stored in register BUF. Register BACKPHAS is then indexed by the content of register MPAC (n) and the complemented content of the appropriate BACKPHAS register is stored in register BUF+1. Thus register BUF contains $\bar{c}(\text{PHASETAB}+n)$ and register BUF+1 contains $\bar{c}(\text{BACKPHAS}+n)$. The content of the accumulator, which is $\bar{c}(\text{BACKPHAS}+n)$ at this time, is complemented and masked against

the content of register BUF and the result is left in the accumulator. In this way the $\overline{c}(\text{PHASETAB}+n)$ is compared against the $c(\text{BACKPHAS}+n)$. Program control is then transferred to Routine ZEROTEST with the return address to Routine PHASECHK stored in Register Q.

16-111. Routine ZEROTEST tests the content of the accumulator to determine if the two phase words in registers PHASETAB+n and BACKPHAS+n were the same. If the $c(A)$ does not equal plus zero, the two phase words do not agree; program control is transferred to Routine NOGO; and the fresh start function assumes control of the program section. (Refer to paragraphs 16-104 through 16-106 for a description of this operation.) If the $c(A)$ equals a positive zero, the two phase words agree and program control is returned to Routine PHASECHK at location PHASECHK+10. At this location register PHASEBAR is indexed by the contents of the accumulator and the $\overline{c}(\text{PHASEBAR}+n)$ is stored in register BUF+2.

16-112. Five more masking operations are performed by Routine PHASECHK and, when each mask is completed, program control is transferred to Routine ZEROTEST to determine if the phase words being tested agree. Table 16-5 lists the six tests performed and the phase words involved in each test. Notice each different pair of phase words are compared twice, thus the six comparisons. First the complemented form of one phase word is compared against the true form of a second phase word. Then the complemented form of the second phase word is compared against the true form of the first word. This double comparison is necessary because it is possible for phase word disagreement to go undetected if only one comparison is performed. A simple example of how disagreement escapes detection is shown in figure 16-4 where a 0 indicates agreement and a 1 indicates disagreement. Notice that disagreement undetected in one comparison is detected in the other comparison. Thus the importance of two comparisons can be readily seen. Following the last comparison, program control is transferred to Routine ENDEVOTE.

TABLE 16-5
PHASE WORD COMPARISONS

Number	Phase Word Registers Compared
1	$\overline{c}(\text{PHASETAB}+n)$ and $c(\text{BACKPHAS}+n)$
2	$\overline{c}(\text{BACKPHAS}+n)$ and $c(\text{PHASEBAR}+n)$
3	$\overline{c}(\text{PHASETAB}+n)$ and $c(\text{PHASEBAR}+n)$
4	$\overline{c}(\text{BACKPHAS}+n)$ and $c(\text{PHASETAB}+n)$
5	$\overline{c}(\text{PHASEBAR}+n)$ and $c(\text{PHASETAB}+n)$
6	$\overline{c}(\text{PHASEBAR}+n)$ and $c(\text{BACKPHAS}+n)$

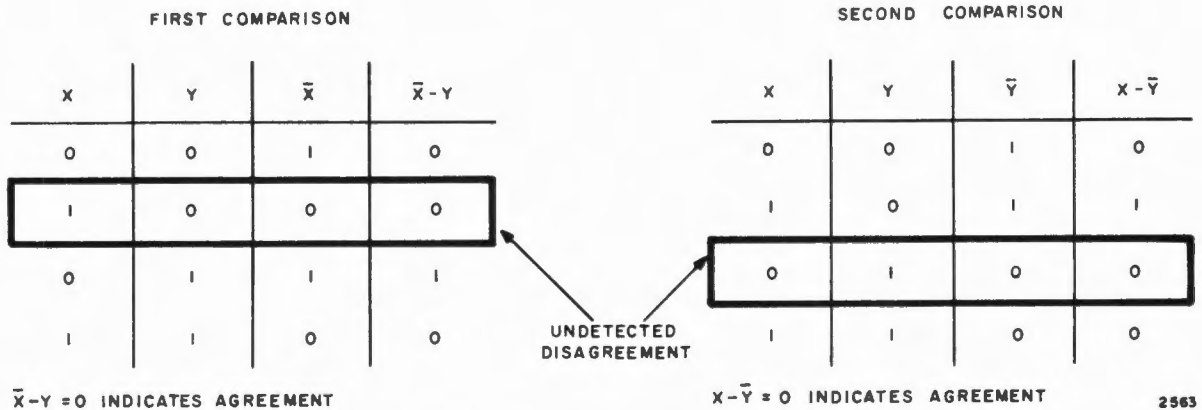


Figure 16-4. Sample Comparison

16-113. Routine ENDVOTE tests the content of register MPAC (00003, 00002, 00001 or 00000) to determine if all four words in the phase tables have been tested for agreement. If the $c(\text{MPAC})$ is greater than plus zero, all the comparisons have not been completed; the $c(\text{MPAC})$ is decremented by one and program control returned to the beginning of Routine PHASECHK. If the $c(\text{MPAC})$ is equal to plus zero, all four phase words have been compared for agreement and program control is transferred to Routine GOJUMP. Thus Routines PHASECHK, ZEROTEST and ENDVOTE continue operating in a loop until all four phase words in the three phase tables have been compared for agreement, or until a disagreement is detected, whichever occurs first.

