Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Naval Research Laboratory Washington, D.C. 20390	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

THE NRL MICROWAVE SPACE RESEARCH FACILITY: COMPUTER SYSTEM SOFTWARE FOR CONTROL OF THE 60-FT X-BAND ANTENNA

4. DESCRIPTIVE NOTES (Type of report and inclusive dates) This is a final report on one phase of the NRL Microwave Space Research Facility. Work on other phases is continuing.

5. AUTHOR(S) (First name, middle initial, last name)

Charles A. Bass and Theodore F. Schmeckpeper

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
January 25, 1971	66	3

8a. CONTRACT OR GRANT NO. NRL Problem R01-36 b. PROJECT NO. XF 48222001.2116.5/2116.11/3116.51 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) NRL Report 7188 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Department of the Navy (Naval Electronic Systems Command) Washington, D.C. 20360

13. ABSTRACT

To extend the Navy's program in satellite communications, a fully steerable 60-ft X-band antenna system has been completed at the Naval Research Laboratory's Microwave Space Research Facility at Waldorf, Maryland. The antenna can be operated in five different modes. In one of these, the digital mode, a general-purpose digital computer has been programmed to provide direct digital control of the antenna position.

Software developed at the site has been organized into two operating systems. A nonreal-time system is comprised of a resident monitor and basic utility programs necessary for efficient operation of a computer installation. The second system, which provides digital control of the antenna in a real-time environment, is of a multilevel, time-shared structure. Four software levels are executed on a priority basis by a real-time executive. Two levels may be utilized for execution of auxiliary programs concurrently with antenna control. Several interrupt processing routines provide linkage to the real-time dynamics of the antenna system.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6801

Security Classification

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Digital computers						
Real time operations						
Computer systems programs						
Time sharing						
Spacecraft tracking						
Digital systems						
Servomechanisms						
Steerable antennas						
X-band antenna						
Celestial tracking						

DD FORM 1 NOV 65 1473 (BACK)
(PAGE 2)

Security Classification

CONTENTS

ABSTRACT

To extend the Navy's program in satellite communications, a fully steerable 60-ft X-band antenna system has been completed at the Naval Research Laboratory's Microwave Space Research Facility at Waldorf, Maryland. The antenna can be operated in five different modes. In one of these, the digital mode, a general-purpose digital computer has been programmed to provide direct digital control of the antenna position.

Software developed at the site has been organized into two operating systems. A nonreal-time system is comprised of a resident monitor and basic utility programs necessary for efficient operation of a computer installation. The second system, which provides digital control of the antenna in a real-time environment, is of a multilevel, time-shared structure. Four software levels are executed on a priority basis by a real-time executive. Two levels may be utilized for execution of auxiliary programs concurrently with antenna control. Several interrupt processing routines provide linkage to the real-time dynamics of the antenna system.

PROBLEM STATUS

This is a final report on one phase of the NRL Microwave Space Research Facility. Work on other phases is continuing.

AUTHORIZATION

NRL Problem R01-36
Project XF 48222001.2116.5/2116.11/3116.51

Manuscript submitted August 19, 1970.

THE NRL MICROWAVE SPACE RESEARCH FACILITY

COMPUTER SYSTEM SOFTWARE FOR CONTROL OF THE 60-FT X-BAND ANTENNA

INTRODUCTION

This is the fourth in a series of reports on NRL's Microwave Space Research Facility, located approximately 6 mi west of Waldorf, Maryland (Fig. 1). The first of these reports dealt with the design and development of the 60-ft X-band antenna (1). Subsequent reports have described the computer and interface hardware (2) and operator control of the antenna system (3).

Among the modes in which the antenna can be operated is the digital mode. In this mode, direct digital control of the antenna position is provided by a closed-loop digital servo employing a Control Data Corporation 924A general-purpose digital computer (Fig. 2). The computer closes the servo loop, in addition to generating position commands for satellites and celestial objects. The computer was acquired and installed at the site in March 1964. Included in the initial purchase were a line printer, card reader, paper tape reader, paper tape punch, and console typewriter. Development of the antenna control software and other computer programs was initiated, and by the Spring of 1966 a fully operational system was brought on-line. The software for this initial system was stored on paper tape. Data input to the system employed punched cards.

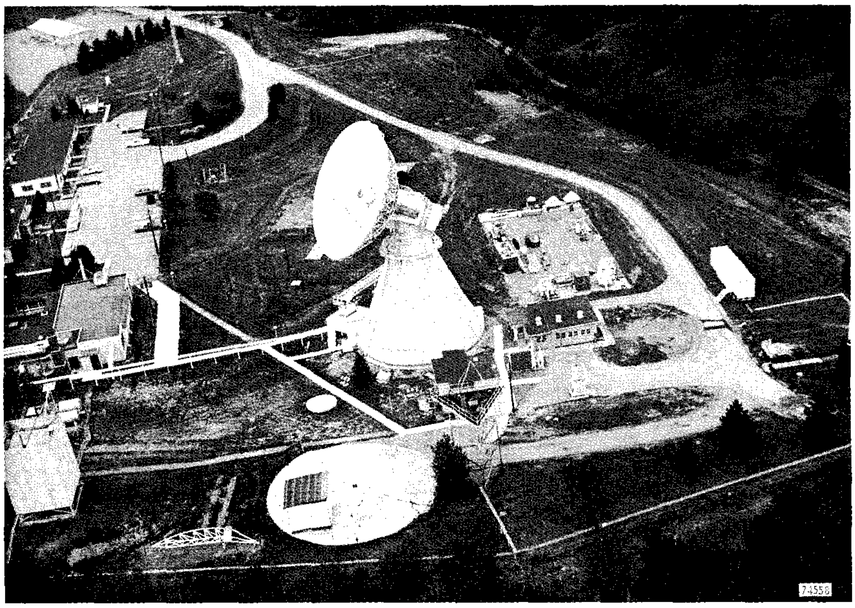

Fig. 1 - Aerial view of the 60-ft X-band antenna at NRL's Microwave Space Research Facility, located at Waldorf, Md.

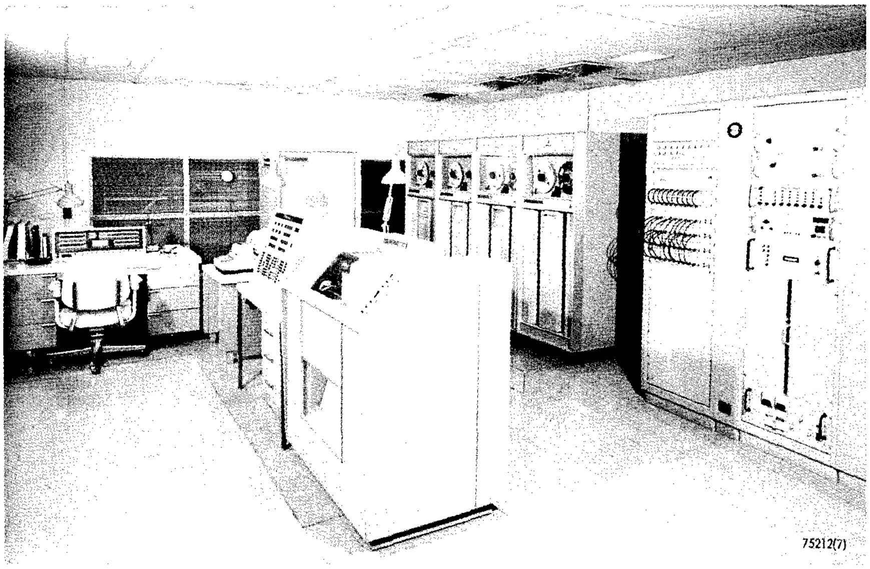

Fig. 2 - View of the CDC 924A general-purpose digital computer. In the digital mode the computer is used to provide direct digital control of the antenna shown in Fig. 1.

Subsequently, work was initiated to modify the system for magnetic tape program and data storage. In addition, the original system, which was input to the computer as a single program, was broken up into smaller units. Only part of the first system would reside continuously in the computers's memory. Additional routines could be easily loaded from magnetic tape as needed, using a newly developed operating system. Thus, more efficient use of the computer's 16,384-word memory would be achieved.

In July 1966 four magnetic tape handlers and a magnetic tape control unit were purchased. The magnetic tape version of the operating system was brought on-line shortly afterward. The software described in this report has been in use since the Summer of 1967.

In the digital mode of antenna control two interfaces are employed, one between the computer and the antenna, and a second between the computer and the system operator. The antenna servo is completely closed through the computer—generated position commands are compared with the encoded antenna position and the resulting errors are output to the antenna servo. Once an initial dialogue with the computer has been completed, however, the antenna operator can forget that the computer is present. Further antenna control is effected not through the computer typewriter, but rather through a number of pushbuttons and thumbwheel switches linked to the computer via the computer sensing of contact closures and the computer control of associated lamps. Thus, the antenna operator may acquire the feeling that he has direct hard-wired control over the antenna, while at the same time being protected by software logic from making operational mistakes.

Although digital control of the antenna is the primary function of the computer, the latter is also used for many unrelated tasks. Two operating systems have been developed for use at the site, thereby obtaining the utility and flexibility required in both areas of computer usage. These systems have been designated as the real-time system (antenna control) and the nonreal-time system (no antenna control) (Fig. 3). All computer programs produced at the site, including the software making up both systems, are written in assembly language.*

The real-time system is a multilevel, multiprogrammed, time-shared software structure. Software on each of four main levels, designated as Class A levels, performs a separate but not necessarily independent function. In contrast to the levels of programming in typical multilevel systems, information is processed in a seemingly independent fashion but in reality is frequently exchanged between levels for further processing. A large library of reentrant routines, always available in memory in the real-time system, may be used by software on any of the Class A levels.

A system of computer interrupts links the software to the real-time world. Typical use of externally generated interrupts is in signalling the program when antenna positions are available for input and providing the structure for execution of the software in a series of timed intervals. Additional internally generated interrupts provide communication between the Class A levels and an executive monitor routine, which controls actual execution of these levels.

Because of economical programming and the speed of the computer, all antenna control functions (the two top priority levels) and system overhead are executed in about one-third of the time available in a given interval, for the worst case combination of command generation and control. More frequently, these functions are accomplished in less than half that time. Hence, software on the two lower priority levels has available a large amount of execution time. A typical real-time scenario might have the antenna executing a programmed scan about a satellite's computed position, periodic computation of new position points for the satellite, and line printer output of another object's positions for requested future times, all performed concurrently.

OPERATIONAL CHARACTERISTICS

Operator control of both the real- and nonreal-time systems is effected through use of the computer console typewriter, which serves both as an input medium for the operator's requests and as an output medium through which the computer supplies necessary information to the operator. The typewriter is also used to establish question-answer dialogues by programs which require additional information to completely define the action desired.

Programs are input to the computer from the system library magnetic tape, on which they are stored in a machine-language format. The first file on this tape is the directory. Each record in the directory is a binary-coded-decimal (BCD) card image containing the name of a program and a set of standard information about the program. The second file contains the basic control program for the nonreal-time operating system, while the remaining files each contain a computer program used at the site.

A second tape, designated as the system data tape, is used in conjunction with the real-time system. This tape also has as its first file a directory to the subsequent files, which in this case contain data stored as BCD card images. This data includes orbital elements of artificial Earth satellites, information defining remote antenna sites, ephemerides of the sun, moon, and planets, and mean positions of various stars and celestial radio sources. Although this data is normally input to the real-time system from the system data tape, it may alternatively be input through the punched card reader.

*Several assembly language routines have been written which, when assembled with an assembly language program, provide the convenience of Fortran-like input and output format specifications (Iw, Fw.d, Ew.d, etc.).

Listings of the system library tape and system data tape directories are given in Appendix A.

PROGRAM CLASSIFICATIONS

Programs are classified in two ways. One method is by the program's "mode." The mode refers to the system under which the program is intended to be executed. Programs are thus referred to as real-time or nonreal-time programs. A program's mode is contained within the directory record of the program.

A program may be confined to operation within the nonreal-time system for either of two physical reasons: it may require more computer memory than is available within the real-time system, or it may use a particular piece of hardware in such a way as to be incompatible with the real-time system. Other programs are designated as nonreal-time because their execution during real-time operation would inconvenience the antenna operator, even though such execution may be physically possible. The mode of the main control program for the real-time system is also nonreal-time, since it must be loaded and executed to initiate operation of the real-time system.

Most real-time programs have functions pertaining to the input of information to the antenna control software. Other real-time programs are used for data acquisition, or the computation of output data involving extensive use of the real-time software. Additional programs not related to these functions may be executed within the real-time system. These include routines such as memory dumps which may be useful during real-time operation and which can be conveniently aborted should one of the former programs be required. The latter routines comprise a third program mode, since they may be executed within either the real- or nonreal-time systems.

A final program mode is necessary for special routines which may be executed only under control of another program. Thus they are neither real- nor nonreal-time programs, but are part of a hybrid system composed of their calling program and the particular operating system within which this program is executed.

A second method of program classification concerns whether the area of core into which the program may be loaded is restricted. Certain programs are always loaded at a particular address in the computer memory, and are thus referred to as absolutely addressed programs. They are stored on the library tape as binary records of their machine instructions. The loading address and size of these programs are contained in their directory records.

The remaining programs are relatively addressed, or relocatable, programs. They can be loaded and executed in any area of memory not reserved for the current system software. The actual area into which they are loaded is specified by the system, the operator, or another program. Relocatable programs are stored on the library tape as a series of relocatable binary card images which are produced by the assembly processor used at the site.

NONREAL-TIME SYSTEM

The software portion of the nonreal-time system consists of a resident monitor program and the various nonreal-time programs contained on the system library tape. All the computer's standard peripheral equipment is available to this software. In addition, much of the hardware referenced through the interface racks (Fig. 3) may be used by nonreal-time programs, although connections are seldom made to this hardware except by programs specifically designed to test the hardware.

The resident monitor (designated hereafter as either the "resident" or the "monitor"), which is initially loaded into core by a short paper tape bootstrap, remains in memory at all times during which either the nonreal-time or the real-time system is in operation. This program, called NACSYS, resides in the upper 2560 locations of memory. This leaves approximately 13,300 words of memory for use by other nonreal-time programs.

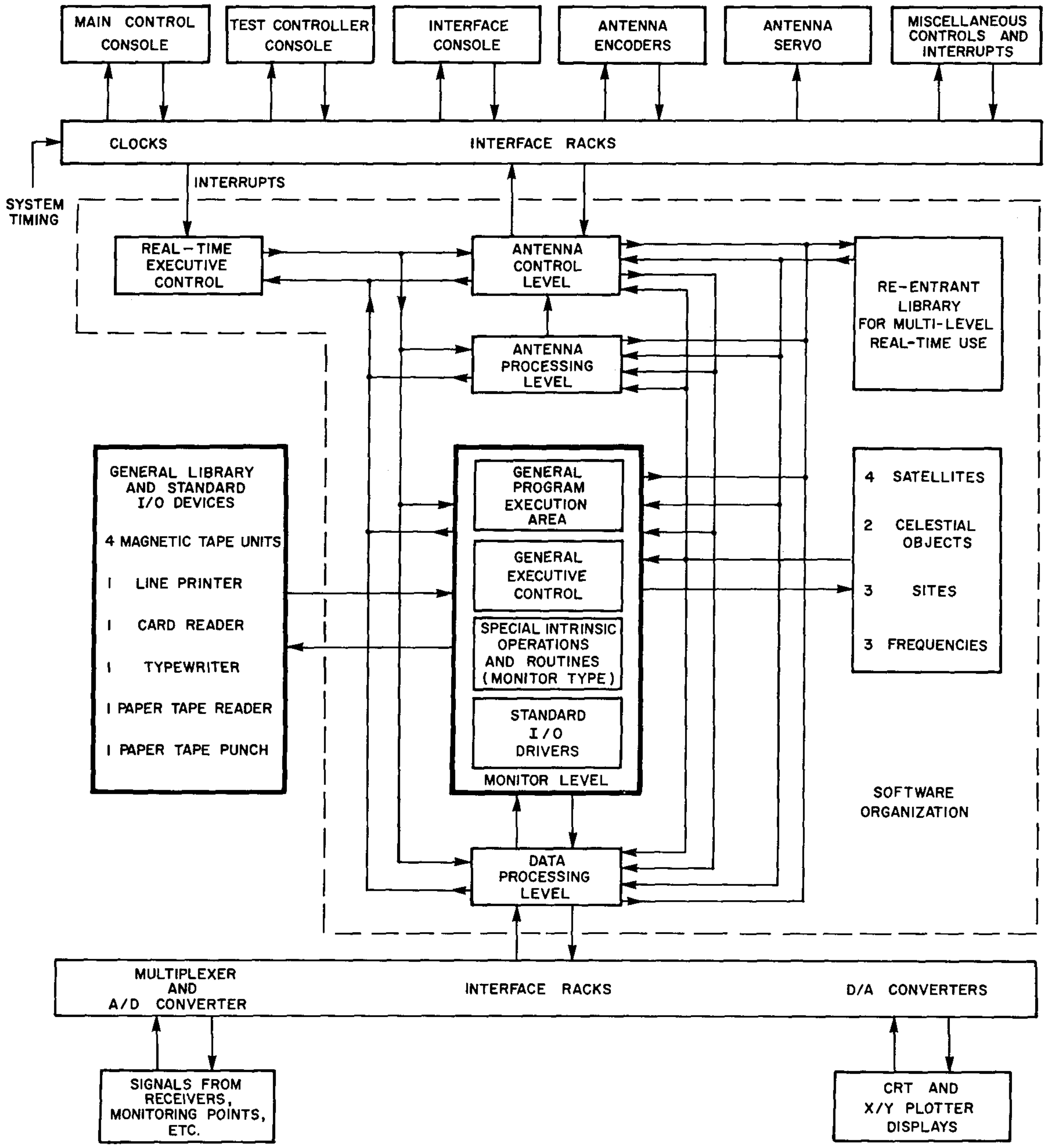

Fig. 3 - Block diagram of the real-time NACTAC computer system. This system consists of hardware and software elements for digital control of the antenna and for acquisition of data for processing and display.

NACSYS provides basic control for the nonreal-time system. The computer operator communicates with NACSYS through the computer console typewriter. Certain general functions may be performed directly by the resident upon operator request. For more specific and complex functions, the resident is requested to load and execute one of the nonreal-time programs from the system library tape.

Among the most important nonreal-time programs are two upon which the rest of the system is based. The first is the assembly processor used at the site. This processor converts programs from symbolic programming language (assembly language) to machine language. Initial input is on punched cards. The magnetic tape output includes printer line images (listing tape) and object deck card images (object tape). Once a listing tape is obtained for a given program it may be used as input, together with punched change cards, for subsequent assemblies modifying the program.

The second program is that used to update the system library tape. Program files to be added are input from a single magnetic tape. Basic control is provided by punched control cards indicating where, on the present library tape, programs are to be added or deleted. Additional card input consists of directory cards containing information to be saved in the directory records for new programs. This program is also used to create and update the system data tape.

Other nonreal-time programs include those used in generating files for the system data tape; various routines, such as memory dumps and tape copying programs, which perform basic data-transfer functions; and hardware diagnostic routines for both peripheral equipments and certain interface gear.

Because of the absence of a Fortran compiler in the site's software repertoire, it is generally inefficient to write data processing and scientific-problem-solving programs of the type typically run at batch processing centers. Of course, some of the real-time software of necessity consists of such programming, and certain routines are available to the nonreal-time system for the processing of data which is collected or generated at the site. One such program is SPECTRAL, which uses the Fast Fourier Transform algorithm to compute periodograms of data samples obtained from an analog source. More programs of this type may be added to the system in the future, but they will be largely confined to reduction of data acquired directly through the system interface in relatively small quantities. For data processing on a larger scale it is anticipated that use of the Laboratory's CDC 3800 computer will be desirable because of that machine's higher speed, additional peripheral devices, and Fortran capabilities.

RESIDENT MONITOR

Processing of Operator Requests

The operator's typed requests are in the form of an alphanumeric code. The code always begins with an alphabetic character and has a maximum length of eight characters. Typing the eighth character of an eight-character code automatically terminates the request and initiates appropriate action by the monitor. For codes of less than eight characters, a period is required to terminate the request. A carriage return typed at any time prior to termination of a request will nullify the request and return the system to a condition in which further input is interpreted as a new request.

The processing of a request is outlined in Fig. 4. The operator may request either a general ("intrinsic") routine capable of execution by the resident itself, or the loading of a program from the system library tape. If one of the resident's intrinsic routines is requested, the operator may be required to type additional information to explicitly define his request.

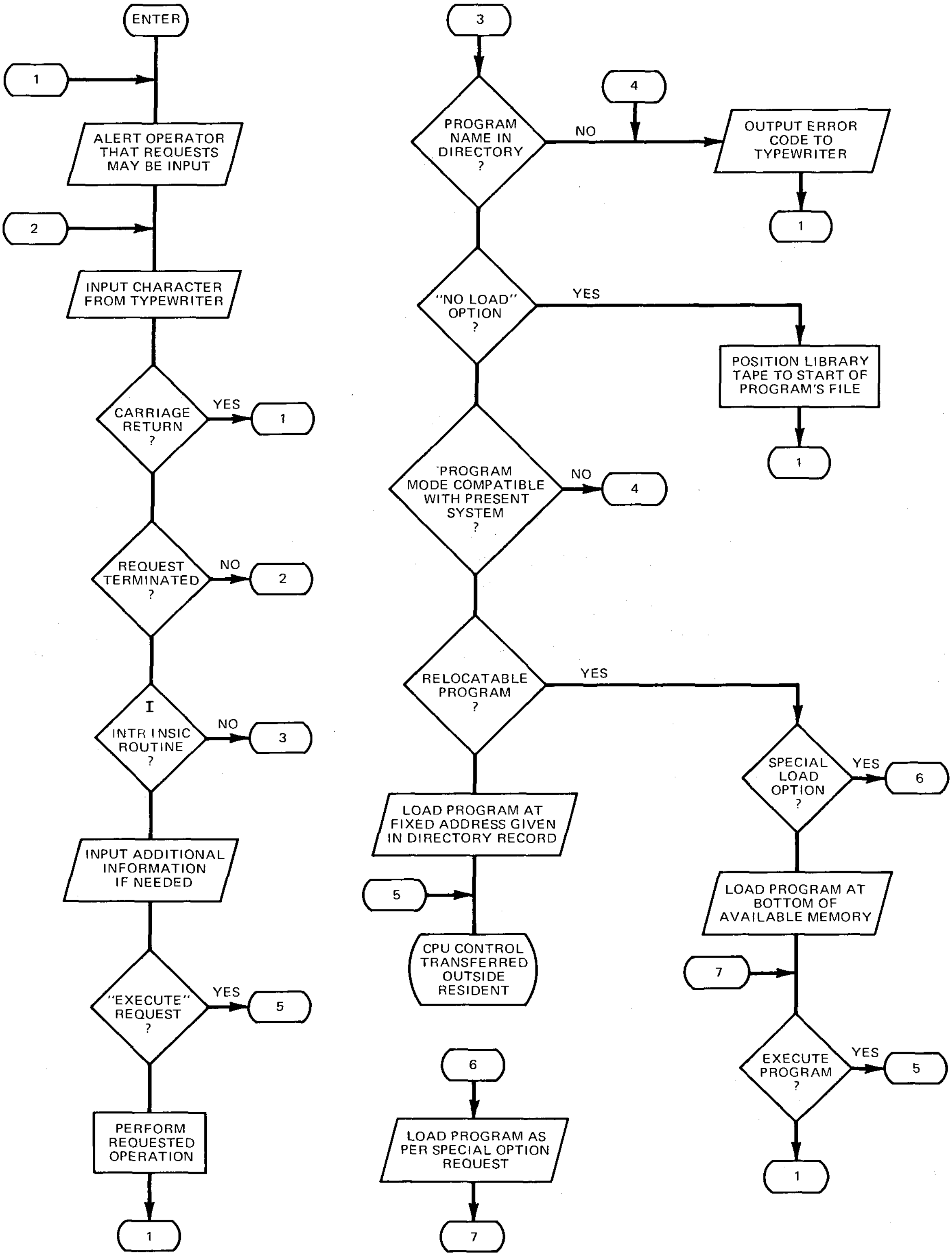

Fig. 4 - Flow diagram of the processing of NACSYS operator requests

If a program from the library tape is desired, the request code which is used is the program name as it appears in the library tape directory. Absolutely addressed programs are, of course, always loaded at the same core location. Relatively addressed programs are generally loaded into the lowest portion of available memory. All programs are normally executed immediately upon completion of loading. However, the operator can, by typing a digit as the first character of his request, employ certain options to specify whether the program is to be executed immediately, and where the relatively addressed programs are to be loaded. In addition, the option to position the library tape at the start of the program's file may be requested, in which case the program is not loaded. Unless this last option is specified, the program mode is always checked to certify that it is compatible with the current system (real time or nonreal-time).

