An Engineering Study of Onboard Checkout Techniques

FINAL REPORT

TASK 2: SOFTWARE

Huntsville

IBM

An Engineering Study of Onboard Checkout Techniques

CR-115129
c1

FINAL REPORT

TASK 2: SOFTWARE

IBM NUMBER 71W-00112

MARCH 1971

Prepared for the
National Aeronautics and Space Administration
Manned Spacecraft Center
Houston, Texas 77058

CONTRACT NUMBER NAS9-11189

Prepared By: D M Welsh

Task Leader

Approved By: J. W. Simmons

J. W. Simmons
Study Manager

IBM Federal Systems Division, Electronics Systems Center/Huntsville, Alabama

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

LIST OF ILLUSTRATIONS

LIST OF TABLES

FOREWORD

This is one of a set of five final reports, each one describing the results of a task performed under Contract NAS9-11189, "An Engineering Study of On-Board Checkout Techniques". The five reports are as follows, all dated March 1971:

Task 1:	REQUIREMENTS ANALYSIS AND CONCEPTS (IBM No. 71W-00111)
Task 2:	SOFTWARE (IBM No. 71W-00112)
Task 3:	ON-BOARD MAINTENANCE (IBM No. 71W-00113)
Task 4:	SUMMARY AND RECOMMENDATIONS (IBM No. 71W-00114)
Task 5:	SUBSYSTEM LEVEL FAILURE MODES AND EFFECTS (IBM No. 71W-00115)

The nine month study was performed by the IBM Federal Systems Division at its Space Systems Facility in Huntsville, Alabama, with the support of the McDonnell Douglas Astronautics Company Western Division, Huntington Beach, California.

Technical Monitor for the study was Mr. L. Marion Pringle, Jr., of the NASA Manned Spacecraft Center. The guidance and support given to the study by him and by other NASA personnel is gratefully acknowledged.

Section 1

SOFTWARE TASK SUMMARY

1.1 INTRODUCTION

The primary objectives of this study task were to determine requirements for language and executive software to accomplish automated On-Board Checkout of a space station.

A summary of the analysis and recommendations derived from the accomplishment of these objectives is provided in the foregoing section. A more detailed description of the software study task is provided in subsequent sections of this report.

The analysis of the Space Station Data Management Subsystem (DMS), and the six other Space Station Subsystems resulted in the recommendation that an interpretive high level language and a multi-level executive be used to satisfy the checkout requirements.

The specific elements which have been defined for this language are summarized in Table 1-1.

The requirements of the multi-level executive which are recommended by this study are summarized in Table 1-2.

The on-board checkout techniques defined for each subsystem, and the software sizing summarizations are included in Appendices A through E to this report.

Table 1-1. Element Summary

ELEMENT	
ALLOCATE	Used to assign a configurable group of components to a program.
BEGIN	Defines the start of a sequenceof elements.
CALL	Accomplishes subroutine linkage.
DEFINE	Defines variables and symbols.
DELAY	Suspends program execution for a specified time period.
DISPLAY	Presents messages on the display.
EJECT	Provides the capability to begin subsequent information on a new page.
END	Defines the end of a sequence of elements.
EXECUTE	Schedules a specified task for execution.
GOTO	Executes an unconditional branch.
IF	Compares and executes a conditional branch.
MEASURE	Measures specified test points.
READ	Retrieves data from auxiliary or main storage.
SEARCH	Processes character strings.
SOLVE	Computes arithmetic expressions.
SPACE	Provides the capability to insert blank lines in the program listing.
STATUS	Obtains or sets the status of a configurable group of components in the data base.
STIMULATE	Applies a specific stimulus to a specified test point.
STOP	Terminates execution of a task.
TIME	Obtains current time.
WRITE	Transmits data to auxiliary or main storage.

Table 1-2. Executive Requirements Summary

Scheduling

- Time- and event-oriented program initiation
- Processor allocation on a basis of relative task importance

Support Services

- Compiler
- Interpreter
- Assembler
- Utility/maintenance programs
- System initialization

System Communication

- Communication between the crew and the Data Management Subsyster
- Provide status information on request
- Provide alarms to advise crew of detected failures
- Provide capability to request program execution from the keyboard
- Provide "desk calculator" capability
- Generate and update programs from the keyboard

Resource Allocation

- Memory allocation strategy which minimizes contention and fragmentation
- Allocation of resources on a basis of relative task priority
- Data Base characteristics
    - Control of access for acquisition or modification of data
    - Hierarchical relationship of content data
- Data Base contents
    - System status

Table 1-2. Executive Requirements Summary (continued)

  - Redundant elements and paths
  - Schedules and procedures
  - Inventory control
  - Association of symbolic test points with absolute addresses

Data Handling

- Exchange of data between computer and external equipment
- Recognize the receipt of data and define the source
- Route the data for subsequent processing or retention

System Recovery

- Maintain current system configuration
- Control redundant hardware elements
- Checkpoint data

Interruption Servicing

- Identify routines to be executed on a basis of internal interruption signals
- Identify the Remote Data Acquisition Unit (RDAU) associated with an I/O interruption, eliminate superfluous interruptions, and select the appropriate program.

1.2 APPROACH

The approach which led to the foregoing conclusions consisted of three major tasks:

- Preliminary System Analysis
- Analysis of each on-board subsystem
  - Structures
  - Guidance Navigation and Control
  - Electrical Power
  - Environmental Control and Life Support
  - RF Communication
  - Propulsion
- Synthesis of Software requirements

1.3 PRELIMINARY SYSTEM ANALYSIS

The preliminary system analysis phase evaluated current concepts, and established baselines for the language and executive system requirement analysis to accomplish on-board checkout. The specifications for the TOOL and OCS-II programming languages, and the results of the Phase B Space Station Studies were supplied by NASA as inputs to this study. This study approach is depicted graphically in Figure 1-1. The results of the preliminary system analysis provided the baseline for the recommended language design concepts.

1.3.1 LANGUAGE SELECTION

The initiation of any project involving extensive software development requires that emphasis be given to the selection of the languages to be used. This emphasis contributes to an efficient and cost effective technical approach. The language selection is primarily dependent upon the objectives to be achieved and the operational environment. This selection can vary from micro-programming to high level language.

The environment in which the language for the space station will be applicable involves both ground and on-orbit facilities. The primary software efforts involving ground operations will be related to factory and prelaunch checkout. The software required for on-orbit operation will also require significant software development capabilities.

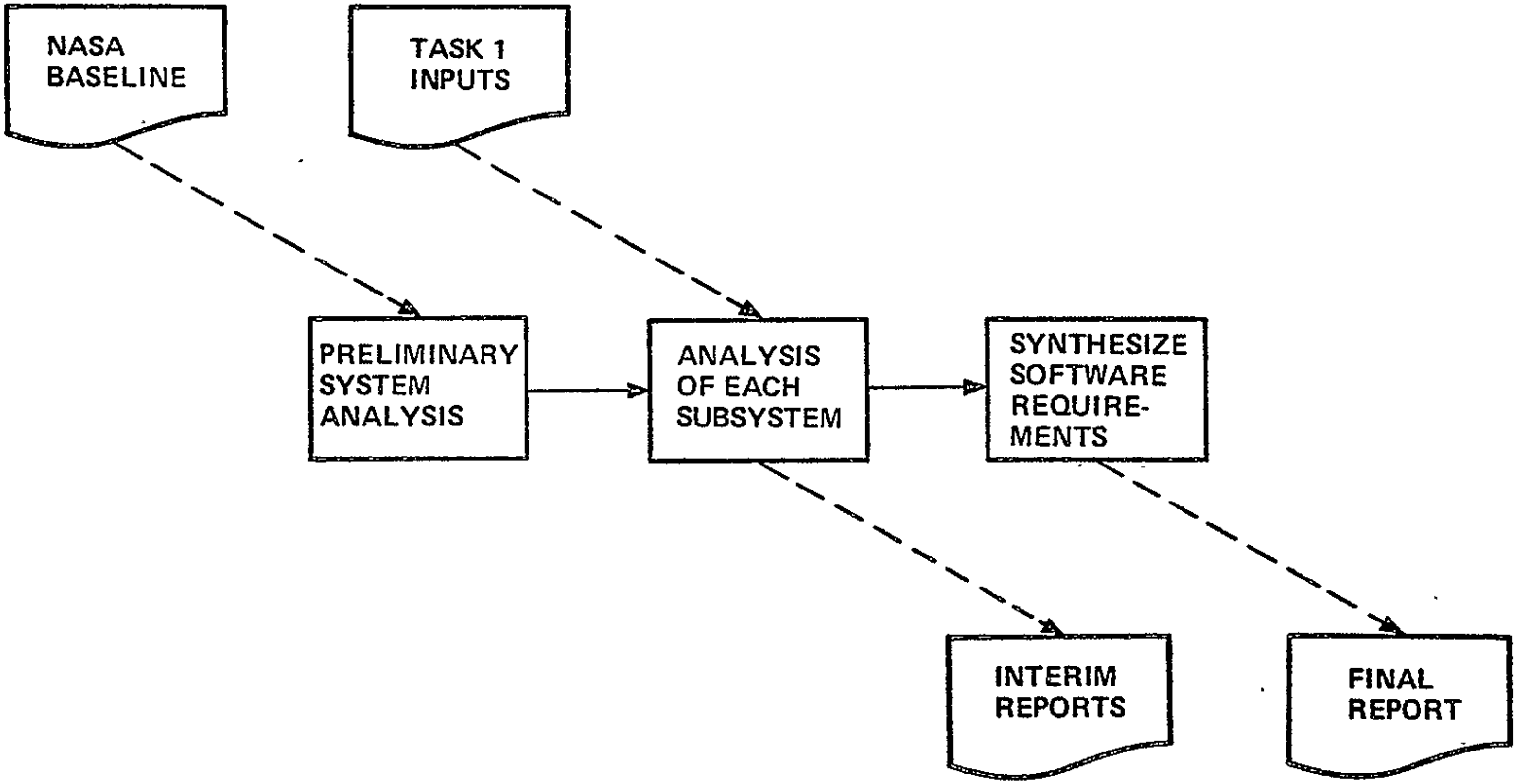


Figure 1-1. Software Task Study Approach

This study recommends that the checkout software be developed and validated in ground-based facilities; however, it is recommended that an on-orbit programming capability be made available as a tool to support scientific experiments to provide for a continual evolvement of the on-board checkout capability,and to provide the capability to respond to unexpected situations which may occur during a mission.

The specific language objective in this study is to meet the requirements for automated on-board checkout in an efficient and cost effective manner. The specific requirements of the language were identified by analysis of the subsystem hardware.

This study recommends the use of a high-level language to accomplish the automated on-board checkout function. The advantages of a high-level language which will contribute directly to the achievement of the stated objectives include:

- Reduced cost associated with the development and maintenance of application programs.
    - Ease of learning for all levels of man-machine interface (factory checkout, prelaunch, on-orbit).
- Increased level of software reliability which can be achieved.
- Independence of the application programs from the physical characteristics of the computers.
    - Implementation compatibility for different computers.
    - Implementation of evolving technological hardware developments with minimum impact on the software.

Additional considerations which influenced the decision in favor of a high level language include:

- The capability to meet current requirements associated with a space application in both an orbital and pre-launch environment.
- The naturalness of the language to the subsystem engineer who is expected to be the primary user.
- The minimal time period in which the user would be able to learn and apply the language.

- Cost considerations relative to the development and maintenance of the language and application software.

- Economical incorporation of evolving requirements through extension of existing capabilities, and creation of new language elements.

- The scope of the language must be compatible for such diversified activities as factory checkout, prelaunch, and on-orbit operation. This includes compatibility with various host computers in the factory checkout environment.

1.3.2 LANGUAGE ARCHITECTURE

The analysis of language requirements for an application as extensive and diverse as the space station required the evaluation of a large number of considerations. Therefore, the requirements for on-board checkout were placed in perspective with the entire space station environment by defining a language which would be appropriate for all on-board systems.

Selection of a suitable architecture for the language was based upon two requirements. The first requirement involved the capability to develop computer programs. The second requirement involved the means for man to communicate with, control, and provide direction to the Data Management Subsystem. These two functions are referred to in this study as the Program Development and System Communication Functions of the language.

It is recommended that the interrelated functions of system communication and program development be accomplished using one common language. This commonalty will permit the syntax (the rules governing the statement structure) of the language used to control the Data Management Subsystem be identical to the syntax of the language used to develop the application programs. A detailed description of the recommended language syntax is included in Section 3 of this report. A cost saving in the required learning curve can be achieved by implementing this concept of commonalty.

The program development and system communication functions of the Space Station Language are divided into subsets, each related to a specific purpose.

A basic subset containing elements which are anticipated to be common to all language applications is included. Additional subsets meet requirements which are unique to a specified area, such as on-board checkout or experiment applications. This concept allows the user to know only the basic subset, and the subset unique to his application in order to meet the programming requirements for his subsystem.

The System Communication elements meet the requirements of controlling both the Data Management System and the application programs.

Several advantages accrue from this subset concept:

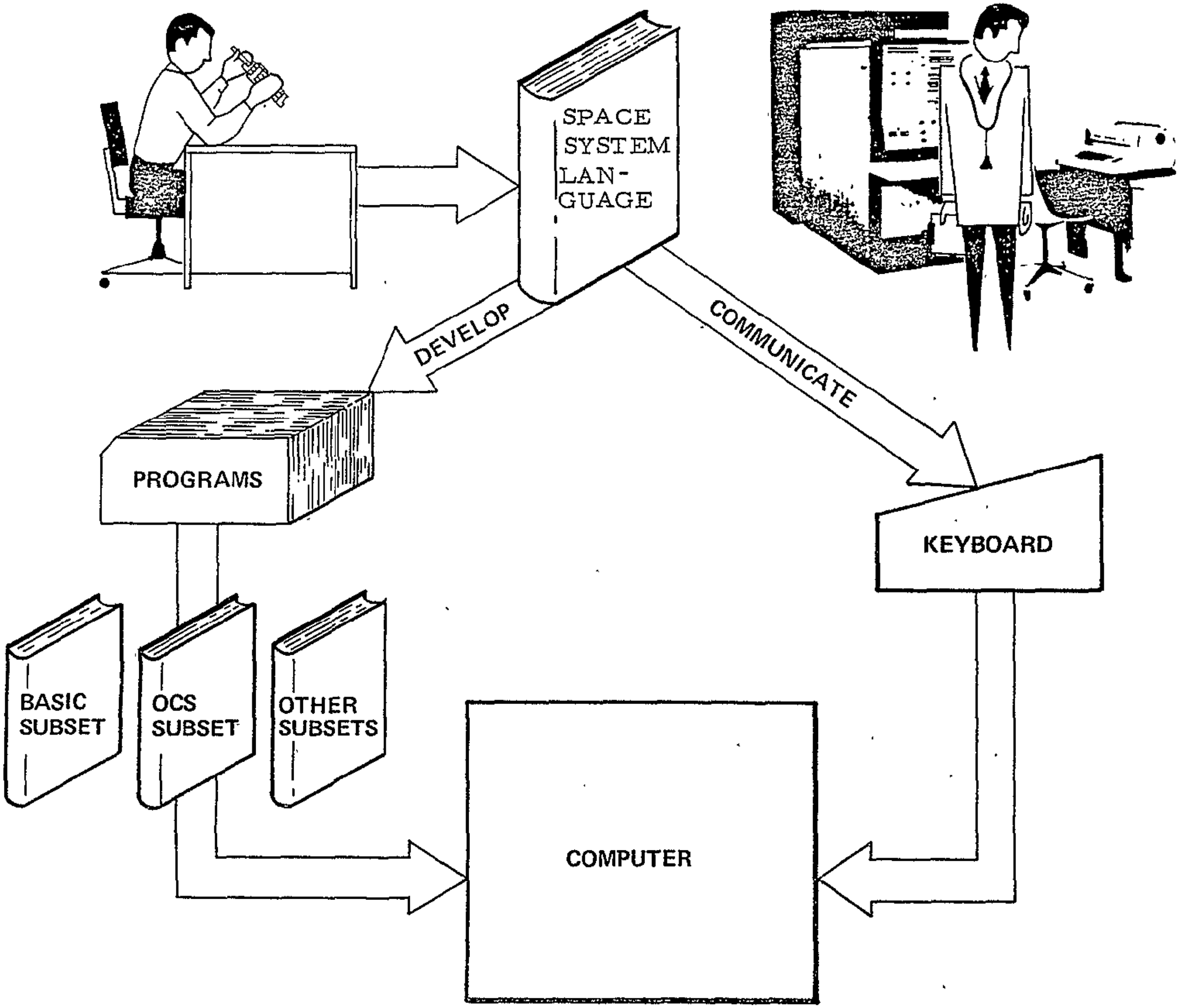
- EASE OF LEARNING - individuals must learn only the basic subset and their own unique subset. Programming expertise is not required of the user.
- INTER-DISCIPLINARY COMMUNICATIONS - all phases of Space Station development will be able to discuss the software in the same frame of reference.
- REDUCED COST - the use of a high level language will reduce the time required for program development and maintenance. In addition, the cost of subsequent enhancements will be reduced, since the languages can be easily expanded to include additional subsets. Individual subsets can be modified without impacting other areas.

Figure 1-2 provides a graphic representation of the recommended language architecture.

The task of learning a language can be extensive, and the amount of time required to achieve a usable level of proficiency is relative to the complexity of the language. The term complexity is used to refer to the combination of the number of elements, the number of element options, and the syntax of the language. It is apparent that the fewer elements and options which must be learned, the easier the task. However, as the number of elements decreases, so does the flexibility (depth, sophistication) of the language.

In order that the language be easy to learn while acquiring the depth necessary for all required applications, the learning level approach is recommended. This concept classifies elements and element options as level 1 and level 2, respectively. Level 1 includes the minimum amount of information for those elements and associated options necessary to successfully develop or modify a program. Level 2 includes element options not included in Level 1, and additional elements. The intent is that the Level 2 features will enhance the flexibility and sophistication of the system. This concept significantly facilitates an incremental learning approach.

Figure 1-3 provides a graphic description of this concept.



LANGUAGE ARCHITECTURE

Figure 1-2. Language Architecture

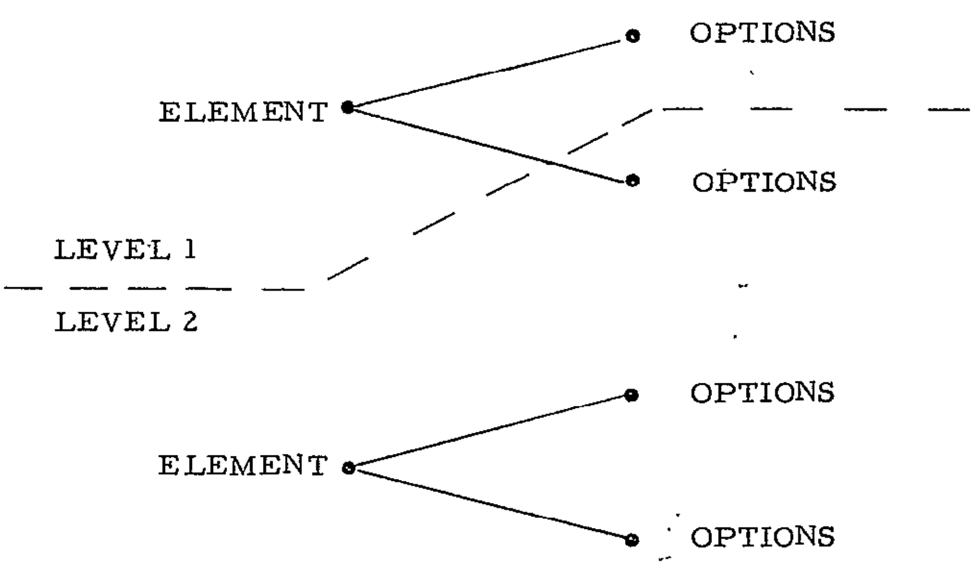


Figure 1-3. Learning Level Concept

1.3.3 BASELINE ELEMENTS

In general terms, the elements of a language incorporate the capability to define data (declarative statements), accomplish specific processes (imperative statements), include the mechanisms for the linkage of statements, and acquire the required data.

With the language objectives established, an an appropriate language architecture selected, a group of baseline elements were selected for their ability to meet anticipated On-board checkout requirements.

The baseline elements were defined using documentation on the TOOL and OCS-2 languages previously developed under NASA Manned Spacecraft Center direction, and other appropriate test languages. The syntax of the elements was defined as a floating format, eliminating positional restrictions. This format delimits input fields by the use of blank spaces and commas, instead of fixed positions on a keyboard/display or input card. In addition, options were associated with each element to efficiently meet the requirements. The selected elements were subsequently divided into OCS and Basic subsets. The rationale for the selection of the baseline elements and the description of each element required to support on-board checkout is contained in Section 3 of this report.

1.4 SUBSYSTEM ANALYSIS

The analysis of each subsystem involved evaluation of a basis of the subsystem definition provided by Task 1 of this study contract. The subsystem analysis activities developed a checkout strategy, defined language and executive requirements, and provided sizing estimates of the software required for automated on-board checkout on a subsystem basis. The results of this analysis for each subsystem were synthesized to define the total subsystem checkout requirements.

1.4.1 CHECKOUT STRATEGY

The recommended software checkout strategy involves a sequence of detecting faults, isolating faults to a failing LRU or LRU's, and reconfiguring the system to continue operation while the failures are being repaired.

This recommendation was developed by evaluating each subsystem with respect to the three general requirements of fault detection, fault isolation, and reconfiguration.

Fault detection incorporates both the recognition of failure occurrence, and the prediction of when a failure can be expected to occur. The Remote Data Acquisition Units (RDAU's) continually check selected test point measurements against upper and lower limits, and notify the executive on an exception basis when a limit is exceeded. This approach avoids occupying the central multi-processor with the low-information task of verifying that measurements are within limits.

Trend analysis is a fault detection technique recommended for predicting the time frame during which a failure can be anticipated. Data is acquired on a basis of time or utilization, and compared with previous history to determine if a "trend" toward degraded performance or impending failure can be detected.

Another checkout requirement evaluated for each subsystem is periodic testing. This type of test is provided to exercise specific components at extended time intervals or prior to specific events, to assure operational integrity. In the event that a failure is detected, the periodic test will isolate to the failing Line Replaceable Unit (LRU) and accomplish recertification after a repair operation.

Calibration of specific subsystem components will be required periodically, or subsequent to a repair and/or replace operation. The techniques involved are unique to the individual component; and, in some cases, require the acquisition of operational data.

Fault isolation is required when a fault is detected. When a particular fault provides an indication that a life critical failure has occurred, the fault isolation routines are automatically initiated. If the failure does not represent an immediate danger to the vehicle occupants, the crew is notified and they will initiate the fault isolation modules at their convenience.

The basic requirements of the fault isolation function is to analyze the available information relevant to a problem, and identify the LRU which is responsible for the anomaly.

Three basic approaches to meeting this requirement were considered. These are:

- Analyze each fault as an independent problem.
- Analyze each fault with a state matrix which defines the possible error states of the subsystem.
- Associate each fault with a specific subsystem, and evaluate that subsystem in detail.

The third approach was selected on a basis of software commonality and cost effectiveness. The complexity associated with the testing can be reduced by localization of the logic associated with the analysis of the subsystem in a unique package. The software commonality will result in reduced software development and maintenance costs, while increasing the reliability of the software.

The fault isolation software is structured modularly for compatibility with the hardware structure of the subsystem. Checkout modules evaluate the performance of a specific portion of the subsystem. A convenient division for this modular structure is at the assembly level or functional area. A program module which can determine and control the sequence in which these checkout modules are executed is also required for each subsystem.

Subsequent to fault detection, the software associated with the subsystem which is most likely to contain the error will be activated.

The subsystem software will analyze the error indication, and initiate a sequence of checkout modules to isolate the problem. If successful, the crew is notified regarding the Line Replaceable Unit (LRU) to be replaced. If an error cannot be identified, the crew is informed of the situation and has an option to execute the periodic test of the subsystem.

After a fault has been isolated, reconfiguration software restores the functional capability of the subsystem. This is most commonly accomplished by exchanging a redundant element for the failing unit, or by defining an alternate path to accomplish the required function.

1.4.2 LANGUAGE AND EXECUTIVE REQUIREMENTS

After analyzing each subsystem in regard to checkout strategy requirements, the area of language requirements was approached. The flowcharts which were developed for selected subsystem components during the development of the checkout strategy were a significant input to this activity. These flowcharts were expanded and analyzed to define the type of operations required to accomplish checkout, and the language elements which could most efficiently meet these requirements.

The baseline elements and associated options were evaluated on a basis of the ease and efficiency with which they could meet the logical, I/O, and arithmetic operations required to accomplish automated on-board checkout, within the constraints established during the preliminary phase of the task. The elements which were selected are summarized in Table 1-1, and are described in detail in Section 3 of this document.

Concurrently with the analysis of language elements for each subsystem, the executive requirements which were associated with each element also received consideration.

1.4.3 PROGRAM SIZING

Each subsystem was also reviewed to provide software sizing estimates for processor time, I/O time, and memory requirements. Table 1-3 reflects a summary of the estimates which would be typical to perform a periodic test on each subsystem.

A significant conclusion which can be drawn from this table is that automated on-board checkout with remote limit checking, will require minimal processor time, and that run time of the program will be highly dependent upon the efficiency of the I/O interface.

1.4.4 SUBSYSTEM SOFTWARE DEFINITION

The analysis of each subsystem resulted in the preparation of preliminary design definitions describe the software which would typically be used to accomplish the checkout of each subsystem. Data for each subsystem includes:

- A description of the required software modules
- Interface descriptions
- Language requirements
- Executive requirements
- Program sizing

These documents were developed as the study progressed, and are included in Appendices A through E for each subsystem to provide background for the overall conclusions.

Table 1-3. Subsystem Periodic Test Characteristics

Subsystem	Time (Minutes)			Memory (Words)	
	CPU	I/O	TOTAL	Main	Aux.
Propulsion					
Low Thrust	.010	2.649	2.659	6K	8.5K
High Thrust	.015	4.745	4.760	5K	10K
RF Communications	.025	5.024	5.049	13.5K	20K
Guidance Naviagation and Control	.022	3.004	3.026	12K	12K
Electrical Power					
Isotope/Brayton	.021	3.310	3.331	15K	15K
Solar Array	.042	5.302	5.344	15K	15K
EC&LS	.020	4.580	4.600	16K	33K
Structure	.005	2.913	2.918	4.5K	4K

1.5 EXECUTIVE REQUIREMENTS

The incorporation of a multi-level executive, consisting of a Master Executive and an Onboard Checkout Executive, is recommended because of its adaptability to modularization of the required program functions. This concept contributes to the simplification of the definition and control of the program modules.

The significance of program modularization is in its contribution to reducing the logical complexity of the system by providing clear decisions of responsibility. The position of a program module in a hierarchical structure is based upon its functional responsibility and the extent of supervisory influence which it exerts.

Figure 1-4 depicts the major hierarchical levels and associated interfaces in the executive structure recommended by this study.

The determination of Executive System requirements was approached by defining seven general areas of executive functional responsibility, and categorizing each identified requirement. Once the requirements were identified and categorized, it was necessary to determine the hierarchical level at which each specific requirement could best be accomplished on a basis of programming and run time efficiency.

The specific areas of functional responsibility which were established are:

- Scheduling
- Support Services
- System Communication
- Resource Allocation
- Data Handling
- System Recovery
- Interruption Servicing

1.5.1 SCHEDULING

A task is a unit of work for the DMS multiprocessor; and task scheduling is the executive function which controls the flow of work. A multiprogrammed scheduling function is recommended for the space station multiprocessor configuration to permit concurrent execution of two or more tasks. A priority dispatching technique will select the tasks on a basis of their relative importance to the operational objectives of the system.

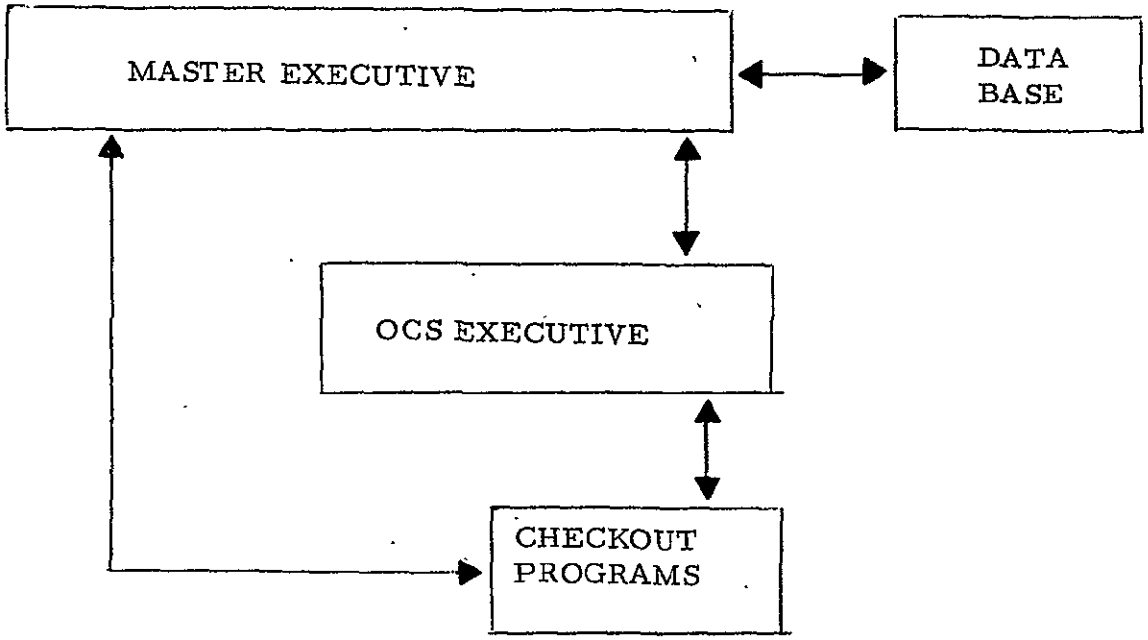


Figure 1-4. Multi-Level Executive Relationships

This strategy utilizes task priority level, memory requirements, and estimated execution time to select tasks, initiate and control task execution, and terminate task execution. This strategy provides the required time and event oriented scheduling capabilities. The initiation of programs on a time oriented basis is required for the trend analysis and selected periodic check routines. Event oriented scheduling requirements include the initiation of critical fault isolation sequences, and programs which are requested by the crew.

1.5.2 SUPPORT SERVICES

The services of an executive program are primarily concerned with providing the software to meet the requirements for program generation and maintenance. The executive support services required as a minimum to support the recommende high-level language include an interpreter and a compiler.

A compiler program, which will execute in both an on-line and off-line environment, will process the source statements and build the appropriate data list tables. An interpreter program will process the data lists and execute the associated operations.

Another system requirement which is defined in this functional area involves the initialization of the system. The software required to accomplish this must contain the logic required to determine the assignment of the multiprocessor to either the operational or experiment functional areas.

1.5.3 SYSTEM COMMUNICATION

The functional area of system communication is integral to the establishment of an efficient man-machine interface. This application requires an executive program to provide the interface between the operator and computer system through a keyboard/display.

The language used to accomplish this interface with the computer is commonly referred to as job control, or command language. The executive interprets the inputs and initiates the designated activity.

Specific requirements identified relative to the execution of checkout application programs include:

- The capability to provide variable data to a program at both load and execution time.

- The ability to specify options to control the sequence of program operation.

- Permit manual response to a detected error situation.

  - Retest
  - Terminate
- Require operator concurrence prior to issuing specific stimuli.

1.5.4 RESOURCE ALLOCATION

The function of resource allocation is responsible for minimizing interference which can result when concurrent tasks share the same resources. The efficient allocation of memory, external devices, and the data base will reduce the processor time which is lost due to memory contention, and the delays associated with input/output operations.

Memory allocation is responsible for determining the memory currently available for allocation to new tasks, and memory which is already in use. The effects of memory contention can be minimized by a strategy which allocates unique memory modules to concurrent requirements. The memory allocation strategy should also minimize the effect of memory fragmentation by evaluating the total memory which is unavailable due to fragmentation, and reassign memory only when system performance is being impacted.

The function which allocates subsystem resources is particularly important to the On-Board Checkout System. It is essential that interference be eliminated between concurrent requirements. An example of this interference would be an attempt to initiate the periodic test for a subsystem which is currently performing a necessary operational function. These considerations prevent the use of a subsystem or component by a program whose requirements could conflict with the current operation of the subsystem. Executive modules which provide this capability must maintain information relative to the current use of each subsystem component where the possibility of interference exists; and must assure that two conflicting requests are not issued.

This study recommends that the allocation of equipment (displays, printers, and subsystems), be accomplished using a priority queuing technique which would sequence requests to insure that the tasks access the equipment on a non-interference basis, and in consideration of their relative priorities rather than arrival order.

1.5.5 DATA HANDLING

The data handling capabilities of the executive must support the system requirements for acquisition, routing, and retention of information. In a multi-processor environment, tables will be updated by one processor and used by another. Precautions are required to prevent one processor from accessing a particular table while it is being updated by a second processor. Specific requirements which have been identified to be incorporated into a data base are as follows:

- System Status
- System Configuration
- Redundant elements and paths
- Schedules and procedures
- Inventory control
- Association of symbolic test points with physical addresses
- Program modules

The data base concept which is recommended to meet these extensive and varied requirements involves hierarchical relationships between the specific data which must be retained, with responsibility for operations on the data base vested in an executive function, instead of with application programs. This permits control to be maintained in a central routine, with the result that data base integrity is enhanced.

Techniques are required to handle the exchange of data between programs and between external equipment. The functional capabilities which are required in this area are:

- Exchange of data.
- Recognition of the receipt of data and source identification.
- Routing of data for subsequent processing or retention.

These functions are commonly accomplished by the executive input/output (I/O) routines which provide the interface between the application programs and the external I/O devices. In general, the functional capability which is required in the executive system to assure compatibility with this design concept includes:

- Transfer of data between subsystems and the computer.
- Identification of the specific subsystem.
- Analysis (limit checking, compaction) of the data.
- Presentation of messages on display units.
- Test point measurement.
- Issuance of stimuli to test points.

When errors occur in the equipment they must be analyzed, and the appropriate response taken. This action, dependent upon the problem, includes a subset of the following items:

- Return error analysis information to the program which requested the I/O operation.
- Maintain error statistics.
- Reset the I/O device.
- Retry the operation.
- Notify the operator.
- Initiate recovery procedures

In addition to the above, the inclusion of special processing routines to meet the data retention requirements of trend analysis, data logging, and checkpointing is required for the Space Station application. These requirements will necessitate allocating areas of auxiliary storage to hold the data, and periodically deleting data which is no longer required.

1.5.6 SYSTEM RECOVERY

The functional area of system recovery relates to the capability of the space station subsystems to remain operative in the presence of failures. The primary responsibility in this area involves the analysis of a failure, and reconfiguring the system to reduce the impact of the problem. Specific requirements related to this area include configuration management, redundancy control, and the interface to assure that the checkpointing of data is accomplished.

Redundancy control refers to DMS control and accountability of system elements which are redundant. It is essential that the DMS have the ability to alternate redundant elements between primary and secondary status, in order to assure operational readiness.

Configuration management routines establish and maintain the information which provides the current system organization description required for redundancy control.

The functional area of system recovery is also responsible for assuring that timely checkpointing of the appropriate data is occurring.

System recovery is also responsible for coordination of program termination when required, prior to successful completion of an application program. This function must assure that any subsystem elements which may have been affected by the program are restored to a proper configuration prior to continuation of other system operations.

1.5.7 INTERRUPTION SERVICING

The real time response requirements of the space station environment are particularly adaptable to interruption servicing. The interruption servicing routine performs the following functions:

- Save machine status (i.e., processor status, register contents) at the point of interruption.
- Identify the interruption source and type.
- Select and execute the appropriate program module.
- Restore machine status and return to the interrupted program.

Specific requirements identified by this study relative to on-board checkout relate primarily to responses to out-of-tolerance indications which are received from the RDAU's. Special emphasis is required to analyze multiple failure indications. The study focuses attention on the problems encountered as the result of an error whose effect will impact other areas. When this situation occurs, measurements associated with downstream components will exceed tolerance limits. These measurements must be evaluated to determine if they are within the scope of the original problem, or represent a unique problem.

The approach which is currently considered appropriate is to disable the RDAU limit check capability, thus eliminating repetitive interruptions. The RDAU scan of those test points which could normally be expected to exceed limits in view of the detected error, is also disabled. When the fault isolation sequences have successfully isolated the problem and repair has been accomplished, limit checking is reinitiated.

1.6 REVIEW

The software task summary has provided an overall identification of the Software Task study results. A more detailed description of the study performance is presented in the subsequent sections and appendices of this report.

Section 2

EXECUTIVE ARCHITECTURE

2.1 INTRODUCTION

This section describes the design requirements of an executive program structure to support the autonomous on-board checkout mission, as well as other data processing missions of the Space Station Data Management Subsystem. The Data Management Subsystem baseline consists of six processors sharing a large capacity main storage and peripherals, with a data bus for acquisition and control of digital and analog data associated with the other Space Station subsystems. Auxiliary storage complements that supplied by shared main storage.

2.2 APPROACH

In order to synthesize an executive design, the architecture and capabilities of the Data Management Subsystem were initially studied to determine the general attributes of an executive program which would be effective in the DMS. The specific executive requirements of on-board checkout were then studied for the following Space Station subsystems:

- Structure (Appendix A)
- Guidance, Navigation, and Control (Appendix B)
- Electrical Power (Appendix C)
- Environmental Control and Life Support (Appendix D)
- RF Communications (Appendix E)
- Propulsion (Appendix F)

Finally, the specific requirements of the subsystems were used to mold the design requirements of the on-board executive program, operating in the DMS multiprocessor.