16-114. Routine GOJUMP uses Routines REPHASE2, PHASESUB and ONSKIP of program section Progress Control to determine if either or both of the two Mission Functions have been terminated. Orbital Integration is tested on the first pass through Routine GOJUMP and Prelaunch Alignment is tested on the second pass.

16-115. Initially Routine GOJUMP enters the octal quantity 00001 (representing Orbital Integration) into register MPAC+1 and executes a RELINT followed immediately by another INHINT. The RELINT/INHINT combination is executed to avoid a RUPT LOCK hardware failure as described in paragraph 16-108. The octal quantity 00001 is entered into register PHASDATA

to indicate to Progress Control which Mission Function is selected. Then program control is transferred to Progress Control at Routine REPHASE2 which, in conjunction with Routine PHASESUB, determines the Phase of the current Mission Function as described in paragraphs 16-49 and 16-50. After the Phase has been determined, program control is returned to location GOJUMP+6 by instruction TC ITEMP2. Register ITEMP2 was previously loaded with the address of location GOJUMP+6 during the execution of Routine REPHASE2.

16-116. At location GOJUMP+6, program control is transferred to Routine ONSKIP which is also part of Progress Control. Routine ONSKIP determines whether or not the selected Mission Function has been terminated, in a manner similar to that described in paragraph 16-46. However, if the Mission Function has been terminated, program control is transferred to Routine GORETURN; otherwise, program control is transferred to Routine DEMANDON via location GOJUMP+8.

16-117. Routine DEMANDON updates the mission function code contained in register MODREG to indicate that the selected Mission Function (Orbital Integration or Prelaunch Alignment) has not been terminated. The mission function code (00001 or 00000) as contained in register PROG is complemented, is placed in the accumulator and is used to index register BIT1. Then the $c(\text{BIT1}-n)$ is placed in the accumulator where n is either 0 or 1 depending upon the Mission Function currently being processed. The relationship is as follows:

<u>Mission Function</u>	<u>c(PROG)</u>	<u>n</u>	<u>c(BIT-n)</u>	<u>c(A)</u>
Orbital Integration	00001	1	00002	00002
Prelaunch Alignment	00000	0	00001	00001

Thus the accumulator contains a ONE in bit position 1 or 2 to indicate which Mission Function is current.

16-118. The $c(\text{MODREG})$, which was previously set to 00000 by Routine STARTSUB, is added to the $c(\text{A})$ and the sum is stored in register MODREG. In this way, the current mission function code is entered into register MODREG for later display on the DSKY's. (Refer to table 16-4 for the significance of the various bit positions in register MODREG.) Program control is returned to Routine GOJUMP at location GOJUMP+9.

16-119. At location GOJUMP+9, the Phase of the current Mission Function, which was loaded into register PHASE by Routine PHASESUB, is stored in

register MPAC. Register GOCADR is indexed with the contents of register PROG (00001 or 00000) and the complete address of the appropriate Mission Function is placed in the accumulator. (Register GOCADR+1 contains the complete address of Orbital Integration and GOCADR contains the complete address of Prelaunch Alignment.) Program control is then transferred to the current Mission Function via Routine SWCALL of program section Interbank Communications.

16-120. Program control is transferred to Routine FFGO in Orbital Integration or Routine PRELGO in Prelaunch Alignment. Both routines are similar and perform the same function for their respective Mission Functions. Initially the priority of the Mission Function is placed in the accumulator for use by the Executive. Program control is then transferred to the Executive at Routine FINDVAC (paragraph 12-32) where a request is made to execute Job FFGO for Orbital Integration or Job PRELGO for Prelaunch Alignment. Program control is returned to Fresh Start and Restart at Routine GORETURN via Routine SWRETURN of Interbank Communication.

16-121. Routine GORETURN tests the content of register MPAC+1 to determine if both Mission Functions have been tested for termination in order to update the Mission Function status code on the DSKY's. If register MPAC+1 contains a quantity greater than plus zero (00001), Orbital Integration has been processed but Prelaunch Alignment has not. Therefore, the content of register MPAC+1 is decremented by one and program control is transferred to Routine GOJUMP at location GOJUMP+1 to allow Prelaunch Alignment to be processed. If register MPAC+1 contains a plus zero, both Mission Functions have been processed. Routine GORETURN then enters the content of register ALARMPR (the priority of Job DOALARM) into the accumulator and enters the Executive at Routine NOVAC (paragraph 12-25) to request the execution of Job DOALARM.