Intrinsic and External Routines

Intrinsic routines are designated as such because the software needed to perform their function is contained entirely within the resident. With few exceptions, the intrinsic routines may be classified by their functions as either tape positioning or information transfer routines.

Tape positioning routines perform such operations as rewinding tapes on specified units and advancing or reversing tapes a specified number of records or files. Information transfer routines perform a variety of operations: single binary records containing the contents of a designated area of the computer memory may be written on a particular tape; binary records can be read into core; binary card images of a program object deck may be loaded from a tape; and tapes containing BCD line or card images can be listed on the line printer. Included among these routines are the basic input/output (I/O) drivers which handle standard I/O operations between the computer and the magnetic tape handlers, line printer, card reader, and typewriter.

One of the intrinsic routines falling outside these two basic classifications is the "execute" routine, which transfers program control from within the resident to a designated location in the computer memory. No further operator requests can be processed by the resident until control is returned to it by the software which has been executed.

External routines are simply the programs contained on the library tape. Use of these programs is similar to the intrinsic "execute" request. After a program has been loaded, control is transferred to it, and the resident ceases its dialogue with the operator until the external program has run to completion, at which time it transfers control back to the resident.

Common Usage of Monitor Routines

Most of the monitor's intrinsic routines, especially the I/O drivers, perform general functions which are very useful to the typical computer program. The NACSYS monitor has therefore been designed to facilitate use of its intrinsic routines by other programs used at the site. This has been accomplished by providing a set of entry points to the intrinsic routines near the beginning of the NACSYS program. Since NACSYS itself is an absolutely addressed program which always resides in the same area of core, these entry points are easily referenced by assembling, with a particular program, a list of equivalence statements defining the entry points by their absolute addresses within the computer (Appendix B).

To use one of these routines from an external program, an indirectly addressed return jump is taken to the appropriate location in the entry point list. Control parameters are passed from the external program to the routine through the computer's operational registers. For a write operation on magnetic tape, for example, the parameters would include the starting address of the output buffer, the number of words to be transmitted, the tape unit number, and the parity to be used. Any output from the intrinsic routine to the external program is also accomplished using the operational registers.

Two intrinsic routines are particularly useful to external programs. One is the routine used by the resident for inputting operator requests. This may be utilized by an external program in fashioning its own monitor control system. The program is then capable of accepting operator requests and performing various functions intrinsic to the *external* program.

The second routine is that used by the resident for locating and loading program files from the system library tape. By inputting a program name to this routine, an external program can load and execute other programs without requiring operator action. This routine is also used to position the system data tape to particular data files by data entry programs in the real-time system.

I/O Switches

A status check is always performed prior to initiating an I/O operation. Associated with the status checking software for each I/O driver is a software switch, called an I/O switch. This is an unconditional jump instruction, the address portion of which may have different values. Under normal nonreal-time operation, an I/O switch is set to cause continual looping through the status check subroutine until a ready status is obtained from the hardware. If paralleling of I/O is desired, however, the switch may be set to other locations by an external program. (The I/O switches are referenced by a technique similar to that by which the intrinsic routines are used by external programs.) By this procedure, data processing or initiation of additional I/O can be accomplished while one I/O operation is being completed, instead of looping through a status check during this time.

REAL-TIME SYSTEM

The purpose of the real-time software is to provide an operator-controlled program tracking capability to the 60-ft antenna. In order to do this, three basic functions must be accomplished by the software. First, it must be capable of computing the position, referred to the antenna site, of an object to be tracked. That is, given orbital elements or ephemeris data for an object, the system must compute the azimuth and elevation of the object at any given time for the local site. Second, the system must be capable of comparing a desired antenna position with the actual position and buffering the resultant position errors to the antenna servo. Third, means must be provided for operator control of the entire process.

Two additional functions have been designed into the real-time system which, although not mandatory, are very desirable. One of these is the capability to enter new data into the system while the three primary functions are being executed. This removes the necessity of restarting the entire system if new parameters are desired. The second is that of acquiring and processing digital data in real time while all the above functions are being fulfilled.

Although all these functions are either necessary or desirable, there is a large discrepancy in how often the various functions must be accomplished. For example, error output to the antenna must be performed on a regular basis many times per second to achieve smooth antenna motion. Computation of tracking data in the form of x, y, z coordinates, however, need be done only every few seconds if accurate interpolation methods are used for times in between these computations. And if the computation points are calculated for several "computation periods" in advance, this function assumes (timewise) a noncritical nature and can be performed on a time-available basis. Again, the two functions described as desirable but not mandatory may be used infrequently and at random intervals.

Because of the great amount of variability in the demand for these functions, a time-shared system was designed which is capable of performing mandatory functions as often as necessary and which still has time available for performing the other operations whenever they are desired.

The basic real-time computer program, NACTAC, consists of the software necessary to perform the mandatory functions mentioned above, plus the control software for the time-sharing system. The two remaining functions, that of data entry to the system and acquisition and processing of digital data, are performed by other real-time programs. In order to make most efficient use of the time-sharing capabilities of the system, the software used, especially that comprising NACTAC, was designed to require a minimum amount of central processing unit (CPU) time. This was the primary consideration in writing the various subroutines within the system—the amount of core used by the subroutines was given a much lower priority.

Under the nonreal-time operating system, primary control is effected by means of the computer console typewriter. This method of control is also used in the real-time system. For certain functions, however, particularly those associated with operator-via-computer control of the antenna, the typewriter is unsuitable as a means of communications. For these functions, control consoles (Fig. 5) were built into the system by which the antenna operator can command the computer's antenna control functions via push-buttons and thumbwheel switches (2).

Position errors are transmitted to the antenna servo at a rate of 128 per second. The time interval of 1/128 sec, one of the basic time units of the system, is referred to as a subsample period. The externally generated interrupt signal which signifies the beginning of such a period is called a subsample pulse. A second basic time unit is an interval of 1/16 sec. This interval is called a sample period and is initiated by an interrupt signal called a sample pulse. Every eighth subsample pulse is accompanied by a sample pulse.

The system is not stringently confined to the particular subsample period of 1/128 sec. It can be changed to 1/64, 1/32, or 1/16 sec by changing one hardware wiring connection and modifying a single software instruction. In these cases there would be respectively four, two, or one subsample periods in a sample period, rather than the present eight.

REAL-TIME SOFTWARE

Multilevel Time-Shared Structure

The basic software configuration of the real-time system (Fig. 3) consists of two computer programs. The first of these, NACTAC, operates as a multilevel time-shared system whose various levels perform different functions in a real-time environment. The second program, NACSYS, the resident monitor program in the nonreal-time system, operates on one of the software levels of NACTAC, providing operator control of the computer and, in general, performing the same functions as it does in the nonreal-time system.

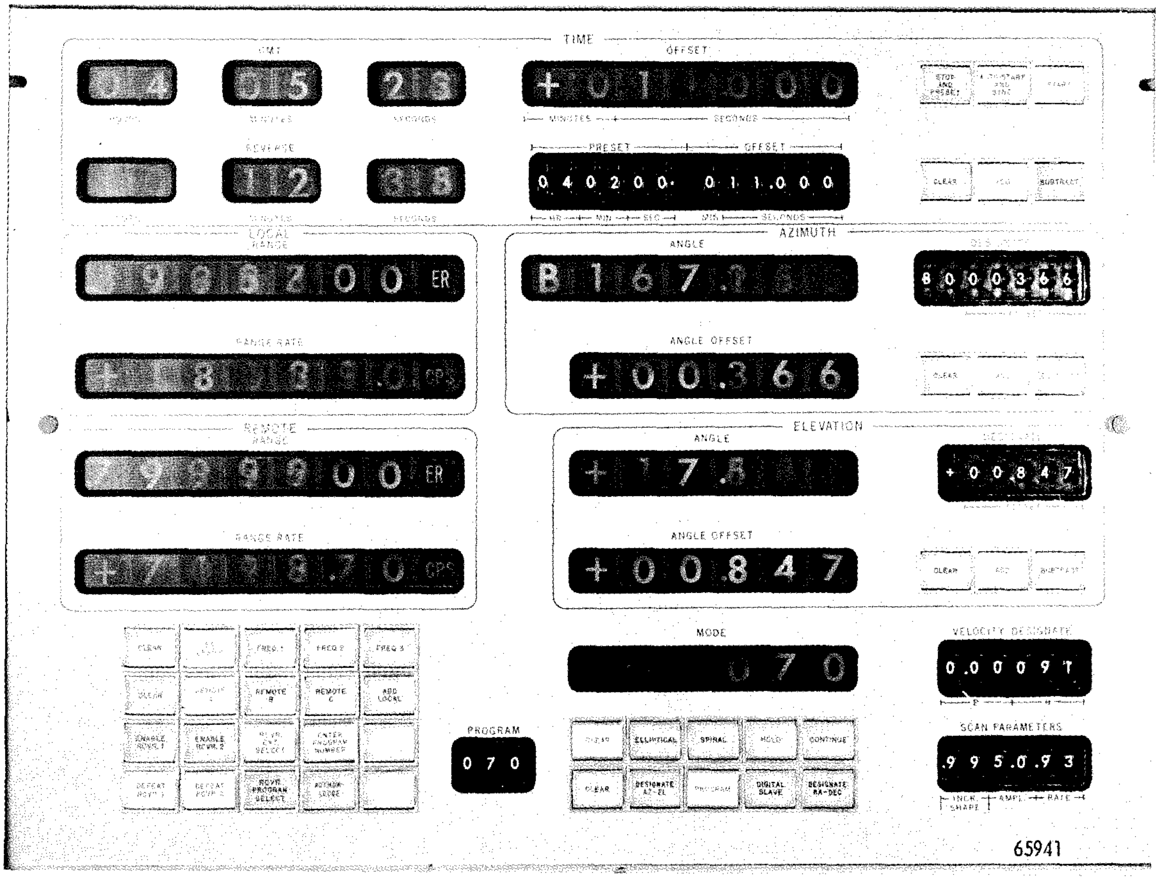

Fig. 5 - By means of control consoles, such as the one shown here, the antenna operator can command the computer's antenna-control functions via pushbuttons and thumbwheel switches

As in the nonreal-time system, NACSYS resides in the upper 2560 words of core. NACTAC occupies the lower 7754 core locations, leaving approximately 5500 words of core as unused. This area is available for loading and execution of the auxiliary real-time programs.

A standard order of priority is assigned to four Class A software levels incorporated in NACTAC. However, this priority scheme may be altered by software on any level should special circumstances make this necessary. Software residing on a Class A level is allocated use of the CPU by a master executive routine responsible for processing requests for execution time and activating the software.

There also exists software in the system which is not executed on any Class A level. Such routines are called Class B routines. They are generally entered on programmed interrupt conditions and are executed with other interrupts locked out. The master executive routine, which is itself a special type of Class B routine, has no "knowledge" of other Class B routines and has no control over their execution.

Associated with each Class A level is a block of cells called a sublist. The sublists are a part of NACTAC, which is an absolutely addressed program; thus the sublists always occupy the same portion of memory. The first part of each sublist contains a block of storage cells. Included in this block are several cells which may be used by the software on that particular level as temporary storage locations, locations which are used by the master executive routine in saving and restoring registers and internal faults when activating or deactivating a level, and special locations in which to save the job status and execution address for the level.

Also included in each sublist are several pairs of cells by which software on that level can utilize various subroutines common to all Class A levels. The first cell of each pair is used to store the return address of the calling program, while the second cell provides entry to the common subroutine. About 80 of these subroutines are contained in NACTAC, each available for use by any of the four software levels. No problems arise if more than one level makes use of the same routine, even though each level may be interrupted while within the routine. This is because the common routines use cells unique to each level, at the beginning of that level's sublist, when temporary storage is needed within the routine, and also because each level has its own entry and exit points to the routines. Finally, every program written for use in the real-time system can operate within this multilevel framework, by having appropriate lists of equivalence statements assembled with it to provide linkage to NACTAC (Appendix B). A program can, in fact, operate on any of the four levels desired, provided that the integrity of the antenna control function of NACTAC is preserved.

Real-Time Executive

The executive routine, called EXEC, provides basic control of the multilevel system as depicted in Fig. 6. It is a Class B routine and as such is always executed with interrupts locked out. There are three specific occurrences which cause control to be passed to the executive routine:

1. At the beginning of each sample period control is transferred to EXEC by a routine which processes the sample interrupt.

2. When a software level is temporarily or permanently finished with its assignment, it passes control to EXEC by causing a forced interrupt. This is made to occur by executing a particular select code, which causes the interface to send an interrupt signal to the computer.

3. The executive may cause reentry to itself in 1/128 sec by arming a hardware strobe, which produces an interrupt after that amount of time has elapsed.

When the executive is entered, it is first ascertained whether the routine was entered via the forced interrupt. If this is the case, the address stored by the computer's interrupt system will have no meaning since it will always be the last instruction of one of the routines which requests the forced interrupt. Therefore, the continuation address for the last Class A level executed is obtained from a cell set by the latter routines and is substituted in the cell containing the usual interrupt address.

The executive then checks the job status of each Class A level, beginning with that level which has top priority and moving down the priority list. If the job status is "complete", no function is to be performed by the software on that level. All the remaining types of job status indicate that the level has something to do, but they differentiate as to when the level is to be activated. A "type 2 deferred" level is not to be activated until the next sample period; a "type 1 deferred" level is to be activated no sooner than 1/128 sec hence; and a "not complete" status indicates that the level is to be executed as soon as possible.

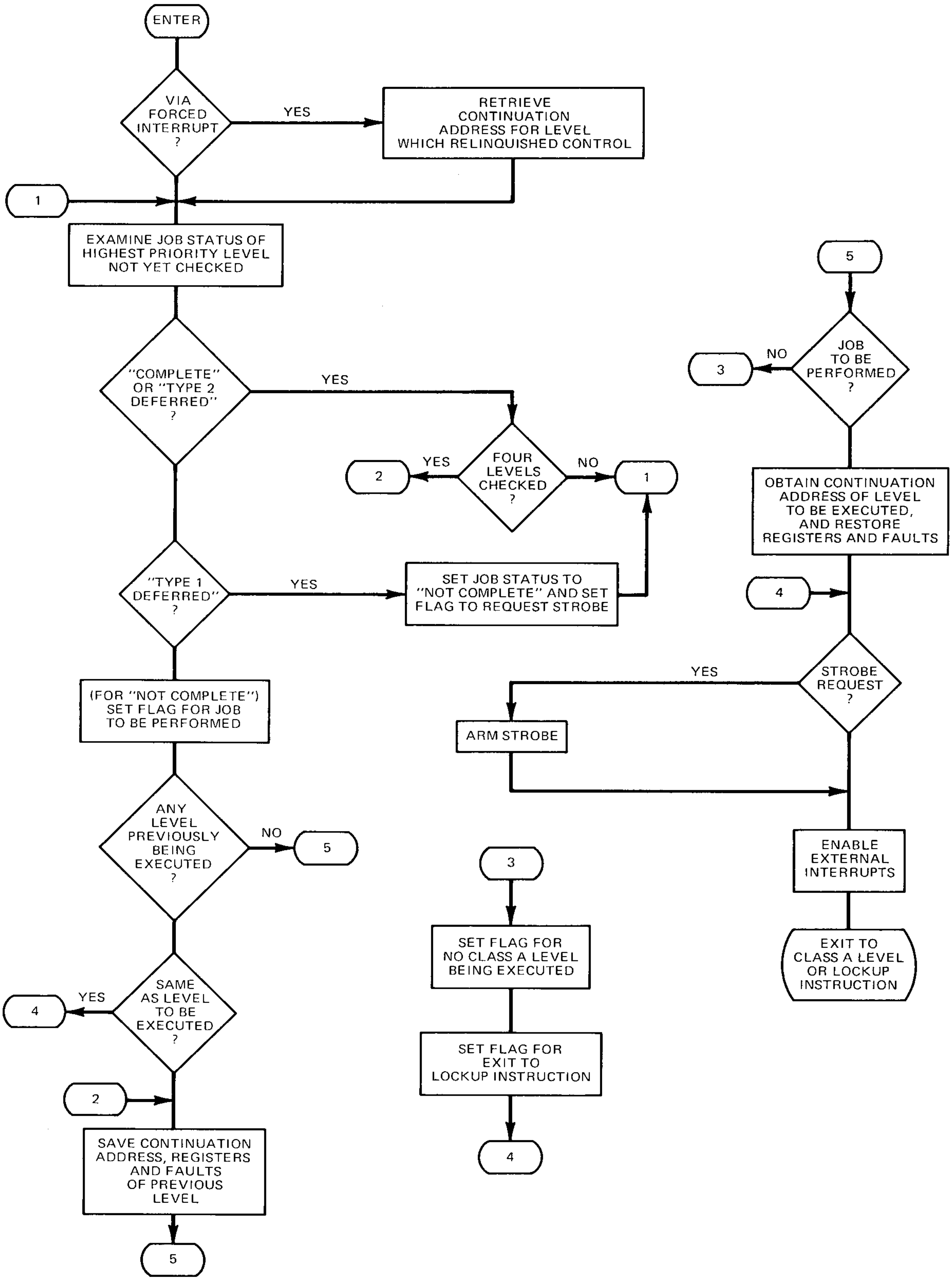

Fig. 6 - Flow diagram of the multilevel master executive (EXEC) routine

The highest priority level having a job status of "not complete" is selected for execution by the executive (no further levels are checked). If this level is different from the last Class A level executed, the continuation address of the latter level is stored in that level's sublist, along with the contents of all working registers and indications of any internal faults which may exist. The registers are then restored for the level to be executed, from that level's sublist, and any faults present when this level was last executed are reproduced. The continuation address for this level is also obtained from the sublist; if any "type 1 deferred" jobs were found, the strobe is armed and, after enabling external interrupts, an exit is taken from the executive.

It may happen that after a level has requested a "complete" or a deferred status, none of the four levels will have a "not complete" status. In this case interrupts are again enabled (after possibly arming the strobe) and a jump is taken to a lockup instruction. The computer then does nothing until a sample interrupt or strobe interrupt occurs (except for perhaps executing certain Class B routines to process other interrupts). Either of these interrupts causes execution of EXEC again, at which time some Class A level will always be ready for execution.

The two routines shown in Fig. 7 (which have been combined into one flow diagram for simplicity) are routines used by all Class A levels to return control to the executive. The ENDOFJOB routine is used when a Class A level has completed its current job and demands no further execution time. In this case, the level will not be activated again unless its job status is changed from "complete" to some other status, either by a Class B routine within the real-time system or by software on another Class A level. The HOLDJOB routine is used when a given level wishes to continue execution at some future time, but has no function to perform at the present time. This is used most often while waiting for an I/O operation to be completed, in order to give execution time to lower priority levels.

Both routines first inhibit external interrupts from being processed. The continuation address of the level calling the routine is stored in a special cell, to be picked up by the executive when it is entered. The level's job status is set to "complete" by the ENDOFJOB routine, and to one or the other type of deferred status, depending on an input parameter, by the HOLDJOB routine. The forced interrupt is then selected, and the inhibition on interrupt processing is removed. Logic flow is then held up until the interrupt occurs and the executive routine is entered.

Class A Levels

The four program levels incorporated in the NACTAC system are referred to as the antenna control, antenna processing, monitor processing, and data processing levels. The levels are here mentioned in the order of standard priority, although this priority scheme can be changed by any level if unusual requirements are present.

The antenna control level software has two main functions. First, it implements operator commands made via the various control consoles and is responsible for maintaining most of the displays on these consoles. Second, if the digital mode of antenna control has been requested, this software computes the azimuth and elevation commands for each sample point, in many cases using information obtained from the antenna processing level. These commands are then modified in accordance with operator requested options and predefined limits on the antenna's motion.

The antenna processing level is used to provide basic position commands to the antenna control level as x, y, and z coordinates of points in space. These commands are computed using the data for the particular object being tracked, which has been previously stored in one of several data tables.

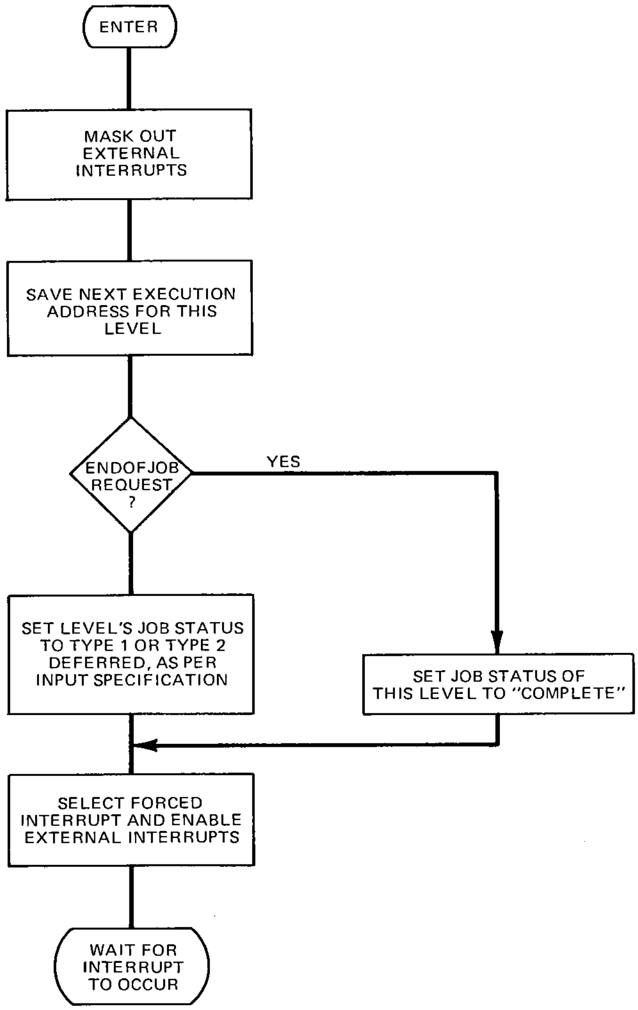

Fig. 7 - Flow diagram of HOLDJOB and ENDOFJOB routines used by all Class A levels to return control to the executive

The basic software residing on the monitor-processing level is NACSYS, the resident monitor for the nonreal-time system. The function of the resident is the same in the real-time system as in the nonreal-time system. That is, it executes various operator requests input via the console typewriter. These requests include both the execution of the resident's intrinsic routines and the loading and executing of external programs from the system library tape.