While the purpose of this study is to examine techniques for the On-Board Checkout System, other systems are considered in the design of the executive. As a result, a multi-leveled executive design has evolved which provides services at the upper levels which are oriented toward specific systems, and which utilize the more general services in the lower executive levels.

2.3 EXECUTIVE FUNCTIONS

The Space Station Executive is a program which supports the on-board checkout functions of status monitoring, fault isolation, reconfiguration, periodic check, calibration, and trend analysis. The requirements of the Master Executive are to service interruptions, provide an interface between hardware and other programs, allocate the processor resource of DMS to tasks on a priority basis, and serve as a repository for routines which have common usage among higher-level executives such as the OCS Executive and the Experiment Executive.

2.3.1 MULTI-LEVEL APPROACH

The executive design is a multi-level one in that functions are stratified with regard to their scope of use. At the lowest level are functions which provide the interface with hardware devices such as the data bus controller. All on-board programs being executed by the multiprocessor make use of the low-level functions, regardless of whether they are checkout oriented or not. Low-level executive services are normally used by the higher level functions, rather than directly by the application programs.

The lower levels of the executive are referred to as the Master Executive because of the central control and interface with the DMS hardware which they provide. The upper levels of the executive are oriented toward individual systems of the Space Station; therefore the OCS Executive functions are generally found in the upper levels of the hierarchy.

Portions of the OCS Executive reside in main storage, and are closely aligned with the Master Executive routines because of their frequency of use. From the standpoint of how processor control is given to certain OCS routines, these routines are indistinguishable from some of the Master Executive routines; however, they are categorized as OCS routines because their function has no wider application than on-board checkout.

2.3.2 CENTRALIZATION VS. DECENTRALIZATION

In order to perform detection, isolation, and reconfiguration in an integrated and autonomous checkout system, a centralized data base is required to achieve coordination. There appears to be no trade involved in the area of whether to centralize or decentralize that portion of the DMS which fulfills the coordination function.

If it was desirable to disperse DMS functions as much as possible, the minimum which would require centralization would be a configuration table reflecting the logical relationships among assemblies of the Space Station. Shared auxiliary storage could be used to contain the configuration table. However, by centralizing, definite advantages relating to equipment utilization and operational capability can be realized.

From the standpoint of equipment utilization, more efficient use of equipment can be achieved in a centralized configuration because other functions can be performed during those intervals when a decentralized arrangement would produce idle time.

From the standpoint of retaining operational capability when a failure occurs, a centralized arrangement can be reconfigured to supply redundant idle components to serve in place of the failed item; and, the switch can be made automatically.

Preprocessors in the GNC Subsystem provide an example of the effect of decentralization. The optical navigation/attitude preprocessor serves to control the operation of five GNC assembly categories. Failure of an LRU in the preprocessor results in loss of capability with regard to the horizon detectors, attitude gyros, horizon sensors, star sensors, and star trackers; a total of nine required and one standby-redundant assemblies. If operational control were vested in the centralized multiprocessor, loss of one of the processors would affect only the function being performed at the instant of failure. Other functions would continue normally. The lost function might be recovered automatically in the centralized environment. However, in a decentralized environment, all the above functions would be lost until manual remove-and-replace activity could be effected.

2.4 MASTER EXECUTIVE DESIGN

The Master Executive is designed to perform the following:

- Handle interruptions
- Supervise tasks
- Control programs in main storage
- Control the Data Management Subsystem itself
- Supervise the interval timers of all processors
- Supervise the interface with man via display units
- Handle data logging
- Handle checkpointing
- Supervise exiting and termination procedures

The functions of the Master Executive are:

- Scheduling
- Support Services
- System Communications
- Resource Allocation
- Data Handling
- System Recovery
- Interruption Servicing

The Master Executive will perform multiprogramming; that is, fulfill two or more separate programming requirements concurrently by making decisions based on various conditions in the Data Management Subsystem. Therefore, it is a goal of executive design that modules will be reenterable. The reenterable attribute is ascribed to a program which is used concurrently in the performance of two or more tasks. The design of the Master Executive is such that it can perform its functions while being used by more than one processor in the DMS multiprocessor.

The basic service of the Master Executive is to provide an interface between software and hardware. In doing so, it enables programs to be largely independent from an evolving hardware environment.

The Master Executive serves to reduce redundant software development and reduce maintenance activities which arise in a dispersed design, by serving as a repository for common functions. Any function having multiple uses across systems (e.g., experiment, checkout), is designed with common interfaces and supplied as a service of the Master Executive.

2.4.1 SCHEDULING

The flow of processor control through the Master Executive is shown in Figure 2-1. Processor control is passed to the Master Executive via an interruption such as supervisor call, I/O, timer, program, or machine check. Any interruption causes processor control to be taken from the interrupted program and given to an interruption handling routine of the Master Executive. The interruption handling routines analyze the requirements for servicing the interruption; and, as a result, may pass control to one of the task specification routines which define a new task and record it in the task queue. The task queue is that portion of the data base containing

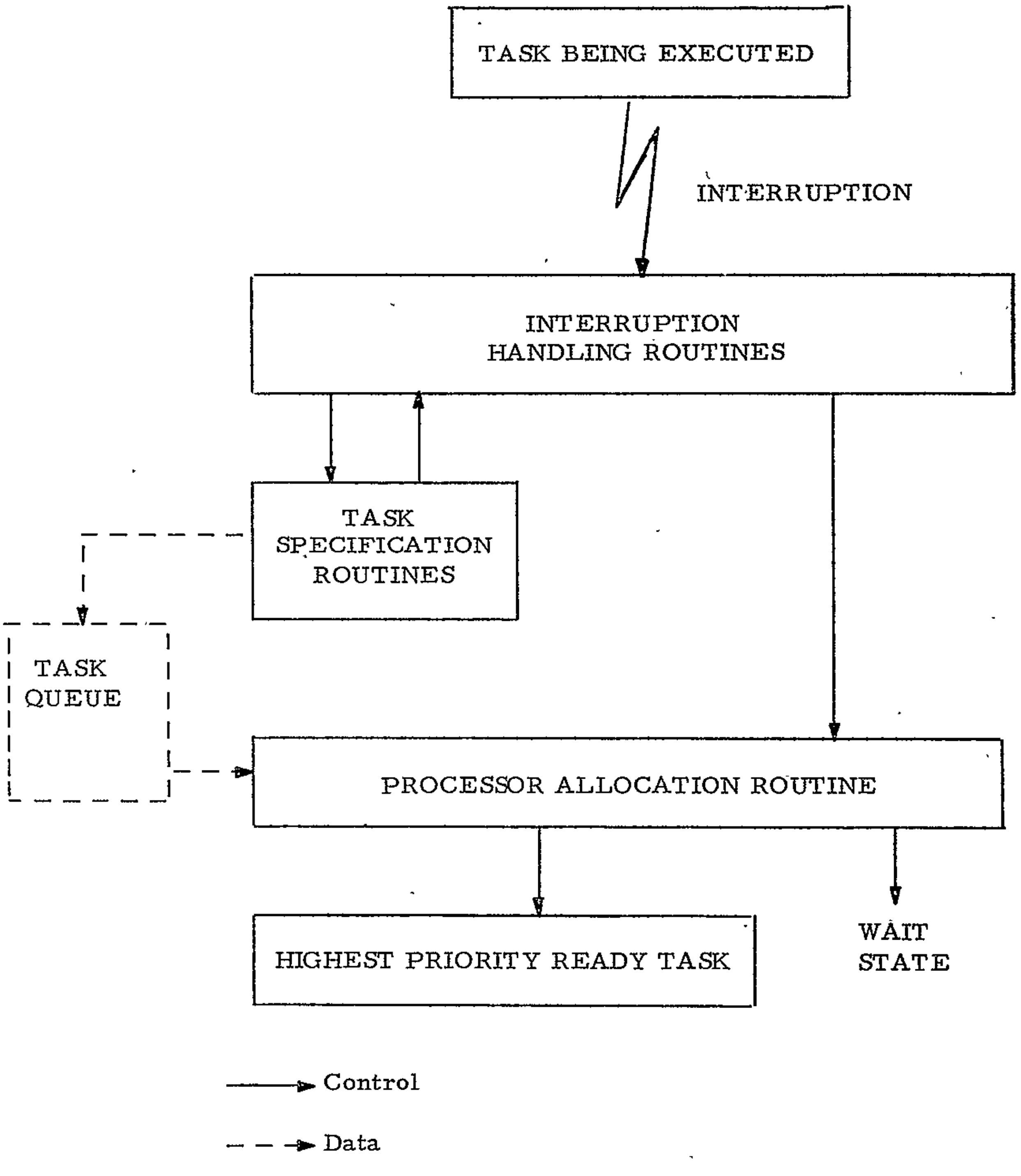


Figure 2-1. Master Executive Scheduling

the identity and characteristics of work being performed by the multiprocessor. After the task specification routine returns control to the interruption handling routine, or in the case where no new tasks are required, control is passed to the processor allocation routine which examines the task queue and chooses the highest priority task which is ready for execution. Control is then passed to this task. If no ready tasks are available, the processor is placed in a "wait" status until another interruption causes the above procedure to be repeated.

The task specification module determines the selection and priority of the tasks to be executed. The task specification function must recognize a request to execute a task, obtain and store enough information for the task to be initiated, and indicate when the task is ready for execution. Task specification identifies, locates, and evaluates in conjunction with the existing job mix, and specifies the order in which processors are to be allocated to each task.

Tasks are recorded in the data base of the Master Executive as a series of elements in the task queue, and ordered as to their priority. The elements record the task states of ready, waiting, and active, as appropriate, so that any processor can examine the queue and determine the highest priority task which is ready for processing. The ready state signifies that execution of the task can resume as soon as possible. The waiting state signifies that execution of the task must be suspended until the completion of a specific operation such as data transfer from the data bus. The active state signifies that a processor is currently engaged in executing the task.

The logic paths in the Master Executive are brought together into the processor allocation routine, which either selects the task which should next be executed, or enters the wait state, if all tasks are active or waiting. This results in a high service attention for tasks in the system.

A distributive task scheduling technique is used because the effects of processor or memory failure can be controlled in such a way that the failure does not disable the entire system. In most cases, remaining elements of DMS can continue to function in a normal way while corrective action and recovery is taking place. The distributive technique allows any processor to execute any DMS task which is ready. As a result, during the lifetime of a task, it will receive the services of several different processors as the task states cycle through the normal sequence of ready, active, and waiting. The distributive technique eliminates the problems of interchanging master and slave roles, while utilizing more of each processor's potential.

Programs are scheduled to be initiated by the following means:

- By the crew or ground using the execute element of the high level language
- By the routine which handles timer interruptions
- By the routine which logs real-time data
- By other programs using either the execute or the call elements of the high level language

Program initiation is performed in the following steps:

a. Bring in program characteristics table

b. Verify that the parameters required by the program are available

c. Allocate memory and other DMS resources required by the program

d. Fill out or complete the program characteristics table

e. Format the working storage used by the program

f. Give control to the program

If the crew or ground requests a checkout program be executed, and one or more of the DMS LRU's required are inactive, an indication is received that the executive is unable to allocate from available resources.

2.4.2 SUPPORT SERVICES

In contrast to the in-line services of the Master Executive which are obtained when an interruption occurs, the support services operate as separate tasks.

2.4.2.1 Language Support

The language support services of the Master Executive consist of a compiler which organizes language statements into data lists, an interpreter which processes the data lists as a program, and a library of routines which provide detailed functions to support the compiler and the interpreter.

2.4.2.1.1 Compiler

Since the high level language is used for system communication as well as programming, the compiler is provided to transform keyboard entries into data lists for efficient processing by the interpreter. The compiler is used for batch processing of language stored on card, tape, or direct access media, as well as for real-time processing of keyboard entries. The data lists which are prepared during the batch process are stored in a compact form for later processing by the interpreter.

When used in the real-time mode, the compiler keeps a low profile in main storage by using overlay techniques which load only the compiler modules needed for the element being processed. As soon as a data list is formed from one language statement, control is given to the interpreter for actual processing.

From the keyboard user's standpoint, there is both a define and an execute mode of the language. When the compiler is in the execute mode, language statements, consisting of elements and modifiers, are passed to the interpreter one-by-one, as they are entered from the keyboard. The interpreter performs the specified operations and returns results, or status to the user in a conversational manner.

When the compiler is in the define mode, the data lists from language statements are accumulated until the user commands that the execute mode be entered. Transition between modes is effected by the BEGIN and the END language elements.

2.4.2.1.2 Interpreter

In general, an analysis of two or more programs written in a high-level language reveals a high degree of redundancy in the instructions which have been prepared for execution by the language translator. These programs may differ only in the parameters or data lists which are to be processed by the instructions. By removing the executable instructions from the high-level language programs and consolidating them in an interpreter, a savings in main storage results from the elimination of the redundancy. In addition, the data lists which now comprise the program can be made independent of the main storage addresses actually used to store the program during execution.

For the Space Station multiprocessor with shared main memory, the above characteristics result in more efficient use of main storage by eliminating redundant instructions which might otherwise appear in concurrent tasks. In addition, should a memory LRU fail, the programs in the LRU can be relocated without having to be restarted. A disadvantage in consolidating the executable instructions is the increase in memory contention, which results when several processors use the same set of instructions in the interpreter.

Two or more tasks are concurrently processed by the interpreter as a result of storage allocation and interpreter design techniques. The interpreter uses a block of main storage allocated to the program for storing the data required in the interpretation process, and the data required by the program itself. The interpreter routines are made reenterable by this storage technique. The reenterable attribute allows use of a routine by subsequent tasks, prior to the time when the initial task completes its use. For example, the interpreter routine which processes the MEASURE element for Task A may start an I/O operation and be waiting for its completion when Task B requires use of the MEASURE routine. Since working storage is used for all changes, Task B may enter the routine before Task A leaves it, thus decreasing the time spent waiting for task processing.

The interpreter provides general processing, such as recognition of the language element represented in the data list, and passing processor control to the appropriate library routine for detailed processing. The interpreter also controls the loading of non-resident library routines from auxiliary storage into a transient area of main storage.

2.4.2.1.3 Library Routines

Library routines of the language support services provide detailed processing necessary for the language elements. Depending on their frequency of use, they may be stored in auxiliary storage and transferred to main storage as required, or kept in main storage. Principal services provided by the language support library are as follows:

- Symbolic Test Point Translation - permitting hardware-independent development and multiple use of program modules. (See Figure 2-2).
- Data Handling - employing a data base definition, storage, and accessing technique for both the high-level language programs and for the interpreter.
- Message Assembly - permitting concise language references to skeleton messages stored in the data base.

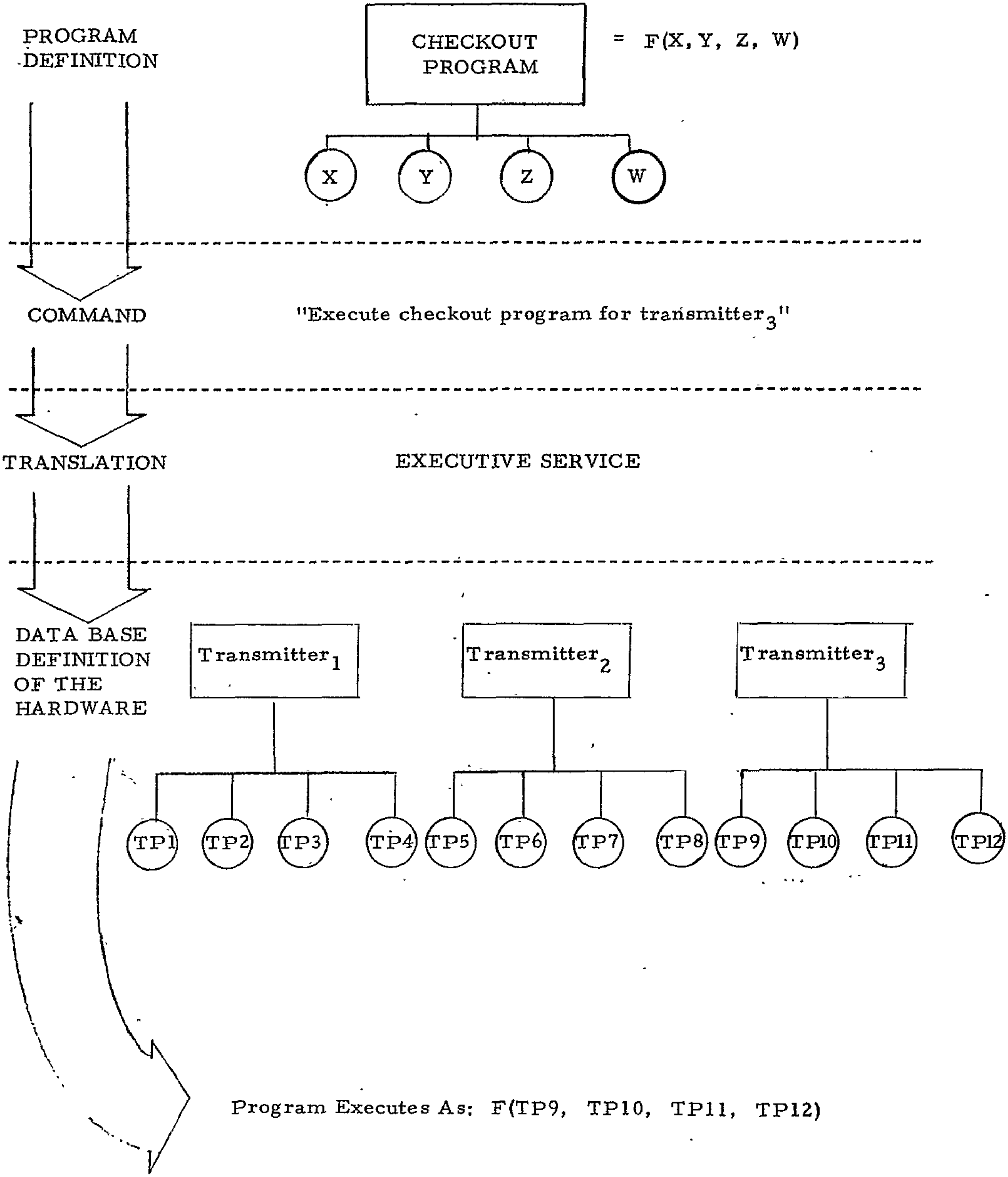


Figure 2-2. Symbolic Test Point Translation

- Arithmetic Processing - in support of the SOLVE element.

- Documentation and Debugging - to provide display, retrieval, and modification of programs; and, to provide facilities for tracing program execution.

2.4.2.2 Time-Based Services

When large numbers of programs must be initiated repetitively on a scheduled basis, a machine is a more reliable initiator than man. Examples of programs of this type are those used for trend analysis, periodic checkout, and data path verification. To relieve crew or ground personnel from the burden of remembering when to execute these tasks, a Master Executive routine called the Pacer is provided which sets the interval timer of a processor to a value which will cause a timer interruption when initiation of a scheduled task is required. A flow chart of the Pacer is shown in Figure 2-3.

2.4.2.3 Utility Services

The utility services are provided by a collection of general purpose programs which are used as required to aid man in establishing the data base, retrieving data, and modifying the Rate Table used by the Pacer.

2.4.2.4 System Initialization

The initialization procedure is invoked by crew or ground, and establishes the division between processors used for operations and those used for experiments. Initialization is invoked by a hardware command to one of the DMS processors. Included as a parameter is the address of the auxiliary storage device which contains the initialization program and the executive. Other processors are activated by the first, using the processor-to-processor communications feature. The initialization function determines the DMS configuration; loads the executive into shared main storage; establishes the initial executive data base; and finally causes each processor to execute pending tasks; or await further communication from crew, ground, or data acquisition path.

2.4.3 SYSTEM COMMUNICATION

The system communication function of the Master Executive is designed to enable man to obtain the services of the Master Executive; and, through those services invoke the functions necessary to provide data processing for the on-board subsystems. It is a design goal to implement the system communication function within the structure of the high-level language, in order to minimize the amount of "non-language" which man must know in order to make use of the Data Management Subsystem.

Entry in the Rate Table:

Event Identification	Δt	$t_i + \Delta t$

Δt = period of initiation
t_i = time of last initiation

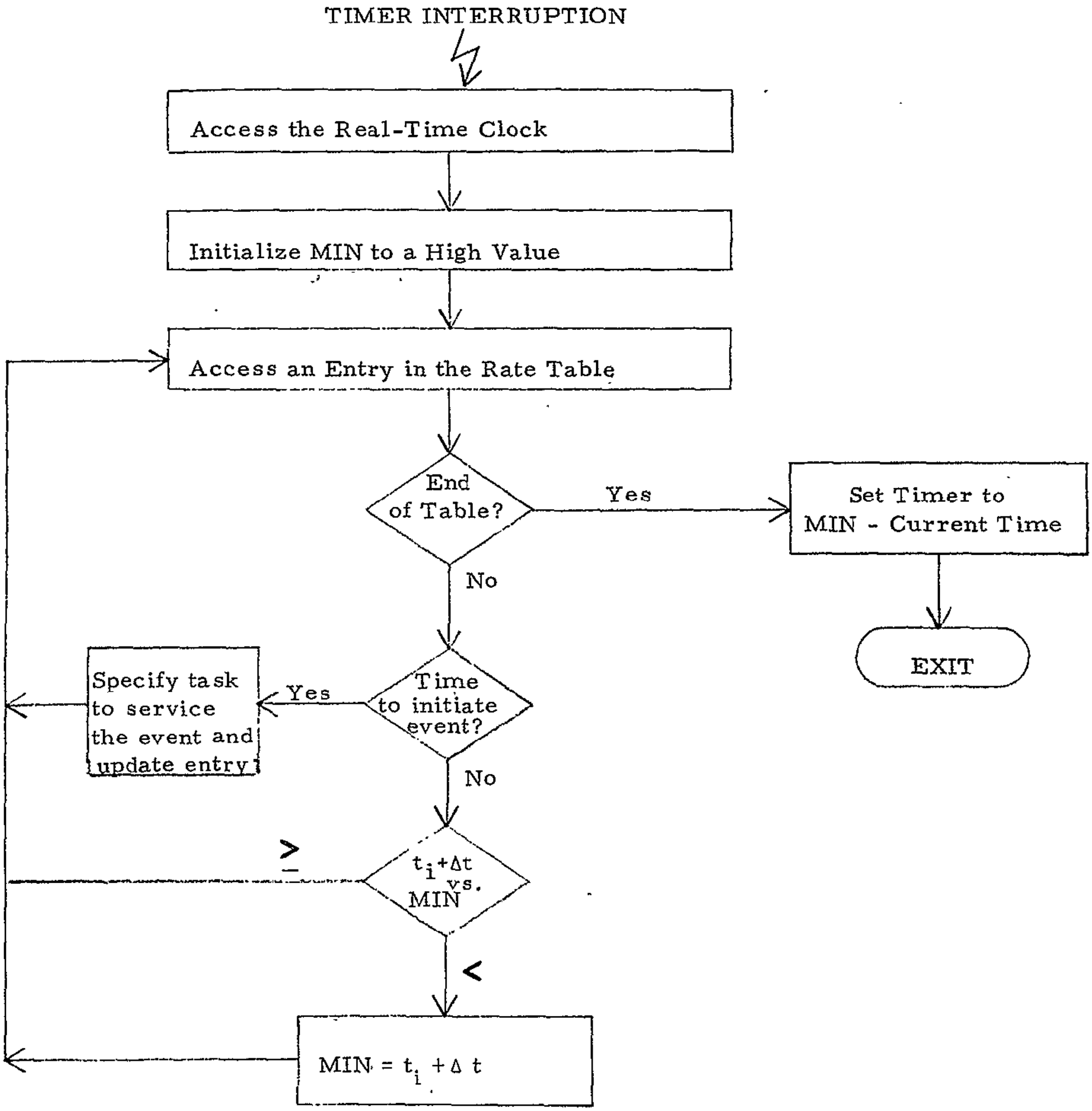


Figure 2-3. Pacer Logic Flow

2.4.4 RESOURCE ALLOCATION

The allocation and use of resources function of the Master Executive is primarily a supervisory one, and the routines that perform this function are collectively referred to as the Supervisor. The routines of the supervisor control the allocation and use of the processors, shared main storage, data bus controller, data bus terminals, and other devices of the Data Management Subsystem. The supervisory functions are as follows:

- Interruption Supervision, which involves analysis using specific routines to handle each type of interruption.
- Task Supervision, which is the recording of tasks currently in the Data Management Subsystem, and their associated priorities, statuses, and the programs they require.
- Shared Main Storage Supervision, which is the allocation and de-allocation of various portions of shared main storage.
- Contents Supervision, which is the loading of programs into shared main storage, and the recording of the characteristics of these programs for use in allocation and recovery procedures.
- Timer Supervision, which is the setting and maintaining of the interruptable timers of all processors in the Data Management Subsystem.

Shared main storage in the DMS is allocated by the Master Executive based on pre-defined program requirements, and requirements which arise during program execution. In order that the effect of a memory failure may be assessed, a memory table of contents is redundantly maintained in separate line replaceable units.

Programs are designed so that during execution, they may be moved about in shared main storage to eliminate fragmentation of unused storage.

2.4.5 DATA HANDLING

When a task of the Data Management Subsystem requires data outside its own working storage, it must use the data handling services of the Master Executive to obtain that data. The reason for this is to control access to the data and resolve possible conflicts. The data handling function of the Master Executive provides a unified interface with using programs so that the data location (main storage or auxiliary storage) is transparent to the programs.

2.4.5.1 Capabilities

The principal rationale for providing extensive data handling capabilities for the Space Station On-Board Checkout Subsystem is that an accurate account of status and configuration must be maintained, at least down to the LRU level. Most high-level language programs are involved indirectly in data handling; however, the interface provided through the READ and WRITE language elements allows data handling specifications to be made in general terms, consistent with the abilities of the language in other areas. Operations such as indexing, buffering, physical storage management, etc., are performed by executive routines. Data base definitions are maintained external to either executive or checkout programs, in order to provide a central reference for all routines gaining access to the data base.

2.4.5.2 I/O Operations

The processing of I/O operations is divided into the processing required to start the operation, and the processing which is required when the operation is terminated.

The use of a READ or a WRITE language element results in control being passed to the language support services routine which determines what further action should be taken. If an I/O operation is required, information required to initiate the operation is gathered, and a supervisor call instruction is executed which causes a supervisor call (SVC) interruption. The SVC interruption handler of the Master Executive gives processor control to the I/O supervisor, which either starts the I/O operation using the start I/O instruction, or enqueues the operation if the data path is busy.

When the I/O operation terminates, an I/O interruption occurs causing processor control to pass, first to the I/O interruption handler, and then to the I/O supervisor which records the fact that the operation is completed in the table provided for that purpose by the routine which requested the operation (usually the language support services routine). If an error is indicated, the appropriate error handling routine is scheduled. The queues are examined to determine if another I/O operation can be started on the data path prior to returning control to the interruption handler.

2.4.5.3 Data Logging

Many potential sources for real-time data exist in the Space Station. In order to respond to the unpredictable arrival of such data, the Master Executive provides a logging and reduction service which records the arrival of real-time data, and performs preliminary compaction prior to logging.

Programs which process real-time data may be scheduled for execution by the arrival of the data, or the accumulation of a threshold quantity of data. The executive service is designed so that processing does not lock out further arrivals; and when no further data is available, the program is checkpointed so that main storage may be used for other purposes.

Specific advantages can be gained in real-time systems which accept the random arrival of data initiated by external devices, by providing routines to accomplish special analysis. Tolerance checking and data compaction typify such requirements. Tolerance check routines compare the data with user specified values, and the operator is notified only when limits are exceeded. The technique of compressing data of like characteristics or values during real-time processing reduces the amount of time and space required for storage.

The log supervisor function of the Master Executive provides the following services:

- Receiving, time stamping, and logging real-time data
- Initiating the program which processes the data
- Retrieving data from the log at the request of the program

The first service continues while the other two are in progress. Note that the application program, not the log routine, coordinates transfer of data from the log to the program for processing. The log routine is responsible for insuring that the program is initiated. It is the program's responsibilility to read the data by requesting retrieval from the log supervisor. If no further data is available, the log supervisor returns an indication to that effect to the program which may wait or terminate, as appropriate.

It is the joint responsibility of the log supervisor and the restart supervisor to record the progress made by the program in processing data in the log.

The association between the data source and the program required to process it is pre-defined and recorded in the data base.

A fixed number of auxiliary storage tracks, divided into several segments, are allocated for each log and used in a cycle as shown in Figure 2-4. The segments contain groups of data which have arrived since a checkpoint of the program was recorded. A log control block is used to record the segment boundaries, and maintain coordination with the associated checkpoints. Although the responsibility for initiating checkpoint procedures lies primarily with the program, the log supervisor will initiate a checkpoint if the latest segment begins to overlay the oldest segment.

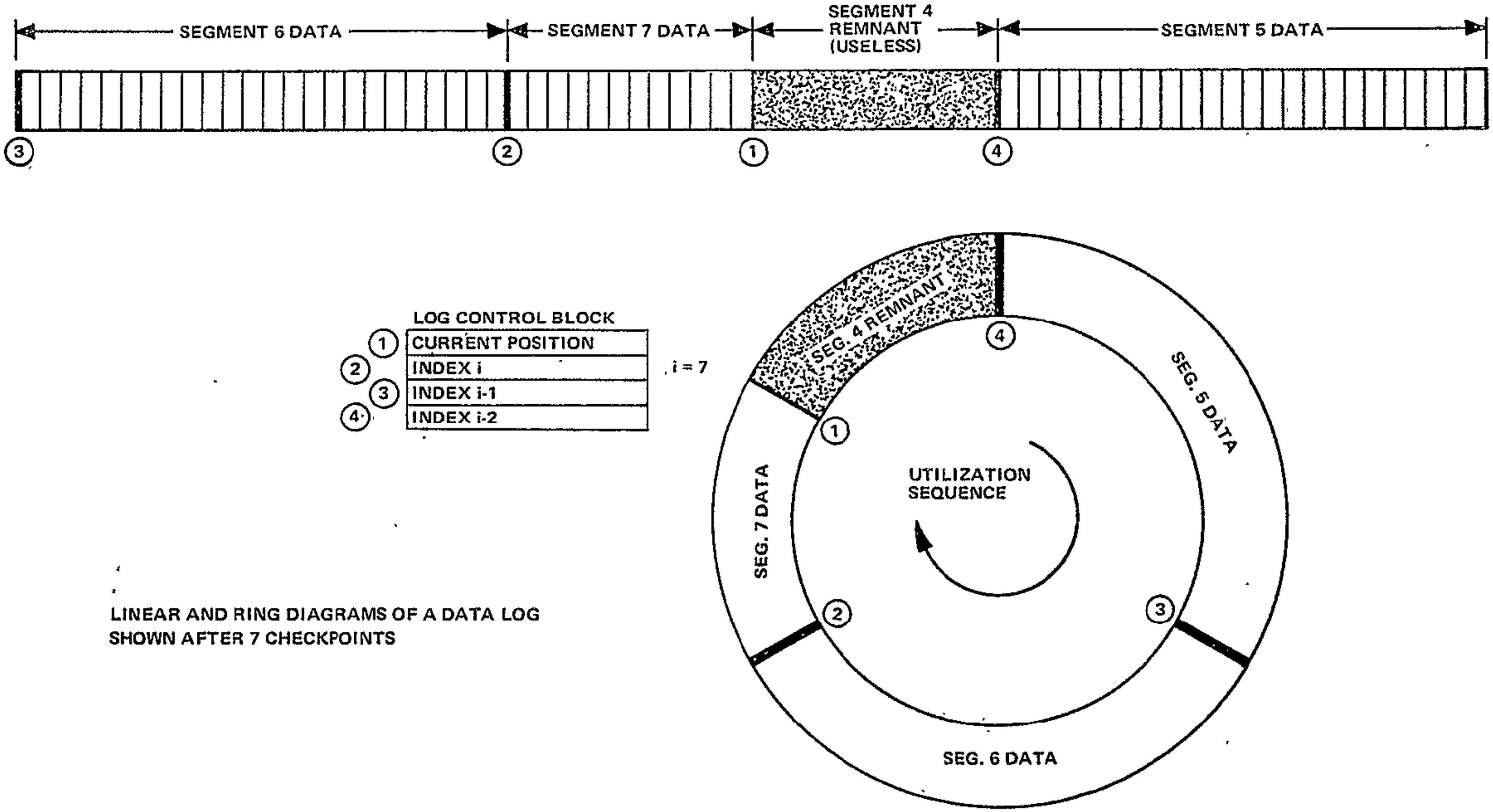
The Data Retention module is a function which must accomplish the retention of data which will be required for analysis at a later time. This can be accomplished by allocating areas of peripheral storage to hold the data, and periodically deleting the data which is no longer needed.

2.4.6 SYSTEM RECOVERY

The system recovery function of the Master Executive is concerned with the activities necessary to maintain a high level of DMS operating capability when a DMS failure occurs. Reconfiguration is performed automatically by making use of redundantly maintained tables which reflect the configuration of active and spare elements at all times; and by changing the configuration control registers of the hardware.

Malfunctions in the processors and in main storage are handled by the recovery management function and initiated by a machine check interruption. Failures in the data acquisition path are also handled by this function; however, initiation is by way of the I/O interruption and subsequent analysis of associated status indications. It is a goal of the recovery management function that system operation may be permitted to continue even though failures have occurred that would otherwise cause the system to stop. If normal retry procedures are unsuccessful, the recovery management function automatically prohibits further use of the failed LRU, so that system operation can be resumed in the

FIGURE 2-4. DATA LOG



LINEAR AND RING DIAGRAMS OF A DATA LOG SHOWN AFTER 7 CHECKPOINTS remaining operative portion of DMS. Recovery management then analyzes the effect of the failure, in order to restart or recover those tasks which are affected.

Recovery from certain memory failures can be achieved by refreshing the contents of the memory cell causing the problem. Separating program text from working storage makes it possible to refresh a program module if a machine check occurs during a reference to the instructions. Having working storage and instructions intermingled reduces the frequency with which this technique can be employed.

Since damage to certain infrequently used data management tables due to program error, may go undetected for a time, a data base audit function is provided to check the integrity of the data base periodically and after the failure of a memory or processor-associated LRU. Programs which provide this function verify that the relationships among components of the data base are correctly established, and that conditions that exist in the hardware are accurately reflected in the data base.

Termination procedures must provide for both scheduled and unscheduled termination of the activities of DMS software and hardware. It is a goal of the termination process to enable a restart of the system without losing task continuity, while providing a means for activities to be selectively resumed by crew or ground specification.

Providing for unscheduled termination increases system overhead. Several methods are employed in existing ground-based systems, including checkpoint, logging of changes, and emergency or backup power supplies. As Space Station requirements evolve, trade studies in this area are indicated.

2.4.7 INTERRUPTION SERVICING

The interruption is the basic means by which hardware and programs obtain the services of the Master Executive. For each type of interruption, a first level routine exists to determine detailed requirements of the requested service and route processor control to the appropriate executive routine.

The interruption is a demand to the Master Executive to recognize that an event has occurred, and to consider that event when scheduling the processor activity. While a keyboard interruption indicates that there is a console entry which should be serviced, an alarm condition may indicate a system failure. It is reasonable to assume that the alarm requires action of a relatively higher priority than the keyboard.

The design of an interruption monitor is heavily dependent upon the hardware interruption system in the computer. The software must complement the hardware design while performing the following functions:

- Save machine status (processor status and register contents if needed) at the point of interruption

- Identify the interruption source and type

- Select and execute the appropriate program module

- Restore machine status and return to the interrupted program

2.5 OCS EXECUTIVE DESIGN

The OCS Executive is designed to provide services which are unique to checkout, and common to more than one subsystem. By locating such services in the OCS Executive, redundant software development is avoided.

The OCS Executive consists of routines closely associated, or in line, with the Master Executive, and routines which operate as separate tasks. The closely associated routines are handled in the same way as the Master Executive's own in-line routines in that they are resident with the Master Executive, or established as transient routines. The separate task routines are invoked as needed by other tasks which require their services.

An example of a separate OCS Executive task is the utility which facilitates RDAU memory management. An example of an in-line OCS Executive routine is the second-level I/O interruption routine which verifies the limit check.

In the following paragraphs, the data base and the services of the Onboard Checkout System Executive are discussed.

2.5.1 DATA BASE

Operations of the OCS Executive are centered about a data base in which the configuration and status of all Space Station assemblies are continuously maintained. By storing the representation of the logical relationships which exist among assemblies of the onboard subsystems, the data base provides the reference for coordinating conflicting tasks. Hardware status can be determined without interrupting the active functions of the hardware in order to make tests, because the data base is maintained in real-time by executive programs. Use of the data base by both the Master and the OCS Executives results in more concise application programs, since the coordination procedures do not have to be provided in each program module.

2.5.2 CHECKOUT SERVICES

The functions which are determined to be associated principally with checkout, yet are not unique to a particular subsystem, are selected for implementation in the OCS Executive. This choice does not restrict in any way the selection of whether or not the high level language is used when programming these functions; rather, the choice of language may be made based on suitability relative to the specific application.

In the following paragraphs, OCS Executive services which support status monitoring, fault isolation, reconfiguration, trend analysis, and checkout program debugging are discussed.