16-122. Upon return from the Executive, Routine ENDFRESH is executed as described in paragraph 16-105. When Routine DUMMYJOB is executed, Job DOALARM (priority 37) is found to be awaiting execution; therefore, Job DOALARM is executed to display the failure or abort code on the DSKY's. After executing Job DOALARM, that Mission Function, which was interrupted by the GO sequence, will be executed beginning at the Phase in which the Mission Function was operating when the interrupt occurred.

ISSUE 16
APPENDIX A

This appendix contains a list of registers set to zero by Routine STARTSUB during a fresh start or a restart. The list includes the registers affected and the significance of each register to the program.

Register(s)	Significance
DSPCNT	Select the initial DSPTAB register (refer to paragraph 17-102).
NEWJOB	Indicates that no Job of priority higher than the dummy Job has been requested. (Refer to paragraphs 12-13 through 12-15.)
MODREG	Indicates that all Mission Functions have been terminated (refer to paragraph 17-484).
CADRSTOR	Indicates that no Jobs are waiting for a data load before continuing execution.
REQRET	Initializes the return address register used following a VERB request (refer to paragraph 17-125).
CLPASS	Indicates that the next data word keyed in from the DSKY's will be cleared if the CLEAR key is depressed (refer to paragraph 17-171).
DSPLOCK	Indicates that Pinball has been released by the operator and is available for internal use (refer to paragraphs 17-217 through 17-224).
MONSAVE MONSAVE1	To terminate monitor activity (refer to paragraph 17-355).
GRABLOCK	Indicates that program section Pinball has not already been "grabbed" by an internal program (refer to paragraphs 17-199 through 17-206).
VERBREG	Indicates that no VERB code is to be displayed (refer to paragraph 17-122).
NOUNREG	Notifies the display routine that no mission function code or NOUN is to be displayed (refer to paragraphs 17-126 and 17-149).
DSPLIST DSPLIST+1 DSPLIST+2	Clears the display waiting list of the complete re-activating address of all internal Jobs that have attempted to use Pinball but found it either already "grabbed" or under operator control (refer to paragraph 17-225).

Register(s)	Significance
STATE STATE+1 STATE+2	Turns off all 45 switches (each bit position is used as a switch) used by program sections using Interpretive Instructions.
PWTCADR PWTCADR+1	Notifies program section Progress Control that no Mission Functions have been suspended by the Executive (refer to paragraphs 16-61 and 16-70).
DESKSET	Indicates that the AGC should not control the IMU CDU's (refer to paragraph 18-49).
DESOPSET	Indicates that the AGC should not control the Optics (refer to paragraph 18-64).
IMUCADR	Notifies Routine IMUSTALL in program section Mode Switching and Mark that no Job is in the dormant state awaiting an IMU mode switch and that the system is prepared to put a Job requesting a mode change into a dormant state until the switch has been completed (refer to paragraph 14-99).
OPTCADR	Same as for register IMUCADR except that it applies to Optics mode switching instead of IMU mode switching (refer to paragraph 14-99).
OLDERR	Clears all old sampled error bits from bit-positions 12, 11, and 10 of register IN2 (refer to paragraph 18-82).
DISPBUF	Indicates to Routine DOWNTMOK of program section Downrupt Processor that there is no relay word to be sent via downlink (refer to paragraph 18-109).
IDPLACER	Causes Routines DATA/ID and IDNEXT of Downrupt Processor to send an ID word downlink prior to the sending of the first of the data words (refer to paragraph 18-112).

Register(s)	Significance
TMINDEX	Causes Routines DATA/ID to send the first data word downlink from that location specified by the address stored in location DNLST1+OCT37. This location contains the address of the first DNLST1 data word (refer to paragraph 18-112).
TMKEYBUF	Signifies to Routine CHARTEST that there is no character word to be sent downlink. This avoids sending false or unwanted words that might be in register TMKEYBUF previously (refer to paragraph 18-111).
MARKSTAT	Indicates that no Mark operations have been requested and leaves the system free to immediately recognize Mark requests (refer to paragraph 14-24).
IN0WORD	Clears register of unwanted data. Register IN0WORD is a temporary storage for the keycode (bits 5 through 1 of register IN0), or a Mark (bit 15 of register IN0).
OUT1	Clears register of signals to the DSKY's and external subsystems.
<p>Note: When the PRIORITY registers in the Job Areas are initialized (except 01000 in Job Area 1 for Routine DUMMYJOB), the information contained in the other registers of the Job Area is preserved. Therefore, since Routine STARTSUB is executed as a subroutine of Job CHARIN, information pertaining to CHARIN is preserved in Job Area 1 when the dummy Job is entered. Information pertaining to Job CHARIN remains in this Job Area until the dummy Job is displaced by a Job of higher priority. (Refer to paragraphs 12-57, 12-58 and 12-63 through 12-68.)</p>	