When the NACTAC program is first loaded and executed it sets a flag in the resident, informing the resident that real-time operation is in progress. Upon sensing this flag, the resident resets all I/O switches to short subroutines. This changes the course of action in the status checking routines so that, instead of constantly looping through a sense instruction while waiting for a ready status on any particular peripheral equipment, the HOLDJOB routine is entered. The monitor processing level is here given "type 1 deferred" job status. Then, when EXEC is entered, the strobe will be armed and the monitor processing level's job status changed to "not complete." When the strobe interrupt occurs 1/128 sec later, control is returned via EXEC to the monitor processing level, if this level is still the highest priority level with a "not complete" job status. If it is not, it must wait until higher priority levels are completed. The effect of the above procedure is that, if an I/O operation is in progress, the peripheral equipment involved is checked for a ready status no more often than every 1/128 sec. The purpose of this is to give the intervening time over to use by the data processing level.

Execution of real-time programs on the monitor-processing level is accomplished as in the nonreal-time system, i.e., an unconditional jump is taken to the execution address of the program after it has been loaded into available memory. When the program has completed its task, a jump is taken back into the resident to process further requests. Unless special operator action is taken, subsequent programs will be loaded into the same core area as the previous program, thus destroying the latter routine.

The data processing level may be used to execute routines which need not have CPU time on a regular basis. Under normal operating conditions, this level is allotted execution time only when the monitor processing level is in the process of an equipment status check, either waiting for a specific I/O operation to be completed or sensing for typewriter input.

When a request is being input on the typewriter (that is, after an initial character has been typed), or when a program or intrinsic operation is being performed on the monitor processing level, a flag is set in the resident. This flag indicates to the data processing level software that no I/O may be initiated by that level at the present time. The only time this level may initiate an I/O operation is when the resident is waiting for the first character of a typed request.

Since all programs loaded by the resident in the real-time system are executed on the monitor processing level, a special procedure is needed to establish a program on the data processing level. The first few instructions of the program are coded to set the job status of the data processing level to "not complete" and the continuation address of this level to the starting location of the data-processing-level coding. Certain cells in the resident defining the limits of available memory must also be changed so that any subsequent programs loaded by the resident will not overlay the data-processing-level software. A jump (on the monitor processing level) is then taken back to the resident, which resumes its normal function. The data-processing-level program will then be given execution time by EXEC whenever none of the three higher priority levels have "not complete" job status.

Class B Routines

They are five Class B interrupt-processing routines in the NACTAC system in its present form. One of these, the multilevel executive routine, has already been discussed. Two other Class B routines will be presented in later sections of the report. One of these processes the digitized antenna position and outputs commands to the antenna servo (see Digital Control of the Antenna); the other routine processes antenna receiver signals during the autoacquisition and autotrack sequences (see Appendix D).

The first of the remaining Class B routines is the sample-interrupt processing routine ANTCNTRL which is summarized in Fig. 8. This routine first buffers data blocks to and from the control consoles. The data blocks provide for communication between the antenna operator and the computer. By means of them, information is displayed at the control consoles, pushbuttons are lighted, and operator requests are sensed.

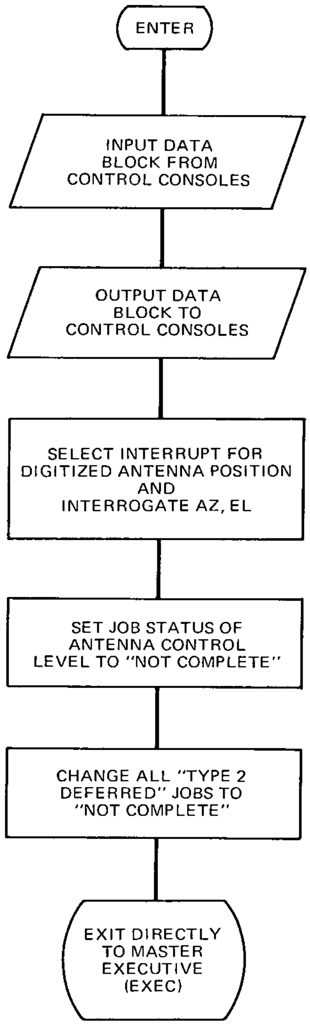

Fig. 8 - Flow diagram of the sample-interrupt processing routine ANTCNTRL, one of five Class B interrupt-processing routines used in the NACTAC system

The computer program then interrogates the antenna position encoders via the interface hardware (2) and enables interruption by the digital encoders once digitalization has been completed. The job status of the antenna control level, which is always "complete" at this point, is set to "not complete." This prepares the system to execute the series of routines comprising the antenna control level during the present sample period.

Finally, the job status of any level which is currently "type 2 deferred" is changed to "not complete." This fulfills the function of the "type 2 deferred" request to the HOLDJOB routine, which is that the level be executed again during the next sample period. An exit is then taken directly to EXEC (with no intervening interrupt), which then passes control to the highest priority level which has a "not complete" job status. This will normally be the antenna control level.

The last of the five interrupt routines is that which processes the subsample pulse interrupt. The only action taken in this routine is to interrogate the antenna position encoders and enable interruption by the encoders, as is done in the sample-interrupt routine.

DIGITAL CONTROL OF THE ANTENNA

The following discussion assumes that the operator has selected the digital control mode designated "program" mode, which is frequently used in connection with digital control.

The first stage in the computation of antenna pointing commands is performed on the antenna processing level. This involves the calculation of an object's x, y, and z site-referenced coordinates at "basic computation points." When an object has been selected for program tracking and the program mode is first requested by the operator, five such points are calculated. They are separated by a constant time interval which is selected according to the type of object (moon, planet, etc.) and, in the case of artificial satellites, the values of eccentricity and mean motion. The time interval is 10, 20, or 60 sec.

The time of the second computation point corresponds to the last Greenwich Mean Time (GMT) which is an integral product of the time interval selected, plus 1/8 sec. For example, if a 10-sec interval were selected, and the time at which the computation process begins were 16:30:35, the five points computed would be for the times 16:30:20.125, 16:30:30.125, 16:30:40.125, 16:30:50.125, and 6:31:00.125.

The antenna processing level is not activated again until the GMT equals the time of the third computation point minus 1/8 sec. This condition is checked during each sample period on the antenna-control level. In the example above, this condition would occur at 16:30:40, at which time the earliest position is discarded. The antenna processing level is then set to "not complete;" when this level is executed it will compute a new position for the time 16:31:10.125.

The next step in the computation process is accomplished on the antenna control level. During each sample period this level is activated to compute the coordinates of the object at a "sample point." The time corresponding to this point is 1/8 sec, or two sample periods, ahead of the present time. That is, if the present time is 16:30:40 plus 1/16 sec, the position of the object is computed for the time 16:30:40 plus 3/16 sec.

The computation of this position is accomplished by means of a four-point interpolation of the x, y, and z coordinates, using the results of the four earliest basic computation points which are currently available. The type of interpolation used, either 2nd- or 3rd-order Bessel, is determined by the same parameters used in selecting the time interval for the basic points. The calculated position is transformed to spherical coordinates of range, azimuth, and elevation, and then modified for operator-requested offsets and scans; it may be further modified to conform to predefined limits on the antenna's position and velocity. The resultant position is saved as the antenna command position.

The final computation is accomplished in the Class B routine ANTENNA, which processes the interrupt denoting that the actual antenna position has been digitized and is ready for input to the computer (Fig. 9). In this routine the antenna's encoded azimuth and elevation are corrected, if necessary, for gross encoder errors. This is done by comparing the encoded angles with predicted angles based on the antenna's motion during the last one or two subsample periods.

If the last subsample pulse occurred at the beginning of a sample period, the last command position computed by the antenna control level (which at this time is the sample point command for 1/16 sec hence) is transferred to special locations to be used for interpolation during the present sample period; the command position formerly occupying these locations is moved to storage cells containing the command position for the present sample point. This command, which was calculated two sample periods in the past, is compared to the encoded angles, and position errors are computed and output to the antenna servo.

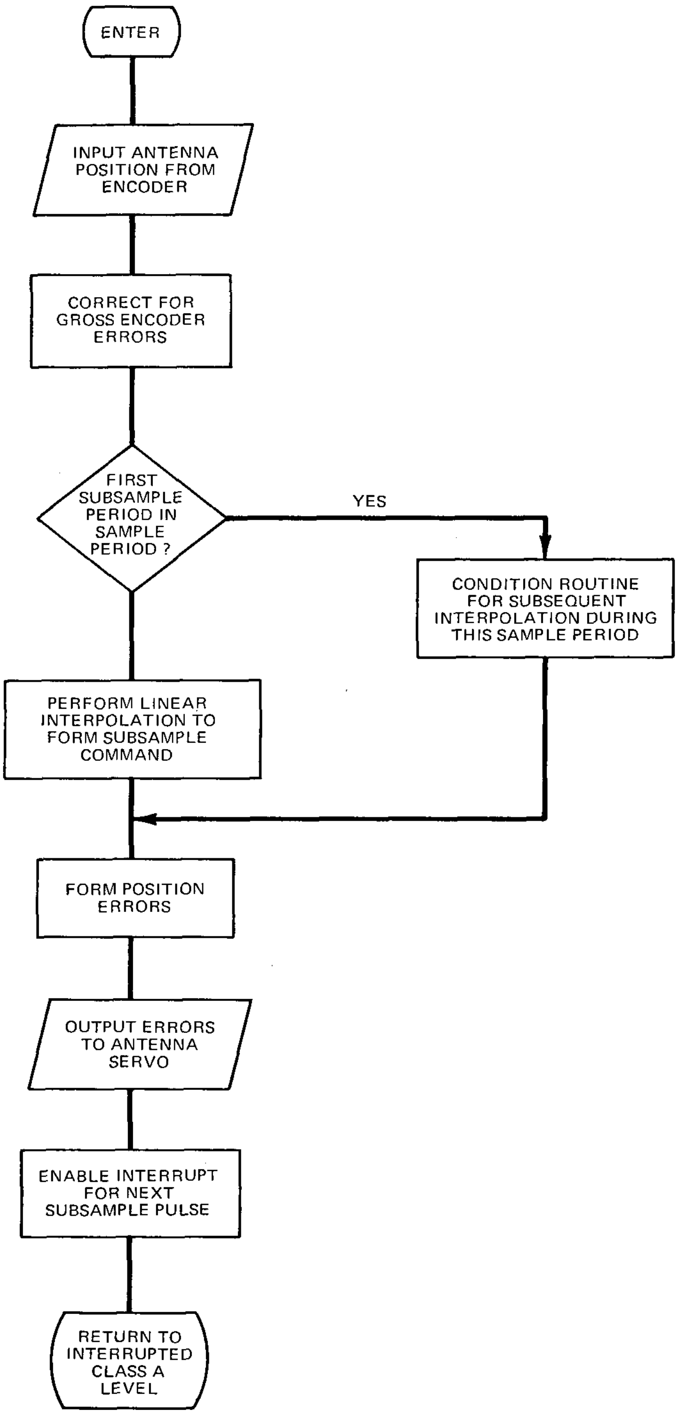

Fig. 9 - Flow diagram of the Class B interrupt-processing routine ANTENNA for digitized antenna position

When ANTENNA is executed during a subsample period which is not the first in a sample period, the subsample command position is calculated by linear interpolation between the sample period commands for the last and next sample points.

A detailed example of the interactions between the Class A and Class B software involved in digital control of the antenna is given in Appendix C.

ANTENNA CONTROL LEVEL

Execution of Operator Commands

Figure 10 outlines the software routines making up the Class A antenna control level. The names enclosed in parentheses on Fig. 10 will hereafter be used to refer to the routines.

The antenna control level assumes its primary function of digital control of the antenna only if the digital mode of antenna control has been selected by the operator. If this is not the case (nondigital mode), the operator may use this level to perform either of two basic functions. The first is control of the automatic acquisition and automatic tracking sequences (Appendix D), which may also be carried out by the software if the digital mode has been selected. The second function allows the operator to obtain displays of the positions of a predefined object at designated times. This function is detailed under the heading GROUP 8 in the present section.

When the digital mode has been selected, the operator may choose any of three submodes by which to control antenna motion:

1. Designate right-ascension/declination—The antenna moves to the designated position and tracks at sidereal rate.

2. Designate azimuth/elevation—The antenna moves to, and holds, the selected position.

3. Program track—The antenna will track an object for which data has been entered into the system.

In either digital or nondigital mode, the antenna processing level maintains communication with the antenna operator. Commands are input to the computer by pushbuttons on the control consoles. The only direct result of pressing a pushbutton is that a binary bit in one of three "system status" words is changed from a 1 to a 0. The system status words are contained in the data block input to the computer by the ANTCNTRL routine at the beginning of each sample period. The software then becomes aware of the command and takes appropriate action; in addition, binary bits in three "computer status" words are set to light or turn off various pushbuttons so as to inform the operator that action has been taken. The computer status words are part of the data block which is output to the control consoles by ANTCNTRL the next time it is executed.

Operator commands are divided into groups, each group being comprised of commands associated with a particular function or option. Commands within a single group are processed by a single software routine, generally one of the routines shown in Fig. 10. The action taken within one of these routines is controlled by a software switch which is set to different locations depending upon the functions to be executed within that command group. A command group is "cleared" if the software switch in the associated routine is set for an immediate exit so that the routine does nothing.

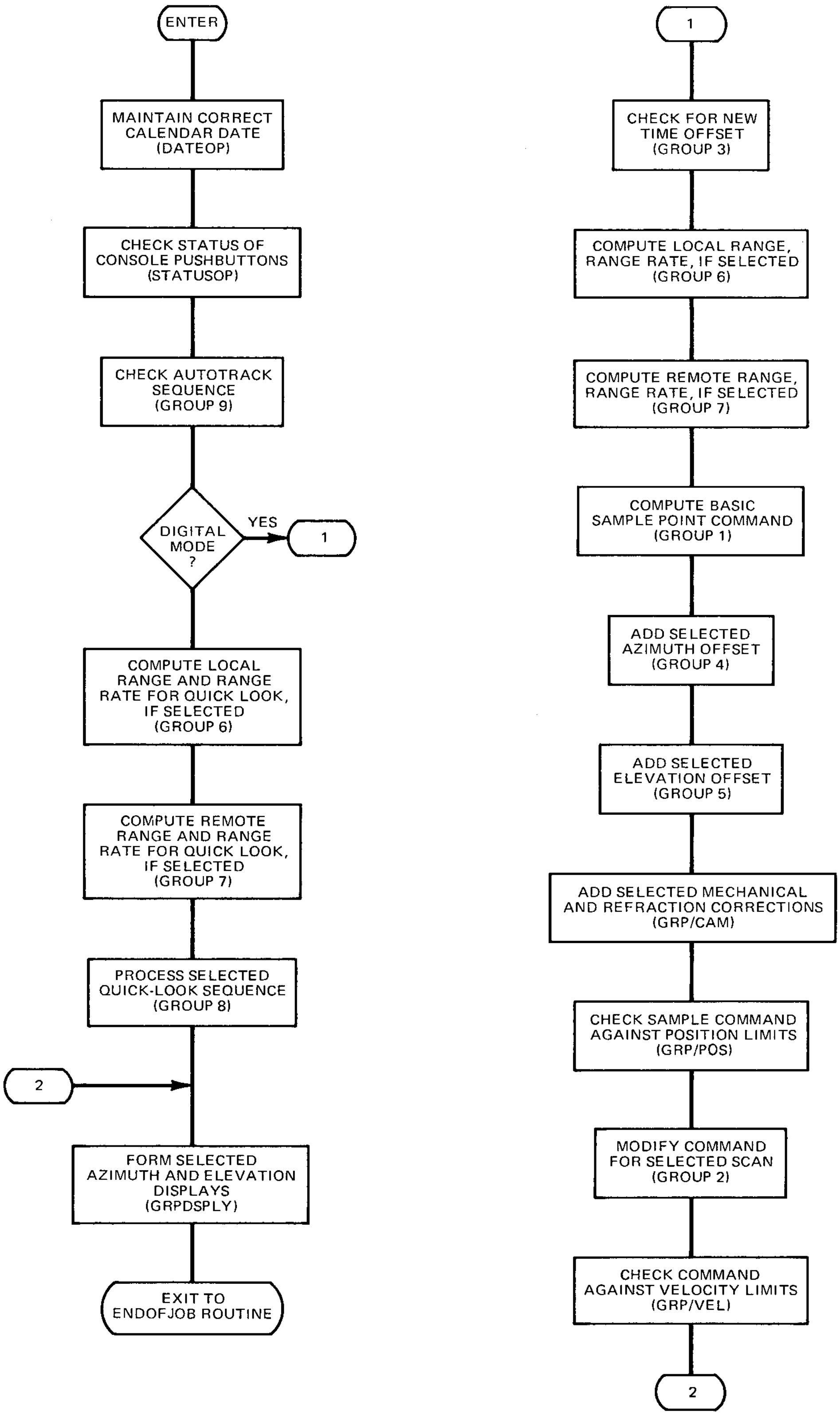

Fig. 10 - Flow diagram of the software routines making up the Class A antenna control level

The function of the pushbuttons, then, is to provide for the setting of the software switches on the antenna control level so that operator commands may be executed. (The operator, of course, need not concern himself with this sequence of events—from his point of view, a pushbutton is pressed to make the *antenna*, not the computer, do something.)

Additional information, such as the values of offsets or scan parameters, is needed to execute some operator commands. Such information is input via thumbwheel switches located on the control consoles. The settings of these switches are included in the input data block along with the status of the pushbuttons; however, no attention is given this data except by the particular routines which need the data to execute a command.

Certain operational parameters are defined by a three-digit octal number called the "program number." This, too, is set by thumbwheel switches and included in the input data block. When an "enter program number" pushbutton is pressed, the current setting of these switches is saved by the software. The program number is used to select several different types of options; by this technique a large number of pushbuttons, which would otherwise be required, is made unnecessary.

There are thus three steps in the execution of an operator command:

1. Hardware indication of the command is input to the computer in the system status words.

2. The command is sensed, and a software switch is set in the appropriate routine for this command group.

3. The latter routine performs the required operation, using additional information contained in the input data block if necessary.

Software Routines

DATEOP—In this routine the GMT is checked to find if a new day has begun. If so, a block of cells containing the current month, day, and year is updated, the Modified Julian Day number* for the new date if set, and the sidereal time is computed corresponding to zero hours universal time of the new day.

STATUSOP—This routine, outlined in Fig. 11, performs the second step in the execution of operator commands. If a command is indicated by one of the system status words, and if the same command was not indicated during the previous sample period (the operator may not have yet released the pushbutton), it is determined whether the command is valid, based on the current configuration of the system. This is intended to preclude the possibility of the antenna operator making a gross error in system operation. If such an error has been made, either no further action is taken by the routine, or the pushbutton the operator has pressed will be flashed on and off indicating that his command cannot be executed. If a valid command has been made, the software switch in the associated command group is set to the correct address for execution of the command, which will then be executed on the antenna control level later on during the present sample period.

The above sequence of events can also be initiated without operator intervention. If the system assumes certain characteristics which dictate that action be taken, other parts of NACTAC can insert dummy status words in such a way that it appears to STATUSOP as if the operator has made a request. For example, if a satellite is lost during an

*The Modified Julian Day number is a more convenient form of the Julian Day number. It is equal to the Julian Day number, less 2400000.5. The Julian Day begins at noon; the corresponding Modified Julian Day begins at the previous midnight.

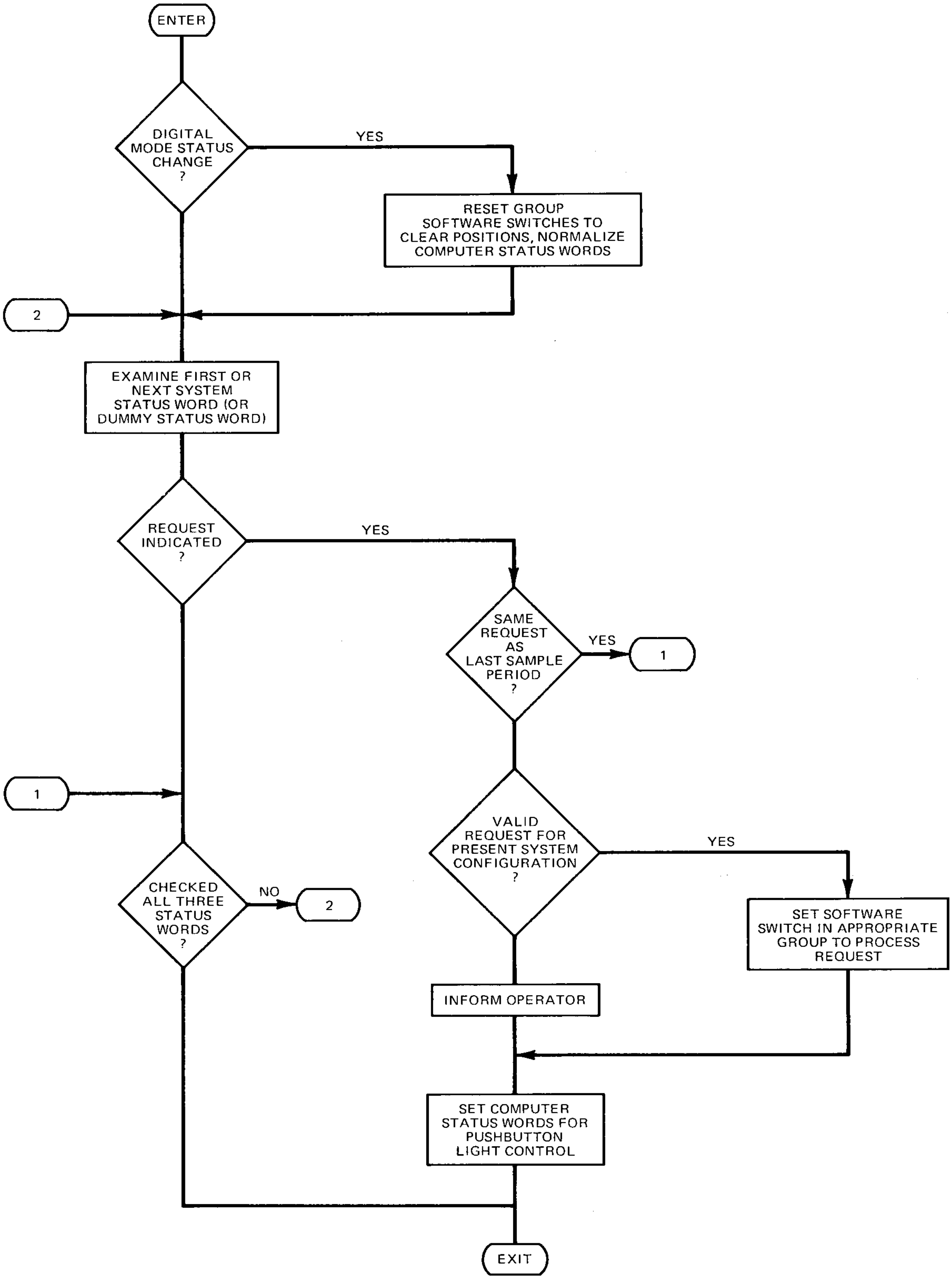

Fig. 11 - Flow diagram of STATUSOP routine

automatic tracking sequence, the portion of NACTAC which senses this can, through the use of a dummy status word, command STATUSOP to start various types of scans in an attempt to reacquire the satellite.

GROUP 9—The GROUP 9 routine is the Class A level software forming part of a complex network of programming embedded in NACTAC. This programming enables the operator to select automatic acquisition and automatic tracking sequences when working with certain artificial satellites. In practice, these sequences have been seldom used, and the usual way in which NACTAC functions can be explained without referral to this software. The details of the autoacquisition and autotrack network have therefore been placed in Appendix D.

GROUP 3—The next routine executed on the antenna control level is the time offset routine. Normally, the software switch for this group is set for no action. Only in the sample period during which one of the pushbuttons associated with this group has been processed by the STATUSOP routine is any action taken.

In this case the time offset to be added or subtracted is obtained from the input data block and converted to the formats used for universal and sidereal time. The program-track digital submode is then set to restart; the time offset selected will then be used in the computation of new basic antenna positions by the antenna processing level.