2.5.2.1 Status Monitoring

Although the continuous measurement of test point parameters and the comparison of their values to upper and lower limits is performed by the Remote Data Acquisition Units RDAU, the OCS Executive plays an important role in status monitoring. When the I/O interruption handler detects an RDAU limit check, it passes control to a module of the OCS Executive known as the RDAU Second-Level Interruption Handler. The function of this module is to verify that the RDAU limit check is a proper one. This is done by accessing the RDAU memory contents stored in shared main storage, and comparing them with the contents of the RDAU memory. The test point which has gone out of limits is then read, and an in-storage limit check is performed to verify that the test point is out of tolerance. Upon verification, the RDAU channels which read that test point are then masked to prevent further interruption until analysis is complete. The identity of the RDAUs involved are placed on a restore queue so that, after analysis is complete and the failure is corrected, the RDAU limit-checking mode can be re-instituted.

A functional flow chart of the Second-Level Interruption Handler is shown in Figure 2-5. Prior to passing control to the subsystem fault isolation routine, an analysis is made to determine whether the test point which is out of limits is within the scope of a fault isolation task already in progress. If this is the case, then no further action is taken. If the test point is not within the scope of existing fault isolation, then a new subsystem fault isolation task is enqueued for execution.

The Second-Level Interruption Handler uses the test point identification as a key to search the data base for the indication that special processing is required for particular test points. In the interest of brevity, this is not shown in Figure 2-5. The most important type of special processing is the analysis to determine if a caution or warning indication is required when the measurement is out of limits. An example of other special processing is that used for certain Electrical Power Subsystem test points which are measured repeatedly after the initial out-of-limit indication. A fault indication is provided only if the measurement remains out of limits for a pre-determined number of consecutive measurements.

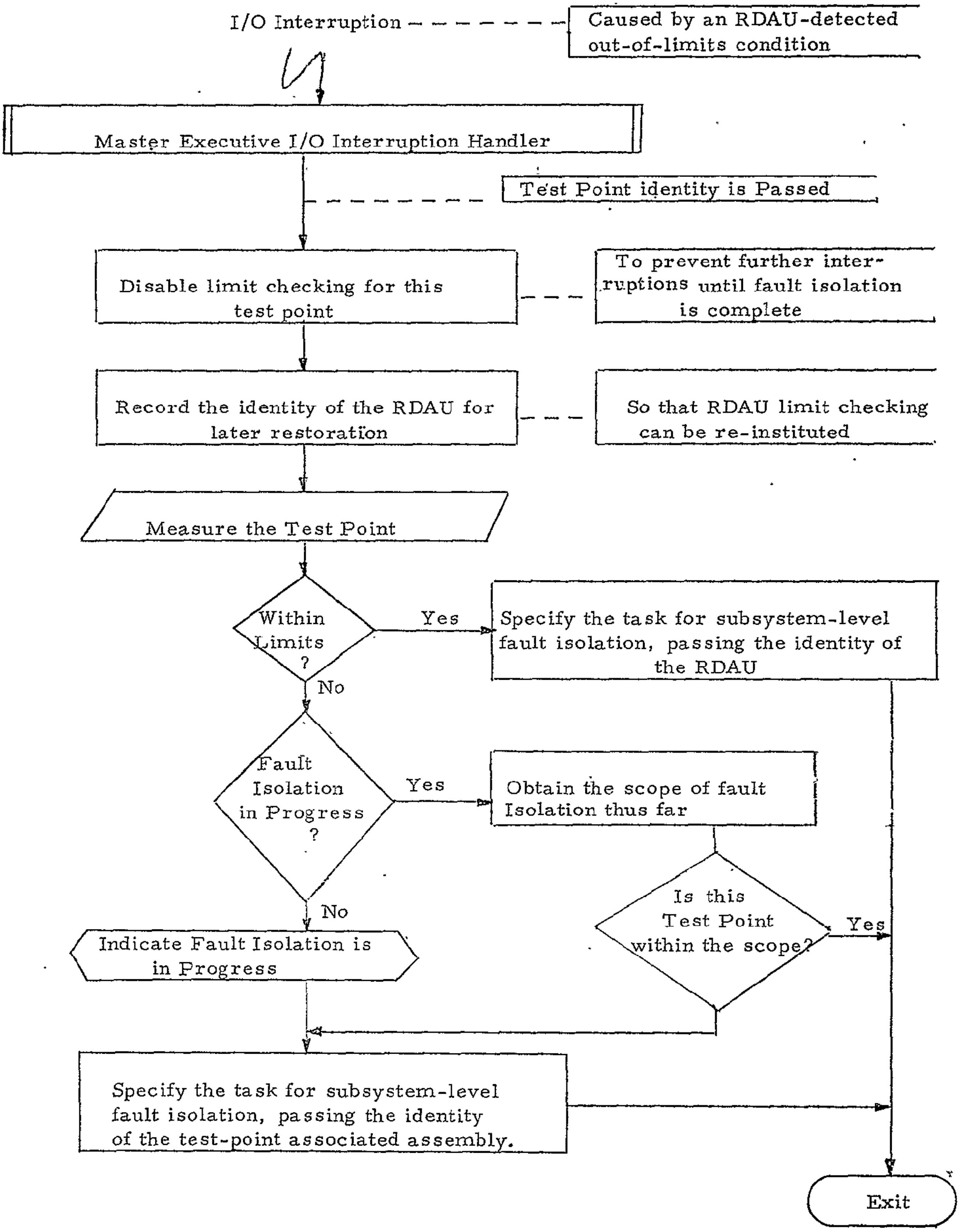


Figure 2-5. RDAU Second-Level Interruption Handler

2.5.2.2 Fault Isolation

When a fault is detected and has been verified by the limit check interruption handler, the OCS Executive routine for fault isolation is given control. This is a routine written in the high-level language which utilizes key test points and the configuration data base to evaluate the performance of each Space Station subsystem, and to determine which sequence of fault isolation programs should be utilized in isolating the fault. Thus, the OCS Executive performs fault isolation at the subsystem level.

Additional functions which support fault isolation are: loss-of-capability analysis, GO-NOGO processor, and the repair rate monitor. These functions are discussed in the following paragraphs.

2.5.2.2.1 Loss-of-Capability Analysis

The loss-of-capability function uses the configuration data base to determine the relationship between an LRU failure and hardware functions and capabilities, which are being used at the time of failure. The relationship is examined with regard to the effect of the failure on vital functions and capabilities.

2.5.2.2.2 GO-NOGO Option Processor

In order to permit the operator some flexibility in dealing with situations where a reading is detected out of tolerance, the application program requires the ability to respond to keyboard direction. Subsequent to detecting an out-of-limit situation, a message is displayed and the program enters the GO-NOGO processor. The available options are presented, the operator enters his selection, and program execution is resumed. Options are provided which permit the program to continue as if no error has been detected, to re-execute the part of the program which detected the error, and to terminate the program.

2.5.2.2.3 Repair Rate Monitor

When a failure has been isolated, the actual remove and replace activity may be delayed in order to optimize crew activities by having all such functions performed according to a schedule. The repair rate monitor function serves as a log to record fault incidences for use in scheduling activities requiring crew intervention. A reminder feature is provided which acts to insure that a failure, once detected and isolated, is not forgotten. Use of the Pacer is made to send a message to the crew if action has not been taken within a pre-determined time interval. Reminders may be rescheduled or cancelled by appropriate manual entries at the display keyboard.

2.5.2.3 Reconfiguration

The reconfiguration update utility program facilitates changing the configuration tables of the data base to reflect changes to the configuration of the subsystems. The utility also is used as a tool when establishing the configuration data base for use by the fault isolation and operational procedures. This function has an extensive interface with man, and is not intended to be used as a real-time update function.

2.5.2.4 Trend Analysis

The trend analysis functions are implemented as a library of several different types of routines which are designed to fulfill all trend analysis requirements of the Space Station Subsystems.

The executive is involved in the following steps when a trend measurement is required:

- The Pacer detects the need to perform a trend measurement
- Restart retrieves the last copy of working storage
- If the program is not already in storage, it is loaded
- Control is passed to the program which makes the measurement, logs it, and updates the working storage
- If the processing results dictate, crew communication is performed
- Checkpoint-and-terminate is invoked to save the updated copy of working storage and relinquish processor control

For certain trend analysis applications, the data is merely stored for later recovery and review. The trend data display function provides the means by which man can retrieve trend data, and control the presentation of it in the form most suitable for his purposes. Facilities are provided for processing the data in tabular or graphical form.

2.5.2.5 Program Verification Facilities

An important aspect of checkout which is too often overlooked is that of software verification. The Executive services capability should contain sufficient software verification techniques to determine the validity of the high-level language test sequences in real time. A requirement further exists for a complete software verification facility to be provided in a ground environment "HangerQueen" to perform checkout verification on both operational and checkout software.

The program verification facilities of the OCS Executive serve to provide an interface with checkout programs indistinguishable from actual hardware, yet controllable by man. The facilities use random number generation techniques to unpredictably assume all possible situations which can arise from a given arrangement of test points, so that program performance can be assessed rapidly and without extensive time consuming breadboard techniques.

Section 3

LANGUAGE REQUIREMENTS AND ANALYSIS

3.1 INTRODUCTION

The Subsystem Analysis performed by the Software Task recommends that a high level test-oriented language be used to accomplish Space Station automated checkout. It is further recommended that this language be expandable for use with all onboard crew systems.

The language concept was initially addressed in the Preliminary System Analysis subtask of this study. A selection of baseline language elements was defined with supporting rationale for their selection. (The preliminary analysis additionally provided the definition of the syntax rules governing the language structure, for a language to meet the Space Station non-language interface requirements.)

A final element selection, with required modifiers, was obtained during the analysis of each subsystem design. It was determined to be essential that the number of language elements be limited to only those required to efficiently perform the station checkout function.

3.2 LANGUAGE SYNTAX

The language developed for a large integrated Space System must be compatible for all phases of development and flight applications. Significant cost savings can be realized by providing commonality throughout the total language utilization functions. Less apparent are the advantages which can be achieved by allowing users in each phase of the project (Factory Checkout, Prelaunch Checkout, and On-Orbit operations), to refer to the software using the same frame of reference.

It is, therefore, essential that at the beginning of language development a set of rules governing the format and control of the language be provided.

3.2.1 SYMBOLS

Syntax is defined as a rigorous statement of what sequences of characters are considered correct in the language and, ultimately, what character strings constitute a syntactically correct program. It is desirous that the language used to write the definitions of a high level language be structured in such a way to eliminate ambiguities that exist in the English language today. To accomplish this a group of symbols have been noted, with their recommended uses for the implementation of the results of this study.

The symbols and their meanings are as follows:

Symbol	Meaning
:=	Is defined as
( )	Surrounds a name being defined or being referenced.
juxtaposition	Concatenation
/	Or
small letters	Language variable (option)
upper-case letters	Terms which are fixed words in the language
null	Absence of a variable
...	Elipsis

3.2.2 CHARACTER SET

The following paragraphs contain lists of recommended standard and special characters to be used in the language for program development and man-machine interface requirements of large scale space systems. The more concise the character set, the easier it becomes to learn and use by the crew.

3.2.2.1 Standard Characters

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z 0 1 2 3 4 5 6 7 8 9

3.2.2.2 Special Characters

'	Apostrophe	*	Asterisk
<	Less than	/	Right slash
=	Equal to	(	Left parenthesis
>	Greater than	)	Right parenthesis
+	Plus	,	Comma
-	Minus		Blank

3.2.2.3 Syntax

In Table 3-1, an example is provided to illustrate how the rules of syntax are expressed. The rules are presented in sequence, which allows subsequent rules to use preceding rules.

Table 3-1. Rules of Syntax

```
(alphabetic) :=A/B/C/D/E/F/G/H/I/J/K/L/M/N/O/P/Q/R/S/T/U/V/W/X/Y/Z
(digit) :=0/1/2/3/4/5/6/7/8/9
(alphanumeric) :=(alphabetic)/(digit)
(label) :=(alphabetic)/(label)(alphanumeric)
(blank) :=
(b̸) :=(blank)/(b̸)(blank)
(element) :=ALLOCATE/BEGIN/CALL/DEFINE/DELAY/DISPLAY/EJECT/END/
            EXECUTE/GOTO/IF/MEASURE/READ/SEARCH/SOLVE/SPACE/
            STATUS/STIMULATE/STOP/TIME/WRITE
(integer) :=(digit)/(integer)(digit)
(sign) :=+/-
(exponent) :=E(integer)/(sign)E(integer)
(number) :=(integer)/(sign)(integer)/(number)(exponent)
(string) :=(alphanumeric)/(string)(alphanumeric)/(string)(blank)
(string-constant) :='(string)'
(keyword) :=(alphabetic)/(keyword)(alphanumeric)/(keyword)=(string-constant)/
            (keyword)=(number)
(modifier) :=(number)/(keyword)/null
(modifier-list) :=(modifier)/(modifier-list),(modifier)/
                  (modifier-list),...,(modifier-list)
(statement) :=(label)(b̸)(element)/
              (label)(b̸)(element)(b̸)(modifier-list)/
              (b̸)(element)/
              (b̸)(element)(b̸)(modifier-list)
```

3.2.3 OPERATORS

This study recommends that the special characters described in the previous paragraphs assume the following operator definitions when used with software development and Data Management Subsystem interface requirements for the Space Station.

3.2.3.1 Arithmetic Operators

+	Addition
-	Subtraction
*	Multiplication
/	Division
**	Exponentiation

3.2.3.2 Relational Operators

<	Less than
<=	Less than or Equal to
>	Greater than
>=	Greater than or Equal to
=	Equal to
<>	Not equal to

3.2.4 FORMAT

The syntax, or the definitions of the statements formed from the elements and modifiers of the language, is critical from a heuristic point of view. The format recommended for this language uses blank spaces and commas to delimit input fields, rather than fixed positions on a keyboard/display or input card.

3.3 BASELINE ELEMENTS

The baseline elements were selected to address the general requirements of a test oriented language to be used by on-board as well as prelaunch checkout personnel. The existing test oriented languages; Test Oriented On-Board Language (TOOL), OCS II, and the Saturn Acceptance Test or Launch Language (ATOLL), were evaluated, as well as high level scientific languages to determine the best qualities of each which were applicable to on-board checkout requirements.

Table 3-2 contains the baseline elements and the rationale for their selection. All the elements which exist in the TOOL and OCS II languages were evaluated. This group of elements represents the initial baseline selected for this study and is not identical to the resulting final language recommendations. Emphasis has been placed on minimizing the number of elements in the language recommended by this study.

The Recommended Action column indicates whether the element was included, modified, deleted, or is a new element independent of the TOOL and OCS II baseline languages developed by NASA MSC. The Source column indicates whether the TOOL or the OCS II system was the source of the element. The elements are listed alphabetically.

2 ⊕

TABLE 3-2. BASELINE ELEMENTS

Element	Recommended Action	Source	Comments
AGAIN	Deleted	TOOL	The iterative function provided by this element, in conjunction with the DO element, will be accomplished using the SOLVE and IF elements.
ALLOCATE	New		This element is used to permit a program to establish control over a "configurable group" (an LRU or group of LRU's) to facilitate the interface between test and operational requirements.
BEGIN	Modified	OCS II TOOL	This element has been incorporated and expanded to provide the capability to: a. Include a heading on the program listing. b. Identify a section of code which can be used to bookkeep the subsystem in the event that it becomes necessary to terminate the program before it completes execution.
CALL	Modified	OCS. II TOOL	This element has been incorporated and expanded to provide increased flexibility for interchange of modules. This was accomplished by: a. Saving and restoring the data cells of the calling program, permitting programs which must interface to define their data cells independently. This eliminates the necessity to predefine data cell contents.

TABLE 3-2. BASELINE ELEMENTS (continued)

Element	Recommended Action	Source	Comments
			b. Incorporating the capability to specify a common communication area, overcoming the problems associated with assigning fixed locations in a system buffer.
CHECK	Deleted	TOOL	The concept of setting flags as the result of an operation is not part of the baseline; therefore, this element has been deleted.
CLEAR	Deleted	TOOL	The function accomplished by this element will be done by the I/O interface; therefore, this element has been deleted.
COMPARE	Deleted	OCS II	This element was deleted because the function is provided by the IF element.
CONNECT	Deleted	TOOL	This element was deleted because its function will be accomplished as part of the I/O interface.
DEFINE	New		The define element will be used to define the contents of memory, and equate values with symbols.
DELAY	Modified	OCS II TOOL	This element has been incorporated in anticipation that program controlled timing loops will be required.
DISCONNECT	Deleted	TOOL	This element was deleted because its function will be accomplished in the I/O interface

TABLE 3-2. BASELINE ELEMENTS (continued)

Element	Recommended Action	Source	Comments
DISPLAY	Modified	OCS II TOOL	The functions of this element have been incorporated. The capability to specify console page number has been added. Options still under consideration include: o Immediate display o Pre-format of messages o Pre-formatting of pages
DO	Deleted	TOOL	The iterative function provided by this element, in conjunction with the AGAIN element, will be accomplished using the SOLVE and IF element.
EJECT	Included	OCS II	The function performed by this element contributes to the readability of the program listings.
END	Modified	OCS II TOOL	This element has been expanded to permit the transfer of control to a specified location.
ENTRY	Deleted	OCS II	The functions accomplished by this element are in the BEGIN and DEFINE elements. All labels associated with a BEGIN element are defined as entry points; any label can be defined as an entry point using the DEFINE element.
EQUATE	Deleted	OCS II	The function accomplished by this element is incorporated in the DEFINE element.

TABLE 3-2. BASELINE ELEMENTS (continued)

Element	Recommended Action	Source	Comments
EVALUATE	Deleted	TOOL	The "SOLVE" mnemonic was selected because it has fewer characters. The functions provided by this operator have been incorporated in the "SOLVE" operator, except for logical operations. This capability will be restored if subsystem analysis indicates that bit manipulation is required.
EXECUTE	New		This element provides the capability to execute a concurrent task. It has the capability to preselect options of the program it is executing, eliminating the need for the entry of keyboard options. In addition, the program run time and priority can be specified.
FORK	New		This element provides the capability to initiate parallel CPU execution, and was added to take advantage of the multi-processing configuration. This element was later determined to not be required.
GOTO	Modified	OCS II	This element has been incorporated to provide an unconditional branching capability, and has been expanded to provide the capability for indirect addressing. In addition, the capability of relative addressing (*+D) has been deleted. The relative addressing fosters conditions which reduce the maintainability of the program.

TABLE 3-2. BASELINE ELEMENTS (continued)

Element	Recommended Action	Source	Comments
IF	Modified	TOOL	The element function and name of the IF element are unchanged. However, the logical and contact closure operations were not included. If subsystem analysis indicates the requirement for bit manipulation, and the capacity of the arithmetic element are inadequate for discrete checks, these options will be restored. Also, the element compares values rather than checking flags which are set by the CHECK element.
INPUT	Deleted	OCS II	The function of waiting for keyboard inputs has been incorporated in the READ element. The feature to continue, if no response is received in a time period, was not included. Either a keyboard entry is required,or it is not.
INTERRUPT	Deleted	OCS II TOOL	This element was deleted since this control is best provided as part of the Executive.
JOIN	New		This element re-unites CPU's which have been executing parallel segments of elements, and was later determined not to be required.
JUMP	Deleted	OCS II	This element has been deleted in favor of the GOTO element.

TABLE 3-2. BASELINE ELEMENTS (continued)

Element	Recommended Action	Source	Comments
LOAD	Deleted	OCS II	The function of this element has been incorporated in the EXECUTE element. The term EXECUTE is considered more appropriate than LOAD.
MEASURE	Modified	OCS II TOOL	This element has been modified to permit the inclusion of a limit check in its function, and to use symbolic test point addressing.
MILESTONE	Deleted	TOOL	The function of this element has been incorporated as part of the DISPLAY element, reducing the number of elements.
READ	Modified	OCS II	This element has been incorporated to access data which is in the data base, or has been stored in a peripheral device by the TRANSFER element. It is also used to access parameter information which is entered with the EXECUTE element and keyboard information. To facilitate the handling of the keyboard and parameter entries, a character manipulation feature has been incorporated.
RECORD	Deleted	TOOL	This element was undefined, and therefore not incorporated.
REMARK	Deleted	OCS II	This element is not required since the function can be accomplished by bracketing character strings with asterisks.
REPEAT FLAG	Deleted	TOOL	This element was considered unnecessary and therefore deleted.

TABLE 3-2. BASELINE ELEMENTS (continued)

Element	Recommended Action	Source	Comments
SOLVE	Modified	OCS II	The Solve operator has been incorporated initially without the "logical" operators. This capability has been restored after subsystem analysis indicated a requirement.
SPACE	Included	OCS II	This pseudo element has been incorporated because it will contribute to the readability of the program listing.
START	Deleted	TOOL	The function of this element has been incorporated in the EXECUTE element.
STATUS	New		This element is used to access the status information relating to configurable groups (an LRU or group of LRUs), which is maintained by the Data Management System.
STIMULATE	Modified	OCS II TOOL	This element has been incorporated, with modification, to use symbolic test point addressing.
STOP	Modified	OCS II TOOL	The function of this element has been incorporated and modified to permit an error response.
TERMINATE	Deleted	OCS II	This pseudo element is not considered a requirement, since its function can be implied from the structure of the input data.

TABLE 3-2. BASELINE ELEMENTS (continued)

Element	Recommended Action	Source	Comments
TIME	New		This element is used to obtain current time, which is considered a necessary function.
TRANSFER	Delete	OCS II	This element is recommended for deletion. This capability is provided by the READ element.

3.4 FINAL ELEMENT REQUIREMENT DEFINITION

The software analysis of each subsystem provided the functional descriptions and resulting requirements for software modules necessary to accomplish the periodic checkout, fault detection, isolation, and reconfiguration functions of each subsystem. These application program modules were then analyzed to determine what language elements and appropriate modifiers were required to accomplish that specific checkout function.

The significant changes from the baseline elements identified in Table 3-2 resulted in enhancements to the modifiers (options) for certain elements. New elements were defined to satisfy certain unique checkout and data management requirements. Certain elements were determined not to be necessary to satisfy the OCS Requirements.

The following paragraphs define the language elements which were determined to be required to accomplish automated on-board checkout. This study recommends a restricted number of elements to accomplish checkout, utilizing a sophisticated high level test-oriented language.

3.5 FUNCTIONAL REQUIREMENTS AND SUBSET STRUCTURE

This study recommends that the language elements be divided into two functional categories, (1) program development, and (2) system communication.

These concepts, as well as the subsequent subset concept, are discussed in Section 1 under Language Architecture.

The system communications function provides the communication path between man and the DMS as well as between programs. The elements for the system communication function are not defined in this study due to their not being unique to checkout. Capabilities such as a "desk calculator" element and an EXECUTE element would comprise the system communication function. The program development function is used to develop the application programs required to accomplish the respective subsystem checkout requirements.

The program development function is subsequently divided into subsets for ease of learning. The Basic subset and the On-Board Checkout subset are defined by this study as they relate to the Space Station checkout requirements. The Basic subset contains those elements which were common to all subsystem general applications. The OCS subset meets the requirements which are unique to the checkout process.

The Support Services paragraphs in Section 2, Executive Architecture, of this document complement the language requirements analysis for the Language Control and Language Services.

3.5.1 PROGRAM DEVELOPMENT FUNCTION

The following listing consists of an element summary of the Basic subset.

ELEMENT	SHORT FORM	DESCRIPTION
BEGIN	BG	Defines the start of a sequence of elements.
CALL	CL	Subroutine linkage.
DEFINE	DF	Defines variables and symbols.
DELAY	DL	Suspends program execution for a specified time period.
DISPLAY	DP	Presents messages on the display.
EJECT	EJ	Provides the capability to begin subsequent information on a new page.
END	EN	Defines the end of a sequence of elements.
EXECUTE	EX	Schedules a specified task for execution.
GOTO	GO	Unconditional branch.
IF	IF	Comparison and conditional branch.
READ	RD	Retrieves data from auxiliary or main storage.
SEARCH	SE	Processes character strings.
SOLVE	SL	Computes algebraic expressions.
SPACE	SP	Provides the capability to insert blank lines in the program listing.
STOP	ST	Terminates execution of a task.
TIME	TI	Obtains current time.
WRITE	WR	Transmits data to auxiliary or main storage.

The following listing consists of an element summary for the On-Board Checkout subset.

ELEMENT	SYMBOL	DESCRIPTION
ALLOCATE	AL	Used to assign a configurable group of components to a program.
MEASURE	MS	Measures specified test points.
STATUS	SA	Used to obtain or set the status of a configurable group of components in the data base.
STIMULATE	ST	Applies a specific stimulus to a specified test point.

The elements of the high-level language are described in a through u below. The short form of the element name appears in parentheses beside the long form; either of which may be used.

Braces { } are used to indicate that a choice must be made from the list which is enclosed.

Brackets [ ] are used to indicate that use of the enclosed item or items is optional.

The use of capital letters indicates a key word which must be coded as shown. Lower case letters represent generic terms whose coded representation is left to the user.

The optional label is a name used by other elements to reference the element which defines it.

a. NAME: ALLOCATE (AL) SUBSET: OCS

FUNCTION: This element is used to assign a configurable group to this program for operational control. This same element is used to de-assign a configurable component.

FORMAT:

$$\left[\text{label}\right] \text{ ALLOCATE} \quad \text{symbref} \left[\begin{matrix}\text{, errloc}\\ \text{, DEASN}\end{matrix}\right]$$

symbref — symbolic reference to identify the configurable component.

errloc	The location where control is to be transferred if the device cannot be allocated.
DEASN	This is a keyword that indicates the identified element is to be de-allocated.

EXAMPLE: ALLOCATE PORT1,ERRO1

This element will attempt to allocate port number 1 to this program. If successful, execution will be resumed at the next sequential element. If unsuccessful, execution will be resumed at ERRO1.

b. NAME: BEGIN (BG) SUBSET: Basic

FUNCTION: This element is used in conjunction with the END element to define a sequence of elements. The label on this first BEGIN element is defined as the task name.

FORMAT: [label] BEGIN [(parameter-list)][,'CCC...'][,labelx]

parameter-list	The list of symbolic test point parameters which will be referred to in the program.
'CCC...'	An alphanumeric character string which will be used as a heading on the program listing.
labelx	A symbolic label which defines the start of a segment which is used in the event that execution must be terminated prior to completion. Failure to specify this parameter indicates that no program bookkeeping is required in the event of a forced test end.

EXAMPLE: POWRCHK BEGIN (A,B,C),'POWER TEST',FTEND

The task name assigned to this program will be "POWRCHK", and the program listing will reflect the heading, "POWER TEST".

The label "FTEND" identifies the segment to accomplish termination bookkeeping, if the test terminated before normal completion

When used at the keyboard, the BEGIN logic enters the "define" mode of control, as indicated in the support services paragraphs in Section 2 of this report.

Exit from the define mode is done by END.

c. NAME: CALL (CL) SUBSET: Basic

FUNCTION: This element accomplishes "subroutine linkage" between program modules. The current data cell contents are saved,. and ten new ones are allocated to the program, The END element will restore the original data cells and return control to the element after the CALL element.

FORMAT: [label] CALL seqname [, (parameter-list)]

seqname	1-8 alphanumeric characters which identify a sequence.
parameter-list	A list of data cells or DEFINEd items for use by the sequence being CALLed. For each parameter specified, there must be a corresponding symbol in the BEGIN statement of the CALLed program.

EXAMPLE: CALL PWRTEST, (COMMON)

The segment identified as "PWRTEST" will be executed. Both the calling program and the "PWRTEST" routine will have access to the area defined as COMMON.

* The EXECUTE element should be used to initiate programs which can be processed concurrently with the initiating task.

d. **NAME:** DEFINE (DF) **SUBSET:** Basic

FUNCTION: This element is used to define the contents of areas of memory, or equate values with symbols.

FORMAT:

```
                        ⎧'ccc...' ⎫
                        ⎪ddd...   ⎪
[label]     DEFINE      ⎨X'hhh...'⎬    [,REP=ddd]
                        ⎪B'bbb...'⎪    [,EQ     ]
                        ⎪labelx   ⎪
                        ⎩ENTRY    ⎭
```

'ccc...'	A string of alphanumeric characters.
ddd...	A decimal number.
X'hhh...'	A hexadecimal number.
B'bbb...'	A bit string.
labelx	A symbol defined in the program.
ENTRY	The label as specified in the element is an entry point for execution of this element.
REP	The repetition symbol indicates the number of times a value is to be repeated. A decimal number 1-999 will be accepted.
EQ	The equate symbol indicates that this definition does not require assignment of storage.

EXAMPLE: ABCD DEFINE X'AC93'

The location, ABCD, will contain the hexadecimal number AC93.

e. **NAME:** DELAY (DL) **SUBSET:** Basic

FUNCTION: Program execution is suspended until the specified time period has elapsed. If the specified interval exceeds a minimum level, the processor will be released for other task performance.

FORMAT:

$$[\text{label}]\quad \text{DELAY}\quad \begin{Bmatrix}\text{M}\\ \text{S}\\ \text{MS}\\ \text{US}\end{Bmatrix} = \text{ddd}$$

ddd	A number indicating the amount of delay. The units for time are expressed as:

US	-	Microseconds
MS	-	Milliseconds
S	-	Seconds
M	-	Minutes

EXAMPLE: DELAY S=50

This statement will result in a 50-second delay.

f. NAME: DISPLAY (DP) SUBSET: Basic

FUNCTION: To write messages on a display unit. Messages are coded with a unique number and are predefined in skeleton form, external to the sequence. A utility program is used to define messages and store them in the data base for use at execution time by the DISPLAY element. If an undefined message is DISPLAYed, the information coded with the DISPLAY element is written, together with the sequence identification and a note to the effect that the message is undefined.

FORMAT:

$$[\text{label}]\quad \text{DISPLAY}\quad \text{message-id}\quad [\text{, (insert}_1\text{, insert}_2, \cdots)]$$

$$[\text{, PAGE=p}]\quad [\text{, LINE=11}]\quad [\text{, FR=dddd}]$$

$$\left[\text{, CONS=}\quad \begin{Bmatrix}(\text{c}_1, \text{c}_2, \ldots)\\ \text{ALL}\end{Bmatrix}\right]$$

message-id	The 1-8 character identification of the predefined message which is to be displayed.
insert_n	The data to be inserted at predefined places in the message. Can refer to data cells or DEFINEd items.
PAGE=p	p defines the page number to which the display should be directed. If this is omitted, the display will be routed to the display page which called the program.

LINE=ll	ll specifies the line number to which the display should be directed. If this is omitted, the display will be routed to the display page which called the program.
CONS=$\left\{\begin{array}{l}(c_1, c_2 \ldots)\\ \text{ALL}\end{array}\right\}$	c_n specifies a specific console number. If this parameter is omitted, the display will be routed to the console from which the program was initiated. ALL indicates that the display is to be presented to all consoles.
FR=dddd	Specifies a micro data assembly frame number in the range 0-6000.

EXAMPLE:

Assume that message number 784 has been defined as follows:

REMOVE AND REPLACE x LOCATED AT y

The following sequence, identified as PGM6, will cause the message to be displayed:

```
PGM6    BEGIN
LRU     DEFINE    'CCCCCCCC'
LOC     DEFINE    'CCCCCCCC'
           .
           .
           .
        CALL   XYZ,(LRU,LOC)   Subordinate seg-
           .                   ment which obtains
           .                   LRU name and
           .                   location.
        DISPLAY   784,(LRU,LOC)
```

The message will appear as follows:

```
MSG784 (PGM6): REMOVE AND REPLACE MOTOR
               LOCATED AT BAY 6:
```

The identification of the message is MSG784, and the source is PGM6. In the message text, "MOTOR" and

"BAY 6" have replaced x and y, respectively. The inserts are varying in length, and the language routine adjusts the position of following message portions accordingly.

If a DISPLAY specifies more inserts than have been defined in the message, the excess items are written following the message so that no information is lost.

If in the above example, message number 784 had not been defined, the following would have been displayed:

```
UNDEFINED MESSAGE 784 (PGM6): MOTOR, BAY6
```

g. NAME: EJECT (EJ) SUBSET: Basic

FUNCTION: Causes the next line of information on the program listing to appear on the first line of the next page allocatable to high level language use.

FORMAT: EJECT

EXAMPLE: EJECT

h. NAME: END SUBSET: Basic

FUNCTION: This element indicates the end of a segment of code, and control is released.

FORMAT: [label] END labelx

labelx	The identity of the sequence being terminated. The identity was established by the use of labelx in a previous BEGIN statement. If the element terminates a Task, control is returned to the Master Executive.

EXAMPLE: END PWRCHK

Control will be returned to the program which invoked the sequence identified as PWRCHK.

When used at the keyboard, END exits from the "define" mode. See BEGIN element description for further definition.

i. NAME: EXECUTE (EX) SUBSET: Basic

FUNCTION: The specified task is scheduled for execution.

FORMAT:

[label] EXECUTE task-name [, EN=enloc] [, ER=erret]
[, TIME=time] [, PARM=($parm_1$, $parm_2$, . . .)]
[, PRTY=priority]

task-name	The name assigned to this sequence of elements at generation time. This parameter will create default values for this program.
enloc	A defined label in the program being called where execution is to begin.
erret	A defined label in the initiating program to which control is returned, in the event the system is unable to schedule the requested program for execution. In the event this parameter is not specified, the task initiating this call will be terminated.
time	The cumulative time period for execution of this segment. If not specified, the value specified at generation will be used. If the program has not completed execution when this timer expires, it is assumed to be in an error status.
$parm_n$	Parameter information for the task.

priority	Permits the user to specify the priority level at which the specified task is to be executed.

EXAMPLE: EXECUTE PWRTEST

The sequence of elements defined as PWRTEST will be scheduled for execution.

j. NAME: GOTO (GO) SUBSET: Basic

FUNCTION: Causes execution of an unconditional branch to the specified label.

FORMAT:

$$\left[\text{label}\right]\ \text{GOTO}\ \left\{\begin{matrix}\text{LABELLOC}\\ \text{Dx}\end{matrix}\right\}$$

labelloc	Any combination of 1-8 letters and characters which are defined in the label field of an element.
Dx	An unconditional branch will be made to the address specified in the referenced data cell. Dx defines a data cell where x = 1-9.

EXAMPLE: GOTO READINPT

Control is transferred to the element identified by the label "READINPT".

k. NAME: IF (IF) SUBSET: Basic

FUNCTION: Performs comparison operations and provides conditional branch logic. If the logical expression is false, the next sequential element is executed. If true, the statement defines the next element to execute.

FORMAT:

$$\left[\text{label}\right]\ \text{IF}\ \left\{\begin{matrix}\text{label}_1\\ \text{Dx}\end{matrix}\right\}\ \text{relational-operator}\ \left\{\begin{matrix}\text{label}_2\\ \text{Dy}\end{matrix}\right\}\left[\text{label}_3\right]$$

$\text{label}_1, \text{label}_2$	Defined constants or variables in the program.
Dx, Dy	Data cells, D0-D9.

relational-operator One of the following:

>	GREATER THAN
>=	GREATER THAN OR EQUAL TO
<	LESS THAN
<=	LESS THAN OR EQUAL TO
=	EQUAL
<>	NOT EQUAL

label_3 The program will branch to this label if the expression is true.

If a label is not specified and the expression is true, the next sequential element is skipped.

EXAMPLE: IF D0 > D2

If the expression is true, (i.e., the value in D0 is greater than the value in D2), the next sequential element will be skipped.

If the expression is not true, the next sequential element is executed.

1. NAME: MEASURE (MS) SUBSET: OCS

FUNCTION: This element measures values at specific test points, and places the result in memory or performs a limit check.

FORMAT:

$$\left[\text{label}\right]\ \text{MEASURE}\ \left\{\begin{matrix}\text{testpoint}\\ \text{table}\end{matrix}\right\}\left[\begin{matrix}\text{, LC=(in, low, high)}\\ \text{, labeloc}\\ \text{, Dx}\end{matrix}\right]\left[\text{,INDEX=Dy}\right]$$

testpoint A symbolic reference which identifies the test point to be measured.

table	A symbolic address for a list of test points. The interpreter differentiates between the test point and table symbolic on the basis of the presence or absence of the INDEX parameter.
labelloc	A defined memory location to store the result of the measure.
Dx	A data cell, where x = 0-9, which is used to store the result of the measurement.
LC	This keyword is used to specify the limit check option.
in, low, high	Control is transferred to these symbolic addresses if the reading meets the respective criteria for the test point in the RDAU tables. in - within limits low - below limits high - above limits
INDEX	Used to identify the relative position in the tables associated with identifying the test point and the area to store the data. This is managed through the use of the elements SOLVE and IF.
Dy	A data cell, where y = 0-9, which is used to index a list of test points.

EXAMPLE: MEASURE TP32, D4

This statement will measure the value on the specified test point and store the result in data cell number 4.

m. NAME: READ (RD) SUBSET: Basic

FUNCTION: Used to obtain keyboard entries, and to retrieve data which was placed in storage using a transfer element.

FORMAT: $\left[\text{label}\right] \quad \text{READ} \quad \left\{\begin{matrix}\text{symbref}\\ \text{KEYBD}\end{matrix}\right\} \text{, memref}$

symbref	Is a symbolic reference to identify the data to be retrieved.
KEYBD	Indicates that an entry is expected from the keyboard. Execution will be suspended until received.
memref	A symbolic memory reference where the data is to be stored.

EXAMPLE: READ KEYBD, INDATA

The data which was entered at the keyboard will be transferred into the area defined as INDATA.

n. NAME: SEARCH (SC) SUBSET: Basic

FUNCTION: Used to search strings of characters for the specified data.

FORMAT: [label] SEARCH location,'ccc...',match-address

location	Is a symbolic address in memory containing a character string.
'ccc...'	Character string which is being searched for.
match-address	The symbolic address to go to if the search is successful. If the search is not successful, the next sequential element is executed.