GROUP 6—The action taken in this and the following routine is summarized in Fig. 12. The GROUP 6 software computes and stores output data associated with the local (Waldorf) site antenna, if requested by the operator. The first type of data which may be computed for display is the range and range rate of the object being tracked. This is done by interpolation, using four-point range and range-rate tables established by the antenna processing level. When this request is first received, the program track submode is restarted as in the time offset group; this action is an indirect means of establishing the above tables since they are not set up by the antenna processing level if they have not been requested. The type of interpolation used is dependent upon the type of object being tracked and/or the characteristics of the orbital elements for artificial satellites.

The second type of data available from this routine is the free-space attenuation and doppler shift at a particular frequency for the object being tracked. This data is derived from values for range and range rate. Therefore, range and range-rate tables are established when either type of data is requested. The frequency used must first have been input to the system on the typewriter, through use of a real-time program from the library tape. Up to three distinct frequencies may be stored within the system using this program. The operator selects the frequency he wants by means of pushbuttons on the main or test control consoles.

This and the following routine may be executed even if the system is not in digital mode, as indicated by Fig. 10. This usage is associated with a special option which is discussed under the heading GROUP 8 in this section.

GROUP 7—This routine computes the same type of data as the previous one, the only difference being that data is computed for a remote site rather than the Waldorf site antenna. That is, the range and rate rate, or free-space attenuation and doppler shift, for a particular frequency are computed for the object being tracked and a remote location. Site information must have been entered into the system through previous use of a real-time program. Data for up to three sites may be stored within the system. The site for which information is to be displayed is then designated by the appropriate pushbutton on the main or test control consoles.

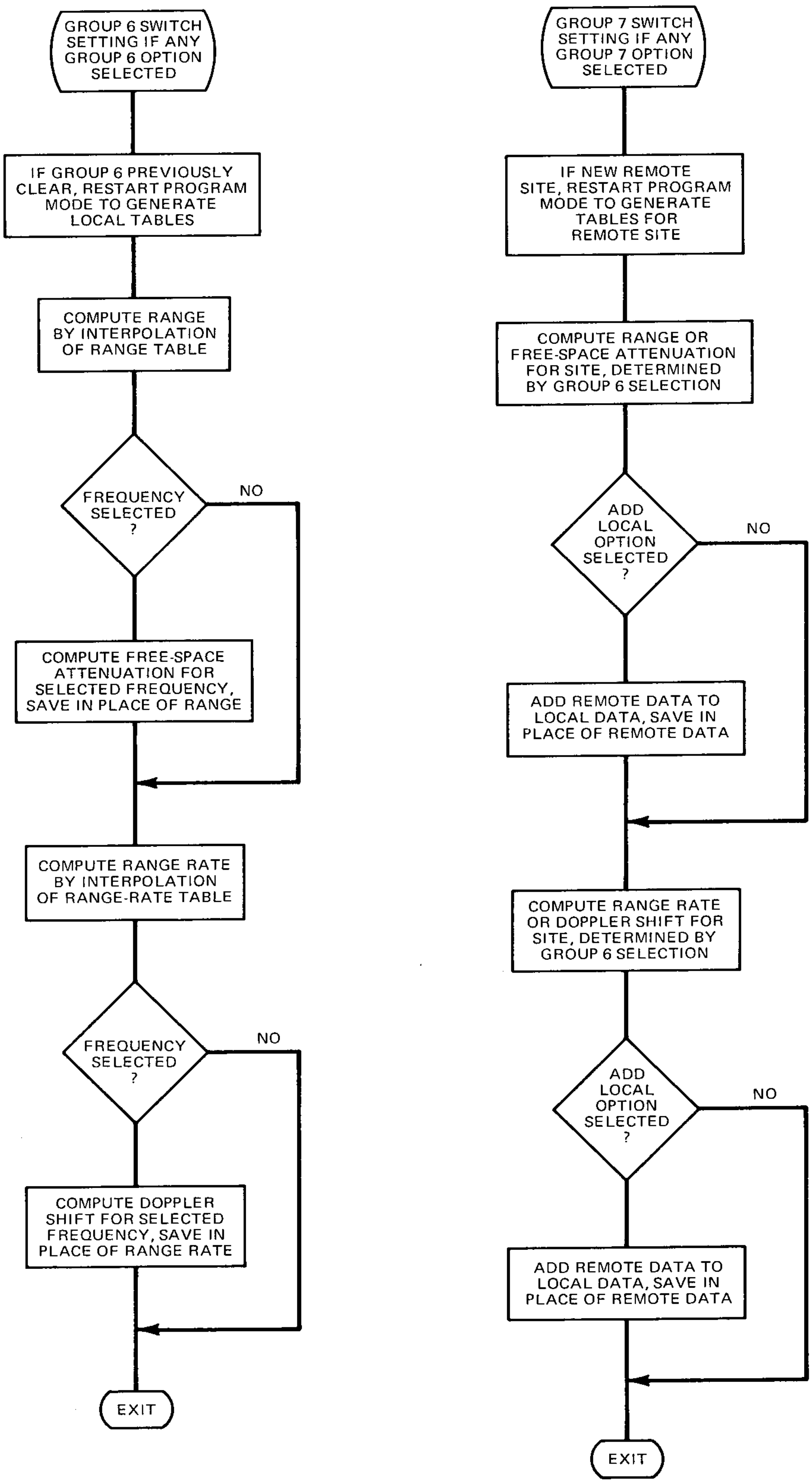

Fig. 12 - Flow diagram of local and remote range and range-rate subroutines, designated as the GROUP 6 and GROUP 7 routines

The operator may also request that the local and remote site data be summed and the resultant displayed, rather than the remote data itself. This is equivalent to computation of the data over the path from the local site to the object to the remote site.

GROUP 7 is prohibited from functioning if no GROUP 6 option has been selected. This is because the type of data requested is defined by pushbuttons associated with GROUP 6. GROUP 7 pushbuttons simply request the same type of data for a remote site as has been requested for the local site.

GROUP 1—The software in this routine is responsible for forming the basic sample period commands when the system is in digital mode. There are three submodes used in the digital mode. The most frequently used of these submodes is the program-track submode. In this submode, the operator designates a particular object to be tracked, for which tracking data has been previously entered in the system. The first time that GROUP 1 is entered after processing the operator's request for the program submode, the basic computation period and type of interpolation are established. If the object to be tracked is an artificial satellite, the basic computation period is set for either 10 or 20 sec, the latter value if the satellite's epoch mean motion is less than 4 revolutions/day. The type of interpolation used is either second- or third-order Bessel interpolation, with the latter being chosen if the satellite's epoch eccentricity is greater than or equal to 0.5. For other types of objects, the computation period and interpolation type are set to 1 min and second-order Bessel, respectively. Having established these parameters, the routine sets the antenna-processing-level job cell to "not complete" and requests five initial basic computation points for the object. This operation will commence when the antenna control level is complete for the present sample period.

In succeeding sample periods, the GROUP 1 software checks a flag to ascertain whether the five initial points have been computed by the antenna processing level. If not, antenna commands are generated which coast the antenna along its previous velocity vector, and an exit is made from the routine.

The object's position at the initial five computation points are given in x, y, and z coordinates in a reference system centered at the local site. If information has been requested from the GROUP 6 routine, range and range-rate tables will also be established for times at the basic computation points. Only the four earliest entries in the tables are used for interpolation purposes.

When computation of the five points has been completed, the GROUP 1 software switch is set to a new position, and the logic outlined in Fig. 13 is thereafter executed during each sample period. The GMT is first checked to find out if it is time for another basic sample point to be computed. If so, the antenna processing level is set to "not complete" to accomplish this task. The tables containing the x, y, and z coordinates of the object (and local and remote range and range-rate tables, if present) are then updated. That is, the earliest entry is discarded and the second through fifth entries are used, while new fifth entries are being computed by the antenna processing level. The sample-point interpolation argument is set to zero.

During succeeding sample periods, the interpolation argument is incremented by the appropriate amount, depending on the computation interval being used, and the sample-period command is formed by the interpolation method being used.

A second digital submode is called "designate right-ascension/declination." When this request is processed, the sidereal time is computed for the current date and time plus a time lead of two sample periods. The settings of the right-ascension and declination thumbwheel switches are then obtained from the control console input block and converted to their binary equivalents. The GROUP 1 software switch is set to execute the routine shown in Fig. 14.

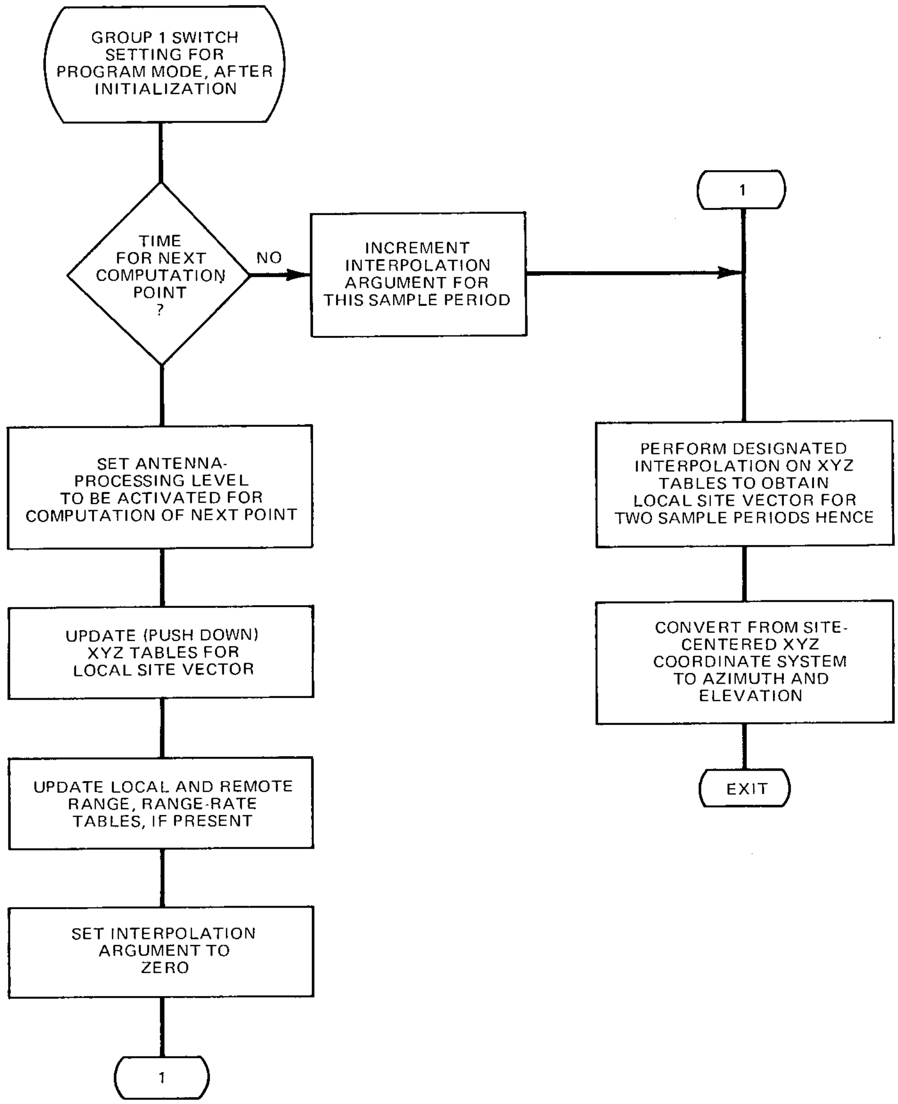

Fig. 13 - Flow diagram of the program track submode, part of the main digital-mode GROUP 1 routine

In each sample period the current sidereal time is added to the designated right ascension to obtain the Greenwich hour angle. The hour angle and the designated declination are converted to x, y, z coordinates in an Earth-centered orthogonal coordinate system. These coordinates are then translated to a site-centered x, y, z system, and finally converted from this system to local azimuth and elevation. The latter coordinates become the basic sample-period command.

The final digital submode is called the "designate azimuth/elevation" submode. To execute this command, the designated coordinates (input via thumbwheel switches) are converted to binary form and saved as the basic sample-period command. The GROUP 1 software switch is then set for no action in succeeding sample periods, since this command is to remain unchanged until further operator action.

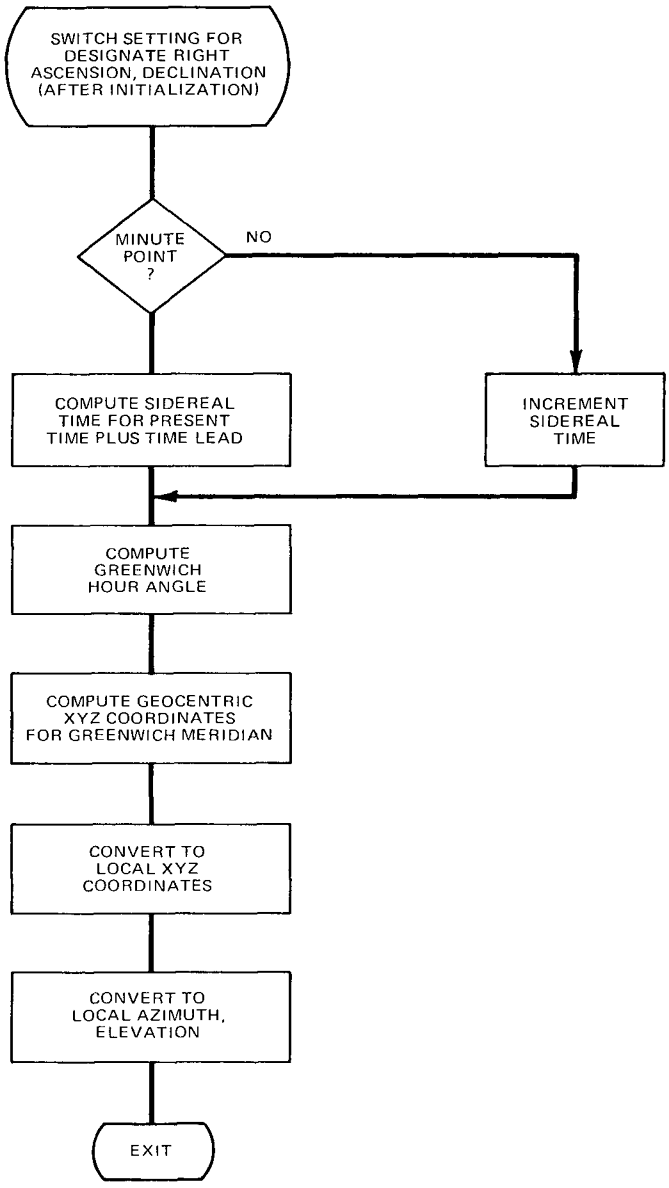

Fig. 14 - Flow diagram of the "designate right-ascension/declination" submode, part of the main digital-mode GROUP 1 routine

GROUP 4—This routine processes commands for adding or subtracting operator-designated angular offsets to the basic sample-period azimuth command. When an offset is first requested, a check is made to ascertain whether the offset would initially generate an azimuth command outside the azimuth limits of the antenna. If so, the operator is informed by a flashing light that his request is unacceptable. If the request is acceptable, the azimuth offset is saved, and subsequently combined with each basic sample-period azimuth command formed by the GROUP 1 routine.

GROUP 5—The GROUP 5 software is exactly analogous to the above routine, the only difference being that this routine processes desired offsets to the sample-period elevation command.

GRP/CAM—Action taken in the GRP/CAM routine is controlled solely by the program number thumbwheel switches, unlike most of the previous antenna-control-level routines which are controlled by the software switch setting associated with the routine. The routine provides options for three corrections to the antenna command.

The first of these modifies the elevation command to correct for errors due to atmospheric refraction. Temperature, barometric pressure, and relative humidity must have been previously entered into the system via the typewriter. The program REFR, which initiates requests for this data, sets up two tables of the elevation corrections within the NACTAC system. The primary table contains corrections for thirty-two equal intervals from 0 to 90 degrees, while the secondary table contains corrections for eight intervals from 0 to 2.81 degrees. If the command elevation is in the latter range, the correction for the actual elevation is calculated using a second-order Bessel interpolation and the entries in the secondary table. In the former range, the primary table is used, with second-order Bessel interpolation being utilized if the command is below 22.5 degrees; otherwise, linear interpolation is used.

The other two corrections are intended to provide freedom from basic pointing errors in the antenna due to mechanical inaccuracies. Data necessary for these corrections is assembled with the NACTAC program—it needn't be entered each time the system is used. This data is the result of pointing accuracy tests conducted over the full hemispheric coverage of the antenna. The azimuth correction data is stored in a table of sixty-four intervals over the range 0 to 360 degrees, while the elevation data is stored for thirty-two intervals from 0 to 90 degrees. In both cases, linear interpolation is used to find the correction for the current command.

GRP/POS—The GRP/POS routine, like GRP/CAM, is controlled by the program number. The cam-corrected antenna commands are checked against predefined azimuth and elevation limits.

There are two special options the operator may select. The first of these is the selection of wide limits, which may only be done while the antenna is in the "designate azimuth/elevation" digital submode. If wide limits have been selected, any elevation command between -2 degrees and +92 degrees will be declared within limits. Otherwise, the elevation limits are 0 and 90 degrees.

Azimuth limits are imposed on the system by the configuration of the various electrical cables which pass through the antenna's primary (azimuth) axis. The maximum rotation of the antenna about this axis is determined by the amount of slack which the cables have. Normal azimuth limits have been set at ±405 degrees—that is, from the central azimuth position of the antenna, the antenna may be moved in azimuth through 1-1/8 revolutions either clockwise or counterclockwise. Wide azimuth limits are set at ±450 degrees, or 1-1/4 revolutions in either direction.

Identical checks are performed on both the azimuth and elevation commands. If the command is found to be within limits, it is not modified; however, if the command is outside the limits, the position-limited command computed during the preceeding sample period is used for the present sample period also, effectively stopping antenna motion around the axis concerned. In addition, several pushbuttons on the control console are flashed and the GROUP 1 software switch is set to the "clear" function to prevent generation of any new sample-period commands.

An exception to the above sequence is made if a "limit hold" option has been selected. In this case, the position-limited command is set to the limit exceeded and operation proceeds normally. This option is useful if, for example, it is desired to begin tracking an object as soon as it moves above the horizon. In this case, the elevation commands generated while the object is below the horizon will be changed to 0 degree, but as soon as elevation commands greater than 0 degree are generated, indicating the object has moved above the horizon, the antenna will begin tracking.

GROUP 2—The function of this routine is to add operator- or computer-selected scans to the position-limited sample-period command. Spiral and elliptical scans have been provided.

Scan parameters for operator-initiated scans are entered via thumbwheel switches on the main control console. For an elliptical scan, which may be requested in the "designate right-ascension/declination" and "designate azimuth/elevation" digital submodes, the scan parameters include the shape of the ellipse, the length of the semimajor axis (in degrees), the scan rate (in degrees per second), and a velocity designate vector. The latter parameter defines the expected rate and direction (with respect to the horizontal axis) of the object being searched for. This vector is used directly only if an automatic acquisition sequence is requested in addition to the elliptical scan, as explained in Appendix D. Indirectly, however, it is used to define the inclination of the major axis of the ellipse to the horizontal axis. The two axes are defined to be perpendicular, so that the ellipse extends as far as possible on either side of the velocity vector.

Spiral scans may be requested in any digital submode. The input parameters are the spiral increment (separation between loops of the spiral) and the amplitude and rate.

If a scan is being performed, the first action taken in the routine is to check the position-limited command and the scan amplitude to find if the antenna's position limits may be exceeded by addition of the scan offset. If this is the case, the scan amplitude is reduced (for this sample period only) to an acceptable value. The azimuth and elevation amplitudes of the scan used in this check are recomputed each sample period for a spiral scan, depending upon the current status of the spiral.

The scan position is then computed for the current sample period; this is done by updating a phase (angle) accumulator each time the routine is executed. The resultant azimuth and elevation offsets produced by the scan are then added to the position-limited sample-period command.

The scan may be held and continued at the operator's discretion. A provision is also made for an automatic scan offset option. This feature, when a "hold scan" pushbutton is actuated, will automatically add the current offsets produced by the scan to any existing azimuth and elevation offsets which may be stored in the system. If no such offsets exist, it enters these as new offsets. The scan group is then cleared, and the combined offsets will be subsequently added to the sample-period commands by the GROUP 4 and GROUP 5 routines.

GRP/VEL—The final modification of the sample-period command is performed by this routine. The difference between the scan-modified command produced during this sample period and that computed during the previous sample period is found and checked against predefined velocity limits stored within the system. (The velocity limits are not under the operator's control.) This is done for both the azimuth and elevation commands. If the velocity limit is exceeded, the command position is changed so as to conform to the limits. The velocity-limited command is then stored as the final command position for two sample periods hence.

GRPDSPLY—This is the final routine executed on the antenna control level. It formats various information available in the system for output to the main azimuth and elevation displays. The actual information displayed is selected by the operator via the program number.

Decimal representations of six different data sets may be selected for display. Each data set consists of two angles, one associated with azimuth, the other with elevation. Among these data sets are included the antenna position, antenna velocity, and the position error output to the antenna servo mechanism.

GROUP 8—One additional routine is included in the antenna-control-level software. This routine processes operator pushbutton requests which are associated with a "quick-look" option. As shown in Fig. 10, it is only executed when the system is in nondigital mode.

The purpose of the routine is to provide a convenient means of ascertaining the position of an object at a specified date and time. After entering tracking data on the object into the system through use of a real-time program and directing the system to "track" the object, the operator presets the time of interest on thumbwheel switches at a test control console. He then presses a "set" pushbutton associated with this group. This is processed by the STATUSOP routine, which sets the software switch associated with GROUP 8.

When the routine is executed, it buffers into the computer the preset time. The universal and sidereal times for the present date and specified time are then computed and stored for use by the antenna processing level. Time increment storage cells used for this option are set to zero, and the antenna-processing-level job status is set to "not complete."

When the antenna processing level is executed by EXEC, it computes the local azimuth and elevation of the object for the specified time plus the time increment. The first time the GROUP 8 routine is executed after completion of this computation, the GRP/CAM routine is called to modify the position for operator-selected refraction and mechanical corrections. The azimuth and elevation are then converted for display. The GROUP 8 software switch is set for no action, and nothing further is done unless new requests are received from the operator.

At this point the operator may have the information he desires. On the other hand, if additional data is needed, he has two options. He may preset a new time on the thumbwheel switches and again press the "set" button. This results, as above, in the display of one computed position.

His second option is to enter a time interval on the thumbwheel switches and press an "increase" or "decrease" pushbutton. This will cause the GROUP 8 software to read in the time interval and store it in the special time increment cells as a positive or negative value. After setting the software switch for this group to a new value, the antenna processing level is again activated.

The GROUP 8 routine thereafter displays a new position once per second. The position displayed, computed by the antenna processing level, will be for the previous computation time plus the positive or negative time increment. Such action will continue until the system runs out of computation data for the object (for objects other than Earth satellites) or until the operator pushes a "hold" pushbutton. In the latter case, the time increment is set to zero, and although the software actually continues to compute a position each second, the display output remains unchanged.

The local and remote range and range-rate routines may also be used once a quick-look sequence has been initiated. As with the pointing angles, this data is updated once per second.

It must be emphasized that the quick-look option is used for informative purposes only and has nothing to do with digital control of the antenna. It may be used in nondigital mode only, and although the antenna may be moved in this mode, it is done entirely under operator control with the computer playing no role. In addition, it should be noted that although quick-look has much software in common with the "program" digital submode, there are many differences in the processes employed. The three main dissimilarities are these: (a) Under the program track submode, the antenna processing level is executed once every 10, 20, or 60 sec, while under quick look it is executed once every second. (b) The time increment between computation points under program track is equal to the interval at which the antenna processing level is executed. Under quick look, this increment is under operator control, and may range from 1 sec up to several hours. (c) For program track, the antenna processing level sets up four-point interpolation tables of x, y, z coordinates for the object and additional tables for local and remote range and range rate, if requested. Under quick look, no such tables are produced and the object's position is given immediately in terms of azimuth and elevation.