EXAMPLE: SEARCH LOC1,'FI',LOC2

The character string stored in location LOC1 is searched for the character string 'FI'. If the search is successful, execution continues at LOC2. If the search is not successful, the next sequential element is executed.

o. NAME: SOLVE (SL) SUBSET: Basic

FUNCTION: This operator performs algebraic operations on data. The result is placed in the specified location. In evaluating expressions, the priority order of the operators is:

1 ** (exponentiation)
2 *, / (multiply, divide)
3 +, - (add, subtract)

If two or more operators of the same priority appear in the same expression, the order of priority of those operators is from right to left.

FORMAT: SOLVE expression

EXAMPLE: SOLVE D2=(XYZ*3+10)/(D7-3**2)

Assuming D7=14, and XYZ=5, D2 will be set equal to 5.

p. NAME: SPACE (SP) SUBSET: Basic

FUNCTION: This element will insert up to 9 blank lines in a program listing produced by the compiler.

FORMAT: SPACE d

d	Specifies the number of blank lines to be inserted and can vary from 1-9. If this parameter is omitted, a default of 1 will be assumed.

EXAMPLE: SPACE 3

Three blank lines will be inserted in the program listing.

q. NAME: STATUS (SA)

FUNCTION: This element is used to obtain or set the status of configurable components from a data base. This element can also be used to indicate agreement with a predetermined status.

FORMAT: $\left[\text{label}\right]$ STATUS symbref, $\left\{\begin{array}{l}\text{location}\\ \text{CLR}\\ \text{Dx}\\ \text{FAIL}\end{array}\right\}$

symbref	Symbolic reference to identify the configurable component.

location	A symbolic program location to place the status.
CLR	Clears a fail indication.
Dx	A data cell where x = 0-9, where the status is to be placed.
FAIL	This indicates that the specified component should be marked with a "failed" indicator in the data base.

EXAMPLE: STATUS HATCH1, D0

This element will obtain the status of hatch number 1 and place the status in data cell D0.

r. NAME: STIMULATE (ST) SUBSET: OCS

FUNCTION: This element will apply a specified stimulus to a specific test point.

FORMAT: [label] STIMULATE testpoint, options

testpoint	Identifies the test point to be activated.
options	To be defined.

EXAMPLE: STIMULATE TP32, OPTIONS

The test point identified by the symbolic reference TP32 is activated in accordance with the specified options.

s. NAME: STOP (ST) SUBSET: Basic

FUNCTION: To stop execution of a task which is an execution, or scheduled for execution.

FORMATS: [label] STOP taskname, errorloc

taskname	Name of a task currently scheduled or in execution.

errorloc	Location to which control is returned if the specified task is neither scheduled nor in execution.

EXAMPLE: STOP PWRTEST

The task identified as "PWRTEST" is terminated.

NAME: TIME (TI) SUBSET: Basic

FUNCTION: Obtains current time from real time clock and stores binary value in memory.

FORMAT:

$$\left[\text{label}\right]\ \text{TIME}\ \left\{\begin{matrix}\text{location}\\ \text{Dx}\end{matrix}\right\}$$

EXAMPLE: TIME D7

The current time is obtained from the real time clock and stored in the symbolic location TIME.

NAME: WRITE (WR) SUBSET: Basic

FUNCTION: This element is used to place data on peripheral storage or in main storage tables.

FORMAT:

$$\left[\text{label}\right]\ \text{WRITE location, symbref}\left[\begin{matrix}\text{,CREATE}\\ \text{,REPLACE}\\ \text{,DELETE}\\ \text{,CKPT}\end{matrix}\right]\ \left[\text{,LOOP=x}\right]$$

location	Is a symbolic reference to a memory location which contains the data to be transferred to a peripheral device.
symbref	The symbolic reference which identifies the destination of the referenced data.
CREATE REPLACE DELETE	The create parameter indicates that a new record is to be established. The replace parameter indicates that an existing record is to be replaced. In the event that a record identified for replacement cannot be located,

	a new record will be created. The delete parameter indicates that an existing record is to be removed. If none of the specified parameters are identified, the replace option will be assumed.
CKPT	This parameter indicates that the data being saved is checkpoint data and should be maintained in the manner prescribed for checkpoint data.
LOOP=x	x indicates the number of successive copies of a file to be retained.

Section 4

RECOMMENDATIONS FOR IMPLEMENTING STUDY RESULTS

4.1 SUMMARY OF RECOMMENDATIONS

The On-Board Checkout software techniques defined by this study for the 33-foot diameter Space Station are also applicable to the Modular Space Station, and the current Space Shuttle design.

Prior to completing Phase C, it is recommended that the following advanced study tasks be performed relative to the Space Station; and that these tasks use the output of this study as primary inputs.

- Perform detailed design of test point utilization and placement for the 33-foot station.
- Design an On-Board Checkout System for the Modular Station.
- Perform detailed design and breadboard implementation for the Space System Language.
- Study requirements for expanding the recommended high-level test-oriented language to be utilized in experiment and other systems.
- Perform a trade study, and provide simulation software to determine the number of copies of the Executive which should be maintained in main memory in a Space Station multiprocessor environment.
- Develop an integrated configuration data base design, and generate subsystem design data for the data base to include subsystem hardware, LRU, and assembly level components.
- Perform a detailed study to determine the level of On-Board Checkout which could be accomplished during Shuttle missions, versus what should be accomplished during prelaunch, or in a docked environment.

It is important that emphasis be given to checkout as a prime consideration in the Phase C hardware and software designs of either Space Shuttle or Space Station. An efficient checkout design is significant when considering cost effectiveness over the total mission life.

4.2 DISCUSSION OF RECOMMENDED TASKS

A brief discussion of those tasks recommended in the previous paragraph is provided. Consideration should be given to the utilization of the data available from this study as primary inputs for a On-Board Checkout techniques study and/or design recommendations task for the Modular Station or Shuttle.

4.2.1 TEST POINT UTILIZATION AND PLACEMENT

The Phase B 33-foot Space Station hardware design was used as the baseline for this study. The effect of the number of test points on remote versus central limit-checking requires a more detailed analysis of test point utilization. The rate at which certain functions should be monitored must be re-evaluated at a subsequent design level to more accurately evaluate processor load criteria. The ultimate placement of test points and BITE may alter the checkout technique of any subsystem.

4.2.2 MODULAR STATION ON-BOARD CHECKOUT

The Modular Station will probably not have a total checkout capability initially, because of the current economic environment. Rather, the autonomous checkout capability will evolve for noncritical operational and experiment systems during the ten-year life of the Space Station. This identifies an additional requirement for a high-level On-Board Test-Oriented Language to aid in the development of a total checkout capability.

Although the modular station design is preliminary to date, some compatibility with the current OCS study is evident; and provides a base to determine the extent of new checkout techniques and criteria to be designed. Preliminary analysis indicates that the checkout techniques developed during this study are compatible with the modular station. The periodic test techniques may change, and the amount of status monitoring will vary with each respective subsystem. The Power and EC&LS Subsystem will probably reflect the greatest delta in checkout requirements. Initial design indicates that the Solar Array Power Subsystem will be used for the Modular Station. It appears feasible to significantly reduce the test point requirements that are defined in this study for the Solar Array Power Subsystem. The EC&LS Subsystem should require separate thermal control for each module. This could increase the total system hardware and power requirements by a significant amount. The low thrust propulsion system will probably disappear from the modular station; however, the number of total thrustors should increase due to the decentralization of the vehicle mass. The GN&C Subsystem could reflect changes in the landmark tracking and star tracking assemblies. These may even be deleted from the modular station because of cost requirements.

4.2.3 HIGH LEVEL LANGUAGE

Upon completion of the preliminary Phase C Space Station design, a detailed design of the language requirements and subsequent program sizing should be performed to determine all DMS requirements unique to checkout. This design should be breadboarded to determine its validity, specifically for the man-machine interface characteristics.

Initial data indicates that other subsystems, specifically in the experiment applications, could benefit from the study of a high-level test-oriented language. Language elements applicable to these systems should be defined. A study is advisable to determine the requirements and specific applications of a high-level language for program development and communication for all Space Station applications.

4.2.4 EXECUTIVE MAIN STORAGE UTILIZATION

A multiprocessor environment, such as that which exists on the current station design, demands considerable study to reduce memory contention. It is recommended that a simulation and trade study be performed to determine the most efficient number of copies of critical executive program modules that should be kept redundantly in main storage.

4.2.5 INTEGRATED CONFIGURATION DATA BASE

It is recommended that a data base be developed to aid in the Space Station or Space Shuttle design, as well as the checkout fault isolation sequence.

Careful design of an information processing system, whether for On-Board Checkout or for broader purposes, entails design of the data base stored in main and auxiliary storage. Such data base design substantially influences operational effectiveness, particularly when a high-level language is employed for problem solving. Types of checkout problems which can be solved by appropriate data base concepts are:

- Loss-of-capability assessment in the event of a fault.
- Identification of reconfiguration paths for determining what can be used in place of a failed LRU.
- Resource allocation and resolution of on-line interference between concurrent operations.
- Identification of the scope of fault isolation which is in progress as succeeding faults occur.
- Real-time assessment of system status.
- Identification of LRU location and nomenclature.
- On-board inventory control.
- Reduction of the impact of hardware changes on checkout, operations, and experiment programs.
- Management of repair schedules and maintenance procedures.

This study indicates a significant requirement for a data base on-board the Space Station. Similar requirements would exist for Shuttle checkout.

4.2.6 SPACE SHUTTLE

The mission definitions and redundant hardware design currently available for Shuttle indicate that only limited checkout functions will be performed during flight. The significant checkout requirements exist during the pre-launch activities. The techniques indicated in this study are applicable to Shuttle On-Board Prelaunch Checkout. The total Shuttle Checkout concept should be studied to make the Shuttle checkout design as autonomous as feasible.

The principle differences to date between Shuttle and Space Station design that impact On-Board Checkout are as follows:

- Station has indicated the need for remote limit checking (e.g., at the Acquisition Control and Test (ACT) unit or a Remote Data Acqusition Unit (RDAU), while Shuttle has not established the need for such features.
- Station has only one level of redundancy with remove-and-replace maintenance, while Shuttle has sufficient redundancy for a mission without maintenance. This means Station must recertify after a remove-and-replace activity, while Shuttle must reconfigure only.
- Station automatically announces a "limit exceeded" condition via a hardware polling operation and a processor I/O interruption. Shuttle transmits all measurements to the processor for limit-checking and any subsequent action (critical signals are checked at intervals frequent enough that failures can be tolerated in the inter-sample period).
- DMS checkout is implemented differently due to DMS hardware definitions and mission constraints.

4.2.7 SOFTWARE TASK APPLICABILITY

It is recommended that the techniques presented herein be applied toward the software analysis of On-Board Checkout for any space system. In addition, the data available in this study can be used as sizing inputs to other studies and subsystem design integration.

Appendices A through F provide definitions of the subsystem software design approach, and sizing criteria developed by this study.

APPENDIX A

STRUCTURE SUBSYSTEM CHECKOUT PROGRAM DESIGN

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF ILLUSTRATIONS

LIST OF TABLES

1.0 SCOPE

The basic Space Station structure provides the necessary pressurized habitat, equipment support, meteoriod protection, radiators, insulation, and docking interfaces to meet operational requirements.

The fault detection function required for the Structures Subsystem is accomplished by tables which are monitored by the OCS executive program. The tables contain the parameters which must be monitored to assure subsystem performance. These tables are transferred to the Remote Data Acquisition Unit (RDAU) via the master executive program, and limit checking is accomplished. Figure 1-1 provides a graphic description of this function.

The program described by this document is required for periodic checkout and fault isolation.

Initiation of the periodic checkout function is accomplished as the result of a keyboard entry by a crew member. It is anticipated that periodic checkout will be accomplished prior to a scheduled event such as docking, artificial "G" deployment, or during specific operational times such as shift changes where status information pertaining to hatch and docking port positions are of interest to the on-coming crew.

The fault isolation utilizes the same software modules as the periodic checkout; however, it is anticipated that analysis of the detected error will permit selection of the appropriate module to begin the required fault isolation.

2.0 APPLICABLE DOCUMENTS

- Space Station MSFC-DRL-160 Line Item 8, Volume V Book Mechanical, McDonnell-Douglas Astronautics Co.
- Baseline Subsystem Descriptions, Interim Report RA-1, August 1970, Contract NAS9-11189, IBM.
- Line Replaceable Unit Definition, Interim Report RA-6, September 1970, Contract NAS9-11189, IBM.
- Subsystem Test Descriptions and Measurement Stimulus List, Interim Report RA-8, November 1970, Contract NAS9-11189, IBM.

3.0 REQUIREMENTS

This program meets the periodic testing and fault isolation requirements for the Structure Subsystem.

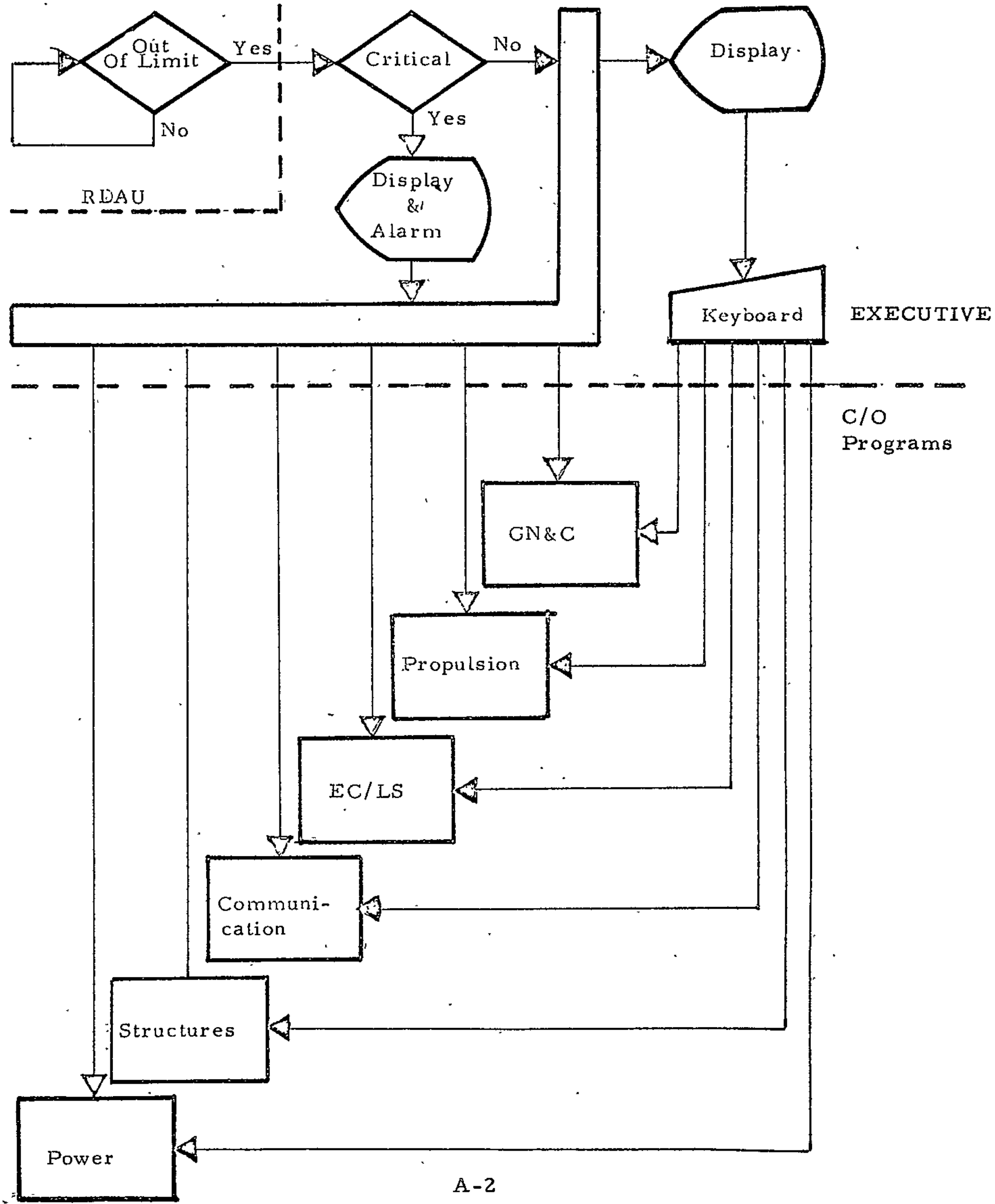


Figure 1-1. Fault Detection Logic

3.1 PERFORMANCE

Four specific functional areas of the subsystem require automated checkout. They are:

- Docking Mechanisms
- Antenna Deployment Mechanisms
- Basic Structure
  - Spacecraft Access
  - Hatches
  - Hanger
  - B/I Power System Handling
- Artificial "G" Considerations

Figure 3-1 provides a block diagram of the functional areas of this subsystem.

The varied requirements of program execution are met by providing the operator with the capability to select specific options of the program.

The operation of the options of this program is depicted in Table 3-2. These options include the capability to provide the current operational status, docking mechanism checkout, antenna operations checkout, and to check out the provision included for an artificial "G" environment.

3.1.1 SYSTEM REQUIREMENTS

This section describes system constraints and requirements which have influenced design.

3.1.1.1 Subsystem Definition

This program specification is based upon the subsystem definition which is available as a result of this study contract. The test points for this subsystem are currently defined at the assembly level; and consequently every failure which is detected cannot be identified with a special LRU (Refer to Table 3-1).

3.1.1.2 Docking Mechanism Checkout Module

There is no test point available to extend or retract the docking ring. The existence of a test point labeled "Docking Ring Pressure Control" has been assumed. This stimuli will permit extension and retraction of the docking ring.

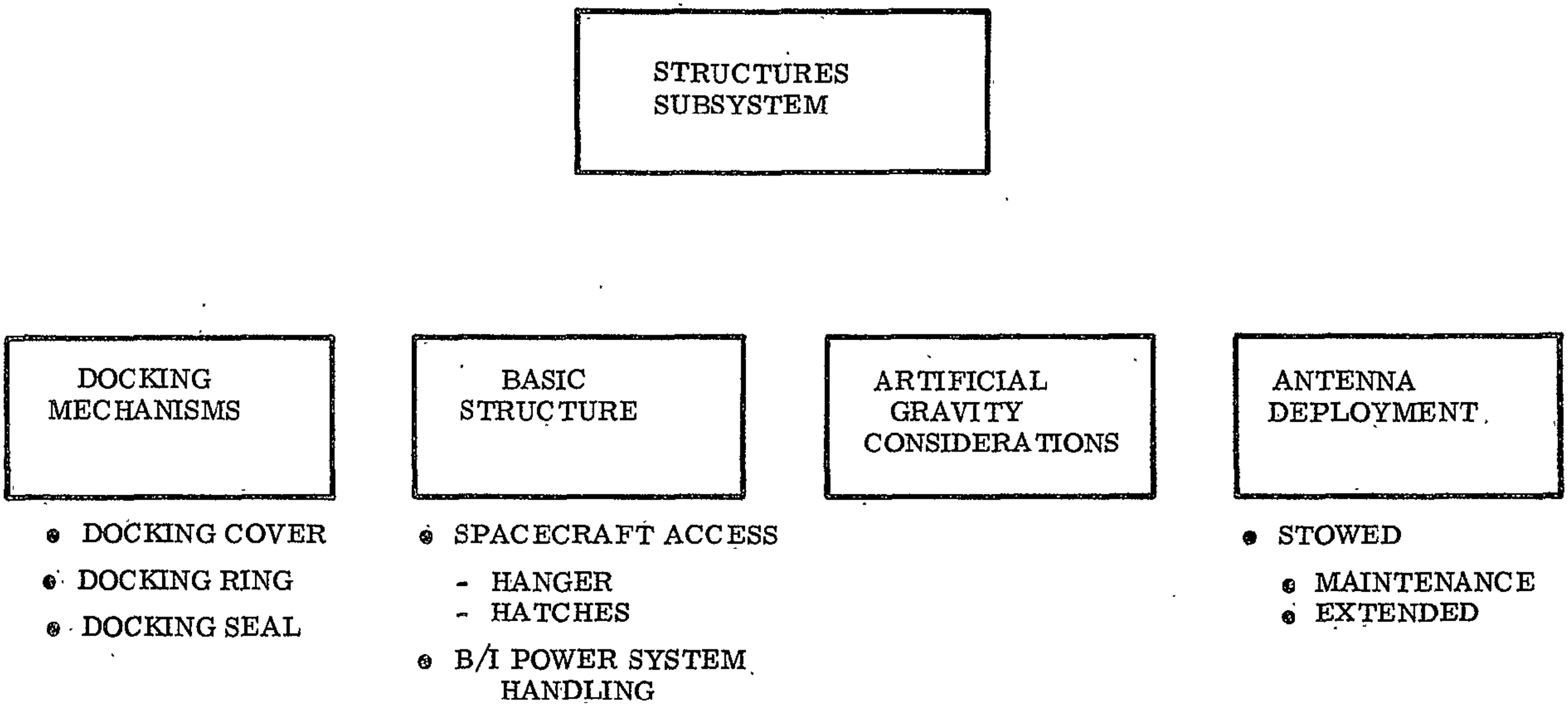


Figure 3-1. Structure Subsystem Functional Areas

This table includes a list of LRU types which make up the structures subsystem. Those which can be associated with specific test points are indicated by asterisks.

Table 3-1. LRU Identification

Line Replaceable Unit	Identifiable
Docking Mechanism Shock Struts	*
Docking Port Inflatable Seals	*
Hatches and Airlock Doors Inflatable Seals	
Inflatable Airlock	
Viewport Window Assembly	
Hatch Temperature Indicators	
Hatch Pressure Indicators	
Hatch Assembly	*
Despin Module Drive Unit	*
Cable Deployment Module Drive	*
Docking Port Seal Latches	
Antenna Boom	
Antenna Boom Drive Unit	*
Cargo Handling Hoist	
Cargo Hoist Cable	
Electric Drive Unit, Isotope System Handling	*
Handling Aids, Isotope System	

3.1.1.3 Artificial "G" Checkout Module

The periodic and fault isolation test requirements indicated in the measurement and stimulus list include test points labeled "Cable Deployment Drum Travel". The information supplied by these test points can only be evaluated during an actual deployment. Consequently, these test points have been omitted for periodic and fault isolation testing. It is anticipated that additional test points will be identified in the future to assure the continuity of these measurements.

There are controls provided in the system for the operation of the S-II latches. These latches are used to hold the S-II stage and are released during the deployment sequence. Because of the problems that could occur if the latches failed to relock during a test, these components are not exercised as a function of the periodic or fault isolation test.

The checkout of the design module requires limit checking on the basis of operational information to be acquired from the operational system. Lack of definition regarding specific data and algorithms prevents consideration of this requirement in the software analysis.

3.1.1.4 Allocable Equipment

The checkout which this program must accomplish requires that the following components be independently identifiable, and dedicated to this program for a specific time interval.

Docking Ports
Hanger
Hatches

Antennas
De-spin Module
Cable Deployment Module

3.1.2 OPERATIONAL REQUIREMENTS

This program specification defines specific operational requirements for periodic testing and fault isolation of the following areas:

- Docking Mechanism
- Hatches, Airlocks, and Viewports
- Artificial Gravity Experiment Provisions
- Antenna Deployment
- Isotope Power System Handling and Replacement Equipment

The specific program modules which meet these requirements are:

- Option selection

- Operational status
- Docking mechanism checkout
- Spacecraft access module
- Antenna fault isolation
- Isotope/Brayton power handling
- Artificial "G" checkout

A module block diagram of the program is provided in Figure 3-2.

3.1.2.1 Option Selection

This function processes the selection of options to permit the selective checkout of specific functional areas.

3.1.2.1.1 Source and Type of Inputs

The options selected will be determined from the parameter data supplied to the program. These options are defined in Table 3-2.

3.1.2.1.2 Destination and Types of Outputs

The outputs of the option selection are used by the program to select the proper options for execution. Tutorial displays are also required.

3.1.2.1.3 Information Processing

This program module processes the input parameter information, and executes the program module specified by the options defined in Table 3-2.

The parameter information passed to the program will be examined. If none was included, the program will execute the programmed default option of checking the operational status of specific test points. (Refer to paragraph 3.1.2.2.)

If the keyword OPTN is included in the parameter data, the program displays the available options and waits for the operator to enter his selection.

Any combination of keywords may be entered; however, they must be separated by a comma. The introduction of a blank character will be used to terminate the input string.

As the keywords are identified, an associated flag is set to indicate the function to be accomplished. The program will then examine each flag and transfer control to the appropriate module for execution.

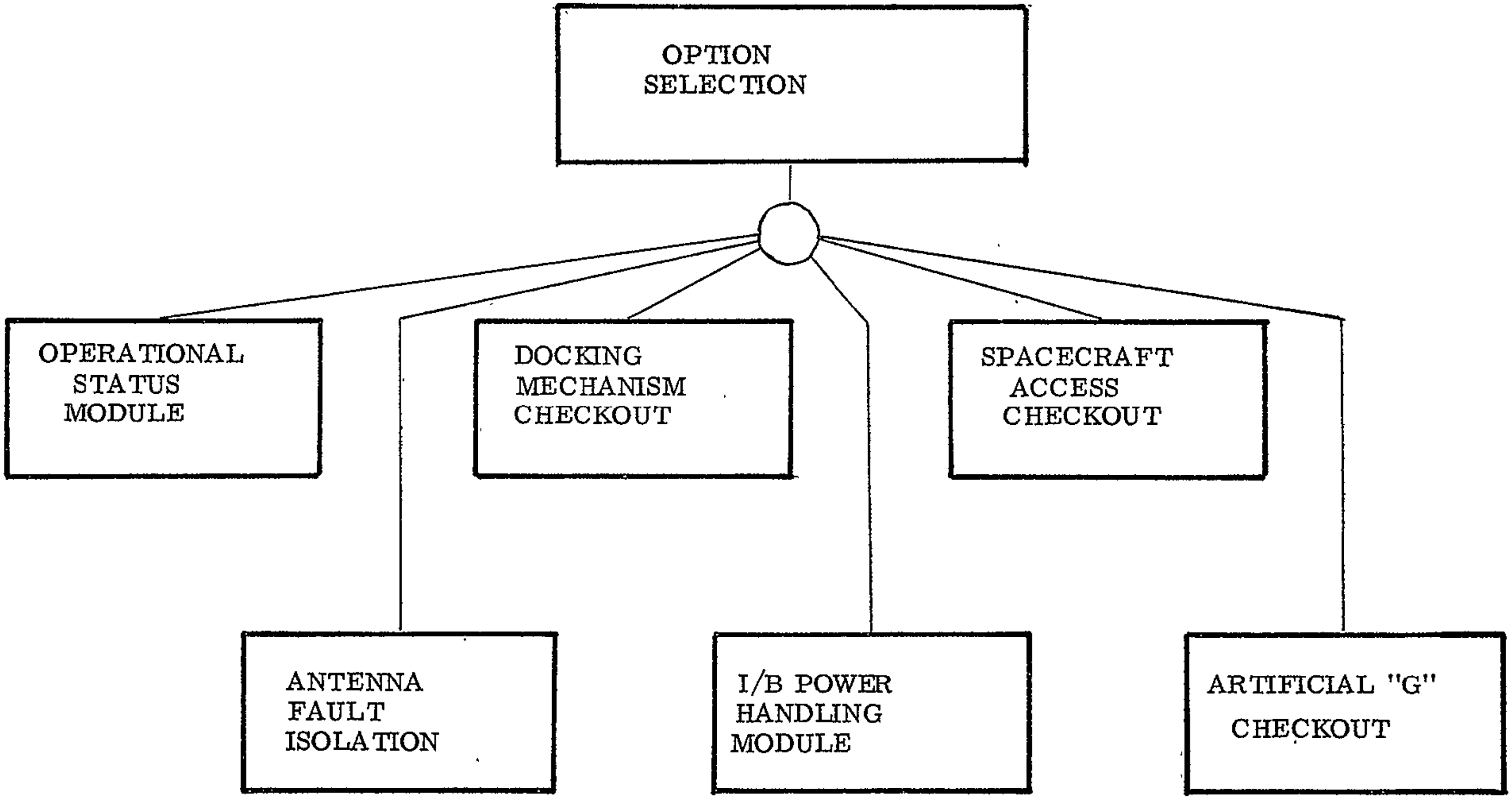


Figure 3-2. Structures Subsystem Checkout Program Modular Block Diagram

Table 3-2. Initial Program Options

Keyword	-	Program Response
STATUS	-	Execute the Operational test point status module
DOCK	-	Execute the Docking mechanism checkout module
ANTENNA	-	Execute the Antenna Fault Isolation module
SPAC	-	Execute the Spacecraft access checkout module
OPTN	-	Display Options
IBP	-	Execute the Isotope/Brayton Power handling module
ATRG	-	Execute Artificial G checkout module
FI	-	Execute fault isolation routines in addition to periodic check.

3.1.2.2 Operational Status

This module is executed to provide the crew with information relating to the current status of the docking ports, hatches, and hanger door positions.

3.1.2.2.1 Source and Type of Inputs

The inputs for this function are obtained from the data base table which the executive monitors to keep track of the status of various system components. This information is based on data received from the following test points:

- Docking Ring Position - This test point indicates if the docking ring is extended or retracted.
- Docking Port Cover Position - This test point indicates if the docking port cover is open or closed.
- Hatch Position Open/Close - Two measurements are required for each hatch to determine if it is opened or closed. One test point indicates the hatch is open, the other indicates the hatch is closed.
- Hanger Door Position Open/Close - Two measurements are required to determine if the hanger is opened or closed. One test point indicates an open status; the other a closed status.

3.1.2.2.2 Destination and Type of Outputs

The status of the test points identified in Table 3-3 are displayed on the display console.

3.1.2.2.3 Information Processing

This routine accesses the data base to determine the positions of the docking ring, the docking port cover, the hatches, and hanger door. This information is displayed at the console which requests execution of the program.

3.1.2.3 Docking Mechanism Checkout

The periodic and fault isolation test requirements for the docking mechanism are identical. This routine is executed to assure the operational status of the docking mechanism.

3.1.2.3.1 Source and Type of Inputs

The primary inputs for this function consist of the subsystem test point data.

Table 3-3. Operational Status Test Points

FUNCTIONAL AREA	STATUS TEST POINTS
DOCKING MECHANISM	DOCKING RING POSITION DOCKING PORT COVER POSITION
SPACECRAFT ACCESS	HATCH POSITION HANGER DOOR POSITION

Specific test points are:

- Docking Port Seal Pressure - The two analog pressures associated with docking port seals are examined to assure that they are within established limits.
- Docking Port Cover Position - This binary test point is used to determine the position of the Docking Port Cover (open/closed).
- Docking Port Cover Control - This test point is used to command the the docking port cover to the open or closed position.
- Docking Ring Pressure Control - This test point is used to extend or retract the docking ring.
- Docking Ring Position Extended/Retracted - There are two test points for each docking port. One signal indicates an extended position while the other signal indicates that the docking ring is retracted.
- Docking Ring Strut Pressure - When the docking ring is extended, this test point is examined to determine whether the strut pressure is within limits.

Additional inputs are received as a result of the operators response to the GO-NOGO options.

3.1.2.3.2 Destination and Type of Output

A descriptive message relative to test points, which indicates an apparent error condition, is presented on the display console. When possible, the associated LRU is also identified.

3.1.2.3.3 Information Processing

This routine uses a subroutine to check the docking seal pressure at each docking port.

The program then attempts to allocate a docking port to prevent altering one of the mechanisms which may be functioning in an operational status.

The program then extends and retracts the docking port ring. Each ring is left in its initial position.

All position information which is obtained from test points is checked by comparison with the current status being maintained in the data base.

For each error that is detected, a failure is indicated for the component, and the operator is presented with an appropriate display.

3.1.2.4 Spacecraft Access Control

This function is used to check the operational status of the spacecraft hatches and hanger door, for both fault isolation and periodic testing requirements.

3.1.2.4.1 Source and Type of Inputs

The inputs for this function consist of the subsystem test point data. Specific test points are:

- Hanger Door Position Open/Close - There are two points for the hanger door. One indicates the open position; the other a closed status.
- Hanger Door Control - This stimuli is used to open and close the hanger door.
- Hanger Pressure - There are two stimuli points for the measurement of the pressure inside the hanger.
- Hanger Temperature - There are two analog points available to measure the temperature inside the hanger.
- Hatch Position Open/Close - There are two stimuli for each of the 16 hatches. One signal indicates the open position; while the other indicates the closed position.

3.1.2.4.2 Destination and Type of Outputs

A descriptive message relative to test points which indicate an apparent error condition are presented on the display console. When possible, the associated LRU is also identified.

3.1.2.4.3 Information Processing

In order to efficiently achieve the requirements for both periodic testing and fault isolation for the spacecraft access components, the two were combined into one module and execution is at the operator's option.

The program assures that both test points which provide hanger door positions are in agreement with each other, and with the status being maintained by the DMS. If disagreement is detected, the operator is provided with an appropriate display. When the points and status are in agreement, the hanger door is moved through both the open and closed positions; and the test points and DMS are again examined for consistency. The operator is notified by a display if the door fails to operate, or the positional indicators provide conflicting information.

Subsequent to verifying the hanger door operation if the fault isolation option has been selected, the temperature and pressure in the hanger are limit checked. In the event of a failure, the hanger is configured non-operational and the operator is notified by a display.

The test points which indicate hatch positions are compared against each other and with the DMS status to assure compatibility. In the event of a discrepancy, the hatch will be configured non-operational and the operator is notified by a display.

For each error that is detected, a failure is indicated for the component; and the operator is presented with the appropriate display.

3.1.2.5 Antenna Fault Isolation

The program requires the capacity to analyze the antenna system for purposes of fault isolation.

3.1.2.5.1 Source and Type of Inputs

The inputs for this function consist of the subsystem test point data. Specific test points are:

- Boom Position - This point is used to determine if the antenna is stowed, extended, or retracted for maintenance.
- Launch Storage Position Lock Status - This test point is used to determine the lock position when the antenna is stowed.
- Antenna Launch Stowage Lock Control - This test point is used to assure that the stowage lock can be locked and unlocked to permit deployment of the antenna.
- Boom Deployment Position Lock Status - This test point is used to determine the boom lock status when the boom is extended.
- Boom Deployment Lock Control - This test point is used to assure that the boom locks can be locked and unlocked when the boom is extended.
- Boom Maintenance Position Lock Status - This test point is used to determine the boom lock status when the boom is in the maintenance position.
- Boom Maintenance Position Lock Control - This test point is used to lock and unlock the boom while it is in a maintenance position.
- Drive Unit Power Monitor - This test point is used to be sure that the power monitor is operating within limits.
- Drive Unit Control - This test point is used to issue a test level to the drive unit power monitor, which is then limit checked.

3.1.2.5.2 Destination and Type of Output

A descriptive message relative to test points which indicate an apparent error condition are presented on the display console. When possible, the associated LRU is also identified.

3.1.2.5.3 Information Processing

This program module performs fault isolation for each antenna. The boom position is read from the specified test point and compared with the current position being maintained by the Data Management System, to insure compatibility. The antenna is in a stowed, extended, or maintenance position. The lock, which is associated with that position, is then checked and again compared with the Data Management System's records for it. If everything is in agreement, the lock is unlocked and relocked to be sure of its operational capacity.

Any discrepancy between positions or failure to lock or unlock results in a message being provided to the operator.

The drive unit power monitor is then limit checked as the final part of this module operation.

For each error which is detected, a failure is indicated for the component; and the operator is presented with the appropriate display.

3.1.2.6 Isotope/Brayton Power Handling Module

This program module is required to check the Isotope/Brayton power supply for purposes of fault isolation.

3.1.2.6.1 Source and Type of Inputs

The inputs for this function are the test point values for the power.

3.1.2.6.2 Destination and Type of Outputs

A descriptive message relative to the power test point is presented on the display console when an error is indicated.

3.1.2.6.3 Information Processing

This module performs fault isolation by limit checking the drive unit power monitor for the Isotope/Brayton power supply.

3.1.2.7 Artificial "G" Checkout Module

This program module is used to checkout the considerations in the Structures for the artificial "G" environment. It is anticipated that it will be executed prior to both spin up and spin down activities.

3.1.2.7.1 Source and Type of Inputs

The inputs for this function consist of the subsystem test point data. Specific test points are:

- Spoke Pressure Supply - This test point is used to assure that the pressure supply for the spoke is within limits.
- Spoke Internal Pressure - This test point is used to assure that the internal pressure within the spoke is within limits.
- De-spin Module Rotation Control - This test point is used to place the de-spin module in rotation.
- De-spin Module Rate - This test point is used to measure the rate of the de-spin module's rotation.
- De-spin Module Power - This test point is used to assure that the power being supplied to the de-spin module is within limits.
- S-II Stage Latches Latched/Unlatched - Two test points are used to ascertain the status of the S-II stage latches.
- Cable Deployment Module Strut Ring Position - This test point is used to determine the strut position. In a zero "G" environment, it should be retracted. In an artificial "G" environment, it should be extended.
- Cable Deployment Module Power Monitor - This test point is examined to be sure that the module power monitor is within limits.

3.1.2.7.2 Destination and Type of Outputs

A descriptive message relative to test points, which indicate an apparent error condition, is presented on the display console. When possible, the associated LRU is also identified.

3.1.2.7.3 Information Processing

This program module is used to accomplish periodic checkout and fault isolation of those components associated with the artificial "G" environment. The internal pressure supply for the spoke and the internal pressure in the spoke are tolerance checked to assure the habitability for the crew in the spoke. The program then limit checks the power to the de-spin module to assure it is within limits. A de-spin module rotational command is issued to the de-spin module. This command is based upon access to operational data which provides the spin rate of the space station. Using this data, rates are computed which the module is expected to attain. The de-spin module rate is evaluated at one second intervals for a sample period and compared to the precalculated limits.

Status checks are then made on the cable deployment module. The S-II latch positions are status checked; and in the event of a positional discrepancy, the display reflecting the S-II latch positions is presented. The strut position is then status checked to assure that it is in the proper position for the current environment.

3.1.2.8 Termination Routine

This module is included to assure the orderly return of control to the executive at the completion of execution.

3.1.2.8.1 Source and Type of Inputs

The inputs to this module indicate a successful or unsuccessful completion of program execution. In the event termination is required due to operator intervention, or a detected failure, additional inputs as to the status of the associated components are required.

3.1.2.8.2 Destination and Type of Outputs

None

3.1.2.8.3 Information Processing

This module must accomplish the functions required to assure the subsystem is properly configured before control is returned to the executive.

Upon entry, this module determines if the program has been terminated as a result of operator intervention or successful completion of the test.

In the event of operator intervention, the bookkeeping required to assure that the subsystem is properly configured must be accomplished.

A message indicating completion of execution is presented, and control is returned to the executive.

3.1.3 PROGRAM FLOWCHARTS

The program flowcharts are provided in this section. The chart ID position on the diagrams is used to distinguish between the levels of detail in the flowcharts. A chart ID of A is used to depict the overall information flow of the program and is considered functional. A chart ID of B is intended to provide more detailed information.

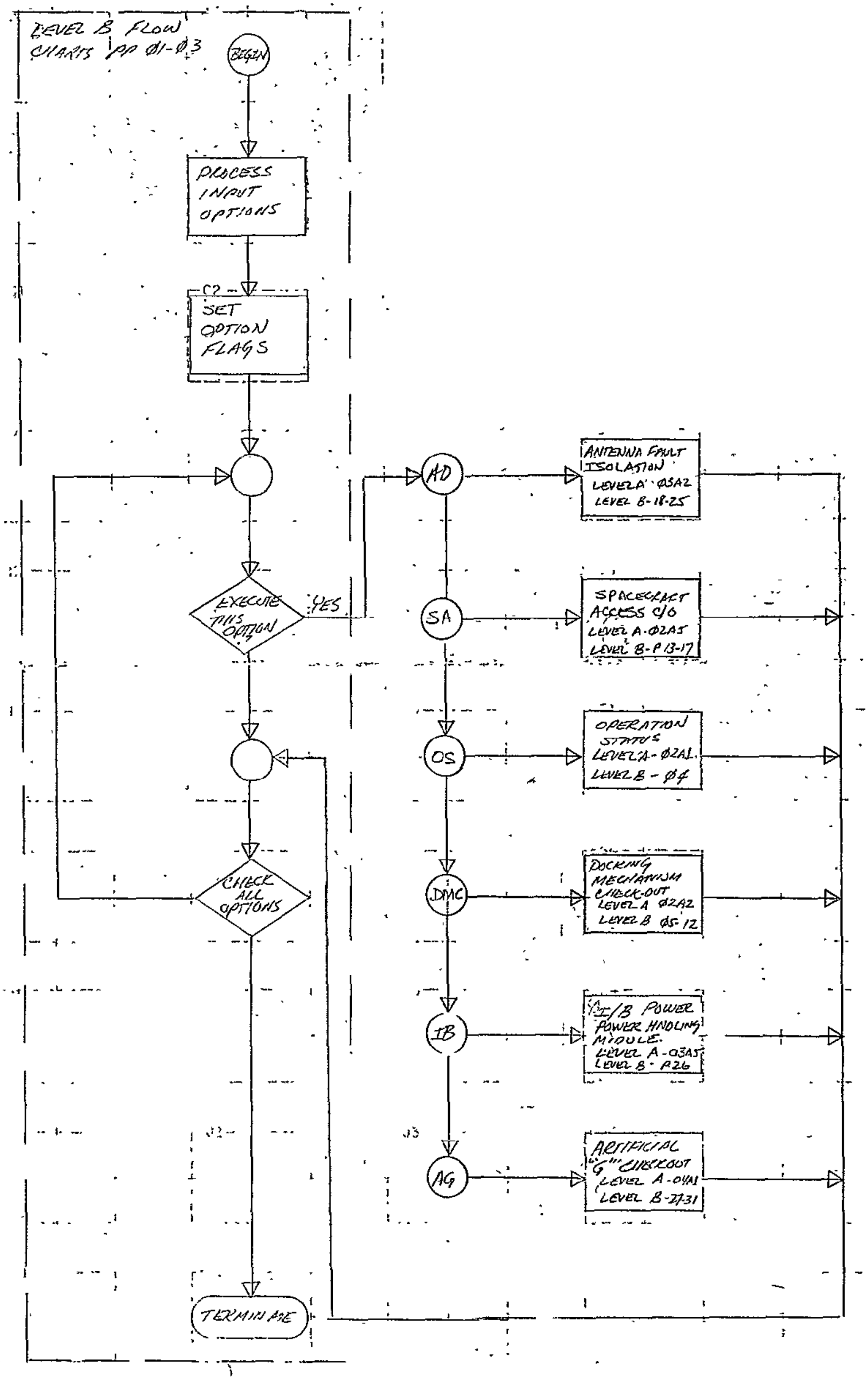


Figure 3-3. Functional Flowchart (Sheet 1 of 4)

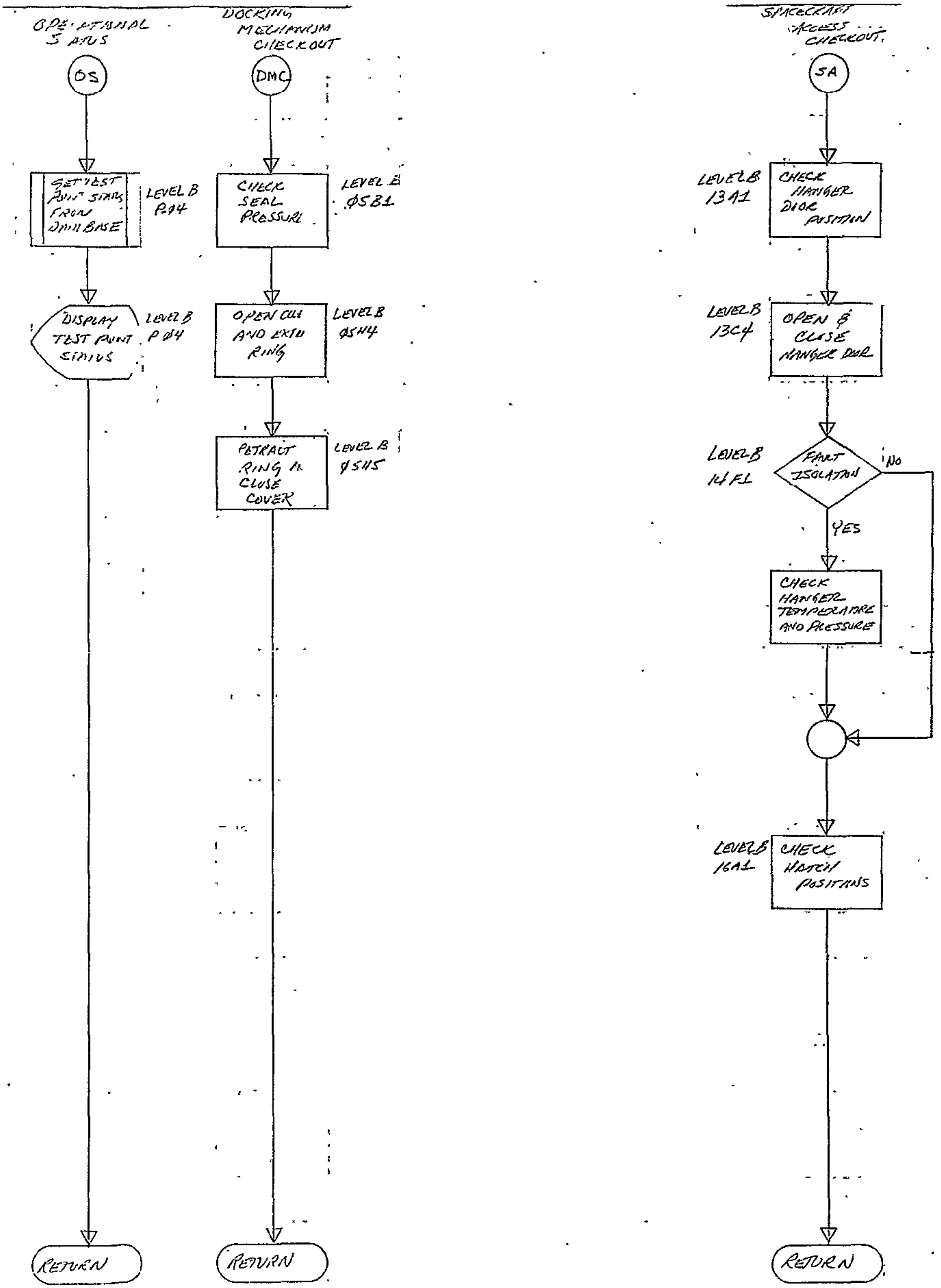


Figure 3-3. Functional Flowchart (Sheet 2 of 4)

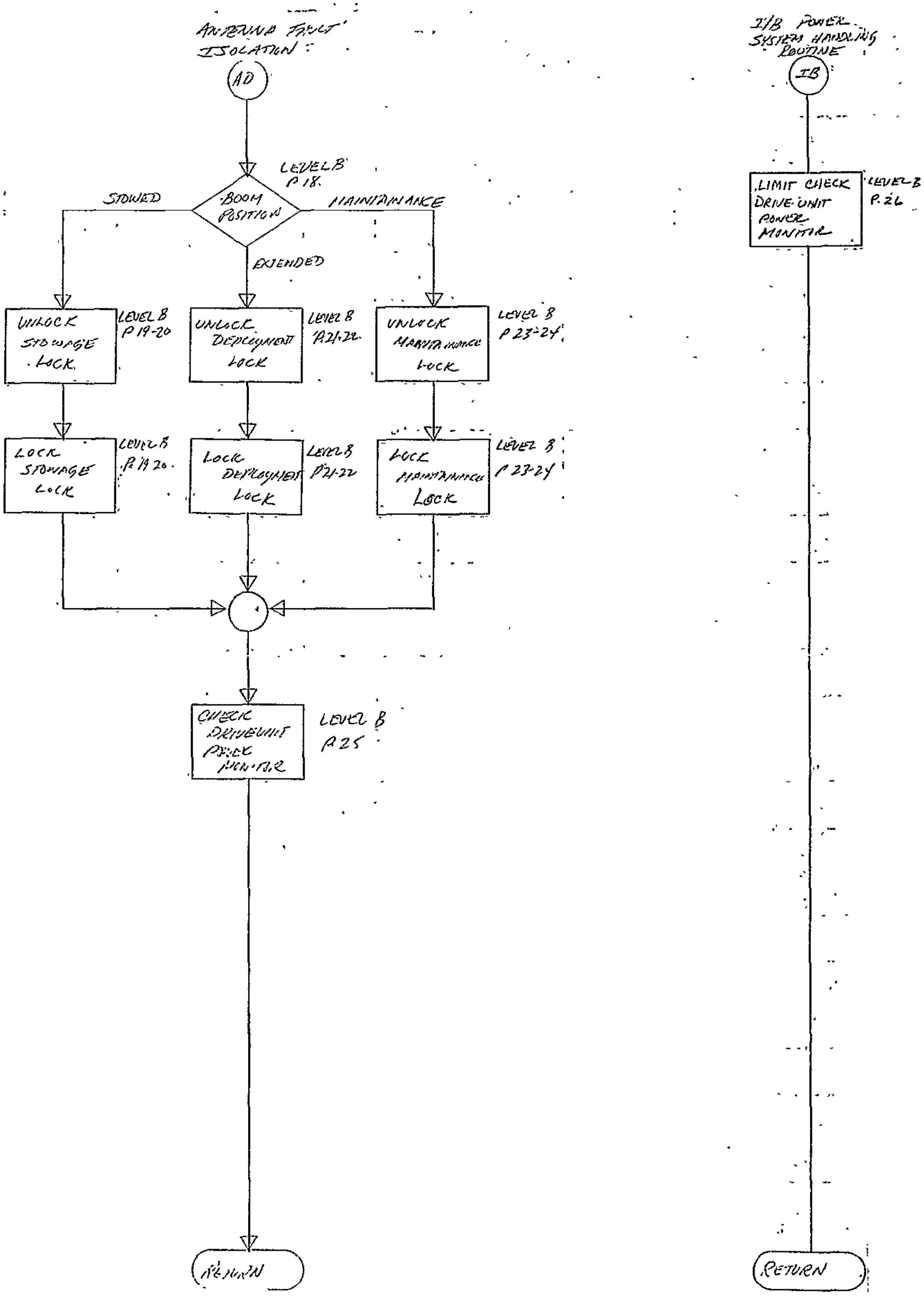


Figure 3-3. Functional Flowchart (Sheet 3 of 4)

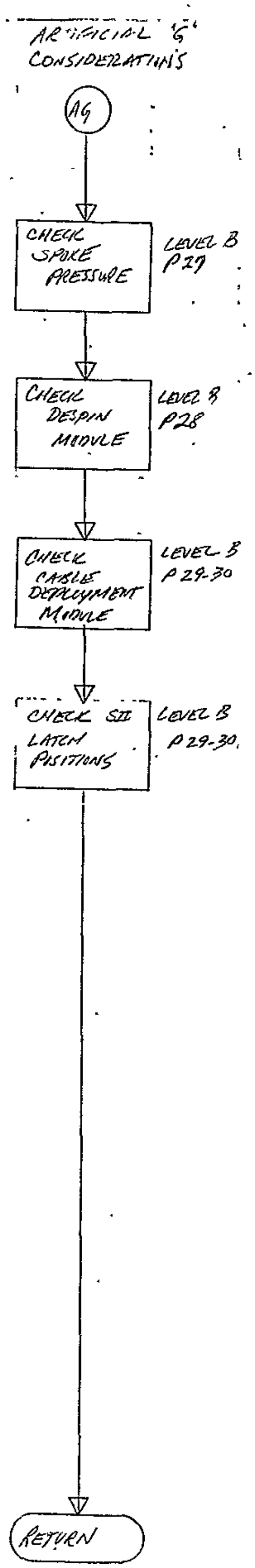


Figure 3-3. Functional Flowchart (Sheet 4 of 4)

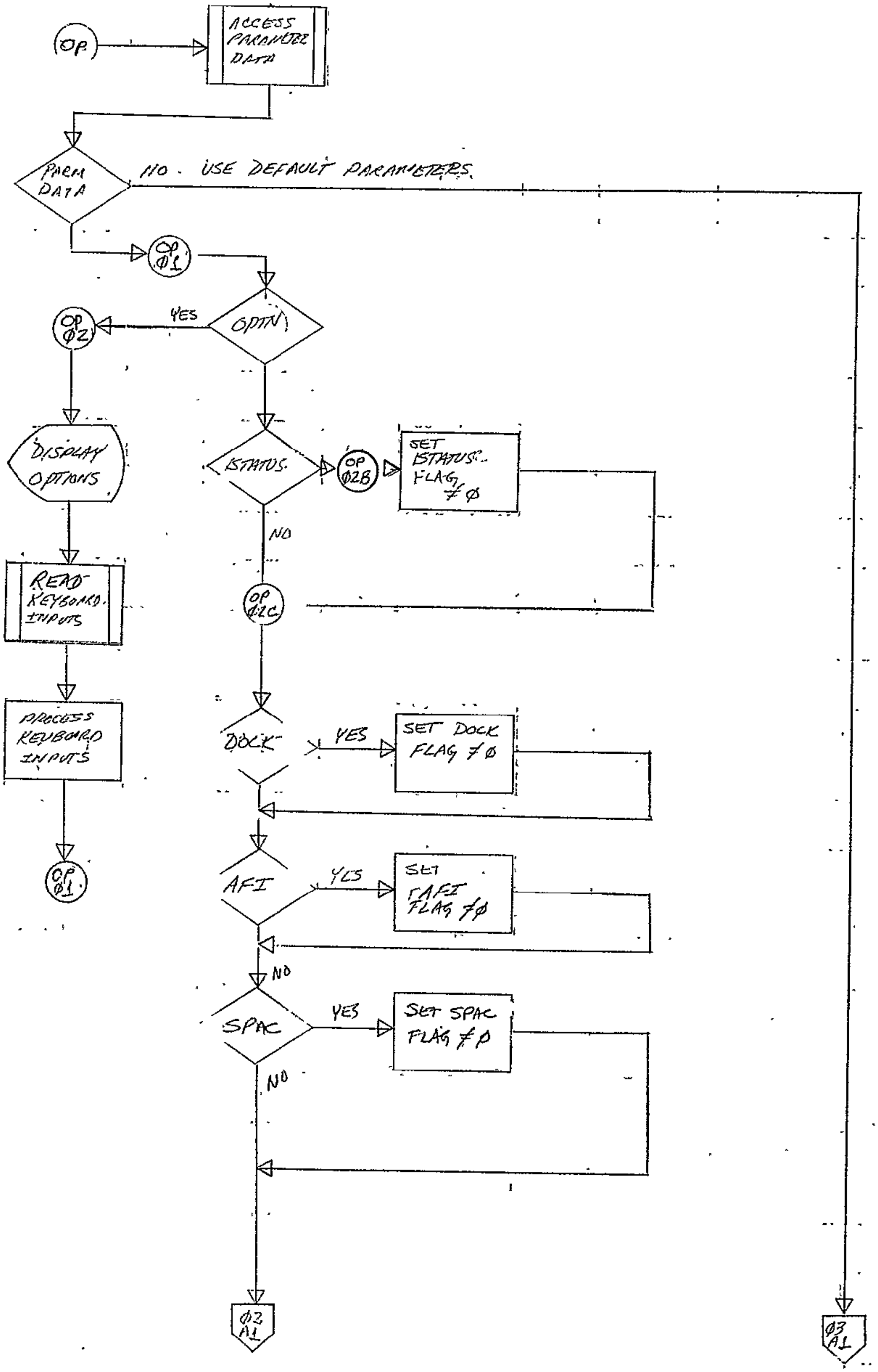


Figure 3-4. Option Selection (Sheet 1 of 4)

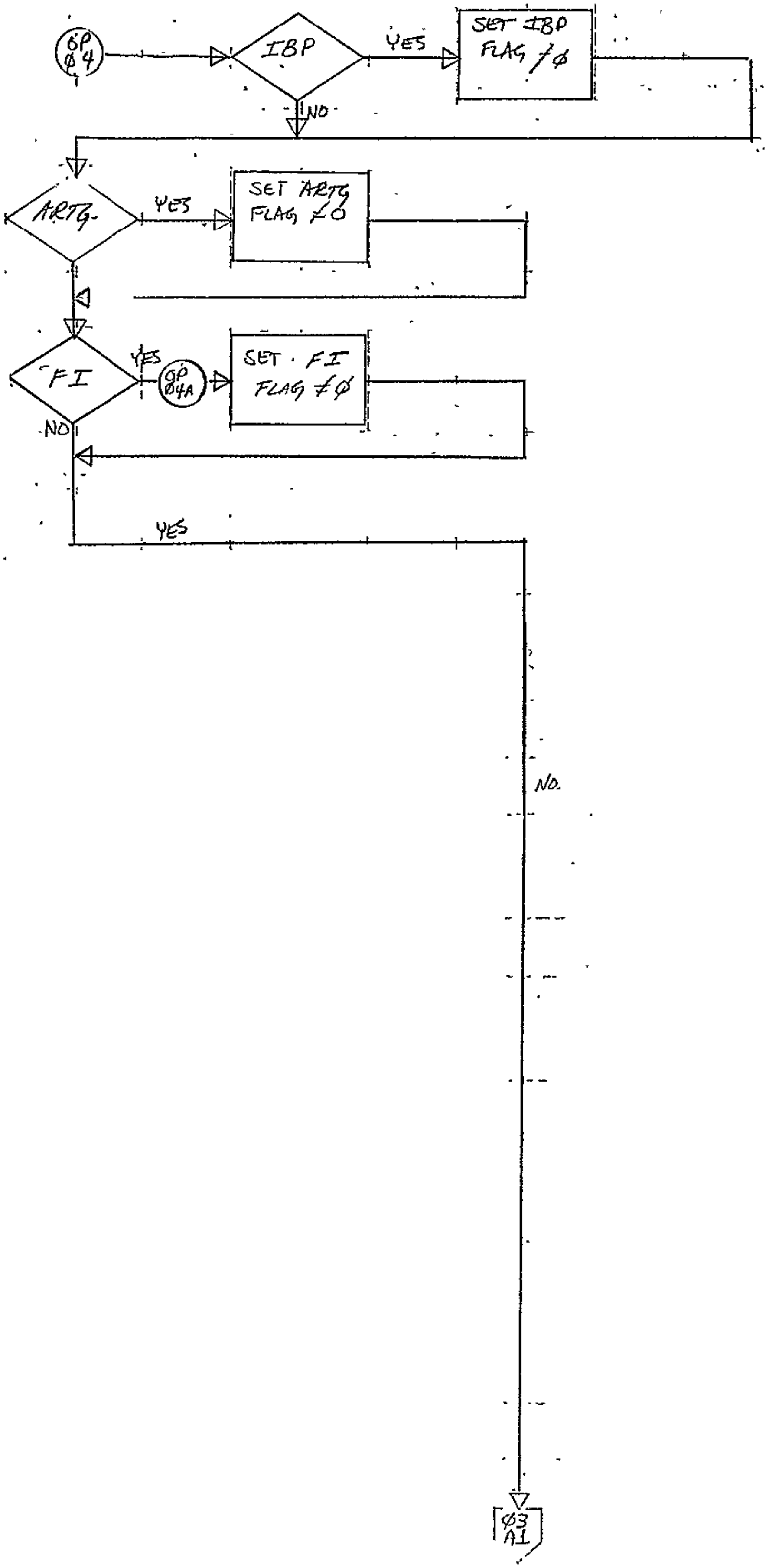


Figure 3-4. Option Selection (Sheet 2 of 4)

3

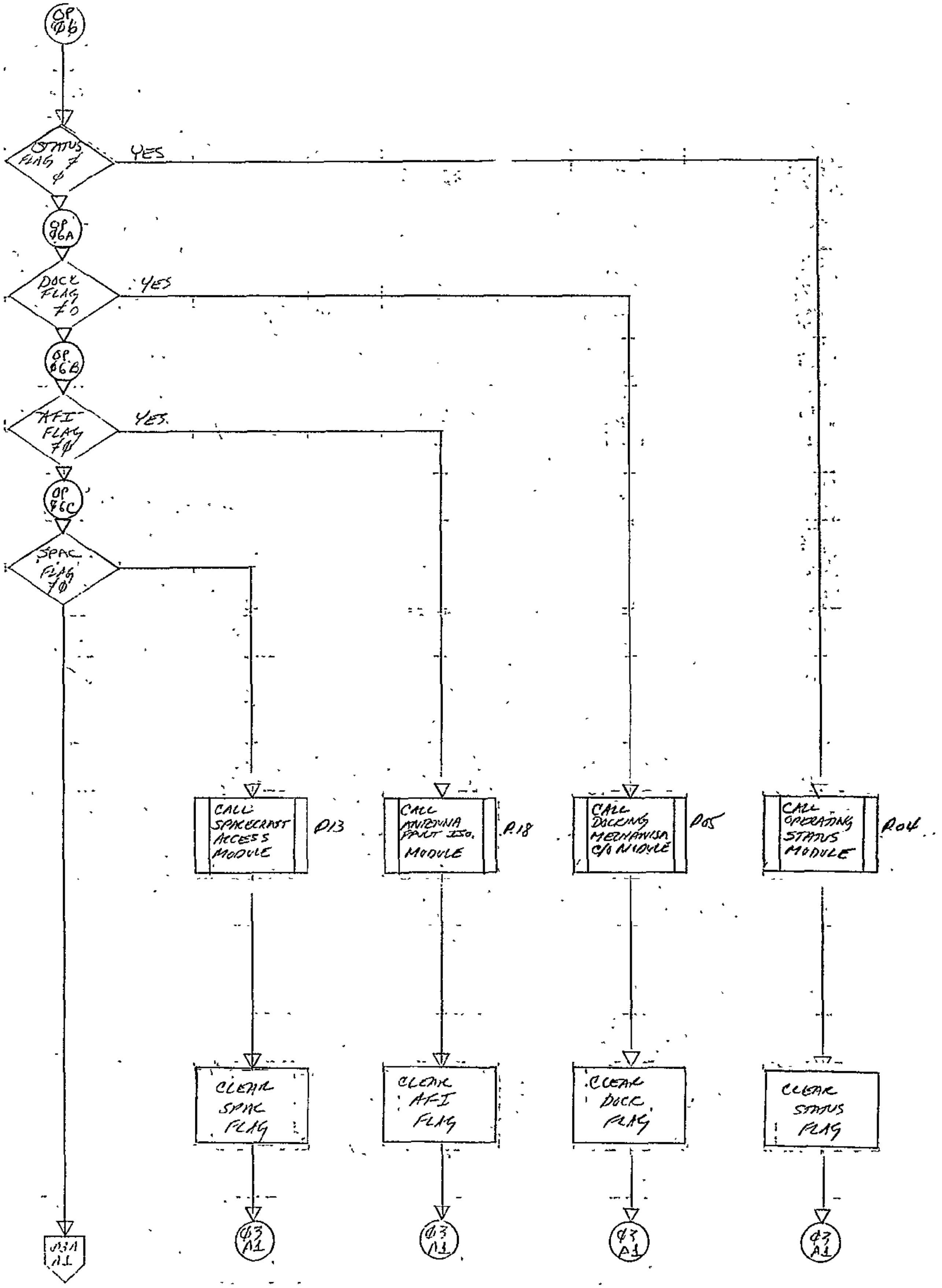


Figure 3-4. Option Selection (Sheet 3 of 4)

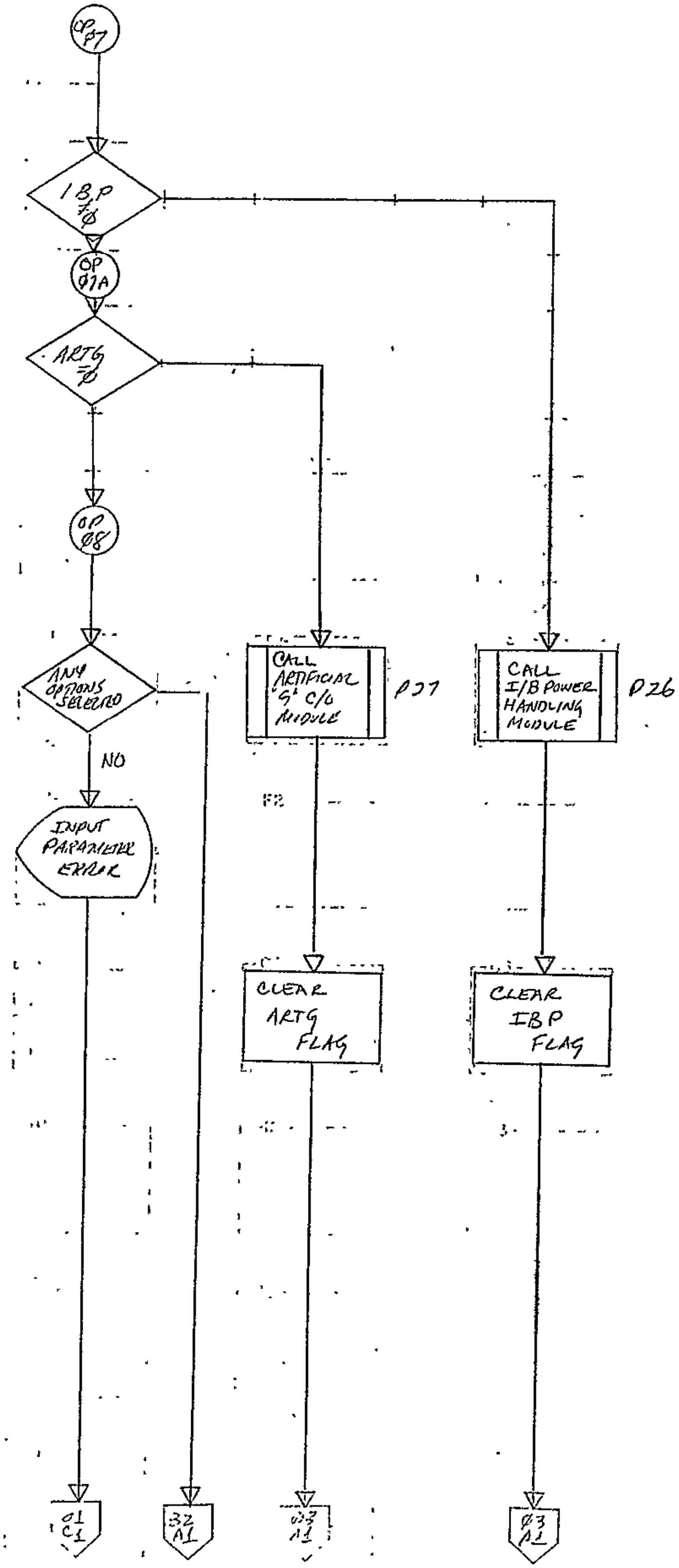


Figure 3-4. Option Selection (Sheet 4 of 4)

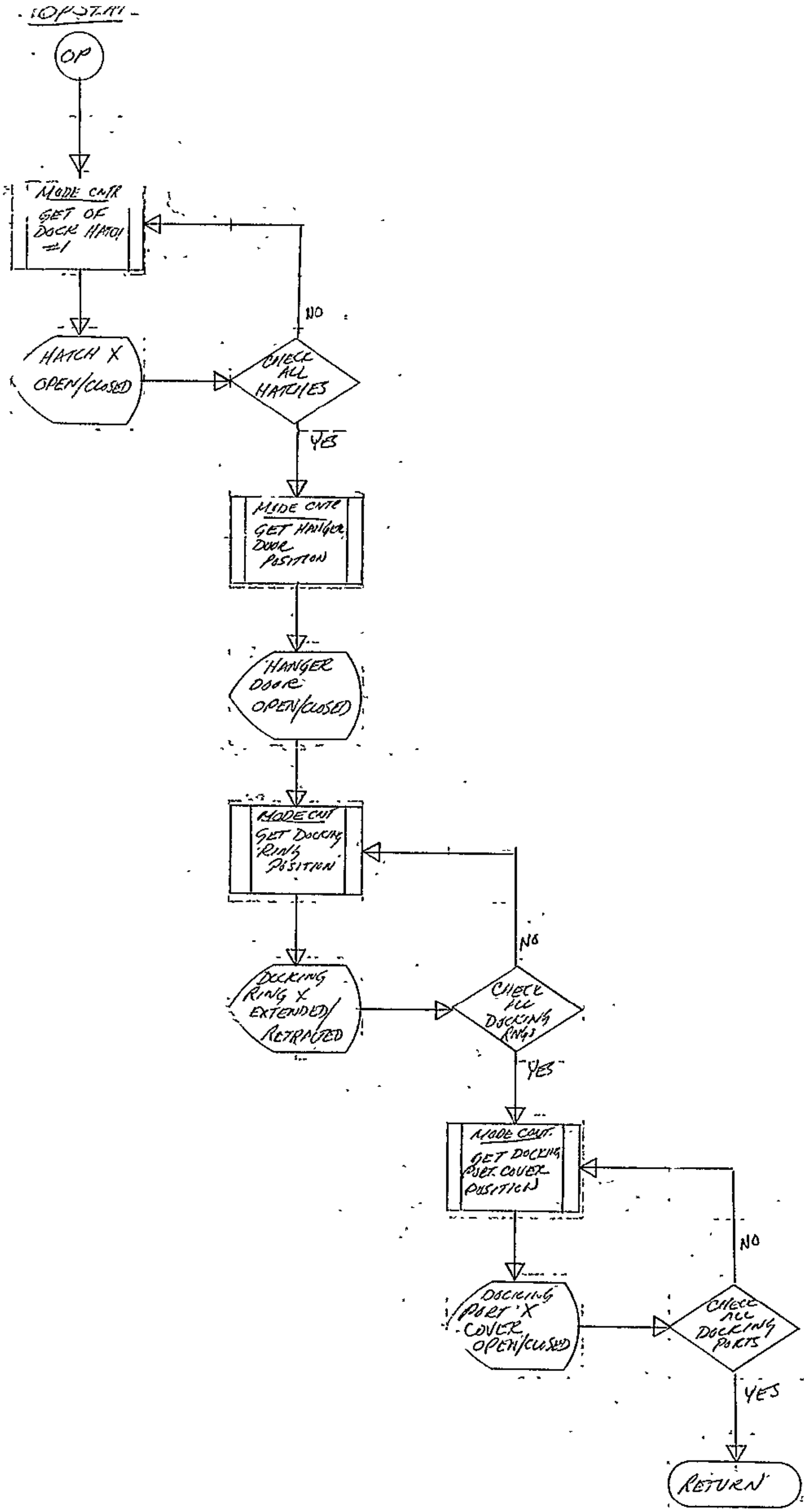


Figure 3-5. Operational Status

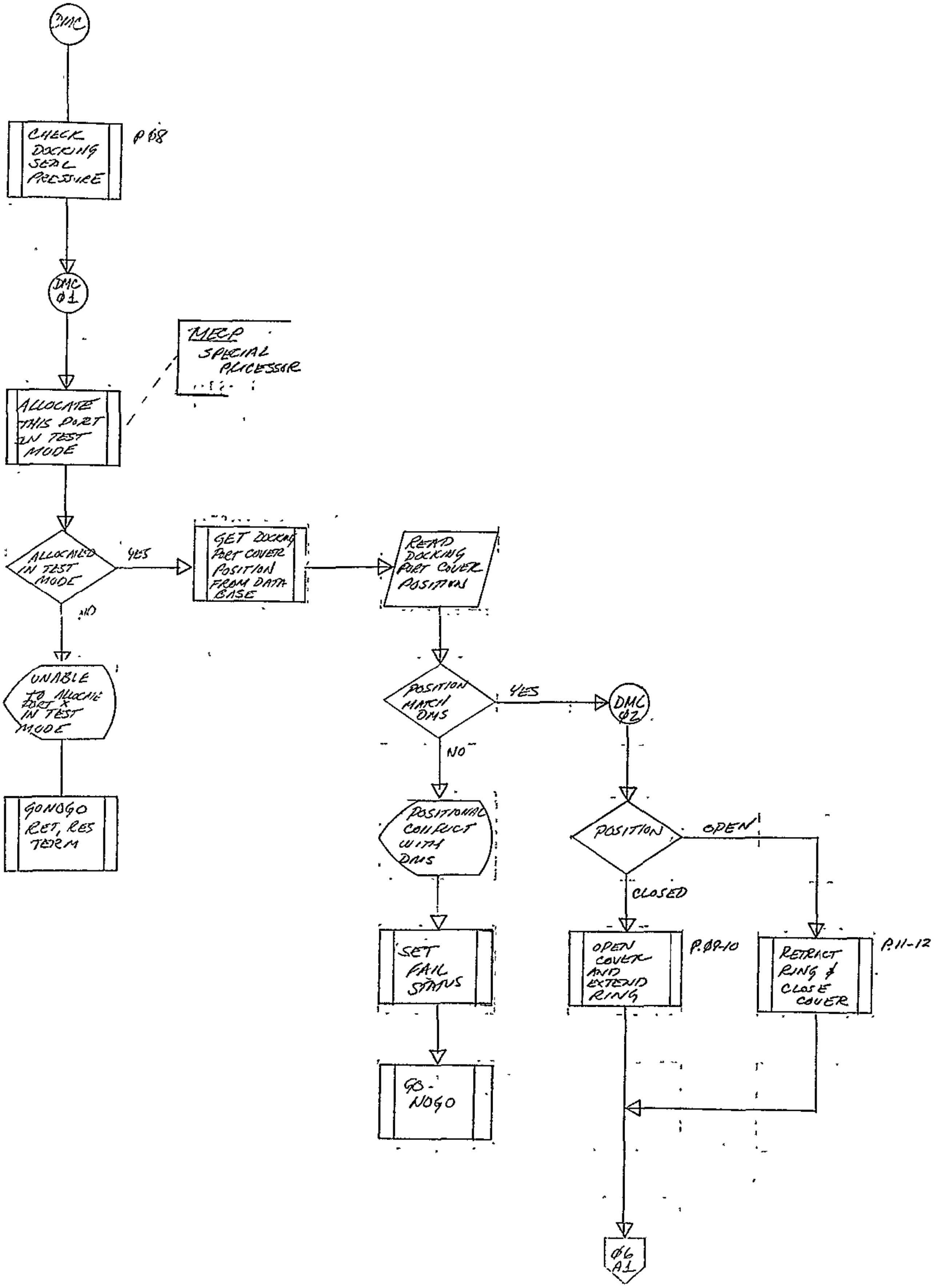


Figure 3-6. Docking Mechanism Checkout (Sheet 1 of 7)

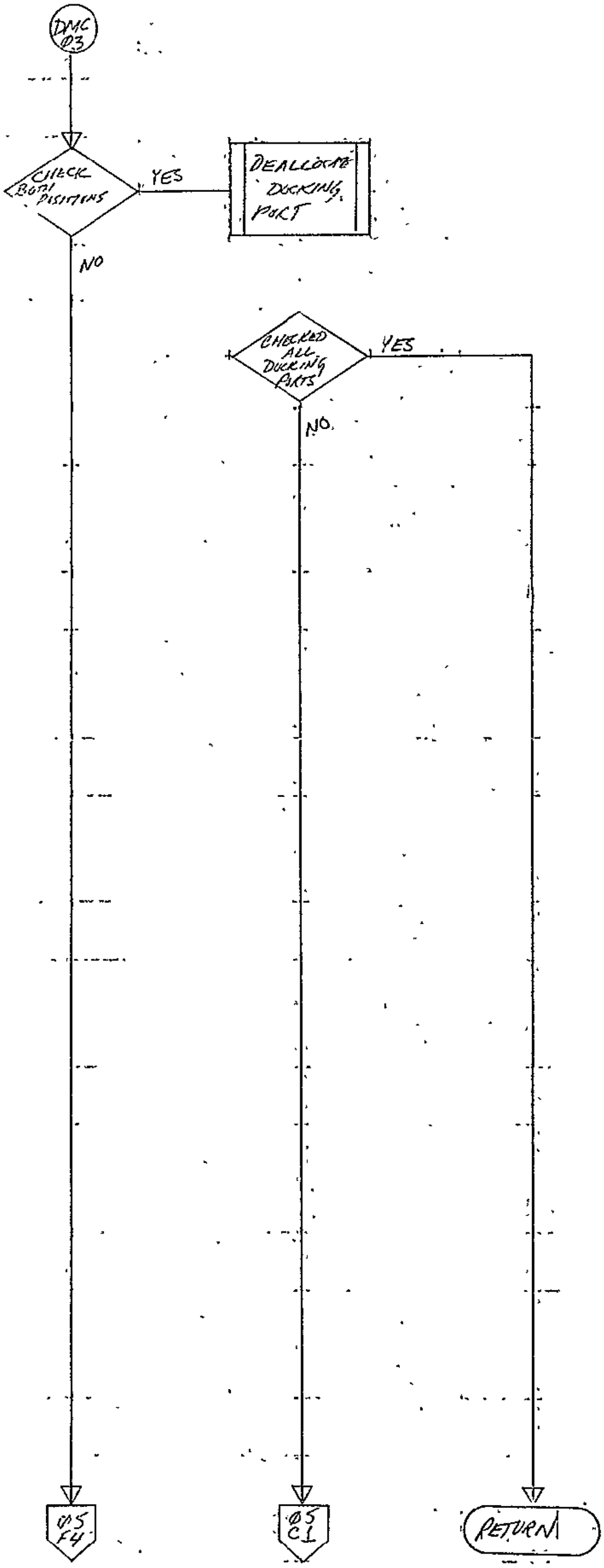


Figure 3-6. Docking Mechanism Checkout (Sheet 2 of 7)

SUBROUTINE: CHECK DOCKING SEAL PRESSURE.