For line printer output of similar information, the program LOOK may be loaded and executed on the monitor processing level.

ANTENNA PROCESSING LEVEL

The logic incorporated on the antenna processing level is summarized in Fig. 15. For antenna processing level use other than quick-look computations, universal and sidereal times for a new basic computation point are computed. If an initial data request is present, five points are needed to establish the various tables. In this event the first value of time computed by the subroutine is for the first point in the table, which will be somewhere between one and two basic timing intervals in the past. The time computed will be incremented by two sample periods to satisfy the requirements of the antenna control level. Subsequently, whenever a new computation time is requested, it will be computed as the last computation time plus the time increment. Operator-requested time offsets, if any, are added to the computed time.

The tracking elements for the object being processed are updated for the new computation time. This is done differently for artificial satellites and celestial objects. For updating of satellite elements the data used includes the basic orbital elements of the satellite and the necessary time derivatives of these quantities. (The time derivatives used vary from first to third order.) The routine uses a Taylor series expansion to update the elements from the epoch of the information to the desired computation time. The updated elements produced include the mean anomaly, mean motion, right ascension of the ascending node, argument of perigee, eccentricity, inclination, and the semimajor axis.

For celestial objects, updating is performed by third-order Bessel central difference interpolations on tables of the object's distance from Earth, right ascension, declination, and the first-order derivatives of these quantities. Thus, if the computation time falls outside the range of these tables, no updating is possible. In this case the main digital mode group (GROUP 1) is set to be cleared and the antenna processing level is terminated. No analogous case arises from the updating of satellite elements, for although the accuracy of the updated elements may decay as the time from epoch increases, the method used insures that the quantities may always be computed. It is important to note, however, that although accuracy may decay, precision is maintained through use of double precision arithmetic in calculating the mean anomoly. Thus, even though computations of satellite positions sufficiently removed from the epoch may be inaccurate, precision, and therefore smoothness of antenna control, is guaranteed. Offsets may then be introduced in the previously stated manner to improve accuracy.

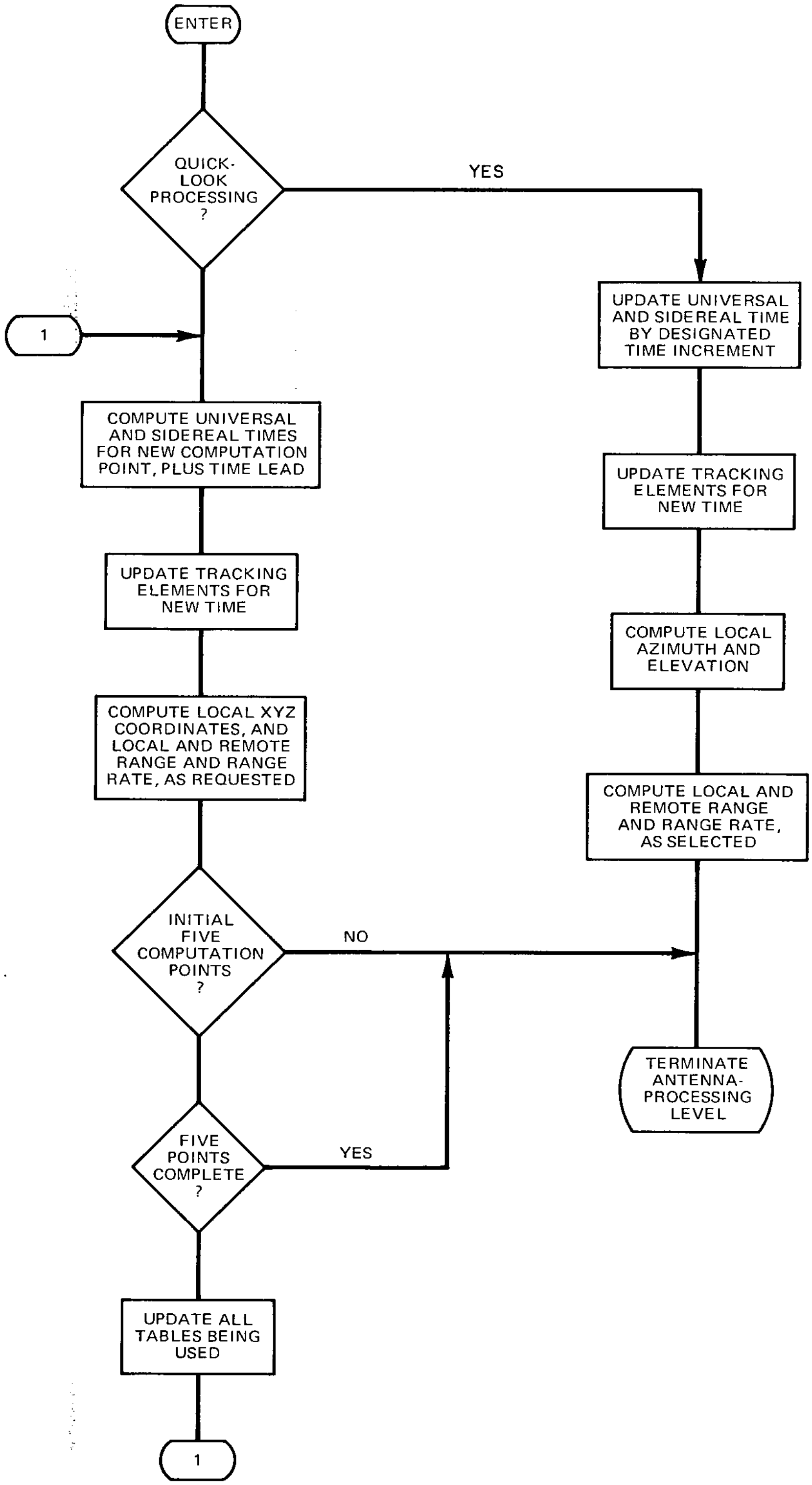

Fig. 15 - Flow diagram of the antenna processing level

The object's local x, y, z coordinates are then calculated. For an artificial satellite, the eccentric anomaly is calculated from the mean anomaly and eccentricity, using an iterative method. The updated elements of the object are then used to compute its position in a rotating orthogonal geocentric system. One axis of this system passes through the equator at 0 degree longitude, and a second axis passes through the North Pole. The coordinates thus derived define a vector from the Earth's center to the object. From this is subtracted a constant vector joining the Earth's center to the local site. This operation results in a vector pointing from the local site to the object.

The local range is then computed, if it has been requested. For a remote site range request, the vector from the remote site to the object is computed using previously entered data defining the site, and the range is calculated from the resultant vector. If requested, the range rate of the object is computed, after first calculating the derivatives of the object's position vector in the previous geocentric coordinate system. This rate of change is combined with the rate of change produced by the Earth's rotation to form the total range rate. The remote range rate, if selected, is computed in a similar manner. Finally, an orthogonal transformation is performed on the local site vector to convert to a site-centered coordinate system, producing the local x, y, z coordinates.

The local x, y, z coordinates and the ranges and range rates, if computed, are stored in positions corresponding to point five of their respective tables. If this was not a request for five initial points, these values will be left in those cells, to be incorporated into the actual interpolation tables (four points) at the next update time by the antenna control level software. On the other hand, if five points were requested, the computed values will be pushed down into the tables for each of the first four points, thereby establishing the complete tables initially.

The branch of the flow diagram showing the quick-look processing is quite similar to that already described and uses much of the same software. The main difference is that the x, y, z coordinates of the object are converted to the local azimuth and elevation by this antenna processing software, rather than by the antenna-control-level routine using this information in digital mode.

REAL-TIME PROGRAMS

Additional programs are necessary to make use of the real-time system. These programs are used to input various types of data to the system and store the data in special tables within NACTAC. Data is normally input from the system data tape, although the capability exists for punched card input.

Three of these programs input actual tracking data for various objects: ORBIT inputs artificial satellite data, PLANET reads ephemeris data into the system for the sun, moon, or any of the planets, and STAR inputs tracking information for various bright visual stars and radio sources. SITE is used to input positions of various remote sites available on the data tape. It is this data which is used in the computation of remote range and range rate.

Of the other real-time programs, two are used to input information directly from the typewriter into the system. FREQ may be used to input a desired frequency into any of three data blocks. This is used when a request for free-space attenuation and doppler shift is processed by the system. The REFR program accepts current weather data; from this, the index of refraction of the atmosphere is computed and the refraction table used by GRP/CAM is established.

The TRACK program is used to command the system to track one of the satellites or celestial objects previously input to the system. This program must be executed before attempting a quick-look operation or before requesting the "program" digital submode to begin operation. TRACK actually doesn't initiate antenna tracking, but rather prepares the system for this by defining the data block to be used by the antenna processing level in computing tracking positions.

LOOK is used to output to the line printer local tracking information for an object previously entered into the system. The start time, time increment, and number of data points desired are input by the operator. Tracking data for a previously defined remote site may also be requested. The azimuth, elevation, range, and range rate for the specified times are output, along with identical information for the remote site, if requested. The function of the program is similar to the quick-look option, but each has features making it more desirable for particular applications. In particular, output from LOOK is not confined to the present day, but may be obtained for any date for which position data for the object is available.

All the above programs are executed on the monitor processing level of the real-time system and are thus capable of being used while the antenna control and antenna processing levels are engaged in any of the functions they perform.

Two additional real-time programs may be mentioned. The first is SPECTRUM, a real-time counterpart of the SPECTRAL program mentioned under the nonreal-time system. The second is DAM (Data Acquisition Module), which operates on both the monitor processing and data processing levels. The program gathers several types of data in real time and outputs it to magnetic tape. An auxiliary program has been written for NRL's CDC 3800 computer to enable these data tapes to be processed by FORTRAN programs.

CONCLUSION

The software developed at the Waldorf Microwave Space Research Facility greatly enhances the usefulness and flexibility of the X-band antenna system. Besides affording convenient operator control of the antenna system, this software provides the means of accurate antenna tracking of a variety of satellites and other celestial objects. The real-time software system has been designed with flexibility and ease of modification as primary objectives; consequently, the system can be, and has been, extended to perform a number of special functions pertaining to real-time data acquisition and data processing.

ACKNOWLEDGMENTS

Special appreciation is extended to Mr. Roy Martinez, who participated in the software development of the first NACTAC system as an employee of the Philco Technical Representative Division.

REFERENCES

1. Bass, C.A., and Townsend, D.H., "The NRL Microwave Space Research Facility: Design and Development of the 60-ft X-Band Antenna," NRL Report 6921, Aug. 1969

2. Bass, C.A., and Stillings, C.J., "The NRL Microwave Space Research Facility: Computer System Hardware for Data Acquisition and Control of the 60-ft X-Band Antenna," NRL Report 7103, June 1970

3. Bass, C.A., and Townsend, D.H., "The NRL Microwave Space Research Facility: Operating Procedures of the Computer Control System for the 60-ft X-Band Antenna," NRL Report 7110, June 1970

Appendix A

SYSTEM LIBRARY TAPE AND SYSTEM DATA TAPE DIRECTORIES

In the system library tape directory immediately following, the program name is followed by the date on which the current version of that program was added to the library tape. This is followed by a two-character code classifying the program. The first character is a number defining the program's mode. The number 1 indicates the program is a nonreal-time program; 2 indicates a real-time program; 3 indicates programs which may be executed within either system; programs having a 4 in this column may only be loaded under control of other programs—they may not be loaded via an operator request to the resident. The second character in the code, A or R, indicates whether the program is absolutely or relatively addressed.

The octal numbers following this code define the length of the binary record containing the program and the locations of memory into which the program is to be loaded, for absolutely addressed programs. For relatively addressed programs, the numbers indicate the amount of core the program may use (but not necessarily the amount it must have) and is, in general, not indicative of the total amount of machine language coding comprising the program.

The octal number following the slash defines the absolute or relative address at which the program is to be executed.

```
001 MANAGE    12/26/68 1A 00100-32777/00100     NACSYS SYSTEM BUILDER
002 NACTAC    02/06/70 1A 00016-06323 06330-17075 17100-17467/17114   REAL TIME
003 NACTAC1   02/06/70 1A 00016-06323 06330-17075 17100-17467/17114   REAL TIME
004 NACTAC3   02/06/70 1A 00016-06323 06330-17075 17100-17467/17114   REAL TIME
005 DAM3      06/25/68 2R 00000-00111/00000     RE-EXECUTE PROGRAM DAM VIA TU3
006 DAM4      06/25/68 2R 00000-00111/00000     RE-EXECUTE PROGRAM DAM VIA TU4
007 TRACK     03/21/67 2R 00000-00255/00000     PROGRAM TRACK DEFINITION
010 SITE      09/06/67 2R 00000-02207/00000     INPUT SITE DATA
011 FREQ      06/09/67 2R 00000-01255/00000     INPUT FREQUENCY DATA
012 ORBIT     09/06/67 2R 00000-03032/00000     INPUT SATELLITE DATA
013 PLANET    06/25/68 2R 00000-04623/00000     INPUT PLANET DATA
014 LOOK      06/09/67 2R 00000-04152/00000     LOOK AT OBJECTS
015 REFR      09/05/67 2R 00000-03523/00000     INPUT REFRACTION DATA
016 SUN       05/09/67 2R 00000-00012/00000     INPUT SUN DATA
017 MERCURY   05/09/67 2R 00000-00012/00000     INPUT MERCURY DATA
020 VENUS     05/09/67 2R 00000-00012/00000     INPUT VENUS DATA
021 MOON      05/09/67 2R 00000-00012/00000     INPUT MOON DATA
022 MARS      05/09/67 2R 00000-00012/00000     INPUT MARS DATA
023 JUPITER   05/09/67 2R 00000-00012/00000     INPUT JUPITER DATA
024 SATURN    05/09/67 2R 00000-00012/00000     INPUT SATURN DATA
025 URANUS    05/09/67 2R 00000-00012/00000     INPUT URANUS DATA
026 NEPTUNE   05/09/67 2R 00000-00012/00000     INPUT NEPTUNE DATA
027 PLUTO     05/09/67 2R 00000-00012/00000     INPUT PLUTO DATA
030 STAR      09/05/67 2R 00000-04105/00000     INPUT STAR DATA
031 DAM       12/26/68 2R 00000-12421/00000     DATA ACQUISITION MODULE
032 REM       11/06/68 4R 00000-00465/00001     REMARKS TO TAPE    * EXECUTABLE
033 CAL       12/26/68 4R 00000-07104/00001     RECALIBRATE MUX    * ONLY BY
034 VAR       12/26/68 4R 00000-02245/00001     CHANGE VARIABLES   * PROGRAM DAM
035 DC/CAL    11/29/67 2R 00000-04770/00000     DC CALIBRATE MUX AND A/D
036 SPECTRUM  01/19/68 2R 00000-11002/00000     SPECTRAL DISPLAY  2**10 MAX PTS
037 ORBTLIST  08/18/67 3R 00000-00237/00000     LIST SATELITE DATA (TU 2)
040 SERVOTST  09/06/67 2R 00000-10425/00000     MEAN, JITTER, RMS, PEAK ERRORS
041 COREDUMP  12/26/68 3R 00000-03171/00000     DUMP CORE, SERVERAL OPTIONS
042 SWITCH    09/18/67 2R 00000-01163/00000     TYPE CONTROL OF CONSOLE SWITCHES
043 CRD/PRNT  09/18/67 3R 00000-00066/00000     CARDS TO PRINTER, ON-LINE
044 CARE      01/13/67 1A 00100-33777/00100     ASSEMBLER
045 COPYHI    01/24/67 1A 00100-32777/00100     COPY TAPE, HI DENSITY
046 ACRONYM   08/20/68 1A 00020-31002/00020     ASSEMBLE OR LIST ACRONYM FILES
047 INDEX     04/16/69 1A 00016-10712/00016     INDEX LISTING TAPE ON TU 3
050 SPECTRAL  01/19/68 1A 00100-27136/00100     SPECTRAL DISPLAY 2**12 MAX PTS
051 DEC/OCT   04/07/67 1A 27000-27607/27000     DECIMAL TO OCTAL CONVERSION
052 ENCDTEST  01/18/67 1A 00100-31017/00100     ENCODER TEST
053 D/ATST    09/18/67 1A 20000-20046/20000     DIGITAL TO ANALOG CONVERTER TEST
054 STARTAPE  08/14/67 1A 00100-03242/00100     STAR CARDS AND DAY NO. TO TAPE
055 DIOG      09/15/66 1A 00020-07126/00020     924-A DIAGNOSTIC ROUTINE
056 MEMTST    09/15/66 1A 00020-00106/00020     MEMORY TEST
057 PRNTST    09/18/67 1A 10000-11007/10000     LINE PRINTER TEST
060 LOOPTST   12/02/66 1A 02000-02470/02000     READER LOOP TEST
061 PUNCHTST  12/02/66 1A 10000-10454/10000     PUNCH TO READER TEST
062 CARDTST   03/03/67 1A 10000-10337/10000     CARD READER TEST
063 TAPETST   07/18/67 1A 00100-06150/00100     603 TAPE UNIT TEST
064 TYPTST    07/18/67 1A 00020-02002/00020     161 TYPEWRITER TEST
065 FINK      09/18/67 1A 04000-15247/05015     FOUR PATTERN MAG TAPE TEST, TU 2
066 CARDTAPE  06/20/67 1A 00100-04410/00100     PACK CARDS ON TAPE
067 COMPFORM  02/01/67 1A 01000-03013/01000     COMPUTER FORMS BUILDER
```

SYSTEM LIBRARY TAPE DIRECTORY

```
001 MANAGE    01/05/67 IA 00100-32777/00100      NACSYS SYSTEM BUILDER
002 CARDTAPE 06/20/67 IA 00100-04410/00100      PACK CARDS ON TAPE
003 SITES     02/20/70                          REMOTE SITE GEODETIC DATA
004 SATELITE 06/21/70                           OSCD SATELLITES JULY 1970 EPOCH
005 SUN       01/01/70                          JAN 1970 - JAN 1971
006 MERCURY   01/01/70                          JAN 1970 - JAN 1971
007 VENUS     01/01/70                          JAN 1970 - JAN 1971
010 MARS      01/01/70                          JAN 1970 - JAN 1971
011 JUPITER   01/01/70                          JAN 1970 - JAN 1971
012 SATURN    01/01/70                          JAN 1970 - JAN 1971
013 URANUS    01/01/70                          JAN 1970 - JAN 1971
014 NEPTUNE   01/01/70                          JAN 1970 - JAN 1971
015 STARS     01/01/70                   145 OPTICAL STARS, 35 RADIO SOURCES
016 DAYS      01/01/70                   IND. DAY NOS. JAN 70 THROUGH DEC 70
017 MOON      01/01/70                          JAN 1970 - JAN 1971
```

SYSTEM DATA TAPE DIRECTORY

Appendix B

EQUIVALENCE LISTINGS

The following listings are reproduced from a listing tape called NACTACOP. When a program is assembled, it is in effect merged with designated portions of the NACTACOP tape through use of a listing tape assembly option provided by the site's assembly processor.

The first part of the listings are equivalence statements providing software connection between the program being assembled and the resident monitor, NACSYS. Both real- and nonreal-time programs are generally provided with these equivalence statements.

The remainder of the listings, beginning near the top of page 42, are assembled with real-time programs to provide communication with the NACTAC portion of the real-time system.

```
                                        DATE    02/12/68

                              IDENT     NACTACOP
                              TOP       REVISION 9
                              SPCL      NSE,NEP

                                        PROGRAMMERS         C. BASS, T. SCHMECKPEPER

00000+  00 0 00000  MARK1     ZRO                           REFERENCE POINT FOR LTA ASSEMBLY
             00016  LOWLIMIT  EQU       16B                 AVAILABLE MEMORY IS .GE. 16B
             33000  MAXLIMIT  EQU       33000B                 AND .LT. MAXLIMIT
             33000  EXIT*     EQU       33000B              JUMP TO NACSYS ENTRANCE
             33001  EXITSW    EQU       EXIT*+1             RESET BY NACSYS TO SYSTART*
             33002  LOWER     EQU       EXITSW+1            CURRENT AVAILABLE MEMORY IS
             33003  UPPER     EQU       LOWER+1                .GE. (LOWER) AND .LT. (UPPER)
             33004  NEXT      EQU       UPPER+1             NEXT AVXILABLE ADDRESS

             33005  REALTIME  EQU       NEXT+1     .LT. 0, REAL TIME, .GE. 0, NOT REAL TIME
             33006  SYSFREE   EQU       REALTIME+1 .LT. 0, INHBT STROBE, I/O UNAVAIL TO DATA*
                                                   .EQ.-0, INHBT STROBE, I/O AVAIL TO DATA*
                                                   .GE. 0, ENABL STROBE, I/O UNAVAIL TO DATA*

             33007  DATE*     EQU       SYSFREE+1           CALANDER DATE FOR REFERENCE

                                        SPECIAL CELLS FOR I/O CONTROL

             33011  TAPECHAN  EQU       DATE*+2             =0, 3 AND 4    =-0, 5 AND 6
             33012  PRNTCHAN  EQU       TAPECHAN+1          =0, 5 AND 6    =-0, 1 AND 2
             33013  PRNTMODE  EQU       PRNTCHAN+1          =0, ONLINE    =-TAPE NO., OFFLINE
             33014  PRNTCODE  EQU       PRNTMODE+1          =0,1,2 FOR 1,2,4 CHAR/WORD
             33015  CARDCODE  EQU       PRNTCODE+1          =0,1,2 FOR 1,2,4 CHAR/WORD BCD
                                                            =0,1 FOR 12,24 BITS/WORD BIN

             33016  READSW    EQU       CARDCODE+1          I/O STATUS SWITCHES
             33017  WRITESW   EQU       READSW+1
             33020  TYPESW    EQU       WRITESW+1
             33021  CARDSW    EQU       TYPESW+1
             33022  PRINTSW   EQU       CARDSW+1

                                        LOCATION OF I/O AND SPECIAL ROUTINES


             33023  ERRORXIT  EQU       PRINTSW+1           ERROR EXIT
             33024  TAPERD    EQU       ERRORXIT+1          TAPE READ
             33025  TAPEWR    EQU       TAPERD+1            TAPE WRITE
             33026  WR2EOF    EQU       TAPEWR+1            WRITE 2 EOF AND BACKSPACE
             33027  BCDCARD   EQU       WR2EOF+1            BCD CARD READ
             33030  BINCARD   EQU       BCDCARD+1           BIN OR HOL CARD READ
             33031  LDBCRDMG  EQU       BINCARD+1           LOAD BINARY CARD IMAGE
             33032  PRINT     EQU       LDBCRDMG+1          PRINT ON/OFF LINE
             33033  MSGTYPE   EQU       PRINT+1             TYPE MESSAGE
             33034  CHARTYPE  EQU       MSGTYPE+1           TYPE CHARACTER
             33035  ONECHAR   EQU       CHARTYPE+1          INPUT ONE CHARACTER FROM TYPE
             33036  NAMEIN    EQU       ONECHAR+1           NAME INPUT VIA TYPEWRITER
             33037  QUESTION  EQU       NAMEIN+1            READY (Y OR EXIT)
             33040  EXITCK    EQU       QUESTION+1          CHECK FOR EXIT TYPED
             33041  ILLEGAL   EQU       EXITCK+1            ILLEGAL TYPED
             33042  RDTAPERR  EQU       ILLEGAL+1           READ TAPE ERROR MESSAGE
             33043  WRTAPERR  EQU       RDTAPERR+1          WRITE TAPE ERROR MESSAGE
             33044  ADRSRQST  EQU       WRTAPERR+1          ADDRESS INPUT VIA TYPE
             33045  ADRSTYPE  EQU       ADRSRQST+1          TYPE OUT ADDRESS
             33046  FETCH*OP  EQU       ADRSTYPE+1          INPUT ROUTINE FROM LIBR TAPE
             33047  MEMORYCK  EQU       FETCH*OP+1          AVAILABLE MEMORY CHECK
             33050  UNPACK    EQU       MEMORYCK+1          CHARACTER UNPACKING
             33051  PACK      EQU       UNPACK+1            CHARACTER PACKING
             33052  LOWCASE   EQU       PACK+1              LOW CASE TYPE TO BCD TABLE
             33053  RESPONSE  EQU       LOWCASE+1           Y OR EXIT

                    *************************************************************************

                    * CODES FOR TAPERD AND TAPEWR INPUT


             00001  BACKSPCE  EQU        1B                 BACKSPACE RECORD CONTROL
             00002  ADVRCORD  EQU        2B                 SEARCH FORWARD RECORD CONTROL
             00004  BACKFILE  EQU        4B                 SEARCH FILE BACKWARDS CONTPOL
             00010  ADVFILE   EQU        10B                SEARCH FILE FORWARD CONTROL
             00020  REWIND    EQU        20B                REWIND CONTROL
             00040  UNLOAD    EQU        40B                REWIND/UNLOAD CONTROL
```

```
          00100  SETLOW   EQU      100B                 SET LOW DENSITY CONTROL
          00200  SETHIGH  EQU      200B                 SET HIGH DENSITY CONTROL
          00400  BINRCORD EQU      400B                 READ BINARY CONTROL
          01000  BCDRCORD EQU      1000B                READ CODED CONTROL
          00004  FILEMARK EQU      4B                   WRITE FILE MARK CONTROL
          00010  ERASE    EQU      10B                  ERASE (SKIP BAD SPOT) CONTROL
          00400  BINARY   EQU      400B                 WRITE BINARY WITH RECOVERY CNTRL
          01000  CODED    EQU      1000B                WRITE CODED WITH RECOVERY CNTRL
          02000  BINARY*  EQU      2000B                WRITE BINARY WITH NO RECVRY CNTRL
          04000  CODED*   EQU      4000B                WRITE CODED WITH NO RECVRY CNTRL

00001+ 00 0 00000  MARK2    ZRO                           REFERENCE POINT FOR LTA ASSEMBLY
                                   NACTAC LOCATIONS

          00001  RESTART  EQU      1B

          00002  CH1LWA   EQU      RESTART+1            CHANNEL 1 LAST WORD ADDRESS+1
          00003  CH1FWA   EQU      CH1LWA+1             CHANNEL 1 CURRENT ADDRESS
          00004  CH2LWA   EQU      CH1FWA+1             CHANNEL 2 LAST WORD ADDRESS+1
          00005  CH2FWA   EQU      CH2LWA+1             CHANNEL 2 CURRENT ADDRESS
          00006  CH3LWA   EQU      CH2FWA+1             CHANNEL 3 LAST WORD ADDRESS+1
          00007  CH3FWA   EQU      CH3LWA+1             CHANNEL 3 CURRENT ADDRESS
          00010  CH4LWA   EQU      CH3FWA+1             CHANNEL 4 LAST WORD ADDRESS+1
          00011  CH4FWA   EQU      CH4LWA+1             CHANNEL 4 CURRENT ADDRESS
          00012  CH5LWA   EQU      CH4FWA+1             CHANNEL 5 LAST WORD ADDRESS+1
          00013  CH5FWA   EQU      CH5LWA+1             CHANNEL 5 CURRENT ADDRESS
          00014  CH6LWA   EQU      CH5FWA+1             CHANNEL 6 LAST WORD ADDRESS+1
          00015  CH6FWA   EQU      CH6LWA+1             CHANNEL 6 CURRENT ADDRESS

          00016  NTRPEXIT EQU      CH6FWA+1             INTERRUPT RETURN ADDRESS

          00017  NTRLNTRP EQU      NTRPEXIT+1           INTERNAL INTERRUPT

          00020  DEADSPCE EQU      NTRLNTRP+1

          00021  CH1NTRP  EQU      DEADSPCE+1           CHANNEL 1 INTERRUPT
          00022  CH2NTRP  EQU      CH1NTRP+1            CHANNEL 2 INTERRUPT
          00023  CH3NTRP  EQU      CH2NTRP+1            CHANNEL 3 INTERRUPT
          00024  CH4NTRP  EQU      CH3NTRP+1            CHANNEL 4 INTERRUPT
          00025  CH5NTRP  EQU      CH4NTRP+1            CHANNEL 5 INTERRUPT
          00026  CH6NTRP  EQU      CH5NTRP+1            CHANNEL 6 INTERRUPT

          00027  NTRPMASK EQU      CH6NTRP+1            INTERRUPT MASK CONTROL

                                   SPECIAL CELLS AND LOCATIONS

          00030  SUBLIST  EQU      NTRPMASK+1
          00030  ANT      EQU      SUBLIST              ANTENNA CONTROL LEVEL ADDRESS
          00031  ANT*     EQU      ANT+1                ANTENNA PROCESSING LEVEL ADDRESS
          00032  MONT*    EQU      ANT*+1               MONITOR PROCESSING LEVEL ADDRESS
          00033  DATA*    EQU      MONT*+1              DATA PROCESSING LEVEL ADDRESS
          00034  LEVEL    EQU      DATA*+1              CURRENT LEVEL ADDRESS

          00035  EXEC     EQU      LEVEL+1              LOC. EXECUTIVE CONTROL ROUTINE
          00036  FAULTRD  EQU      EXEC+1               LOC. FAULT READ
          00037  FAULTWR  EQU      FAULTRD+1            LOC. FAULT WRITE
          00040  MASKB    EQU      FAULTWR+1            LOC. CLASS B INTERRUPT MASK
          00041  UNMASKB  EQU      MASKB+1              LOC. CLASS B INTERRUPT UNMASK
          00042  MASTER   EQU      UNMASKB+1            LOC. MASTER NORMALIZATION OF CONTROL

          00043  DATEBLOC EQU      MASTER+1             YEAR ANNO DOMINI
                                                        MONTH NUMBER, JANUARY=1, ETC.
                                                        DAY OF MONTH
                                                        HOURS SINCE MIDNIGHT GMT
          00047  DATE     EQU      DATEBLOC+4           MODIFIED JULIAN DATE, DAYS BI 8
          00050  ST.UTO   EQU      DATE+1               SIDEREAL TIME FOR 0 HRS UT OF DATE
          00051  EQ.EQ    EQU      ST.UTO+1             EQUATION OF EQUINOXES
```

```
00052  SATELITE EQU      EQ.EQ+1              *START LOCATION OF DATA BLOCK FOR
                                                SATELLITE TO BE PROG. TRACKED
                                              =0 FOR NO SATELLITE TO BE TRACKED**

00053  CELSTIAL EQU      SATELITE+1           *START LOCATION OF DATA BLOCK FOR
                                                CELESTIAL OBJECT TO BE PROG. TRACKED
                                              =0 FOR NO CELESTIAL OBJECT TO BE
                                                TRACKED**


00054  SITE     EQU      CELSTIAL+1           REMOTE SITE FOR WHICH RANGE AND RANGE-
                                              RATE INFORMATION IS COMPUTED.

00055  FREQ     EQU      SITE+1               FREQUENCY BEING USED IN RANGE AND RANGE-
                                              RATE COMPUTATIONS.

00056  PROGNUMB EQU      FREQ+1               INTEGER PROGRAM NUMBER

00057  SAMPLTST EQU      PROGNUMB+1           =0 FOR SAMPLE POINT, =-0 FOR NOT
00060  READTST  EQU      SAMPLTST+1           =0 FOR READ ANGLES, =-0 FOR NOT
00061  DIGMDTST EQU      READTST+1            =0 FOR DIGITAL MODE, =-0 FOR NOT
00062  TRACKTST EQU      DIGMDTST+1           =0 FOR AUTO TRACK SUBMODE, =-0 FOR NOT

                         ** SET BY MONITOR CALLED ROUTINES


00063  NORMALIZ EQU      TRACKTST+1           LOC. ANT CONTROL NORMALIZATION
00064  SAMPLESW EQU      NORMALIZ+1           LOC. SAMPLESW
00065  ERRORSW  EQU      SAMPLESW+1           LOC. ERRORSW


00066  SATADRS  EQU      ERRORSW+1
00066  S.1      EQU      SATADRS              START LOC. SATELLITE 1 DATA
00067  S.2      EQU      S.1+1                START LOC. SATELLITE 2 DATA
00070  S.3      EQU      S.2+1                START LOC. SATELLITE 3 DATA
00071  S.4      EQU      S.3+1                START LOC. SATELLITE 4 DATA

00072  CELADRS  EQU      S.4+1
00072  C.1      EQU      CELADRS              START LOC. CELESTIAL 1 DATA
00073  C.2      EQU      C.1+1                START LOC. CELESTIAL 2 DATA

00074  MOONADRS EQU      C.2+1
00074  M.1      EQU      MOONADRS             START LOC. MOON DATA


00075  SITE.L   EQU      M.1+1                START LOC. LOCAL SITE DATA
00076  SITE.A   EQU      SITE.L+1             START LOC. REMOTE A SITE DATA

00077  SITE.B   EQU      SITE.A+1             START LOC. REMOTE B SITE DATA
00100  SITE.C   EQU      SITE.B+1             START LOC. REMOTE C SITE DATA


00101  FREQ.1   EQU      SITE.C+1             START LOC. FREQUENCY 1 DATA
00102  FREQ.2   EQU      FREQ.1+1             START LOC. FREQUENCY 2 DATA
00103  FREQ.3   EQU      FREQ.2+1             START LOC. FREQUENCY 3 DATA


00104  AZCAMTBL EQU      FREQ.3+1             START LOC. AZIMUTH CAM TABLE
00105  ELCAMTBL EQU      AZCAMTBL+1           START LOC. ELEVATION CAM TABLE
00106  FRCAMTBL EQU      ELCAMTBL+1           START LOC. PRIMARY REFRACTION TABLE
00107  FRSUBTBL EQU      FRCAMTBL+1           START LOC. SECONDARY REFRACTION TABLE


00110  LDCNTWD1 EQU      FRSUBTBL+1           LOC. LOAD CNTRL WORD 1 FOR STATUSOP
00111  LDCNTWD2 EQU      LDCNTWD1+1           LOC. LOAD CNTRL WORD 2 FOR STATUSOP
```

```
                              BIT LIST

                              PURPOSE    STANDARD TABLE OF BITS


00112   POSZERO   EQU         LOCNTWD2+1
00113   BIT0      EQU         POSZERO+1
00114   BIT1      EQU         BIT0+1
00115   BIT2      EQU         BIT1+1
00116   BIT3      EQU         BIT2+1
00117   BIT4      EQU         BIT3+1
00120   BIT5      EQU         BIT4+1
00121   BIT6      EQU         BIT5+1
00122   BIT7      EQU         BIT6+1
00123   BIT8      EQU         BIT7+1
00124   BIT9      EQU         BIT8+1
00125   BIT10     EQU         BIT9+1
00126   BIT11     EQU         BIT10+1
00127   BIT12     EQU         BIT11+1
00130   BIT13     EQU         BIT12+1
00131   BIT14     EQU         BIT13+1
00132   BIT15     EQU         BIT14+1
00133   BIT16     EQU         BIT15+1
00134   BIT17     EQU         BIT16+1
00135   BIT18     EQU         BIT17+1
00136   BIT19     EQU         BIT18+1
00137   BIT20     EQU         BIT19+1
00140   BIT21     EQU         BIT20+1
00141   BIT22     EQU         BIT21+1
00142   BIT23     EQU         BIT22+1


00143   NEGZERO   EQU         BIT23+1
00144   NBT0      EQU         NEGZERO+1
00145   NBT1      EQU         NBT0+1
00146   NBT2      EQU         NBT1+1
00147   NBT3      EQU         NBT2+1
00150   NBT4      EQU         NBT3+1

00151   NBT5      EQU         NBT4+1
00152   NBT6      EQU         NBT5+1
00153   NBT7      EQU         NBT6+1
00154   NBT8      EQU         NBT7+1
00155   NBT9      EQU         NBT8+1
00156   NBT10     EQU         NBT9+1
00157   NBT11     EQU         NBT10+1
00160   NBT12     EQU         NBT11+1
00161   NBT13     EQU         NBT12+1
00162   NBT14     EQU         NBT13+1
00163   NBT15     EQU         NBT14+1
00164   NBT16     EQU         NBT15+1
00165   NBT17     EQU         NBT16+1
00166   NBT18     EQU         NBT17+1
00167   NBT19     EQU         NBT18+1
00170   NBT20     EQU         NBT19+1
00171   NBT21     EQU         NBT20+1
00172   NBT22     EQU         NBT21+1
00173   NBT23     EQU         NBT22+1



00142   SIGNBIT   EQU         BIT23
00173   ONE       EQU         NBT23
00141   ONE/2     EQU         BIT22
00140   ONE/4     EQU         BIT21
00137   ONE/8     EQU         BIT20
00136   ONE/16    EQU         BIT19


00142   NEGONE    EQU         BIT23
00172   NEGONE/2  EQU         NBT22
00171   NEGONE/4  EQU         NBT21 .
00170   NEGONE/8  EQU         NBT20
00167   NGONE/16  EQU         NBT19

00143   BIT23/0   EQU         NEGZERO
00173   BIT22/0   EQU         NBT23
00144   BIT23/1   EQU         NBT0
```

```
00141  POS180DG EQU       BIT22
00172  NEG180DG EQU       NBT22
00140  POS90DG  EQU       BIT21
00171  NEG90DG  EQU       NBT21
00137  POS45DG  EQU       BIT20
00170  NEG45DG  EQU       NBT20

00000  COEFF    EQU       0                     REL. LOC. OF COEFFICIENT FOR FLOAT. PT.
00001  EXP      EQU       1                     REL. LOC. OF EXPONENT FOR FLOAT. PT.

00174  FLONE    EQU       NBT23+1               FLOATING POINT ONE
00176  FLONE/2  EQU       FLONE+2               FLOATING POINT 1/2
00200  FLONE/4  EQU       FLONE/2+2             FLOATING POINT 1/4
00202  FLTWOPI  EQU       FLONE/4+2             FLOATING POINT 2 PI
00204  SIX      EQU       FLTWOPI+2
00205  TEN      EQU       SIX+1
00206  HUNDRED  EQU       TEN+1
00207  THOUSAND EQU       HUNDRED+1

00210  OPCODE   EQU       THOUSAND+1            OPERATION CODE MASK
00211  INDEX    EQU       OPCODE+1              INDEX DESIGNATOR MASK
00212  ADDRESS  EQU       INDEX+1               EXECUTION ADDRESS MASK

00213  REVBITS  EQU       ADDRESS+1             AZIMUTH REVOLUTION BITS MASK


                          RESOLVER DEFINITIONS


                          RESOLVER DEFINITIONS AND RELATIVE ADDRESS POSITIONS.

00000  SCALE    EQU       0
00001  ZTEMP    EQU       1
00002  YTEMP    EQU       2
00003  XTEMP    EQU       3
00004  WTEMP    EQU       4
00005  VTEMP    EQU       5
00006  UTEMP    EQU       6

00005  SINTEMP  EQU       VTEMP
00006  COSTEMP  EQU       UTEMP


00003  RNGTEMP  EQU       XTEMP
00001  AZTEMP   EQU       ZTEMP
00004  ELTEMP   EQU       WTEMP

00001  HATEMP   EQU       ZTEMP
00004  DECTEMP  EQU       WTEMP

                          SPECIAL LOCATIONS FOR SATELLITES AND CELESTIAL OBJECTS

00007  ST.ANG   EQU       7

00010  E.RATVCT EQU       ST.ANG+1              EXPONENT OF VECTOR
                          E.RATVCT+1            Z COMPONENT
                          E.RATVCT+2            Y COMPONENT
                          E.RATVCT+3            Z COMPONENT

00014  SINEA    EQU       E.RATVCT+4
00015  COSEA    EQU       SINEA+1

00016  SQRT.ME2 EQU       COSEA+1


00017  SAT.NRNG EQU       SQRT.ME2+1
                          SAT.NRNG+1
```

SATELLITE DEFINITIONS

INPUT BLOCK AND RELATIVE LOCATIONS

```
00000  CATLOG    EQU    0              CATALOGUE NUMBER OF SATELLITE
00001  E.DAYS    EQU    CATLOG+1       EPOCH DAYS, BI 6
00002  E.FOD/64  EQU    E.DAYS+1       EPOCH FRAC OF DAY (FOD)  BF-6
00003  E.M       EQU    E.FOD/64+1     EPOCH MEAN ANAMOLY, FOC  BF 0
00004  E.N       EQU    E.M+1          EPOCH MEAN MOTION, REV/DAY COEFFICIENT
                                             LO-ORDER COEFFICIENT (=0)
                                             EXPONENT
00007  E.DN      EQU    E.N+3                IST DERIV/2, REV/DAY2, COEFF
                                                 EXPONENT
00011  E.D2N     EQU    E.DN+2               2ND DERIV/6, REV/DAY3, COEFF
                                                 EXPONENT
00013  E.D3N     EQU    E.D2N+2              3RD DERIV/24, REV/DAY4, COEFF
                                                 EXPONENT
00015  E.IN      EQU    E.D3N+2        EPOCH MEAN MOTION, REV/DAY, COEFFICIENT
                                             EXPONENT
00017  E.2DN     EQU    E.IN+2               IST DERIV, REV/DAY2, COEFF
                                                 EXPONENT
00021  E.3D2N    EQU    E.2DN+2              2ND DERIV/2, REV/DAY3, COEFF
                                                 EXPONENT
00023  E.4D3N    EQU    E.3D2N+2             3RD DERIV/6, REV/DAY4, COEFF
                                                 EXPONENT
00025  E.RA      EQU    E.4D3N+2       EPOCH RT ASC. ASCEN NODE, FOC  BF 0
00026  E.DRA     EQU    E.RA+1               IST DERIV, REV/DAY, COEFF
                                                 EXPONENT
00030  E.D2RA    EQU    E.DRA+2              2ND DERIV/2, REV/DAY2, COEFF
                                                 EXPONENT
00032  E.W       EQU    E.D2RA+2       EPOCH ARG. OF PERIGEE, FOC  BF 0
00033  E.DW      EQU    E.W+1                IST DERIV, REV/DAY, COEFF
                                                 EXPONENT
00035  E.D2W     EQU    E.DW+2               2ND DERIV/2, REV/DAY2, COEFF
                                                 EXPONENT
00037  E.E       EQU    E.D2W+2        EPOCH ECCENTRICITY,  BF 0
00040  E.DE      EQU    E.E+1                IST DERIV, I/DAY, COEFF


                                                 EXPONENT
00042  E.D2E     EQU    E.DE+2               2ND DERIV/2, I/DAY2, COEFF
                                                  EXPONENT
00044  E.I       EQU    E.D2E+2        EPOCH INCLINATION, FOC  BF 0
00045  E.DI      EQU    E.I+1                IST DERIV. REV/DAY, COEFF
                                                 EXPONENT
00047  E.A       EQU    E.DI+2         EPOCH SEMI-MAJOR AXIS, ER COEFF
                                             EXPONENT
00051  E.DA      EQU    E.A+2                IST DERIV, ER/DAY, COEFF
                                                 EXPONENT
00053  E.D2A     EQU    E.DA+2               2ND DERIV/2. ER/DAY2, COEFF
                                                 EXPONENT
```