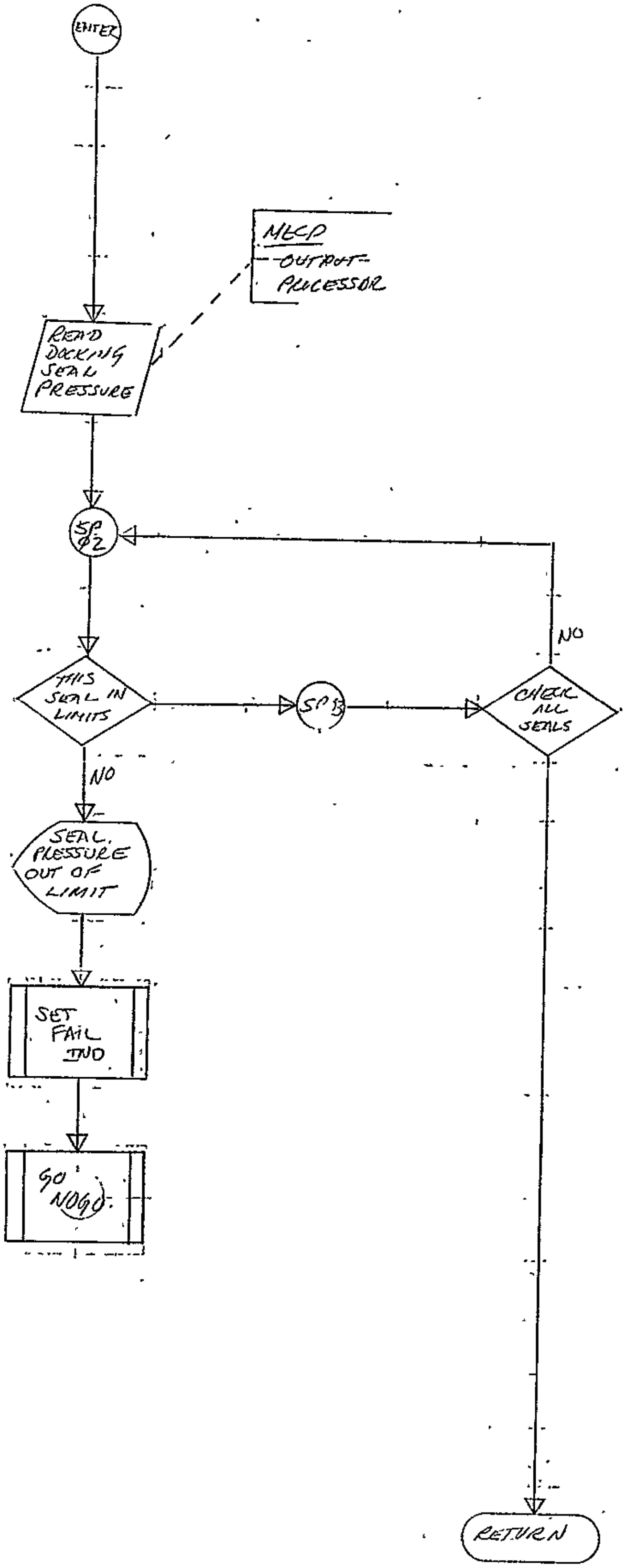


Figure 3-6. Docking Mechanism Checkout (Sheet 3 of 7)

SUBROUTINE:
OPEN DOCKING PORT COVER AND EXTEND RING

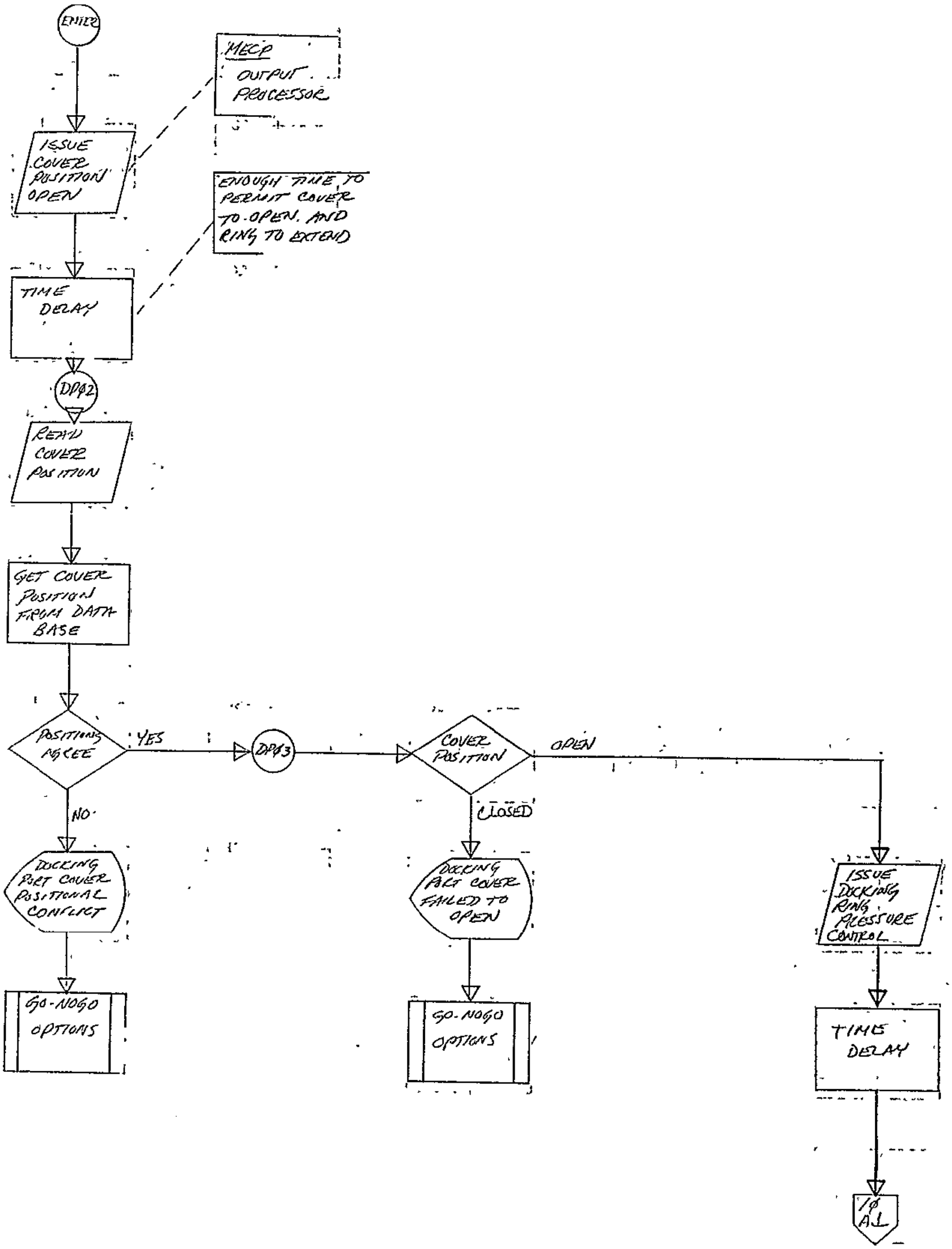


Figure 3-6. Docking Mechanism Checkout (Sheet 4 of 7)

SUBROUTINE

OPEN DOCKING PORT COVER & EXTEND RING (con't)

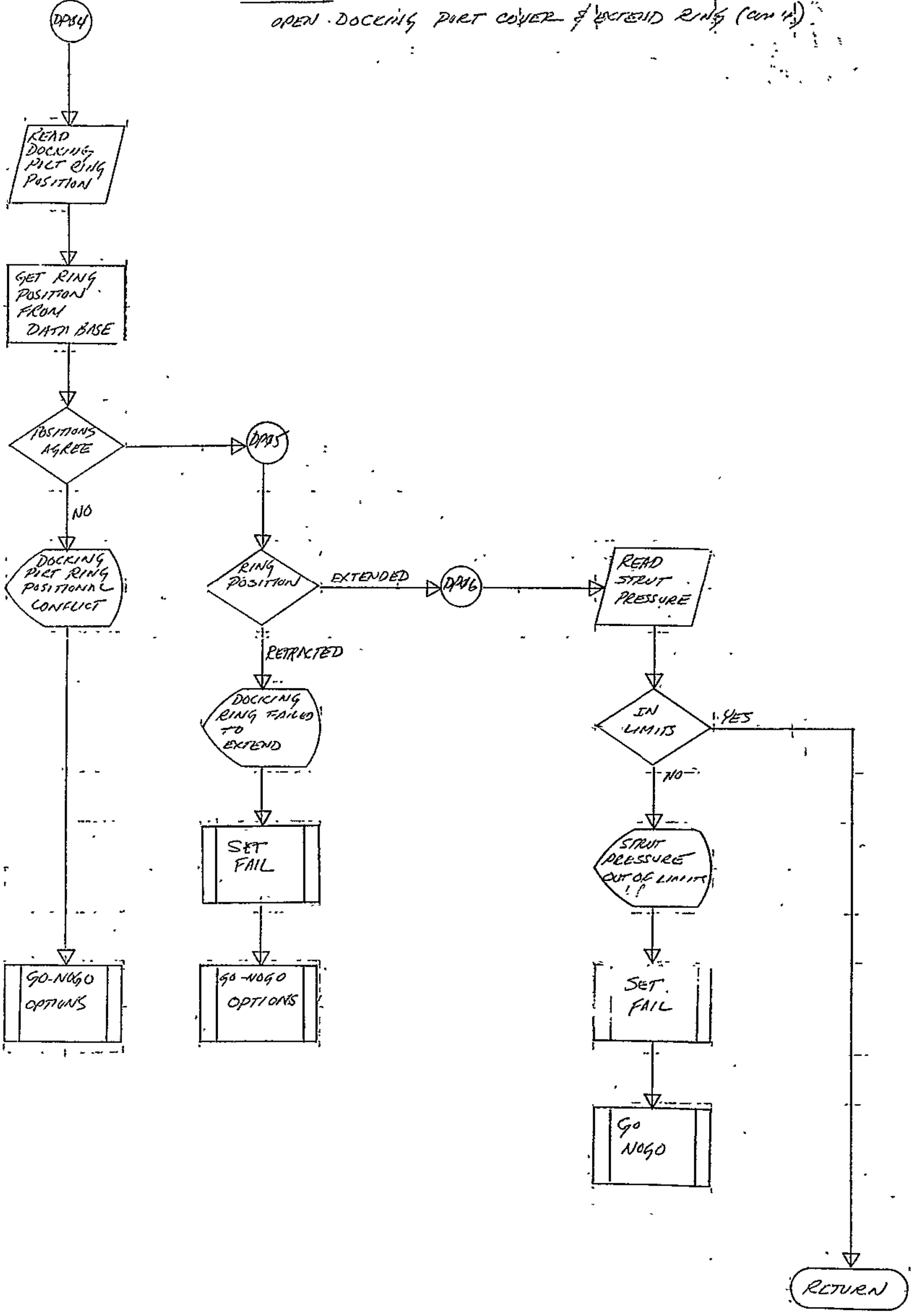


Figure 3-6. Docking Mechanism Checkout (Sheet 5 of 7)

SUBROUTINE:
RETRACT DOCKING RING AND CLOSE DOCKING PORT COVER

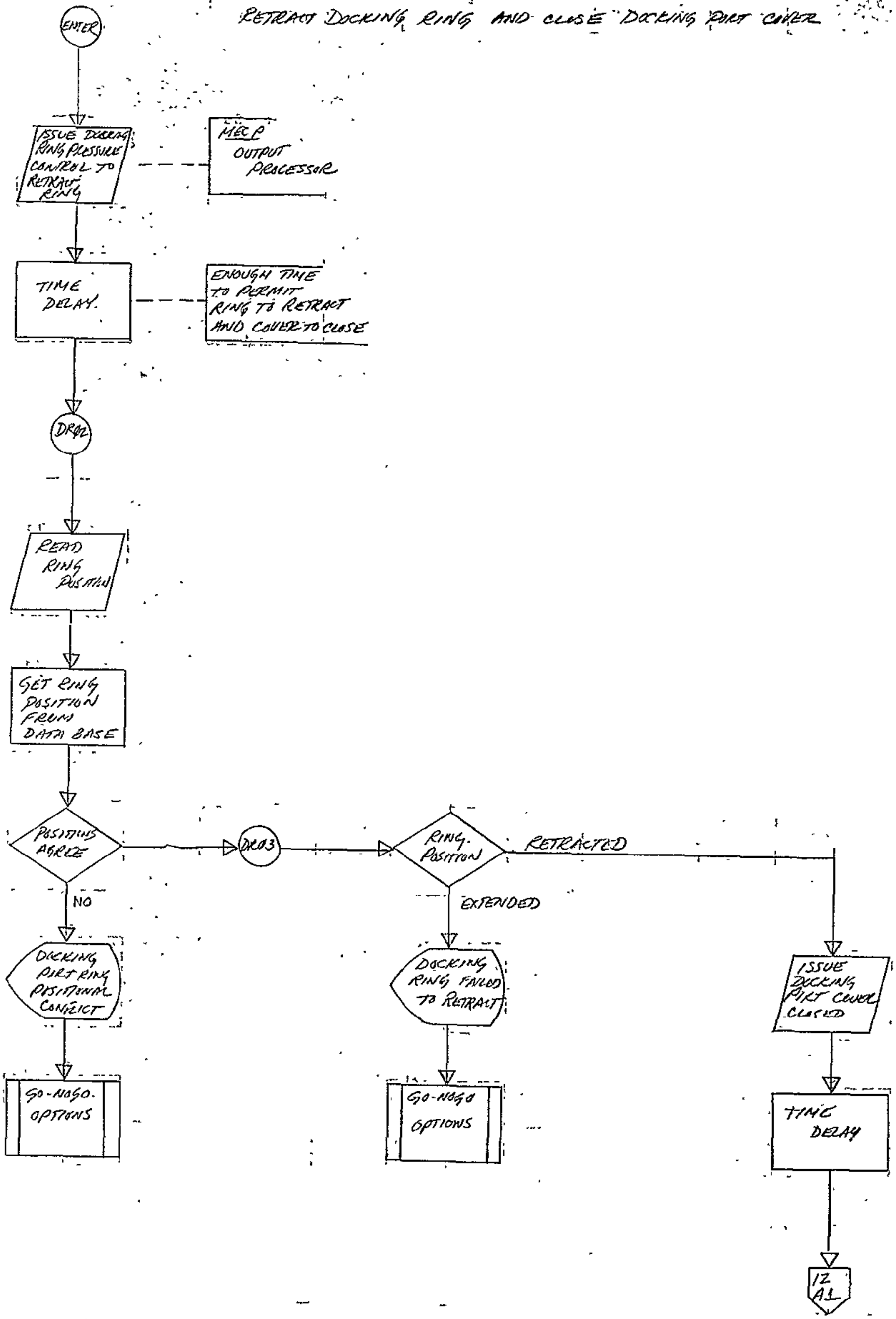


Figure 3-6. Docking Mechanism Checkout (Sheet 6 of 7)

SUBROUTINE:
RETRACT DOCKING RING AND CLOSE DOCKING PORT COVER

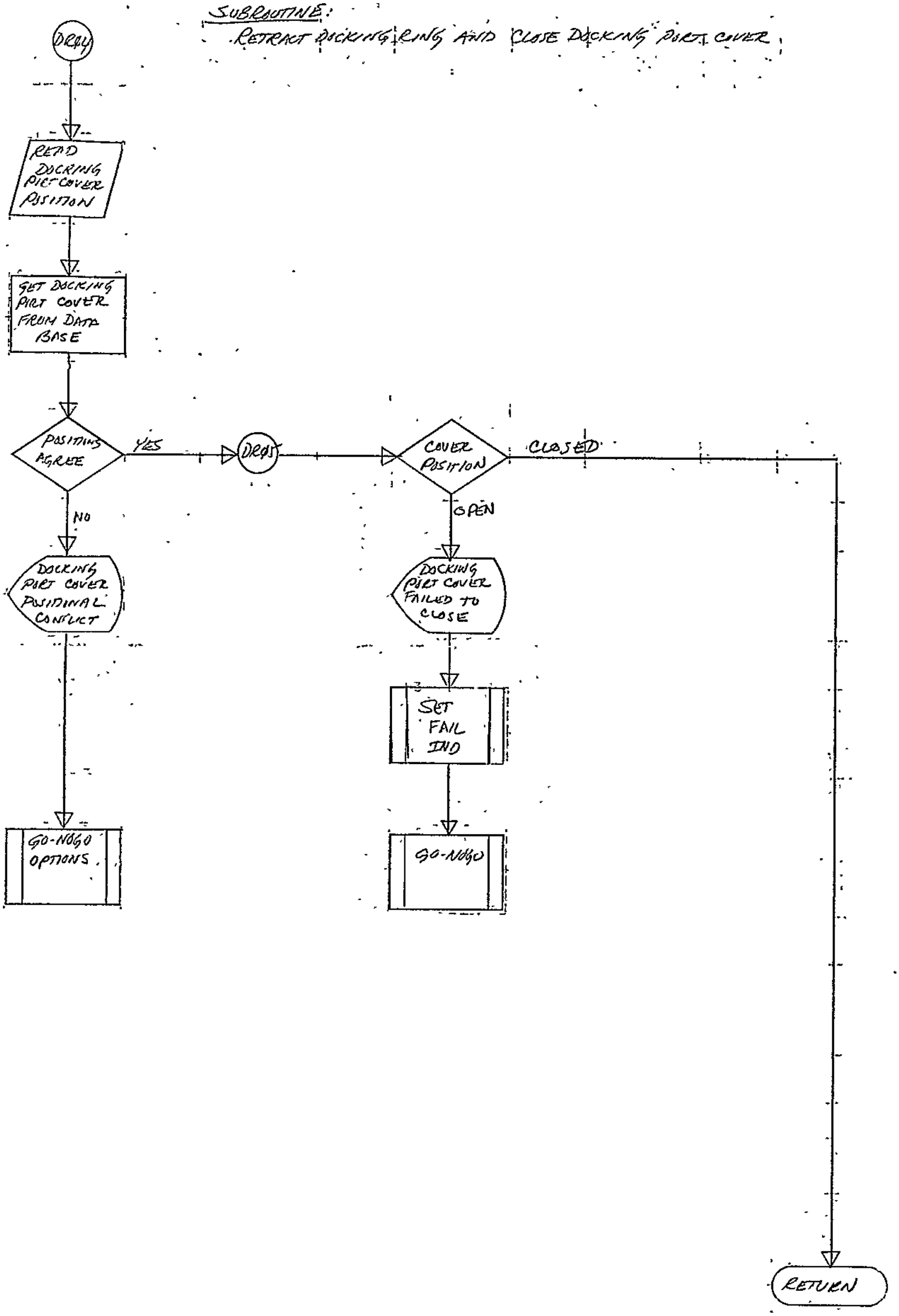


Figure 3-6. Docking Mechanism Checkout (Sheet 7 of 7)

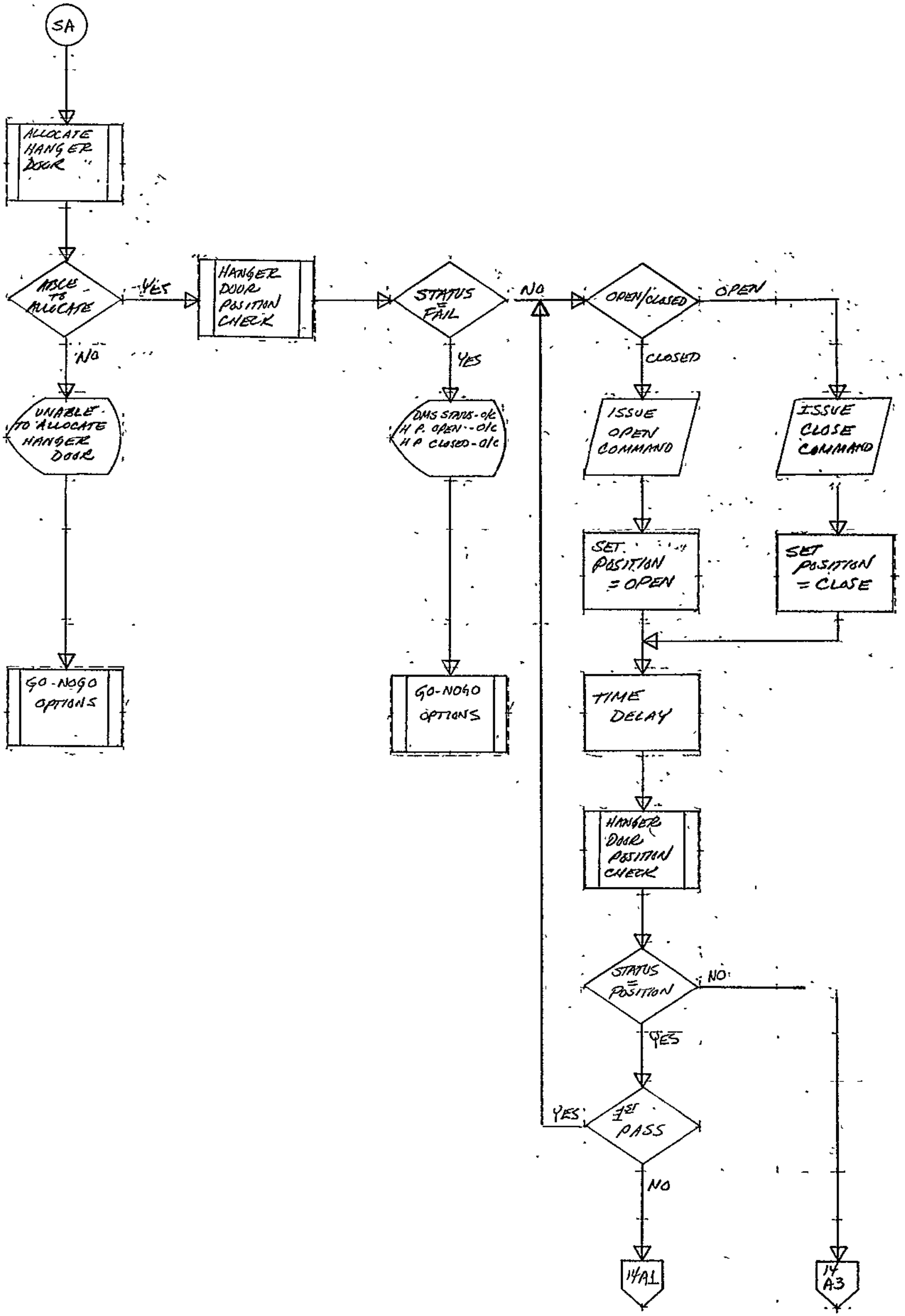


Figure 3-7. Spacecraft Access (Sheet 1 of 5)

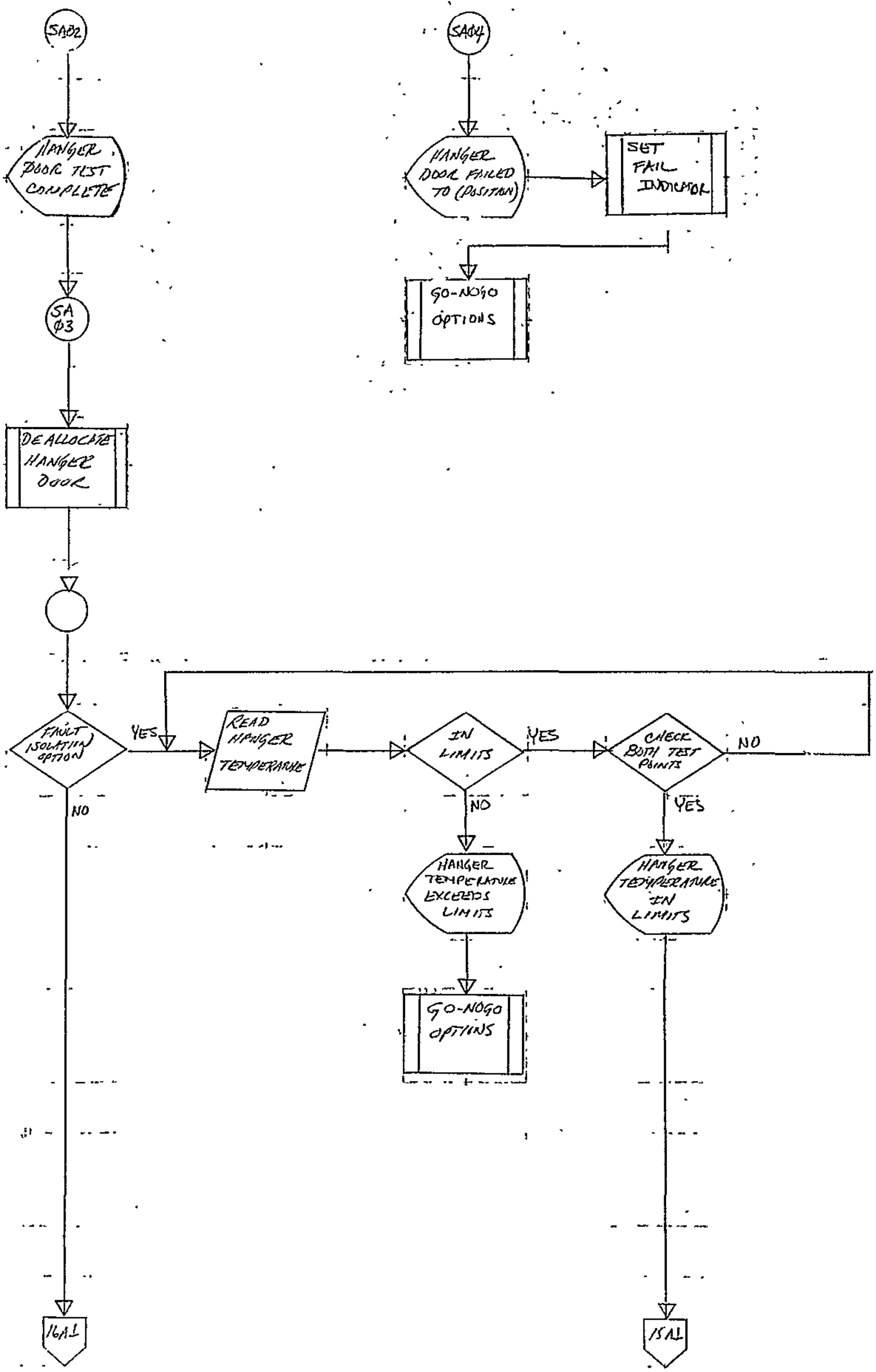


Figure 3-7. Spacecraft Access (Sheet 2 of 5)

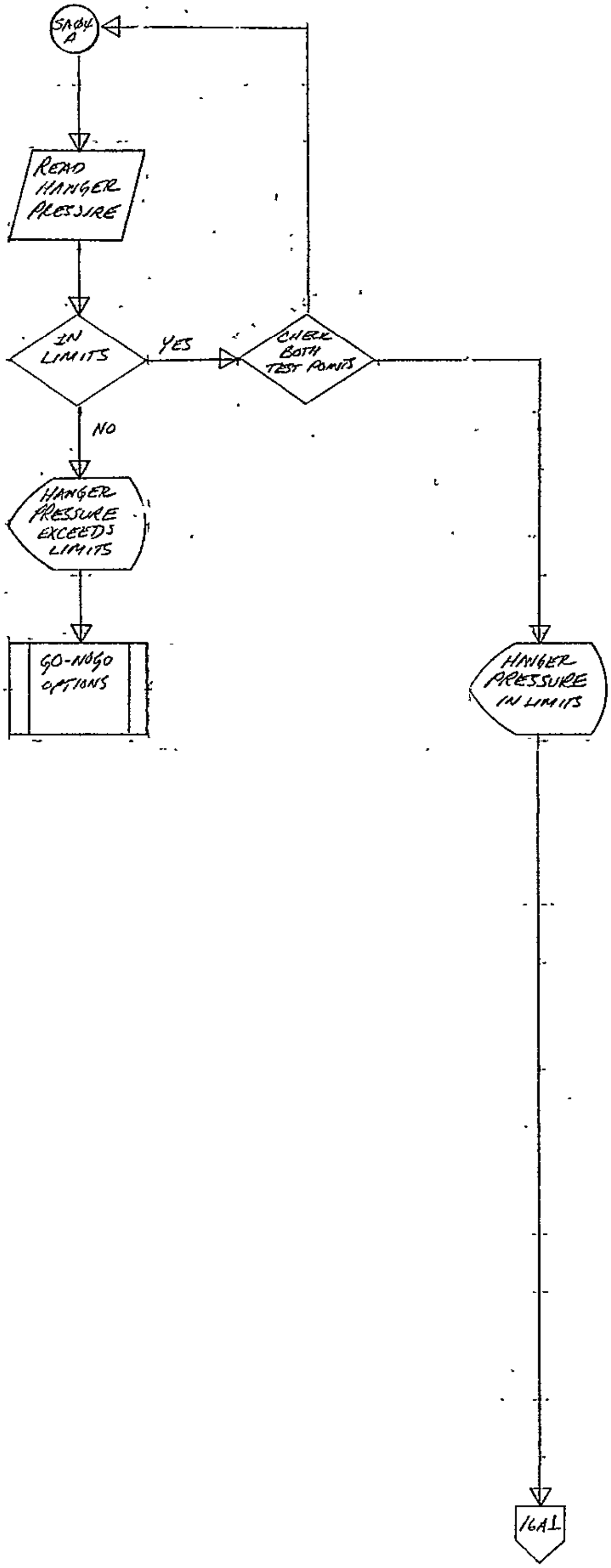


Figure 3-7. Spacecraft Access (Sheet 3 of 5)

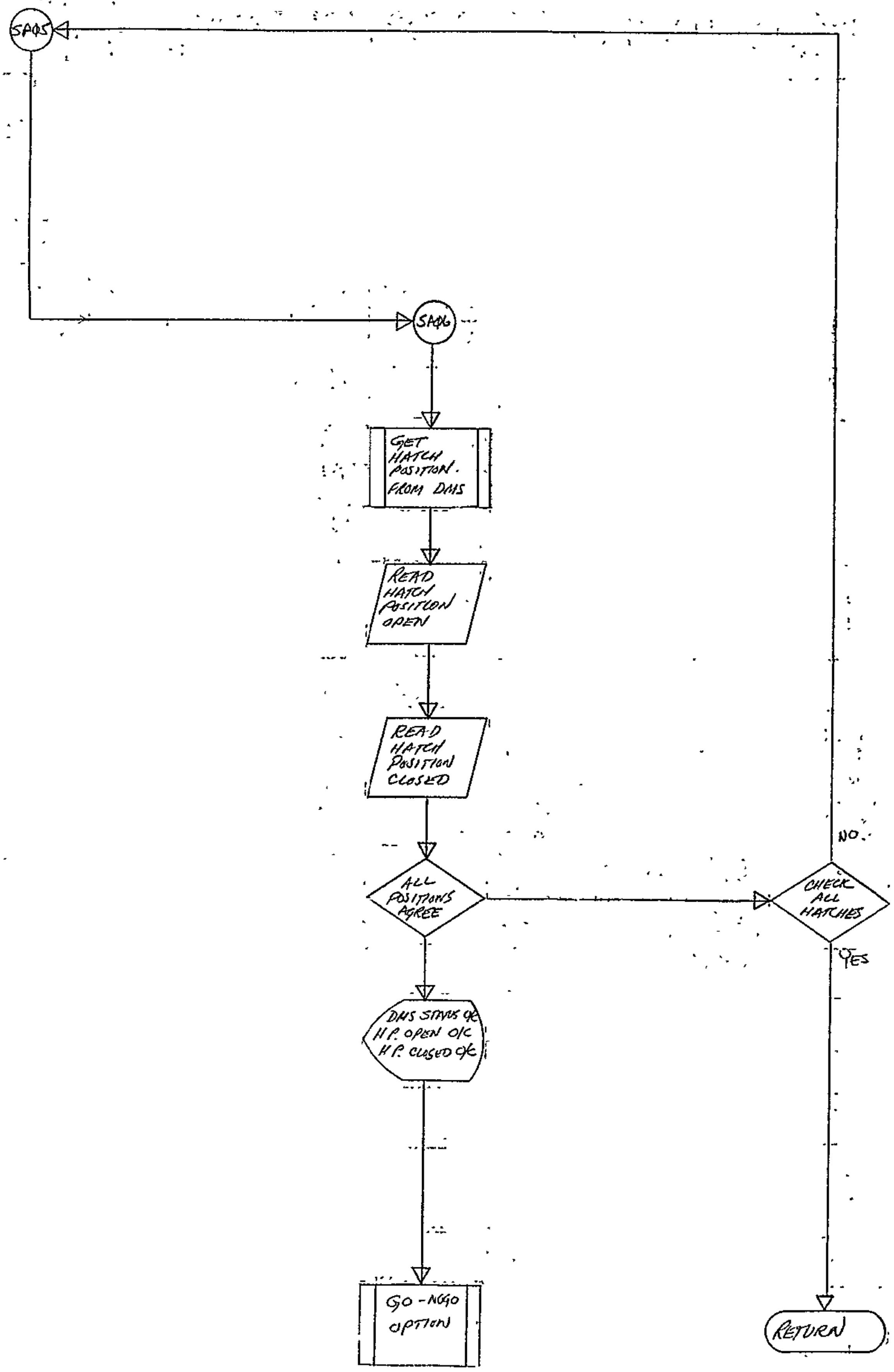
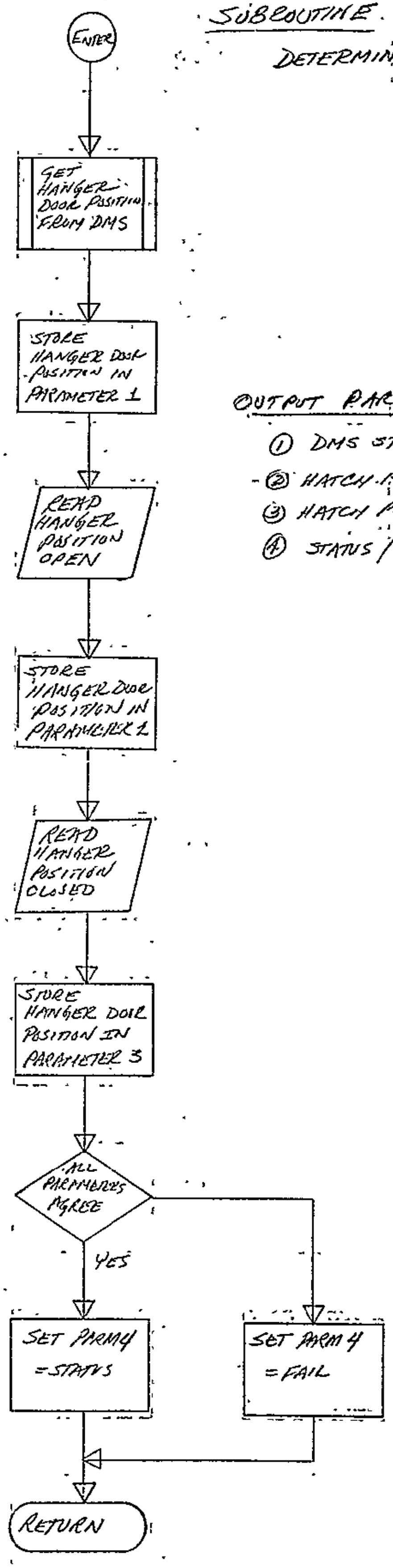


Figure 3-7. Spacecraft Access (Sheet 4 of 5)

SUBROUTINE
DETERMINE HANGER DOOR POSITION



OUTPUT PARAMETERS

① DMS STATUS
② HATCH POSITION OPEN STATUS
③ HATCH POSITION CLOSED STATUS
④ STATUS / FAIL

Figure 3-7. Spacecraft Access (Sheet 5 of 5)

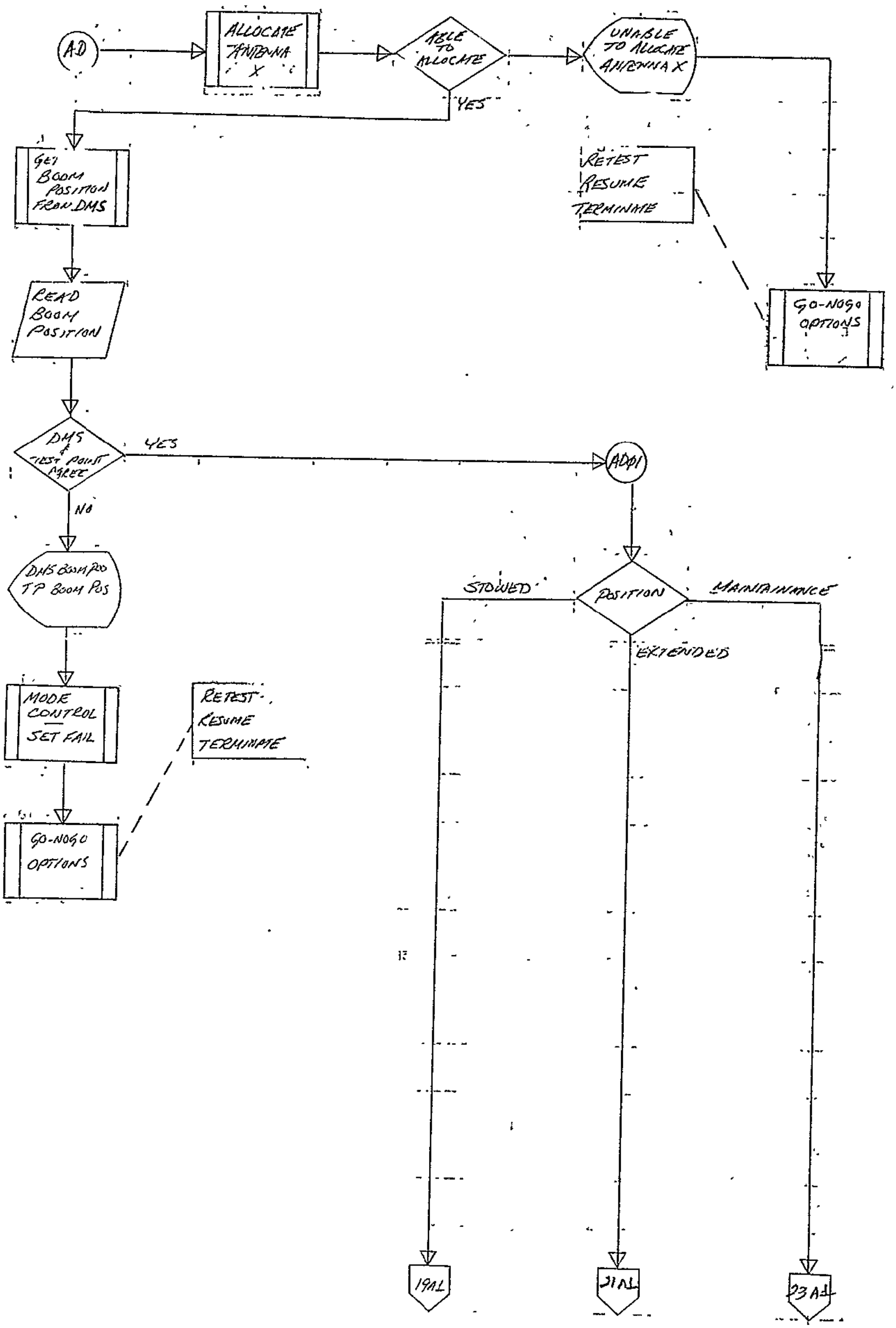


Figure 3-8. Antenna Fault Isolation (Sheet 1 of 3)

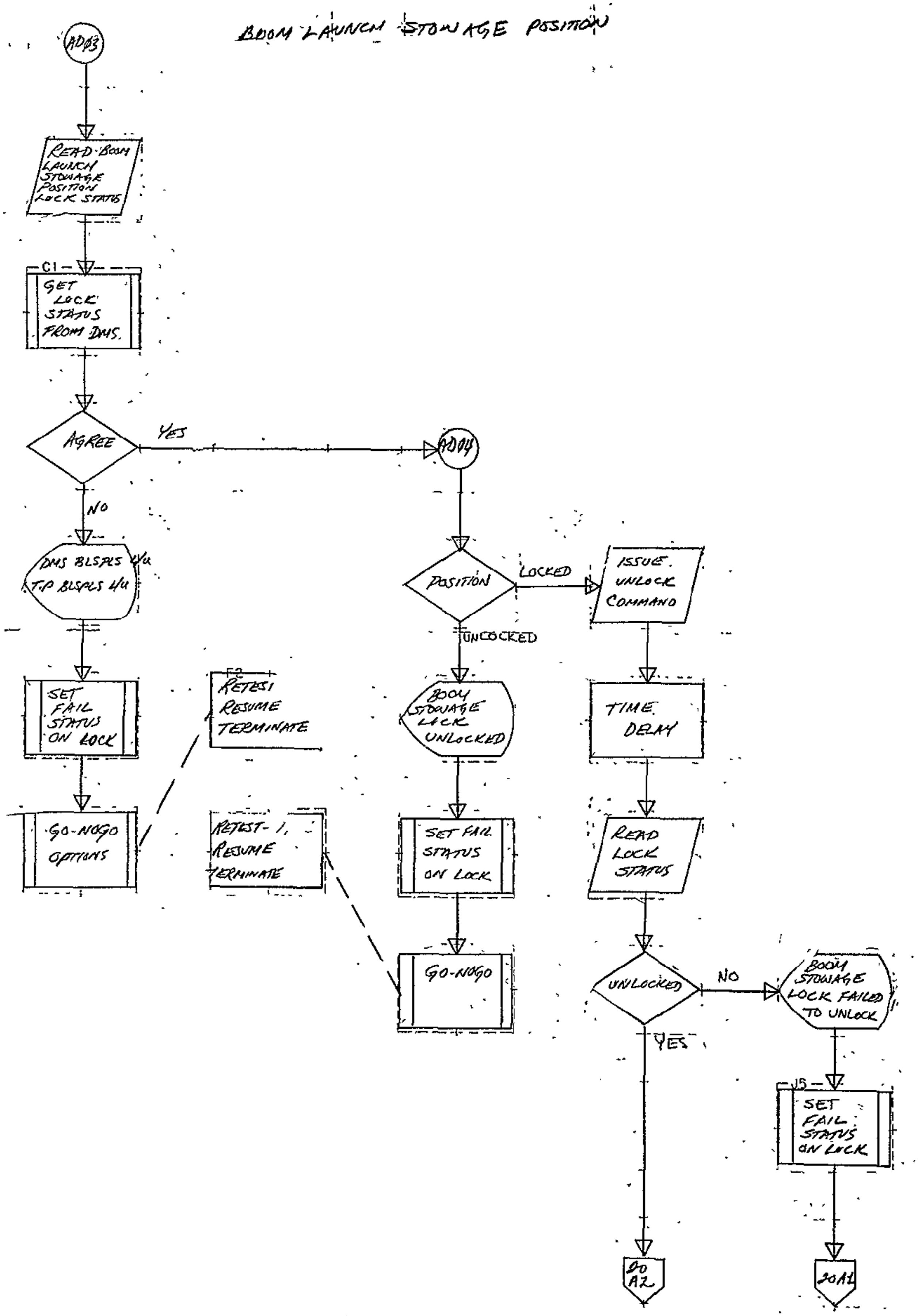


Figure 3-8. Antenna Fault Isolation (Sheet 2 of 3)

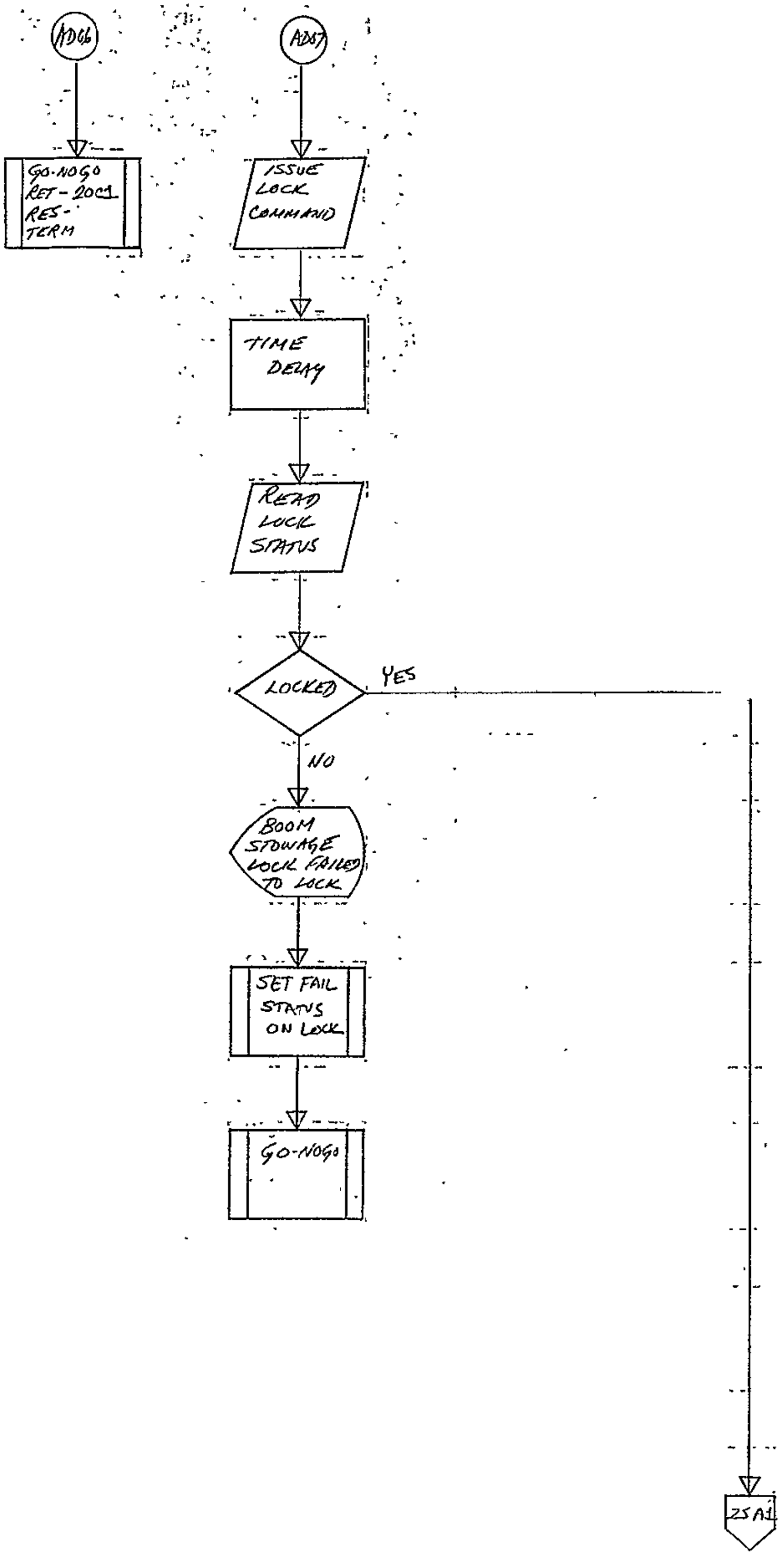


Figure 3-8. Antenna Fault Isolation (Sheet 3 of 3)

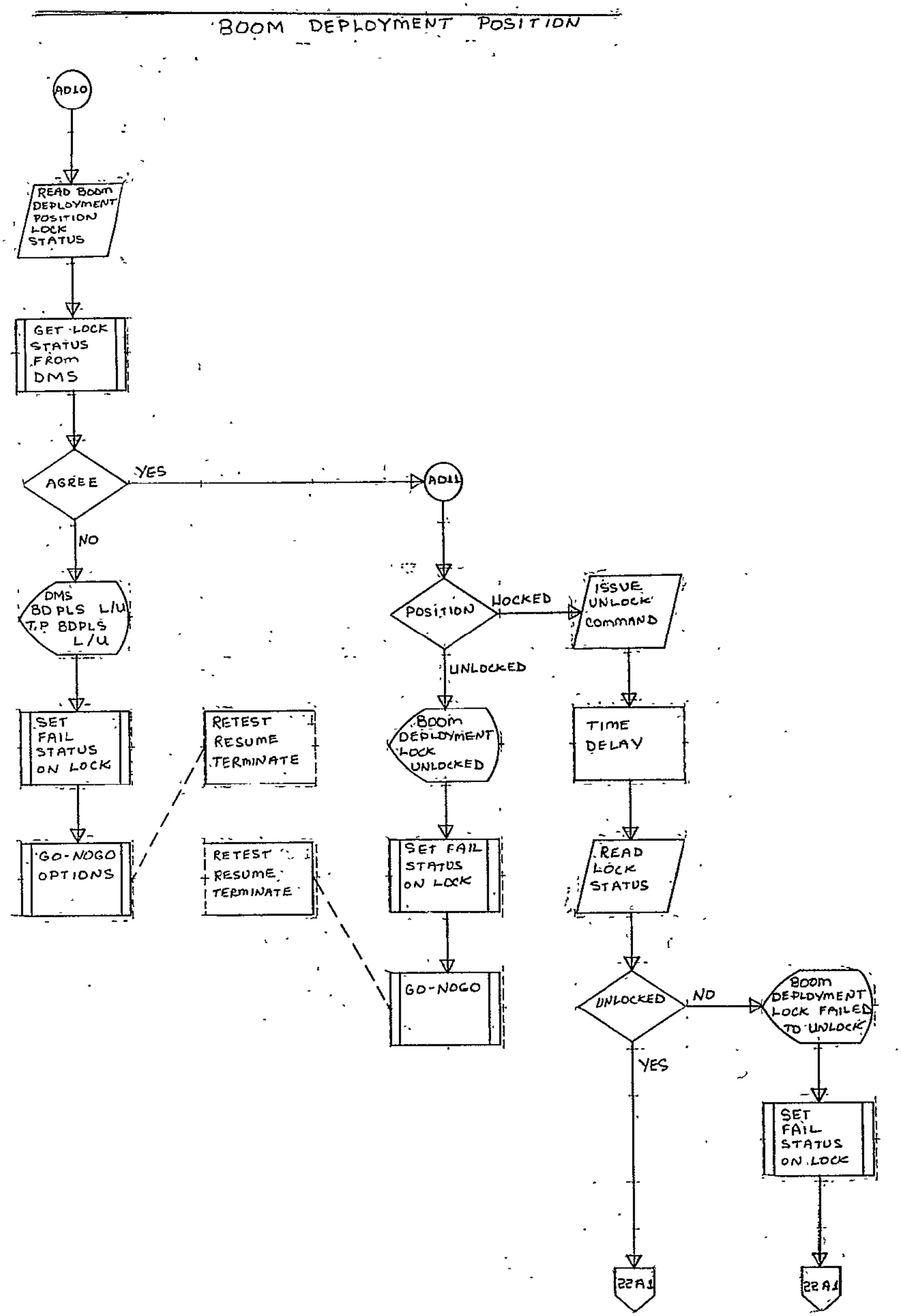


Figure 3-9. Antenna Fault Isolation (Sheet 1 of 4)

BOOM DEPLOYMENT POSITION (CONT'D)

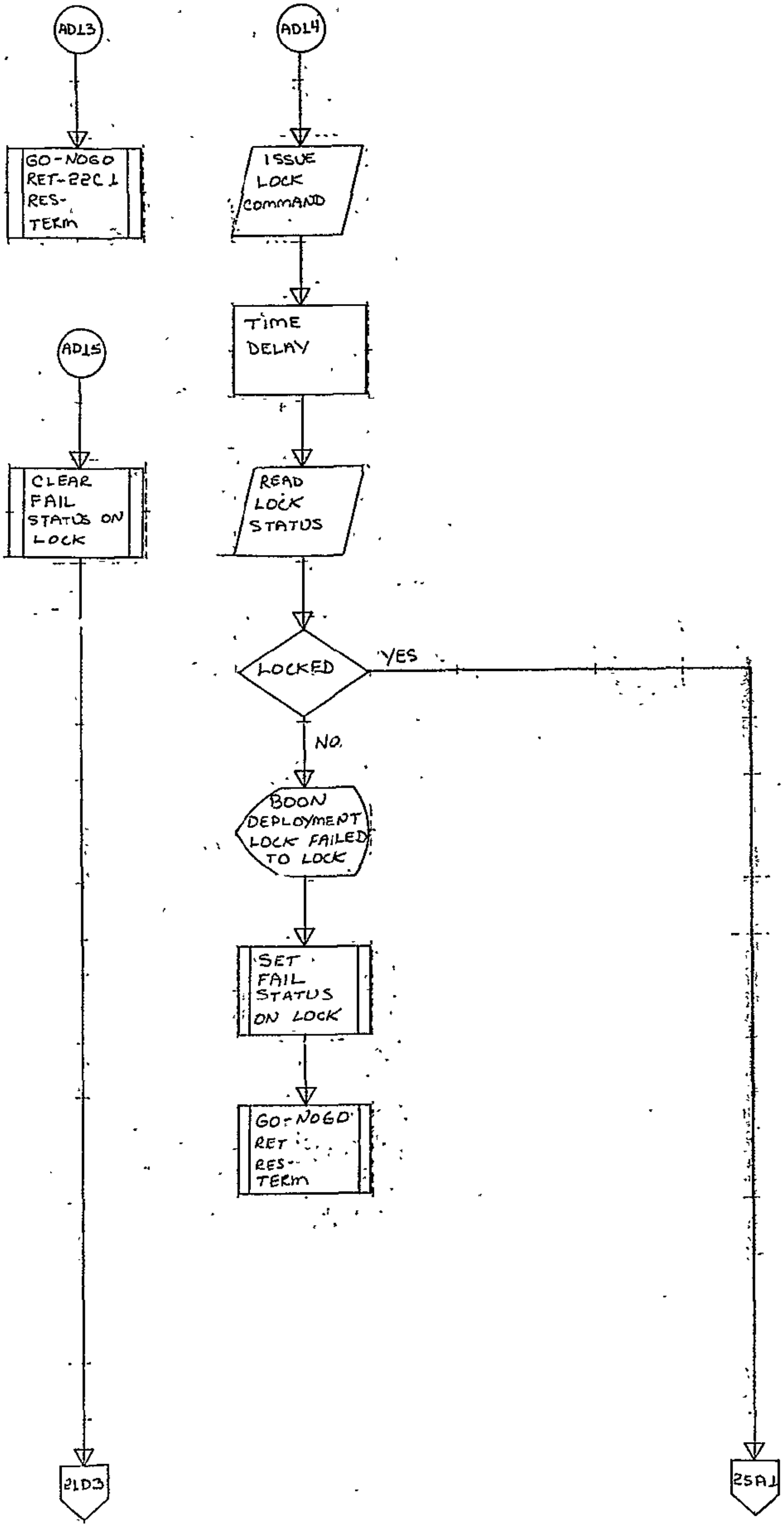


Figure 3-9. Antenna Fault Isolation (Sheet 2 of 4)

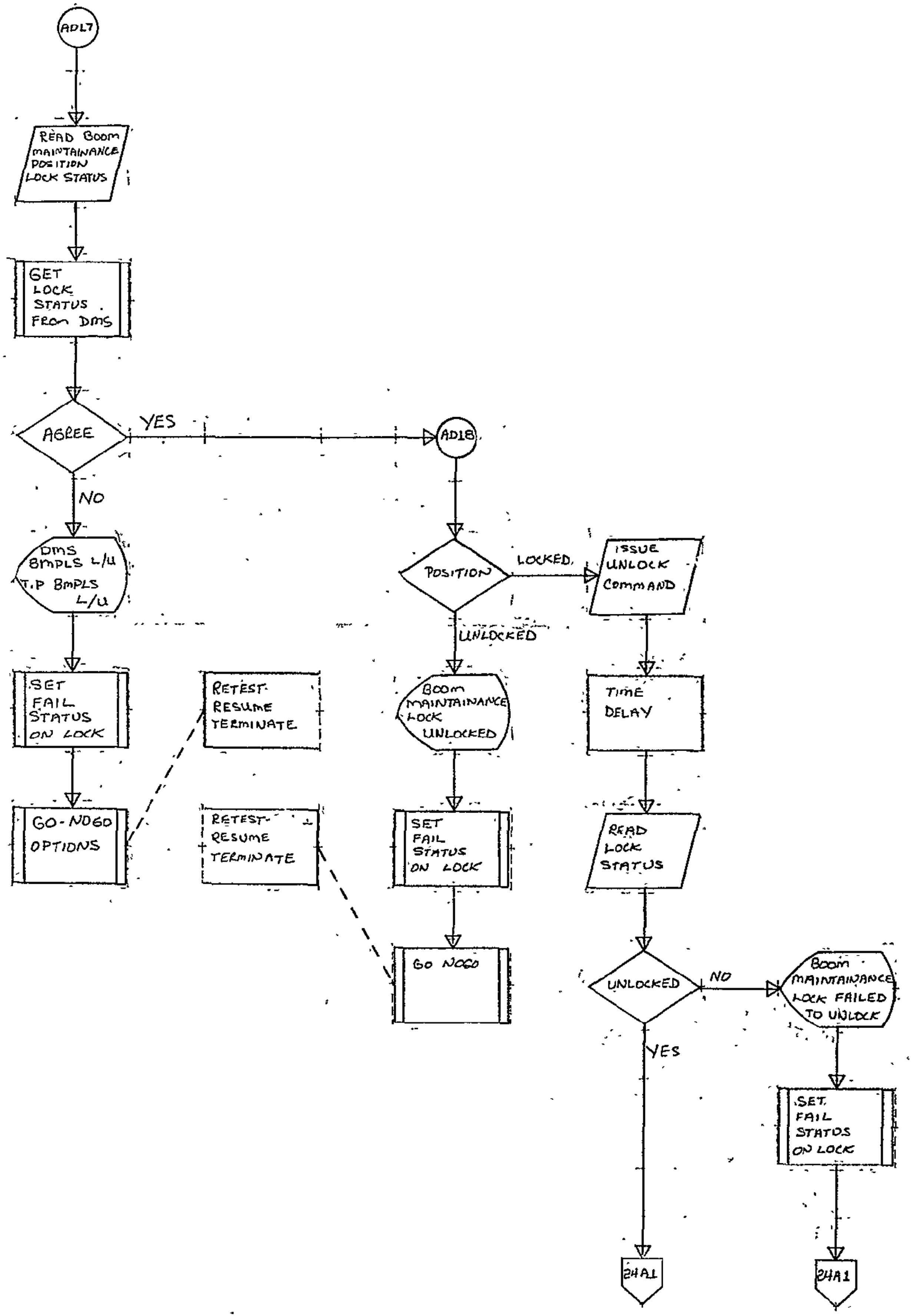


Figure 3-9. Antenna Fault Isolation (Sheet 3 of 4).

BOOM MAINTAINANCE POSITION (CONT'D)

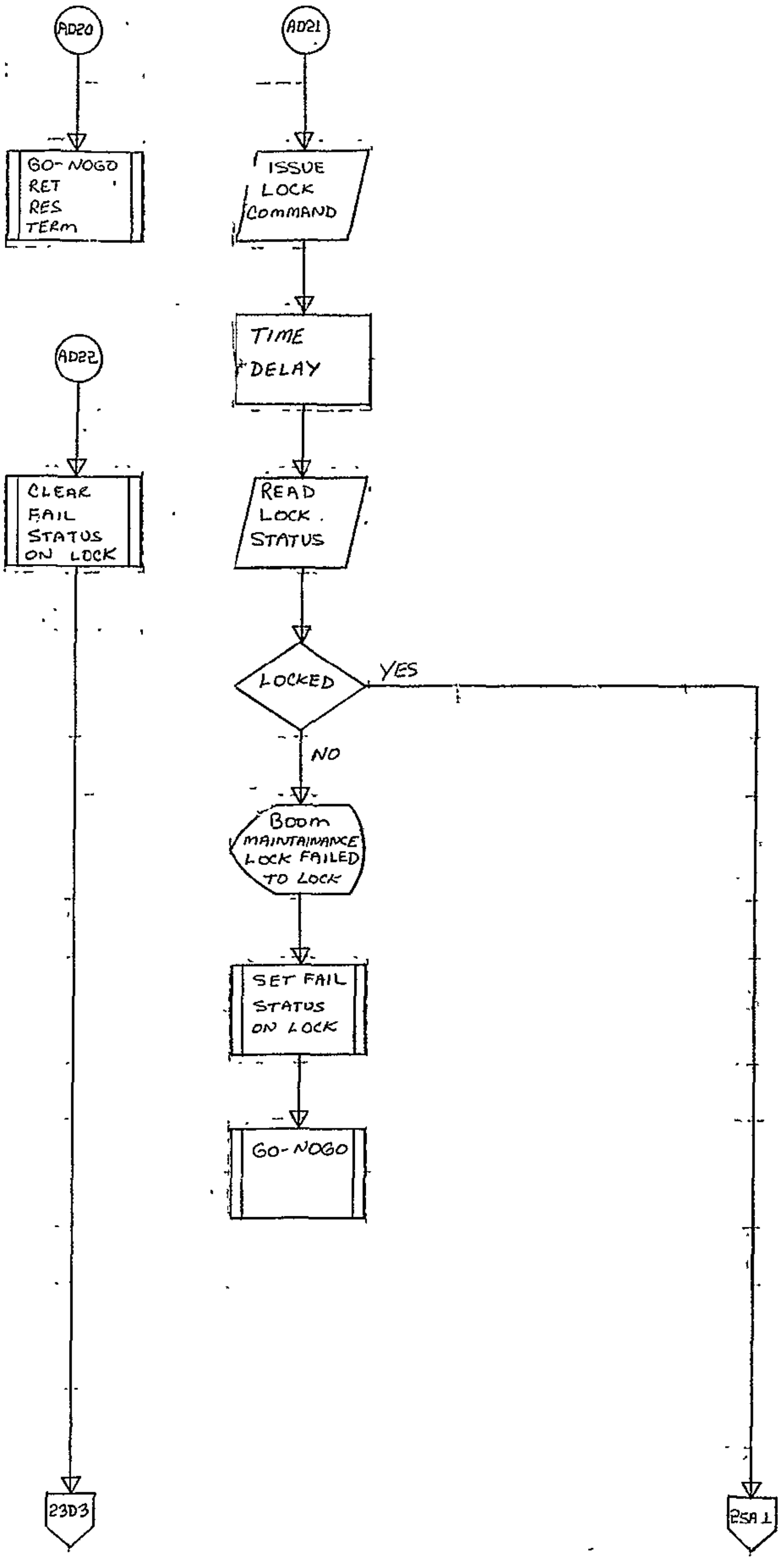


Figure 3-9. Antenna Fault Isolation (Sheet 4 of 4)