OUTPUT BLOCK AND RELATIVE LOCATIONS

```
00055  P.M       EQU    E.D2A+2        UPDATED MEAN ANAMOLY, FOC  BF 0
00056  P.N       EQU    P.M+1          UPDATED MEAN MOTION, FOC/DAY, COEFF
                                                 EXPONENT
00060  P.RA      EQU    P.N+2          UPDATED RT ASC. ASCEN NODE, FOC  BF 0
00061  P.W       EQU    P.RA+1         UPDATED ARG. OF PERIGEE. FOC  BF 0
00062  P.E       EQU    P.W+1          UPDATED ECCENTRICITY, BF 0
00063  P.I       EQU    P.E+1          UPDATED INCLINATION, FOC  BF 0
00064  P.A       EQU    P.I+1          UPDATED SEMI-MAJ AXIS  ER, COEFF
                                                 EXPONENT

00066  P.EA      EQU    P.A+2          ECCENTRIC ANAMOLY, FOC  BF 0

00067  P.DAYS    EQU    P.EA+1         UPDATE TIME, DAYS  BI 6
00070  P.FOD/64  EQU    P.DAYS+1       UPDATE TIME, FOD  BF-6
```

```
                                      CELESTIAL DEFINITIONS

                                      INPUT BLOCK AND RELATIVE LOCATIONS

00000  CEL.ID   EQU       0                     CELESTIAL IDENTIFICATION
00001  C.DAYS   EQU       CEL.ID+1              TIME OF 0 ARGUMENT, MOD. JUL. DAYS  BI 6
00002  C.FOD/64 EQU       C.DAYS+1              TIME OF 0 ARGUMENT, FRAC. DAYS  BF-6
00003  C.POINTS EQU       C.FOD/64+1            NUMBER OF POINTS OF TABLES, LESS OVERLAP
00004  C.DIFF   EQU       C.POINTS+1            TIME INCR. OF TABLES, DAYS  COEFFICIENT
                                                                            EXPONENT
00006  C.DIST   EQU       C.DIFF+2              REL. LOCATION OF DIST. FROM EARTH TABLE
00007  C.RTA    EQU       C.DIST+1              REL. LOCATION RIGHT ASCENSION TABLE
00010  C.DEC    EQU       C.RTA+1               REL. LOCATION DECLINATION TABLE
00011  C.DL/DT  EQU       C.DEC+1               REL. LOCATION DERIV. DIST/ DIST TABLE
00012  C.DRA/DT EQU       C.DL/DT+1             REL. LOCATION DERIV. RT ASCENSION TABLE
00013  C.DDC/DT EQU       C.DRA/DT+1            REL. LOCATION DERIV. DECLINATION TABLE

                                      P.DIST+1, P.DL/DT+1, P.DRA/DT+1, AND P.DDC/DT+1 OF OUTPUT
                                      BELOW ARE ALSO INPUT QUANTITIES.

                                      UNITS OF TABLES ARE...FOR DIST      EARTH RADII, ER
                                                                RTA       FOC  BF 0
                                                                DEC       FOC  BF 0
                                                                DL/DT     ER/DAY
                                                                DRA/DT    RAD/DAY
                                                                DDC/DT    RAD/DAY


                                      OUTPUT BLOCK AND RELATIVE LOCATIONS

00014  P.DIST   EQU       C.DDC/DT+1            UPDATED DIST. FROM EARTHS CENTER
                                                       EXPONENT
00016  P.RTA    EQU       P.DIST+2              UPDATED RIGHT ASCENSION
00017  P.DEC    EQU       P.RTA+1               UPDATED DECLINATION
00020  P.DL/DT  EQU       P.DEC+1               UPDATED DERIV. DIST./DIST., COEFF
                                                       EXPONENT
00022  P.DRA/DT EQU       P.DL/DT+2             UPDATED DERIV. RIGHT ASCENSION, COEFF


                                                       EXPONENT
00024  P.DDC/DT EQU       P.DRA/DT+2            UPDATED DERIV. DECLINATION, COEFF
                                                       EXPONENT

                                      UNITS OF ABOVE ARE SAME AS FOR CORRESPONDING INPUT TABLE.

00026  P.IARG   EQU       P.DDC/DT+2            INTERGER ARGUMENT FOR UPDATE TIME
00027  P.FARG   EQU       P.IARG+1              FRACTIONAL ARGUMENT FOR UPDATE TIME


                                      CLASS A TEMP DEFINITIONS AND RELATIVE ROUTINE LIST

                                      THE FOLLOWING DEFINITIONS ARE RELATIVE TO THE (B6).

                                      TEMP LOCATIONS ARE INTENDED TO BE USED AS WORKING LOCIT905
                                      WITHIN A ROUTINE, INCLUDING SUBROUTINES USED, AND SHOULD NOT
                                      BE USED TO CARRY INFORMATION OUTSIDE A CLASS A LEVEL ROUTINE.

00000  TEMP0    EQU       0
00001  TEMP1    EQU       TEMP0+1
00002  TEMP2    EQU       TEMP1+1
00003  TEMP3    EQU       TEMP2+1
00004  TEMP4    EQU       TEMP3+1
00005  TEMP5    EQU       TEMP4+1
00006  TEMP6    EQU       TEMP5+1
00007  TEMP7    EQU       TEMP6+1
00010  TEMP8    EQU       TEMP7+1
00011  TEMP9    EQU       TEMP8+1
00012  TEMP10   EQU       TEMP9+1
00013  TEMP11   EQU       TEMP10+1
00014  TEMP12   EQU       TEMP11+1
00015  TEMP13   EQU       TEMP12+1
00016  TEMP14   EQU       TEMP13+1
00017  TEMP15   EQU       TEMP14+1
```

```
00020  JOB       EQU   TEMP15+1        =0, JOB COMPLETED,  =-0, FOR NOT
                                       .LT. -0, DEFER JOB
00021  SAVE      EQU   JOB+1           =0, NO SAVE,        =-0, SAVE REGISTERS
00022  RESTORE   EQU   SAVE+1          =0, NO RESTORE,     =-0, RESTORE REGIS.

00023  ATEMP     EQU   RESTORE+1       A REGISTER TEMPORARY STORAGE
00024  QTEMP     EQU   ATEMP+1         Q REGISTER TEMPORARY STORAGE
00025  B1TEMP    EQU   QTEMP+1         INDEX B1 TEMPORARY STORAGE
00026  B2TEMP    EQU   B1TEMP+1        INDEX B2 TEMPORARY STORAGE
00027  B3TEMP    EQU   B2TEMP+1        INDEX B3 TEMPORARY STORAGE
00030  B4TEMP    EQU   B3TEMP+1        INDEX B4 TEMPORARY STORAGE

00031  B5TEMP    EQU   B4TEMP+1        INDEX B5 TEMPORARY STORAGE
00032  B6TEMP    EQU   B5TEMP+1        INDEX B6 TEMPORARY STORAGE
00033  FAULTEMP  EQU   B6TEMP+1        FAULT TEMPORARY STORAGE

00034  EXIT      EQU   FAULTEMP+1      LEVEL RETURN ADDRESS TEMPORARY STORA75

                       RELATIVE ROUTINE LIST DEFINES THE RELATIVE POSITIONS OFJUMPS
                       TO GENERAL SUBROUTINES AND JUMPS FOR THEIR EXITS WITHIN EACH

                       THIS PROVIDES MEANS OF RETAINING INFORMATION PERTINENT TO
                       A LEVEL OF PROGRAMMING WHEN INTERRUPTED FOR ANOTHER LEVEL
                       WHICH USES THE SAME GENERAL SUBROUTINE AS THAT INTERRUPTED
                       IN THE FIRST LEVEL.

                       CALLING SEQUENCE FOR A GENERAL SUBROUTINE IS BASED UPON
                       HAVING PREVIOUSLY SET B6 TO THE STARTING ADDRESS OF THE
                       SUBLIST FOR THAT LEVEL OF PROGRAMMING.  THE CALLING INSTRUC-
                       TION FOR EXAMPLE IS THEN      URJ   6   ADP  .

00035  HOLDJOB   EQU   EXIT+1          DEFER JOB
00037  ENDOFJOB  EQU   HOLDJOB+2       TERMINATE JOB

00041  MASKA     EQU   ENDOFJOB+2      CLASS A INTERRUPT MASK SET
00043  UNMASKA   EQU   MASKA+2         CLASS A INTERRUPT MASK CLEAR

00045  ROUNDOFF  EQU   UNMASKA+2       ROUNDOFF TO GIVEN BIT

00047  ADP       EQU   ROUNDOFF+2      ADD DOUBLE PRECISION
00051  SDP       EQU   ADP+2           SUBTRACT DOUBLE PRECISION
00053  MDPF      EQU   SDP+2           MULTIPLY DOUBLE PRECISION
00055  DPOPEN    EQU   MDPF+2          OPEN FORMAT DOUBLE PRECISION
00057  DPSAME    EQU   DPOPEN+2        INSURE SAME SIGN DOUBLE PRECISION

00061  FAD       EQU   DPSAME+2        FLOATING ADD, SINGLE PRECISION
00063  FSB       EQU   FAD+2           FLOATING SUBTRACT
00065  FMP       EQU   FSB+2           FLOATING MULTIPLY
00067  FDP       EQU   FMP+2           FLOATING DIVIDE

00071  OVFLO     EQU   FDP+2           FLOATING OVERFLOW PROCESSING
00073  FSR       EQU   OVFLO+2         FLOATING SQUARE ROOT
00075  FLOG      EQU   FSR+2           FLOATING LOGARITHIM

00077  DPFAD     EQU   FLOG+2          DOUBLE PREC. FLOATING ADD
00101  DPFSB     EQU   DPFAD+2         DOUBLE PREC. FLOATING SUBTRACT
00103  DPFMP     EQU   DPFSB+2         DOUBLE PREC. FLOATING MULTIPLY
00105  DPOVFLO   EQU   DPFMP+2         DOUBLE PREC. FLOAT. OVERFLOW PROCESS

00107  ADD/FL    EQU   DPOVFLO+2       FIXED ADD AND FLOAT SUM
00111  SUB/FL    EQU   ADD/FL+2        FIXED SUB AND FLOAT DIFF.

00113  FL/FX0    EQU   SUB/FL+2        FLOAT TO FIX NO. 0
00115  FL/FX1    EQU   FL/FX0+2        FLOAT TO FIX NO. 1
00117  FL/FX2    EQU   FL/FX1+2        FLOAT TO FIX NO. 2

00121  DROP      EQU   FL/FX2+2        DIRECT DIGITAL RESOLVER
00123  DR*CP1    EQU   DROP+2          INVERSE DIGITAL RESOLVER NO. 1
00125  DR*OP2    EQU   DR*OP1+2        INVERSE DIGITAL RESOLVER NO. 2

00127  SINCOS    EQU   DR*OP2+2        SINE AND COSINE
00131  SINE      EQU   SINCOS+2        SINE
00133  COSINE    EQU   SINE+2          COSINE
00135  SINSUB    EQU   COSINE+2        SPECIAL SUBROUTINE FOR SINE, COSINE, ST3
00137  ARCTAN    EQU   SINSUB+2        ARCTANGENT
```

```
00141  SQRTSMSQ EQU     ARCTAN+2        SQUARE-ROOT-SUM-SQUARE
00143  SQROOT   EQU     SQRTSMSQ+2      SQUARE ROOT

00145  MOP      EQU     SQROOT+2        M MATRIX OPERATION
00147  M*OP     EQU     MOP+2           M* MATRIX OPERATION

00151  VCTADD   EQU     M*OP+2          VECTOR FLOAT ADD
00153  VCTSUB   EQU     VCTADD+2        VECTOR FLOAT SUBTRACT
00155  VCTMUF1  EQU     VCTSUB+2        PRDCT FLOAT. PT. VCT AND FLOAT. PT. CNST
00157  VCTMUF2  EQU     VCTMUF1+2       SCALAR PODUCT TWO FLOAT. PT. VECTORS
00161  VCTMAG   EQU     VCTMUF2+2       MAGNITUDE FLOAT. PT. VECTOR
00163  VCTNORM  EQU     VCTMAG+2        NORMALIZE FLOAT. PT. VECTOR


00165  VCTXFER  EQU     VCTNORM+2       TRANSFER FLOAT. PT. VECTOR
00167  VCTOVFLO EQU     VCTXFER+2       VECTOR FLOAT. OVERFLOW PROCESS

00171  VCTRULE  EQU     VCTOVFLO+2      SINGLE PREC. SATURATE RULE
00173  DPVCTRUL EQU     VCTRULE+2       DOUBLE PREC. SATURATE RULE

00175  ANGLRULE EQU     DPVCTRUL+2      ANGLE RULE
00177  P.ANGREP EQU     ANGLRULE+2      POSITIVE ANGLE REPRESENTATION
00201  N.ANGREP EQU     P.ANGREP+2      NEGATIVE ANGLE REPRESENTATION
00203  ANGLS180 EQU     N.ANGREP+2      ABSOLUTE ANGLE  LE  180 DEG

00205  ANGTRP.1 EQU     ANGLS180+2      LINEAR INTERPOLATION ANGLE
00207  ANGTRP.2 EQU     ANGTRP.1+2      2ND ORDER BESSEL INTERP. ANGLE
00211  ANGTRP.3 EQU     ANGTRP.2+2      3RD ORDER BESSEL INTERP. ANGLE
00213  VCTRP.1  EQU     ANGTRP.3+2      LINEAR INTERP. NON-ANGLE
00215  VCTRP.2  EQU     VCTRP.1+2       2ND ORDER BES. INTERP. NON-ANGLE
00217  VCTRP.3  EQU     VCTRP.2+2       3RD ORDER BES. INTERP. NON-ANGLE

00221  ANGDIF.3 EQU     VCTRP.3+2       3RD ORDER NUMERICAL DIFFERENTIATION
00223  VCTDIF.3 EQU     ANGDIF.3+2      3RD ORDER ANGLE NUMER. DIFFERENTIATION


00225  INCRDATE EQU     VCTDIF.3+2      RELATIVE DATE BLOCK CONTROL

00227  UPDATE.1 EQU     INCRDATE+2      UPDATE METHOD 1 OF SATELLITE ELEMENTS
00231  UPDMULT  EQU     UPDATE.1+2      SPECIAL MULT SUBROUTINE FOR UPDATE.1
00233  UPDADD   EQU     UPDMULT+2       SPECIAL ADD SUBROUTINE FOR UPDATE.1
00235  M/E.ANOM EQU     UPDADD+2        MEAN TO ECCENTRIC ANAMOLY
00237  SAT.XYZ  EQU     M/E.ANOM+2      SATELLITE ELEMENTS TO XYZ COORDINATES
00241  SAT.ERAT EQU     SAT.XYZ+2       VECTOR PRODUCT EARTH AND SATELLITE V3TS
00243  SAT.DXYZ EQU     SAT.ERAT+2      DERIV. SATELLITE POSITION VECTOR
00245  SXYZ/XYZ EQU     SAT.DXYZ+2      SATELLITE XYZ TO EARTH XYZ

00247  UPDATE.2 EQU     SXYZ/XYZ+2      UPDATE MOON AND PLANENTS
00251  CEL.XYZ  EQU     UPDATE.2+2      CELESTIAL OBJECT TO XYZ
00241  CEL.ERAT EQU     SAT.ERAT        VECTOR RRODUCT EARTH AND CELESTIAL VCTS
00253  CEL.DXYZ EQU     CEL.XYZ+2       DERIV. MOON AND PLANENTS POSITION VCT


00255  RAE/XYZ  EQU     CEL.DXYZ+2      RNG, AZ, EL TO XYZ COORDINATES
00257  XYZ/RAE  EQU     RAE/XYZ+2       XYZ COORDINATES TO RNG, AZ, EL
00255  RHAD/XYZ EQU     RAE/XYZ         RNG, HR ANG, DEC TO XYZ  COORDINATES
00257  XYZ/RHAD EQU     XYZ/RAE         XYZ COORDINATES TO RNG, HR ANG, DEC

00261  ISOTROP  EQU     XYZ/RAE+2       DB ISOTROPIC ANT. EFFEC. AREA
00263  SPCATTEN EQU     ISOTROP+2       DB SPACE ATTENUATION FACTOR

00265  PRIORITY EQU     SPCATTEN+2      SET PRIORITY OF LEVELS


00241  OBJ.ERAT EQU     SAT.ERAT
```

```
                                TEMP LOCATIONS ARE INTENDED TO BE USED AS WORKING LOCATIONS
                                WITHIN A CLASS B LEVEL ROUTINE (INTERRUPT PROTECTED) AND
                                SHOULD NOT BE USED TO CARRY INFORMATION OUTSIDE A CLBSS B
                                LEVEL ROUTINE.

00214  ASAVE    EQU             REVBITS+1
00215  QSAVE    EQU             ASAVE+1
00216  B1SAVE   EQU             QSAVE+1
00217  B2SAVE   EQU             B1SAVE+1
00220  B3SAVE   EQU             B2SAVE+1
00221  B4SAVE   EQU             B3SAVE+1
00222  B5SAVE   EQU             B4SAVE+1
00223  B6SAVE   EQU             B5SAVE+1

00224  FALTSAVE EQU             B6SAVE+1


                                INPUT BLOCK

00225  SYSTAT1  EQU             FALTSAVE+1         SYSTEM STATUS NO. 1
00226  SYSTAT2  EQU             SYSTAT1+1          SYSTEM STATUS NO. 2
00227  SYSTAT3  EQU             SYSTAT2+1          SYSTEM STATUS NO. 3
00230  GMT      EQU             SYSTAT3+1          REAL TIME CLOCK, GREENWICH TIME
00231  TIMOFF.I EQU             GMT+1              TIME OFFSET
00232  AZDSGOFF EQU             TIMOFF.I+1         AZIMUTH DESIGNATE-OFFSET
00233  ELDSGOFF EQU             AZDSGOFF+1         ELEVATION DESIGNATE-OFFSET
00234  VELDSG   EQU             ELDSGOFF+1         VELOCITY DESIGNATE
00235  SCANPARS EQU             VELDSG+1           SCAN PARAMETERS
00236  SPAREB   EQU             SCANPARS+1         SPARE B
00237  SPAREC   EQU             SPAREB+1           SPARE C
00240  SPARED   EQU             SPAREC+1           SPARE D


00241  CMPSTAT1 EQU             SPARED+1           COMPUTER STATUS NO. 1
00242  CMPSTAT2 EQU             CMPSTAT1+1         COMPUTER STATUS NO. 2
00243  CMPSTAT3 EQU             CMPSTAT2+1         COMPUTER STATUS NO. 3
00244  ANTAZ    EQU             CMPSTAT3+1         ANTENNA AZIMUTH (FOR DISPLAY)
00245  ANTEL    EQU             ANTAZ+1            ANTENNA ELEVATION (FOR DISPLAY)
00246  TIMOFF.O EQU             ANTEL+1            TIME OFFSET (FOR DISPLAY)
00247  AZOFFST  EQU             TIMOFF.O+1         AZIMUTH OFFSET (FOR DISPLAY)
00250  ELOFFST  EQU             AZOFFST+1          ELEVATION OFFSET (FOR DISPLAY)
00251  RANG1    EQU             ELOFFST+1          RANGE NO. 1 (FOR DISPLAY)
00252  RNGRAT1  EQU             RANG1+1            RANGE-RATE NO. 1 (FOR DISPLAY)
00253  RANG2    EQU             RNGRAT1+1          RANGE NO. 2 (FOR DISPLAY)
00254  RNGRAT2  EQU             RANG2+1            RANGE-RATE NO. 2 (FOR DISPLAY)


                                THE FOLLOWING LOCATIONS AND CONSTANTS ARE ASSOCIATED WITH
                                THE READING OF ANTENNA NUMBER 1 POSITION, FORMATION OF
                                COMMANDS, AND ERROR OUTPUT FOR THE DIGITAL SERVO.

                                FOR ENCODER FORMAT (ENCD FORMAT) AND COMMAND FORMAT (COM
                                FORMAT), SEE INPUT AND OUTPUT FORMATS RESPECTIVELY FOR
                                READANT AND FRMER.1A.

00255  ENCOD.AZ EQU             RNGRAT2+1          AZIMUTH ENCODER ANGLE, ENCD FORMAT
00256  ENCOD.EL EQU             ENCOD.AZ+1         ELEVATION ENCODER ANGLE, ENCD FORMAT

00257  AZ.C     EQU             ENCOD.EL+1         CORRECTED AZ ENCODER ANGLE, COM FORMAT
00260  EL.C     EQU             AZ.C+1             CORRECTED EL ENCODER ANGLE, COM FORMAT

00261  AZ.CM1   EQU             EL.C+1             LAST SAMPLE AZ ENCODER ANGLE, COM FORMAT
00262  EL.CM1   EQU             AZ.CM1+1           LAST SAMPLE EL ENCODER ANGLE, COM FORMAT

00263  AZ.RFOC1 EQU             EL.CM1+1           AZ COMMAND, OFFSET ADDED,  FOC  BF 0
00264  EL.RFOC1 EQU             AZ.RFOC1+1         EL COMMAND, OFFSET ADDED,  FOC  BF 0

00265  AZ.RFOC2 EQU             EL.RFOC1+1         AZ COMMAND, CAM CORRECETED, FOC  BF 0
00266  EL.RFOC2 EQU             AZ.RFOC2+1         EL COMMAND, CAM CORRECTED, FOC  BF 0
```

```
00267  AZ.R      EQU       EL.RFOC2+1           AZ COMMAND, NO LIMITING, COM FORMAT
00270  EL.R      EQU       AZ.R+1               EL COMMAND, NO LIMITING, COM FORMAT

00271  AZ.R1     EQU       EL.R+1               AZ COMMAND, POS LIMITED ONLY, COM FORMAT
00272  EL.R1     EQU       AZ.R1+1              EL COMMAND, POS LIMITED ONLY, COM FORMAT

00273  AZ.R2     EQU       EL.R1+1              AZ FOR 2 SAMPL PER AFTER LAST SAMPL INTR
00274  EL.R2     EQU       AZ R2+1              EL FOR 2 SAMPL PER AFTER LAST SAMPL INTR

00275  AZ.R2M1   EQU       EL.R2+1              AZ FOR 1 SAMPL PER AFTER LAST SAMPL INTR
00276  EL.R2M1   EQU       AZ.R2M1+1            EL FOR 1 SAMPL PER AFTER LAST SAMPL INTR

00277  AZCOMAND  EQU       EL.R2M1+1            SUBSAMPLE AZIMUTH COMMAND
00300  ELCOMAND  EQU       AZCOMAND+1           SUBSAMPLE ELEVATION COMMAND

00301  ERROR.1   EQU       ELCOMAND+1           SUBSAMPLE AZ-EL ERROR

00302  AZ.CVEL   EQU       ERROR.1+1            AZIMUTH ANTENNA VELOCITY, COM FORMAT
00303  EL.CVEL   EQU       AZ.CVEL+1            ELEVATION ANTENNA VELOCITY, COM FORMAT

00304  AZ.RVEL   EQU       EL.CVEL+1            AZIMUTH COMMAND RATE, COM FORMAT
00305  EL.RVEL   EQU       AZ.RVEL+1            ELEVATION COMMAND RATE, COM FORMAT

00306  AZ.RVM1   EQU       EL.RVEL+1            LAST AZIMUTH COMMAND RATE, COM FORMAT
00307  EL.RVM1   EQU       AZ.RVM1+1            LAST ELEVATION COMMAND RATE, COM FORMAT

00310  AZ.SCAN   EQU       EL.RVM1+1            AZIMUTH SCAN, COM FORMAT
00311  EL.SCAN   EQU       AZ.SCAN+1            ELEVATION SCAN, COM FORMAT

00312  OFFAZ     EQU       EL.SCAN+1            OFFSET AZIMUTH  FOC  BF 0
00313  OFFEL     EQU       OFFAZ+1              OFFSET ELEVATION  FOC  BF 0

00314  AZCCWLMT  EQU       OFFEL+1              AZ CCW LIMIT   NARROW
                                                AZ CCW LIMIT   WIDE
00316  AZCWLMT   EQU       AZCCWLMT+2           AZ CW LIMIT    NARROW
                                                AZ CW LIMIT    WIDE

00320  ELDNLMT   EQU       AZCWLMT+2            EL DN LIMIT    NARROW
                                                EL DN LIMIT    WIDE
00322  ELUPLMT   EQU       ELDNLMT+2            EL UP LIMIT    NARROW
                                                EL UP LIMIT    WIDE

00324  AZVMIN    EQU       ELUPLMT+2            MINIMUM AZIMUTH VELOCITY
00325  AZVMAX    EQU       AZVMIN+1             MAXIMUM AZIMUTH VELOCITY

00326  ELVMIN    EQU       AZVMAX+1             MINIMUM ELEVATION VELOCITY
00327  ELVMAX    EQU       ELVMIN+1             MAXIMUM ELEVATION VELOCITY

00330  L*RANGE   EQU       ELVMAX+1             LOCAL RANGE IN EARTH RADII
00332  L*RNGRAT  EQU       L*RANGE+2            LOCAL RANGE-RATE IN HZ/GHZ
00334  R*RANGE   EQU       L*RNGRAT+2           REMOTE RANGE IN EARTH RADII
00336  R*RNGRAT  EQU       R*RANGE+2            REMOTE RANGE-RATE IN HZ/GHZ

00340  OFFDAYS   EQU       R*RNGRAT+2           UT TIME OFFSET DAYS  BI 6
00341  OFFOD/64  EQU       OFFDAYS+1            UT TIME OFFSET FOD  BF-6
00342  OFFSTFOD  EQU       OFFOD/64+1           ST TIME OFFSET FOD  BF 0

                           CONSOLE ALARM....EFFECTIVE IN WAKING UP THE OPERATOR

                           THE FOLLOWING CELLS MUST BE SET ACCORDING
                           TO THE ANTENNA CONTROL-PROCESSING ASSEMBLY

00343  ALARMSW   EQU       OFFSTFOD+1           ALARMSW HAS THREE POSITIONS...
00344  ALARM     EQU       ALARMSW+1            ACKNOWLEDGE PUSHBUTTON HAS FLASHING
                                                  LIGHT
00345  ACKRCVD   EQU       ALARM+1              MAINTAINS LIGHT, SET BY ACTUATION OF
                                                  PUSHBUTTON
00346  ALARMOFF  EQU       ACKRCVD+1            NO FLASHING LIGHT, NO ACTION UPON
                                                  ACTUATION
```

```
                              TEST CONTROL SWITCHES

                              PURPOSE     TEST CONTROL SWITCHES AND JUMP LIST FOR TEST
                                          CONTROL CONSOLE LOCATED SWITCHES....
                                                    A1, A2, A3, A4, A5

00347  TSCNTASW EQU           ALARMOFF+1

00350  .TSCNTA1 EQU           TSCNTASW+1
00351  .TSCNTA2 EQU           .TSCNTA1+1
00352  .TSCNTA3 EQU           .TSCNTA2+1
00353  .TSCNTA4 EQU           .TSCNTA3+1
00354  .TSCNTA5 EQU           .TSCNTA4+1
```

Appendix C

SYSTEM DYNAMICS

Suppose the Greenwich Mean Time to be 12:30:10 and the program-track digital submode to have been previously selected for tracking of an artificial satellite. The following satellite positions would be stored in the computer at this time (assuming that a basic time interval of 10 sec was selected by the software):

Position	For Time
Basic Computation Point 1	12:29:50.125
Basic Computation Point 2	12:30:00.125
Basic Computation Point 3	12:30:10.125
Basic Computation Point 4	12:30:20.125
Basic Computation Point 5	12:30:30.125
Sample Point 1	12:30:09.9375
Sample Point 2	12:30:10.0
Sample Point 3	12:30:10.0625

The basic computation points have been computed by the antenna processing level, which now has a job status of "complete." The sample points have been computed by the antenna control level, which is also complete.

Figure C1 shows the order in which the various Class A levels and Class B routines might be executed during four subsample periods beginning at 12:30:10. The length of each line in the diagram shows approximately how long the given software takes to be executed.

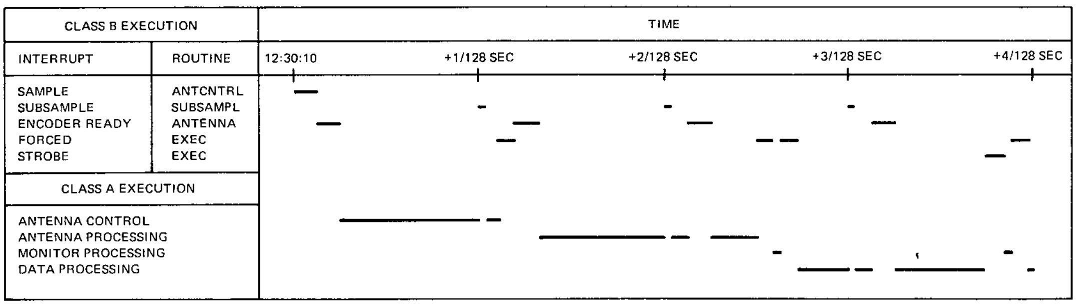

Fig. C1 - System dynamics for real-time operation. This illustration shows the various Class A levels and Class B routines which might be executed during four subsample periods beginning at 12:30:10.

The first occurrence is the sample interrupt, which causes the sample-interrupt processing routine ANTCNTRL to be entered. The antenna position encoders are interrogated and the antenna-control-level job status set to "not complete." The executive routine EXEC is then executed, through which control is passed to the antenna control level.

The antenna position has already been digitized by the encoders, however, so as soon as the EXEC releases interrupt lockout the interrupt for this occurrence is sensed. Interrupt lockout is again in effect, and the ANTENNA routine is entered in order to output position errors to the antenna. Since a new sample period is beginning, this routine updates the positions contained in the sample point table. This table then is arranged as follows:

Position	For Time
Sample Point 1	12:30:10.0
Sample Point 2	12:30:10.0625
Sample Point 3	12:30:10.0625

(Note that no new data has yet been entered for Sample Point 3.) Position errors are computed, based on the present antenna position and the computed command position for the time 12:30:10.0. The errors are output, and the interrupt routine is exited from. Execution resumes at the beginning of the antenna-control-level software.

Since the GMT is an even multiple of the basic time interval selected (10 sec), the decision is made to update the basic computation point table, which then contains the following information:

Position	For Time
Basic Computation Point 1	12:30:00.125
Basic Computation Point 2	12:30:10.125
Basic Computation Point 3	12:30:20.125
Basic Computation Point 4	12:30:30.125
Basic Computation Point 5	12:30:30.125

(As in the previous table, a new position has not yet been computed for Basic Computation Point 5.) The job status of the antenna processing level is set to "not complete," so a new basic computation point (point 5) will be computed. Since the sample point to be computed during this sample period (for 1/8 sec hence) is already contained in the table, no interpolation is necessary. The rest of the antenna control software, concerned with processing operator requests and modifying this position, is then executed. Before it is completed, however, a subsample interrupt occurs.

The interrupt is processed by the SUBSAMPL routine, which again interrogates the antenna position encoders and enables interruption when the positions have been digitized. Control is then returned to the antenna control level.

When this level finally completes the computation of the sample point command, it is inserted in the sample point table as a new third point:

Position	For Time
Sample Point 1	12:30:10.0
Sample Point 2	12:30:10.0625
Sample Point 3	12:30:10.125

The ENDOFJOB routine is then entered. The job status of the antenna control level is set to "complete," after which the forced interrupt is requested.

When this interrupt occurs, EXEC is entered. If standard priority is in effect, EXEC will now select the antenna processing level for execution. Again, however, the antenna position has been digitized and the interrupt sent to the computer. Therefore, when EXEC releases interrupt lockout, the ANTENNA routine is reentered.

This time ANTENNA performs linear interpolation on the azimuth and elevation sample point commands 1 and 2, to calculate the subsample command for the present time (12:30: 10.0 plus 1/128 sec). Position errors are again computed and output to the antenna servo.

The antenna processing level is then executed and a new basic computation point is generated. As seen from the diagram, this process is interrupted twice, once for another subsample interrupt, and then for the position-digitized interrupt. When the process is completed, the basic computation point is stored in the table:

Position	For Time
Basic Computation Point 1	12:30:00.125
Basic Computation Point 2	12:30:10.125
Basic Computation Point 3	12:30:20.125
Basic Computation Point 4	12:30:30.125
Basic Computation Point 5	12:30:40.125

The ENDOFJOB routine is then executed, setting the job status of the antenna processing level to "complete" and requesting the forced interrupt.

The executive routine now passes control to the monitor processing level. Suppose that on this level an I/O operation is initiated and the peripheral equipment is then checked for a ready status, which would denote the operation to be complete. When the status is found to be "not ready" the HOLDJOB routine is entered. The job status of the monitor processing level is set to "type 1 deferred," and the forced interrupt is again selected.

The executive routine, when entered, changes the monitor-processing-level job status to "not complete" and arms the strobe. It then passes control to the data processing level.

This level is engaged in some sort of computation process. It proceeds in this until another subsample interrupt occurs, followed shortly thereafter by another execution of the ANTENNA routine. The next time it is interrupted is by the strobe interrupt.

EXEC is again entered, and control is once more passed to the monitor processing level. The peripheral equipment is still not ready, so the HOLDJOB routine is again executed, after which EXEC passes control back to the data processing level.

This particular example has been chosen to give a basic idea of the dynamic processes of the real-time system; in practice, there is often no job to be performed on the data processing level. In the latter case the time which is shown in the diagram as being used by this level is spent in "dead space," that is, a do-nothing instruction is executed until some interrupt occurs which results in other software being executed.

The monitor processing level may also be different than is shown. For example, a program may be present which has long periods of computation and few I/O operations. In this case, the lines representing monitor processing time would be much longer than those shown in Fig. C1. They would be subject, however, to periodic interruptions for sample and subsample interrupts and antenna position-digitized interrupts. Also, this level might be engaged in waiting for typewriter input, either as part of a program being executed or because the resident's control section may be sensing for operator requests. In this case the typewriter status is not checked every 1/128 sec, but rather only once each sample period; the HOLDJOB routine would set the level's job status to "type 2 deferred," and no strobe interrupts would be used.

The antenna processing level would usually be absent from such a "picture" of software execution. For example, during the next sample period (starting at 12:30:10.0625), no updating of the basic computation point table is necessary. The sample point computed at this time (for 12:30:10.1875) is obtained by interpolation using the first four points already in the table. Therefore, the antenna processing level is left with its job status of "complete." In fact, it will not be set to "not complete" until 160 sample periods from now, at 12:30:20.0, at which time it will compute a new basic computation point for 12:30:50.125.

The diagram would also be quite different if certain unusual programs were being executed on the monitor and/or data processing levels which require a nonstandard priority scheme.

One interrupt condition has not been shown. This is the receiver-above-threshold interrupt, which may occur when the autoacquisition/autotrack submode is selected. This material is covered in Appendix D.

Appendix D

AUTOMATIC TRACKING MODE

Automatic acquisition and automatic tracking of artificial satellites is accomplished through the real-time interaction of various Class A and Class B software within the NACTAC system. The automatic sequences may be executed in either digital or non-digital mode, although nondigital use of these sequences requires a certain amount of manual positioning of the antenna by the antenna operator.

Antenna motion is under control of either the computer or special autotrack hardware (including an autotrack feed which must be inserted in the antenna) dependent upon the previous sequence of events within the system. In general, the autoacquisition and autotrack features function as follows (assuming digital mode has been selected): The antenna operator initiates a scan about the computed position of the satellite, and enables the use of one or both of two tracking receivers. If either receiver picks up the satellite tracking beacon, control of the antenna is given up by the computer and passed to the autotrack hardware. If the satellite is subsequently lost, control is passed to a special hardware device called "rate memory," which moves the antenna along its previous velocity vector in an attempt to reacquire the satellite. If this attempt fails, further action is up to the antenna operator; otherwise, autotrack is again initiated.*

The software providing control of these sequences comprises two computer routines. One of these, a Class A level routine which operates on the antenna control level, is called GROUP 9. The other is a Class B routine called RCVROP, which processes interrupt signals from either of the receivers. Such an interrupt indicates that the level of the received signal has gone above a certain threshold, i.e., that it is probable that the tracking beacon is now being received.

Initiation of an autoacquisition sequence is usually accomplished by the antenna operator pressing an "enable receiver" pushbutton. Upon receiving this command, the STATUSOP routine executes the select codes to enable interruption if the received signal goes above threshold. At this point, however, the software switch controlling GROUP 9 action is still set for an immediate exit from the routine, and remains so until an interrupt from the receiver is processed.

Subsequent action within this routine can vary widely, being basically dependent upon three variables. The first of these is the type of scan which has been initiated by the operator (spiral or elliptical). The second variable is the program number, which in this case differentiates between automatic acquisition and unconditional acquisition sequences. The last variable is the success the system has in acquiring the object, and its subsequent success in keeping locked on to the object while autotrack is in progress.

Figure D1 details the action taken by the RCVROP routine when an interrupt for "receiver-above-threshold" is received. Figure D2 shows the logic employed in the antenna control level GROUP 9 routine. In the latter diagram, the places at which the STATUSOP routine is requested to enable a receiver should be noted. This is done by

*Detailed descriptions of the autoacquisition and autotrack sequences, from the antenna operator's viewpoint, are provided in Ref. 3.

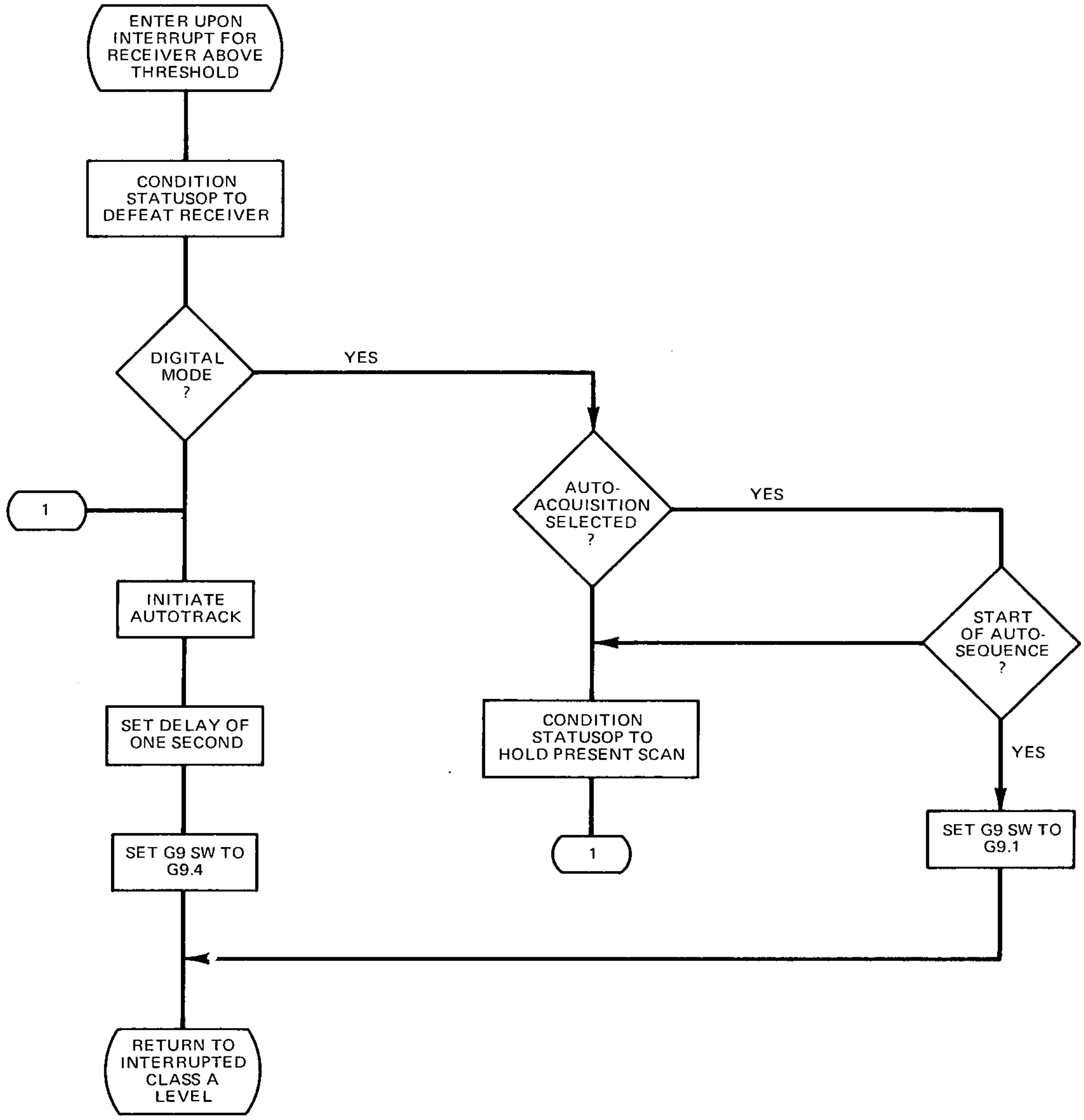

Fig. D1 - Flow diagram of RCVROP interrupt-processing routine for receiver above threshold

substituting the address of a dummy system status word at a special location within STATUSOP. During the next sample period, STATUSOP will execute the select codes to enable receiver interrupts, and will light the corresponding pushbutton on the control console.

This action is taken at two places within GROUP 9. In both cases, the GROUP 9 software continues to be executed during succeeding sample periods. However, it is hoped that during this time the receiver will again go above threshold and deliver an interrupt signal to the computer. In this case, the GROUP 9 software switch will be set to a different value by RCVROP, and subsequent action taken in GROUP 9 will be altered. If no interrupt occurs, GROUP 9 will eventually sense that the attempt to acquire the object has failed, and further action is taken as shown in Fig. D2. In general, the GROUP 9 software controls the times at which receiver interrupts are enabled, while the RCVROP routine sets the GROUP 9 software switch when such interrupts are received.

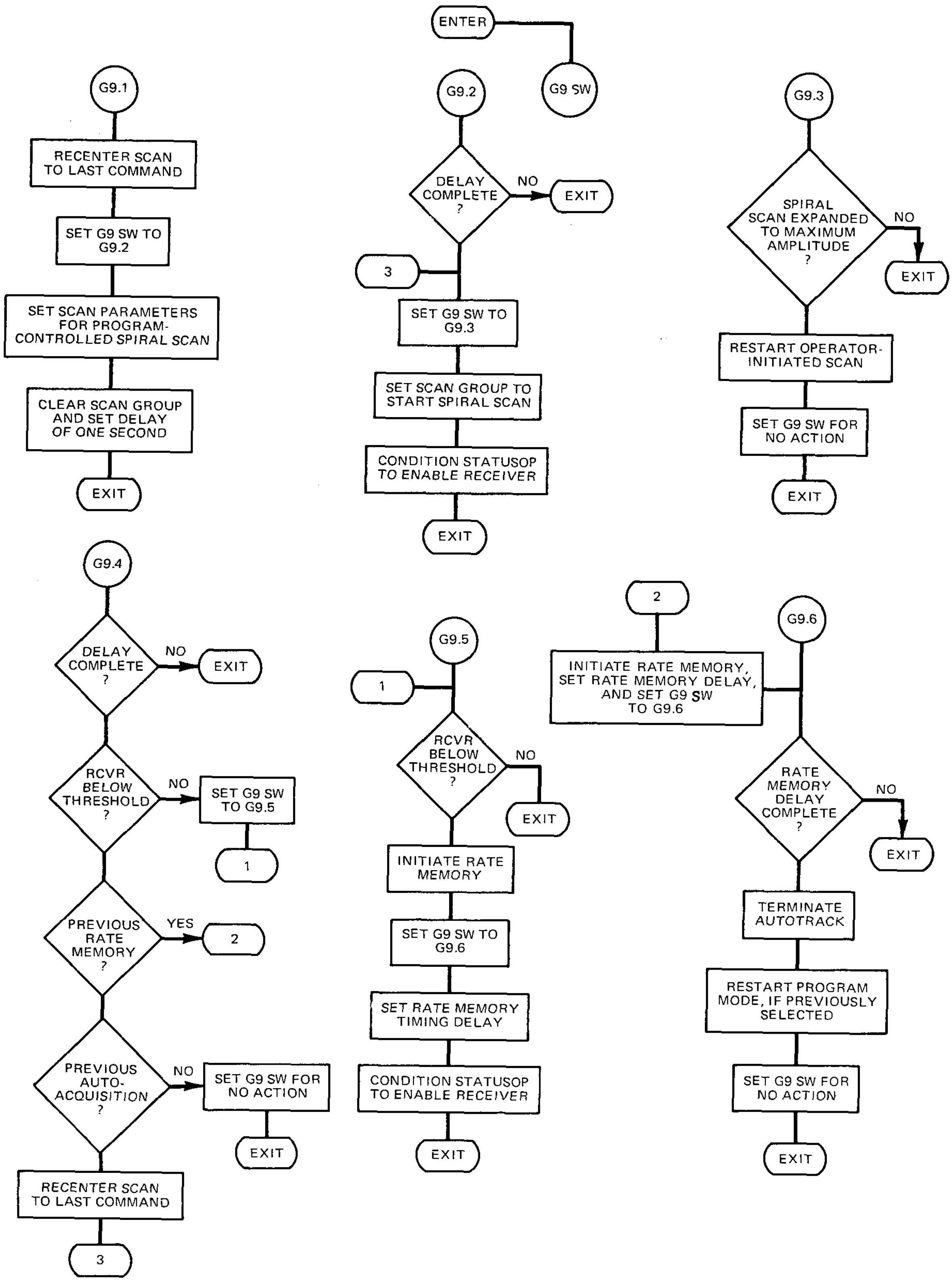

Fig. D2 - Flow diagram of GROUP 9 antenna-control-level autoacquisition and autotrack subroutine

Much of the logic shown in Fig. D2 is never executed if unconditional acquisition is selected by the operator, or if the system is in nondigital mode. In these cases, the first receiver interrupt processed will result in the antenna being put in the autotrack mode and the GROUP 9 software switch being set to G9.4. After a 1-sec delay, GROUP 9 checks whether the receiver is still above threshold. If not, the GROUP 9 switch is set for an immediate exit from the routine and no further attempt is made to acquire the object until the operator again requests the receiver to be enabled.

If the receiver is still above threshold, the object is declared to be locked in, and subsequently GROUP 9 simply checks that the receiver is still above threshold. If the object is lost after being declared locked in, the rate memory mode is initiated and the receiver interrupt is enabled. The software then begins timing out a rate memory delay, the length of the delay being dependent upon which receiver is being used. If no further interrupt occurs, the autotrack mode is terminated and GROUP 9 is set for no further action.

If the object is again found during the rate memory sequence, GROUP 9 logic flow resumes at G9.4. This time, however, if the receiver is below threshold after the 1-sec autotrack delay, rate memory is again initiated.

If automatic acquisition is selected, an additional step is inserted in this process, prior to attempting automatic tracking. When the initial receiver interrupt occurs, the antenna's azimuth and elevation are compared to the satellite's computed position. Offsets are calculated and saved for subsequent addition to sample point commands. A program-controlled spiral scan is then initiated. This is shown in Fig. D2 under the switch settings G9.1, G9.2, and G9.3. An additional procedure, not shown in the diagram, is accomplished if the scan selected by the operator was an elliptical scan. Part of the input for this scan is the velocity vector which the object is expected to have. Every time that GROUP 9 is executed, the antenna command is incremented along the velocity vector, if the GROUP 9 switch is set to G9.2, G9.3, or G9.4.

Figure D3 is a condensed version of the two previous flow diagrams, showing the steps taken for success and failure at various points in the autoacquisition and autotrack processes.

One additional option available to the operator has been incorporated in the GROUP 9 software. This is called the "program correct" feature. It is used when the "program" digital submode has been selected and is requested through the program number thumbwheel switches. When this feature is selected, the initial receiver interrupt processed is used to automatically add azimuth and elevation offsets to the computed position of the object being tracked. These offsets are added to any offsets which the operator may have entered, and the resultant total offsets are displayed on the console. The present scan is cleared, as is GROUP 9, so that this is simply an automatic acquisition feature used to update program track. As such, it may be used without installation of one of the special autotrack feeds.

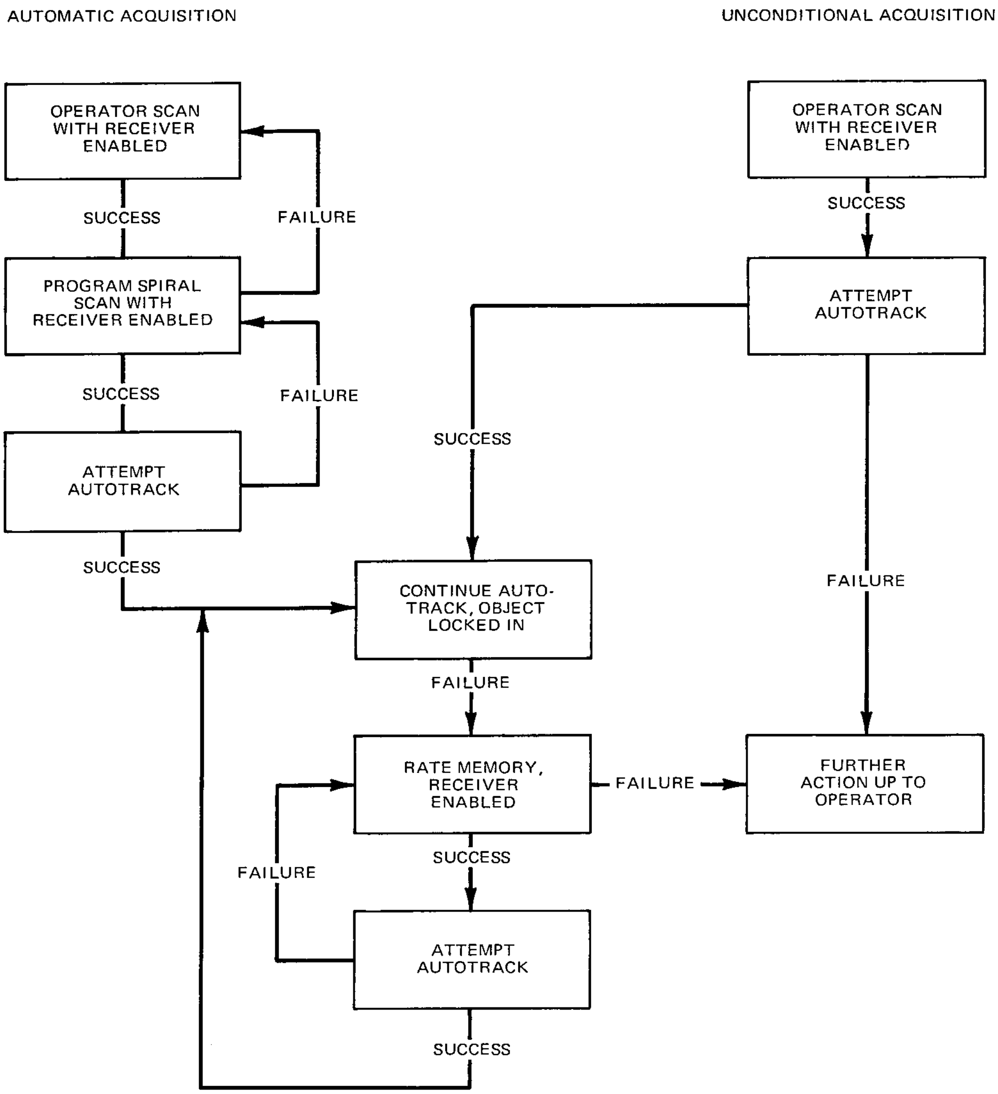

Fig. D3 - Condensed version of autoacquisition and autotrack sequence