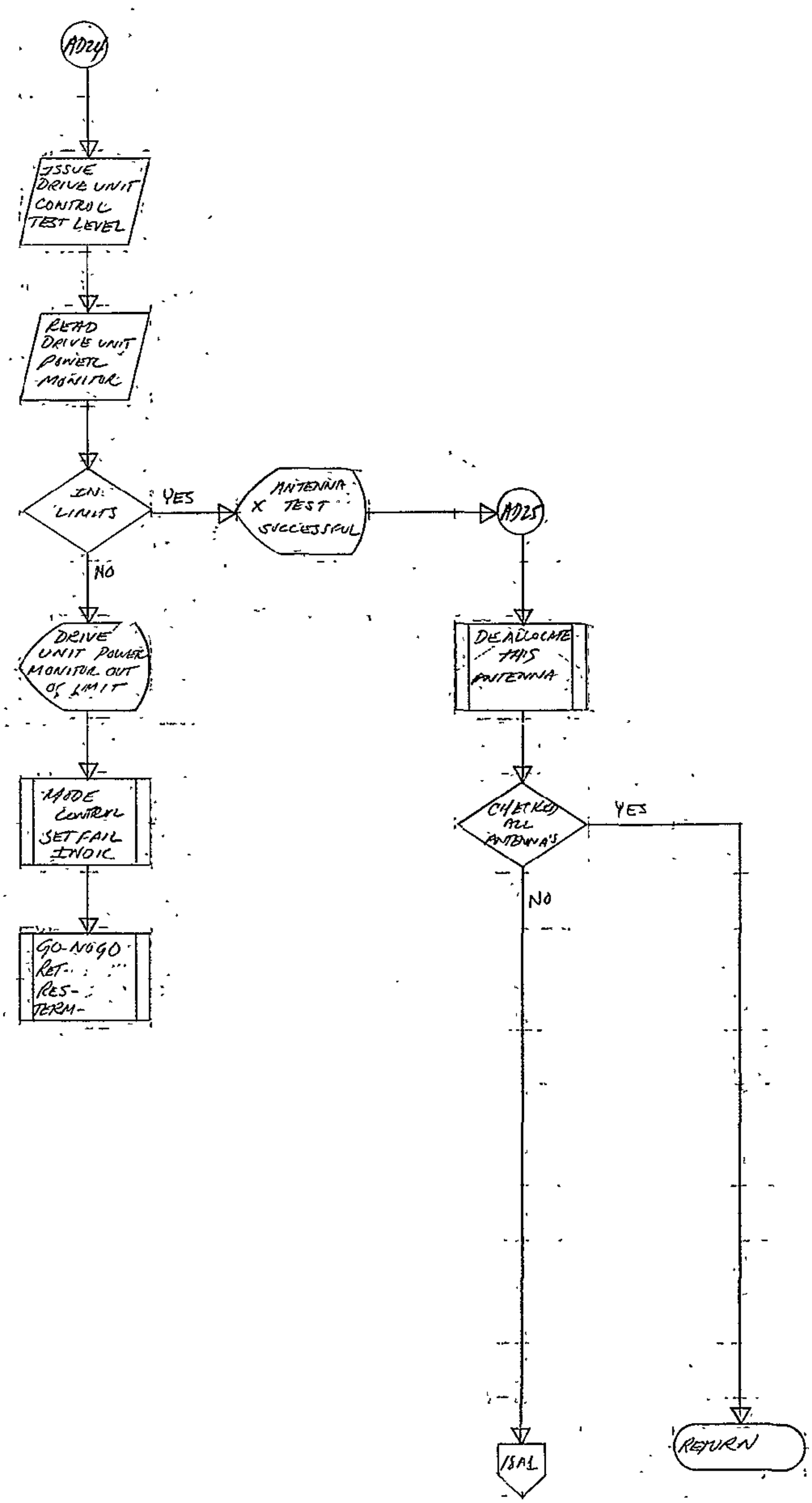


Figure 3-10. Antenna Fault Isolation

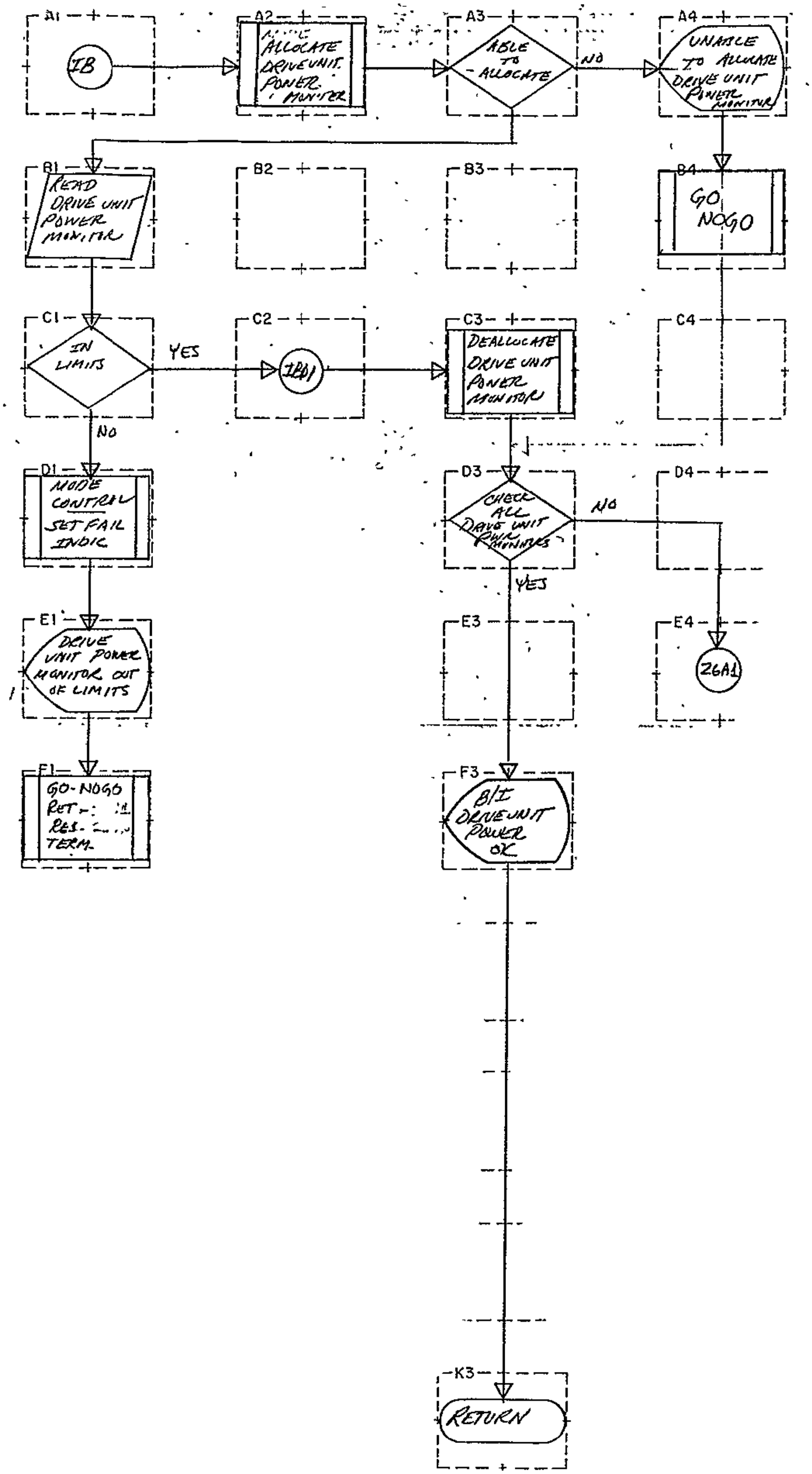


Figure 3-11. Isotode/Brayton Power Handling

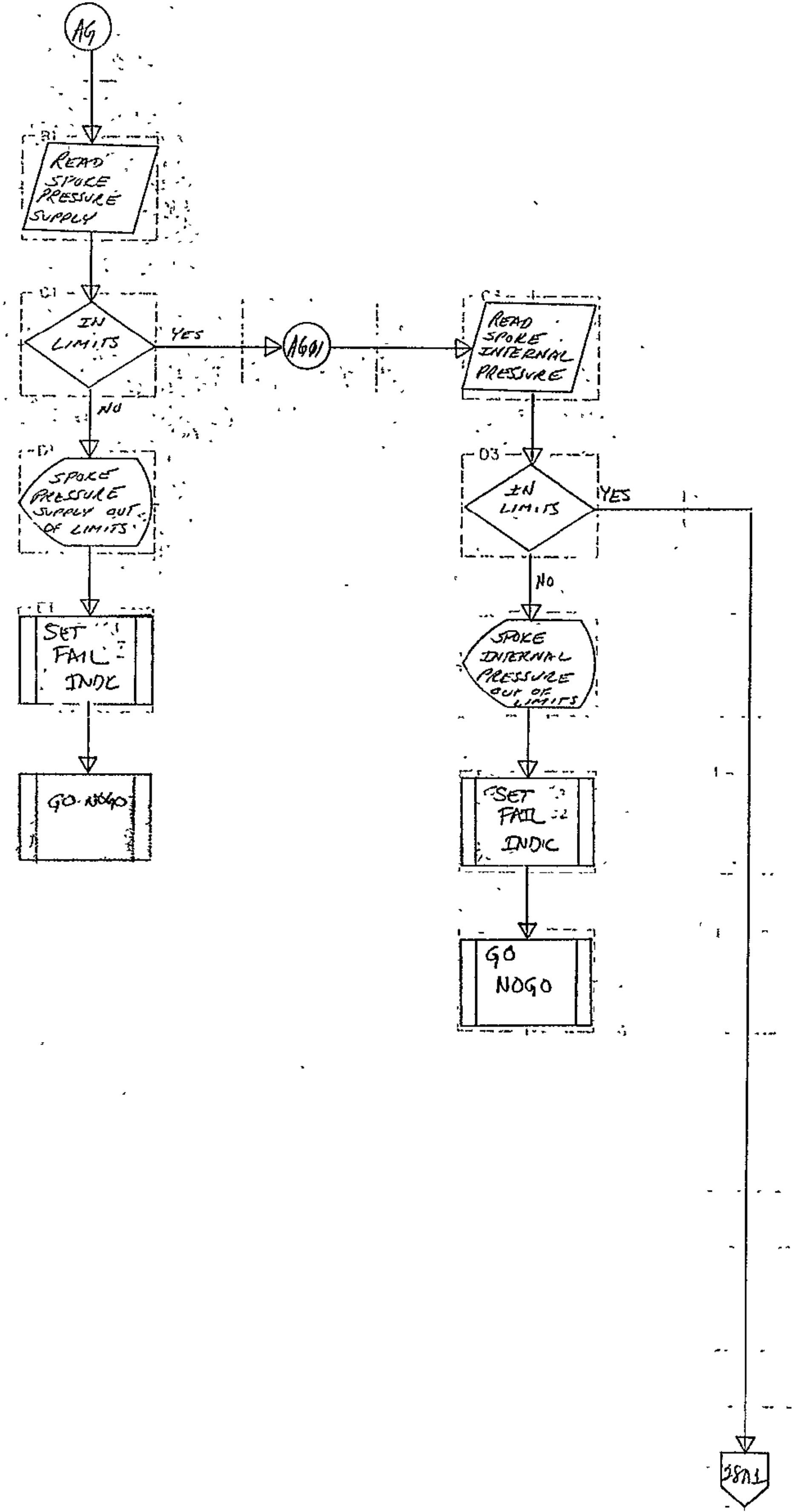


Figure 3-12. Art G Checkout

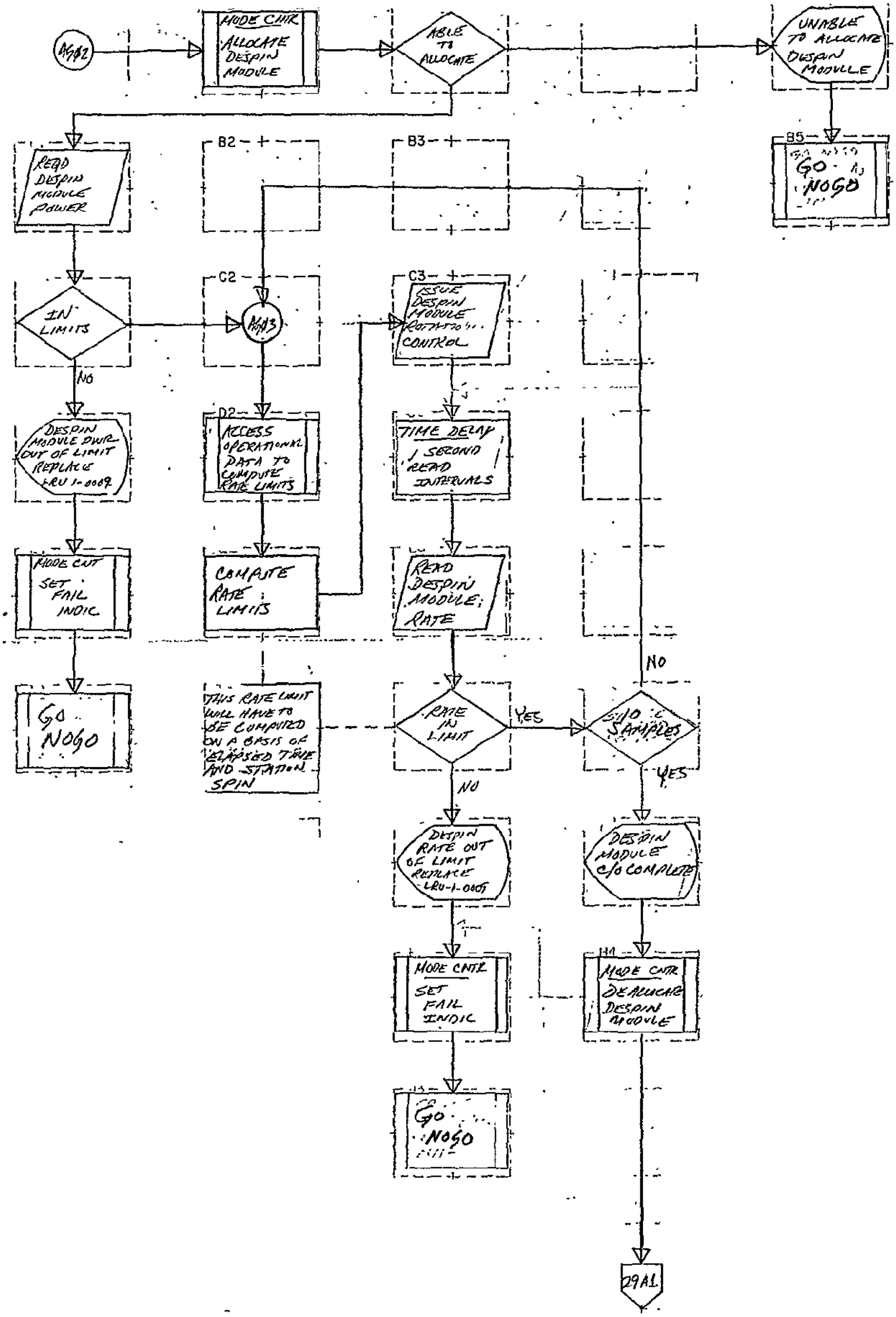


Figure 3-12. Art G Checkout

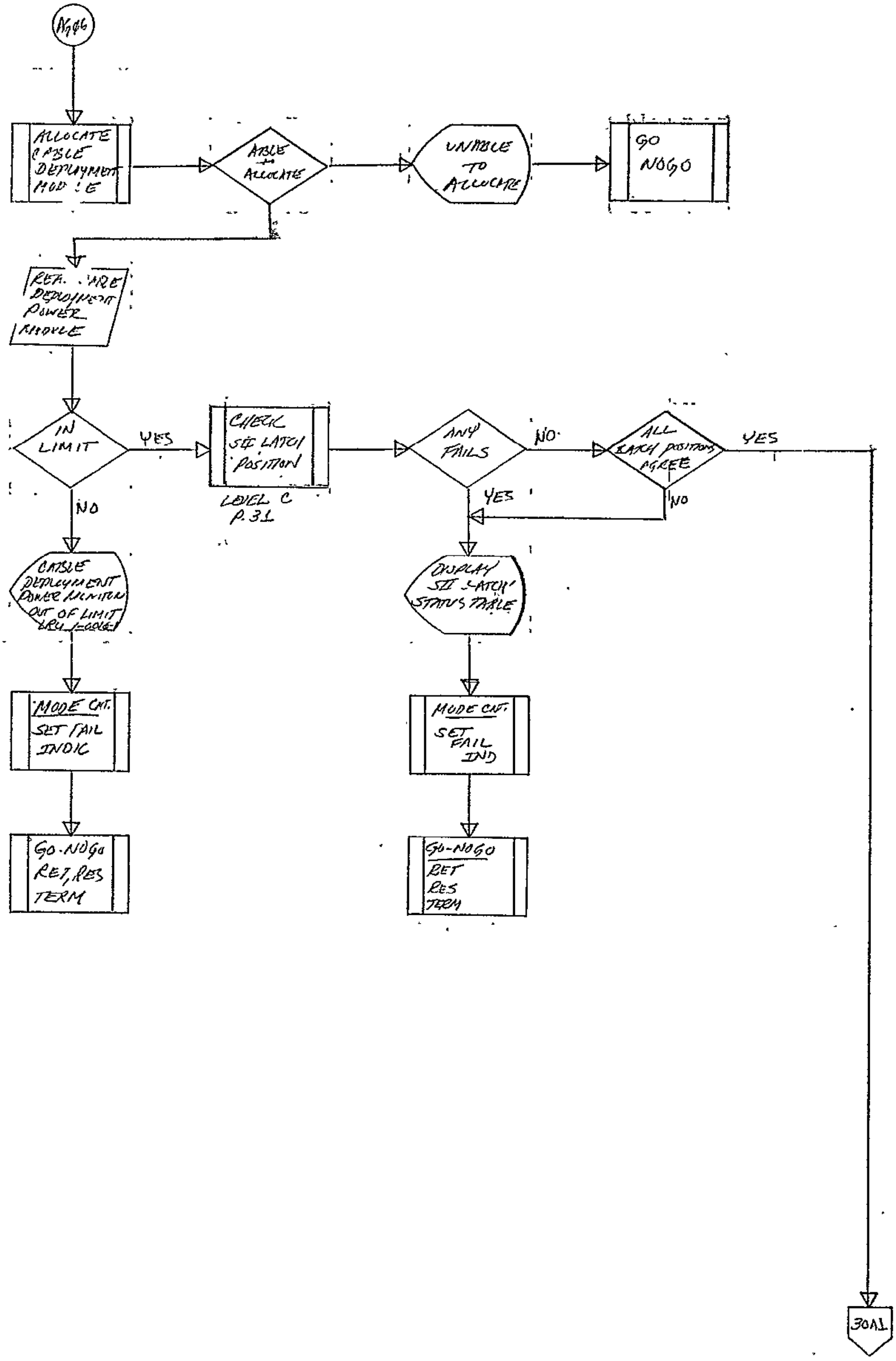


Figure 3-12. Artificial G Checkout

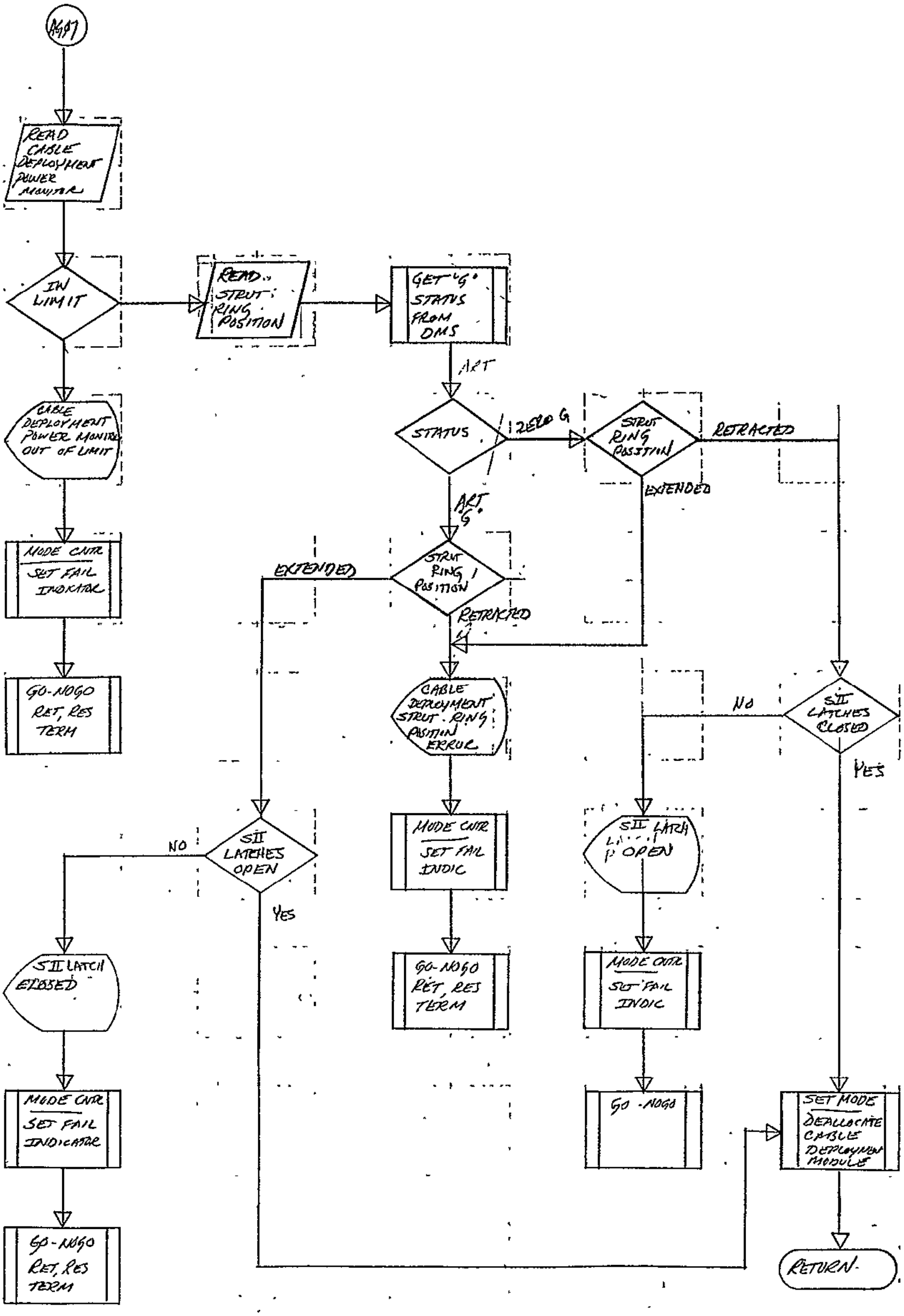
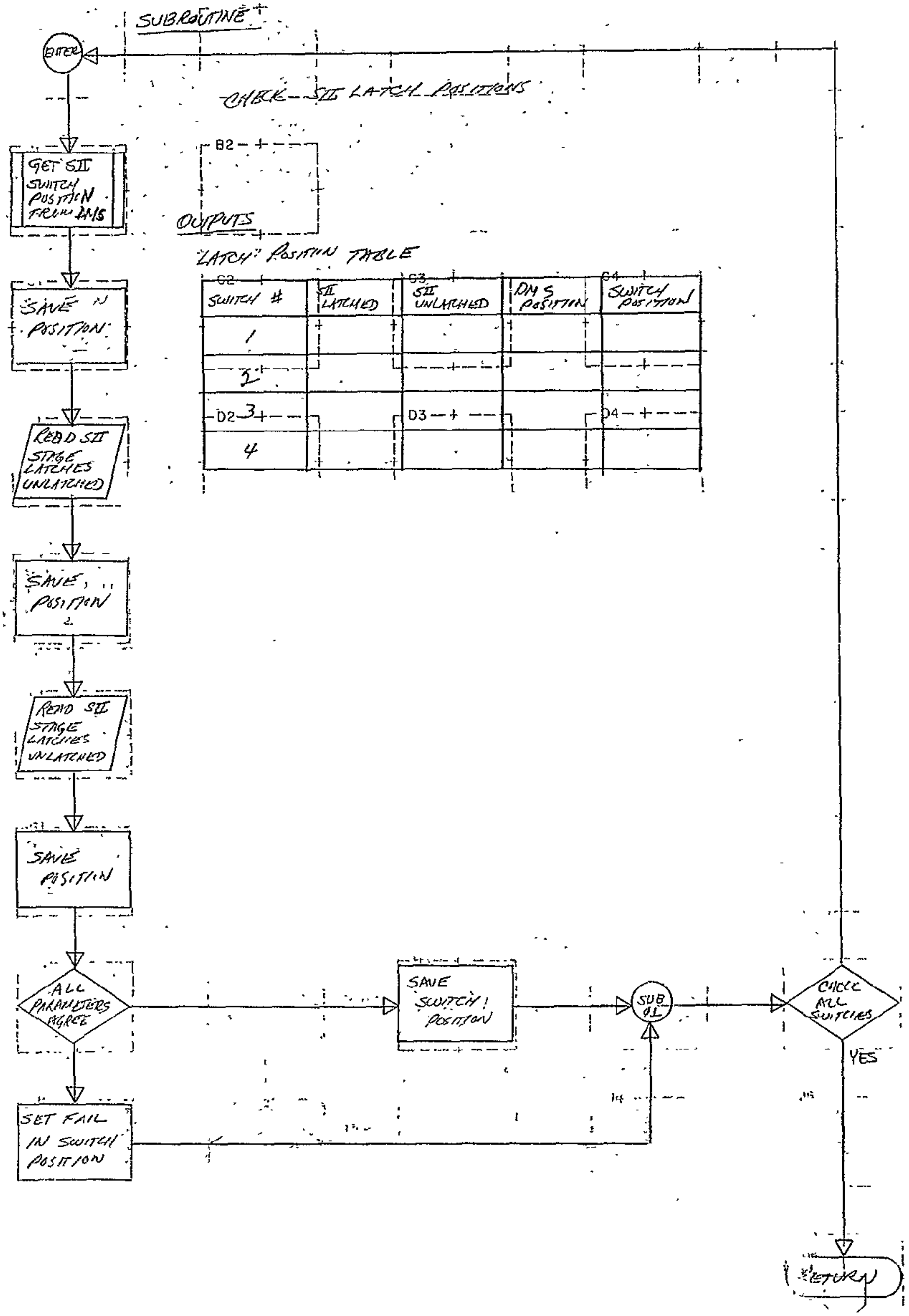


Figure 3-12. Artificial G Checkout



OUTPUTS

"LATCH" POSITION TABLE

SWITCH #	SII LATCHED	SII UNLATCHED	DMS POSITION	SWITCH POSITION
1				
2				
3				
4				

Figure 3-12. Artificial G-Checkout

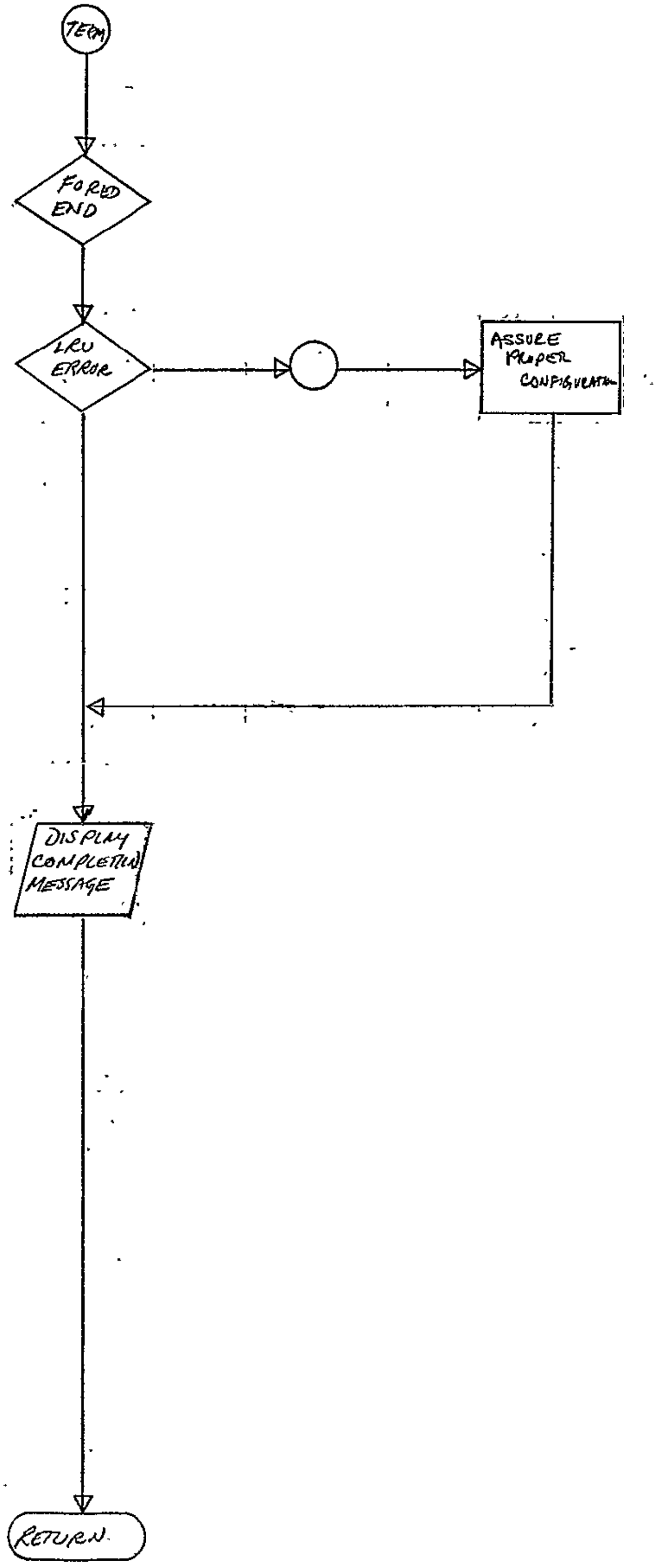


Figure 3-13. Termination Routine

3.1.4 DATA BASE REQUIREMENTS

This program requires access to the data base for the following data:

- Association of symbolic test points with the actual device addresses
- Status and allocation of configurable components

3.1.5 HUMAN PERFORMANCE

The operator is required to communicate with the program to accomplish the desired function. Specifically, the operator must initiate the program using the system communication function. The program may be terminated prior to completion by using the system communication function.

In addition, when errors are detected, the operator is provided with options to control program execution sequence. These options are referred to as GO-NOGO options, and permit the operator to retest the LRU which failed, resume the program execution, or to terminate program execution.

3.2 INTERFACE REQUIREMENTS

This program requires interface with the master executive, OCS executive, and the structure subsystem hardware.

3.2.1 INTERFACE DIAGRAM

The interface between the Structure Subsystem checkout program and the executive architecture is shown in Table 3-4.

3.2.2 DETAILED INTERFACE DESCRIPTION

This section defines the interfaces which are required in the executive architecture to service this program.

3.2.2.1 Input Processor

Keyboard Inputs - This program requires the ability to access parameter information input by the operator from the keyboard to specify the selection of programmed options.

Access Parameter Data - This program requires the ability to access parameter data which is needed to determine the options to be selected.

3.2.2.2 Output Processor

Measure Test Points - This program requires the capability to address specific test points through the Remote Data Acquisition Units (RDAU).

Table 3-4. Structures Checkout Program and Executive Program

		EXECUTIVE PROGRAM															
		Task Specification	Processor Allocation	Task Termination	Memory Allocation	Peripheral Allocation	Data Base Allocation	Interrupt Monitoring	Failure Response	Output Processor	Input Processor	Special Processor	Data Retention	GO-NOGO Processor			
STRUCTURES C/O PGM	Option Control									x	x						
	Operational Status									x		x		x			
	Docking Mechanism									x		x		x			
	Spacecraft Access									x	x	x		x			
	Antenna Deployment									x		x		x			
	I/B Power Handling									x		x		x			
	Artificial "G" C/O									x		x		x			
	Termination			x						x							

Issue Stimulus - This program requires the capability to issue stimuli to specified test points.

Display Control - This program requires the capability to present data at a display console to notify the operator of available options, or to present error messages.

3.2.2.3 Special Processor

Mode Control - This program requires the capability to have exclusive control over particular hardware components to accomplish the required testing. This requires the capability to allocate a component to the program, and to permit the program to indicate a failure status on the device when an error is detected.

3.2.2.4 GO-NOGO Option Processor

This program requires the capability to sequence execution in response to specific keyboard direction.

3.2.2.5 Task Termination

This program requires an entry to return to the executive when it has completed execution.

3.2.2.6 Task Initiation

This program requires that the executive recognize and act upon the request to execute this program.

4.0 SUBSYSTEM SUMMARY

This section summarizes some of the information which has resulted from the analysis of the structure subsystem.

4.1 LANGUAGE ANALYSIS

4.1.1 ELEMENT ANALYSIS

The analysis of this subsystem indicates the following specific considerations:

- This program requires character manipulation capability. This has been met by modification of the read element.
- The program requires the ability to obtain the status of a configurable component from the DMS. This requirement has been met by creating a STATUS element.
- This analysis indicated the need to provide a limit option to the measure element. This has been added to the baseline.

- This program indicates the need for an element to assure that configurable components can be dedicated for its use. The ALLOCATE element has been developed to meet this requirement.

- The analysis of this program indicated the need for the following features to be incorporated into the language:

    - The ability to address multiple test points
    - The ability to index through tables of data
    - The ability to insert variable parameters into predefined displays

4.1.2 SIZING ESTIMATES

The sizing estimates developed for this subsystem are provided on a basis of element utilization, as depicted in Table 4-1. Table 4-2 provides estimates on a basis of processor and memory utilization.

4.2 EXECUTIVE ANALYSIS

The analysis of this program identified requirements for the following executive services:

- Read keyboard inputs
- Access parameter data
- Measure test points
- Issue stimuli
- Display control
- Mode control
- GO-NOGO processing
- Task termination
- Task initiation

Table 4-1. Element Estimates

PROGRAM TITLE: STRUCTURE SUBSYSTEM CHECKOUT PROGRAM

ELEMENT	OCCURRENCE	EXECUTION FREQUENCY
ALLOCATE	12	32
CALL	58	34
DELAY	12	53
DISPLAY	64	45
GO TO	74	53
IF	71	295
MEASURE	37	141
READ	4	12
SEARCH	8	9
SOLVE	58	156
STATUS	101	98
STIMULI	11	44
TOTAL	510	972

Table 4-2. Structures Subsystem

Periodic Test

Program Text	4.0K	
Working Buffer	.5K	
Main Storage		4.5K words
Displays	.5K	
Program code	3.5K	
Auxiliary Storage		4K words
CPU time	.005	
I/O time	2.913	
Time in Main Storage		2.918 minutes

APPENDIX B

GNC SUBSYSTEM CHECKOUT PROGRAM DESIGN

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

1.0 SCOPE

The GNC Subsystem Checkout Programs are used to monitor GNC Subsystem test points in order to verify the proper operation of the functional assemblies of which it is made. Cognizance is taken of the mode of the GNC Subsystem by utilizing data prepared by the GNC Application Programs. When a fault indication is detected, isolation is performed by logically combining the measurements taken from test points of the subsystem. Fault detection is initiated and performed without crew assistance. This does not preclude crew control, however, since a test module may be initiated from the display console keyboard or by ground command at any time. In addition, the rate at which the monitoring modules are initiated may be altered in a similar manner.

The functions identified are those for fault detection, fault isolation, trend analysis, reconfiguration, and calibration. They are implemented in a combination of hardware, multi-level executive, and high-level language programs. The modular programs and executive services are multi-purposed and can be invoked by the crew, ground personnel, or other programs.

2.0 APPLICABLE DOCUMENTS

- "Utility Services", Space Station, MSFC-DRL-160, Line Item 13, Preliminary Systems Design Data, Volume I, Space Station Preliminary Design, Book 4, Contract NAS8-25140, July 1970.
- "Baseline Subsystem Descriptions", Interim Report RA-1, Contract NAS9-11189, August 1970.
- "Subsystem Checkout Strategy", Interim Report RA-5, Contract NAS9-11189, September 1970.
- "Line Replaceable Unit Definition", Interim Report RA-6, Contract NAS9-11189, September 1970.
- "Subsystem Test Definition and Measurement/Stimulus List", Interim Report RA-8, Contract NAS9-11189, November 1970.

3.0 REQUIREMENTS

The GNC Checkout Programs provide for fault detection, trend analysis, fault isolation, reconfiguration, and calibration by a combination of executive services, high level language programs, and coordinated hardware utilization.

3.1 PERFORMANCE

The modules used for fault isolation, reconfiguration, and calibration are capable of being initiated by crew, ground, or another module. Before proceeding on an analysis of subsystem test points, the subsystem mode is ascertained in order to determine which analysis modules should be employed and how the analysis should be performed. Checkout module performance is valid whether initiated as a result of an RDAU limit check, a crew or ground command, or by the Pacer, (a software module which automatically initiates programs at a prescribed rate). A diagram of the Guidance, Navigation, and Control Subsystem is shown in Figure 3-1.

Figure 3-2 reflects the functional breakdown of the subsystem and the hierarchical relationship which exist between the various assemblies.

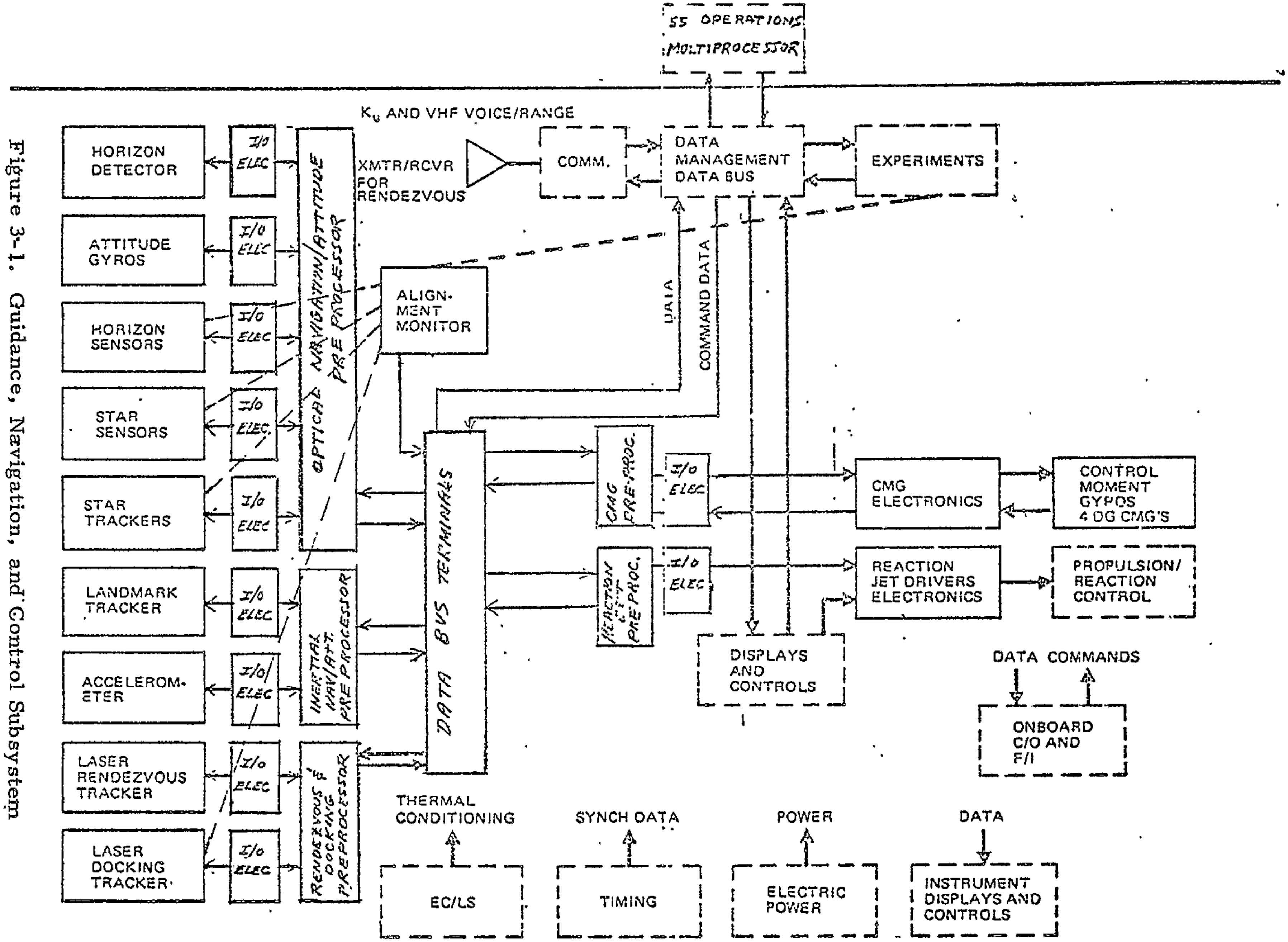


Figure 3-1. Guidance, Navigation, and Control Subsystem

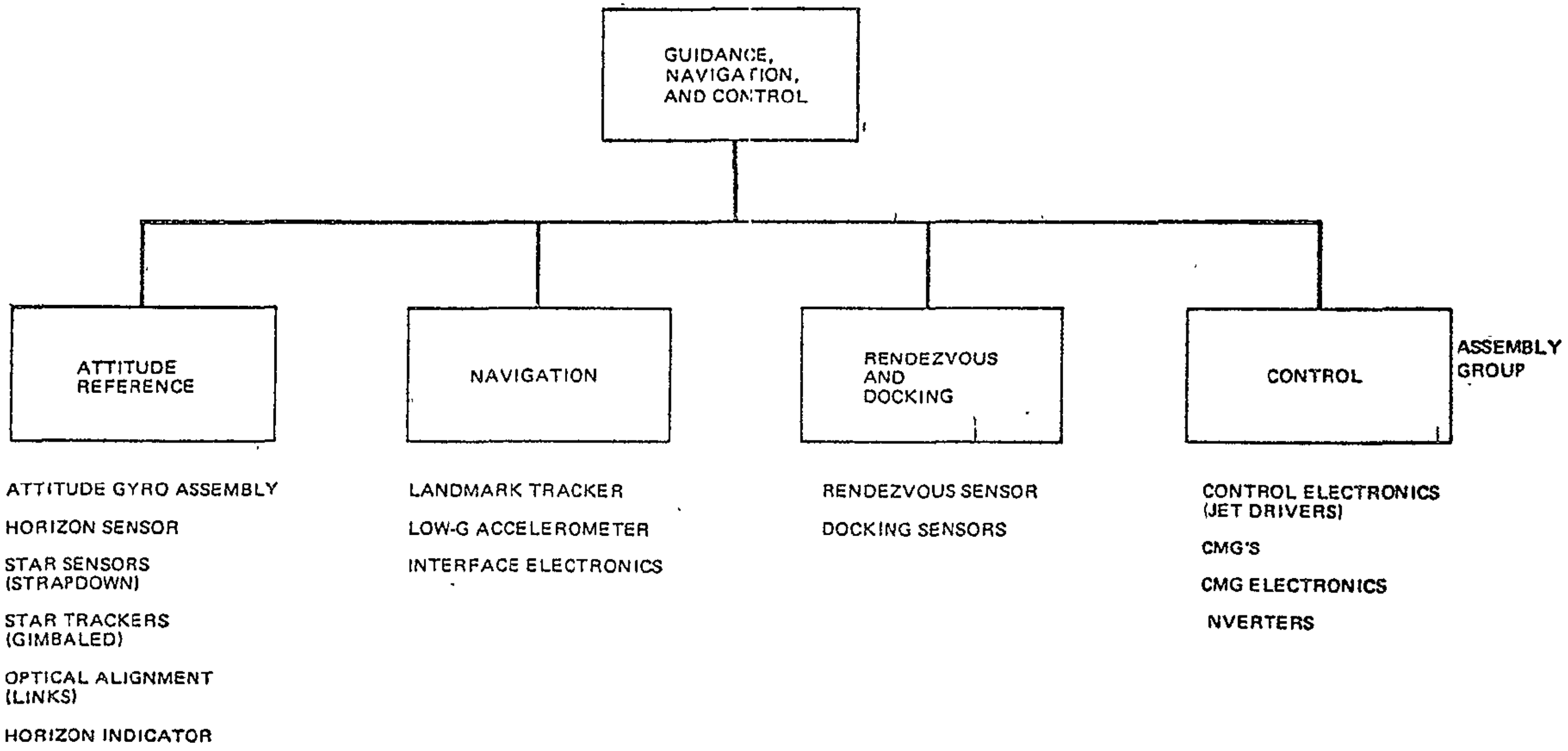
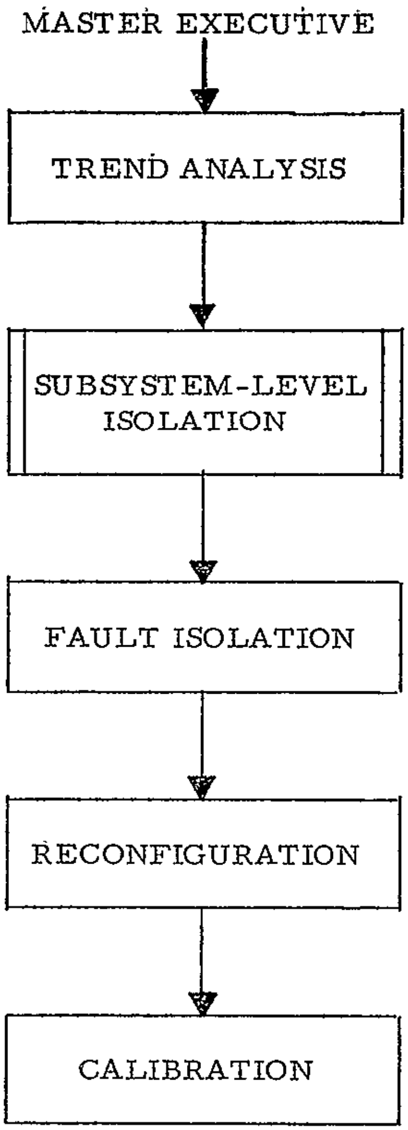


Figure 3-2. GNC Subsystem Assembly Configuration

The GNC Checkout Program Structure is outlined in the following chart:



All functions are initiated as independent modules by the master executive program. When a module requires another program to be executed, it requests this of the master executive.

3.1.1 SYSTEM REQUIREMENTS

The GNC system requires that trend analysis be performed and used as a fault detection method. In addition, interface with crew and ground is required upon detection of an unfavorable trend, so that potential resupply aspects can be considered. Operational data will be used where appropriate for checkout purposes.

The GNC Checkout Programs are written in a high level language so that development and alteration by professionals other than programmers will be feasible. The programs will interface with a multi-level executive, the lowest level of which will also serve non-checkout programs. Upper levels of the executive will perform services unique to the checkout mission.

Note that interface between the Data Management Subsystem and the GNC Subsystem is accessed only through the Master Executive. Checkout programs may interface either directly or indirectly with an executive level.

Fault detection will be accomplished by hardware under the control of software, by GNC application programs, and by trend analysis programs. The most common method will be hardware under control of software. Limits are stored in the memory of the Remote Data Acquisition Units (RDAU's) which continuously check test points, and interrupt the multiprocessor if an out-of-limit signal is received. The rate at which the RDAU checks limits meets or exceeds the highest rate requirement for fault detection sampling.

While continuous orbital monitoring will be performed by RDAU hardware under software executive control, periodic checks will be performed by using the same modules employed during fault isolation.

3.1.2 OPERATIONAL REQUIREMENTS

The GNC checkout modules are required to perform caution and warning, trend analysis, calibration, fault isolation to the LRU level, and reconfiguration of the GNC Subsystem with a modular design which allows employment of various program modules in a variety of configurations upon initiation by RDAU interrupt, crew, ground, Pacer, or other programs. The fault detection control and trend analysis functions are implemented by extensive use of executive modules. The higher level language is used for the fault isolation, reconfiguration, and calibration functions with executive support in the areas of mode analysis and data base management.

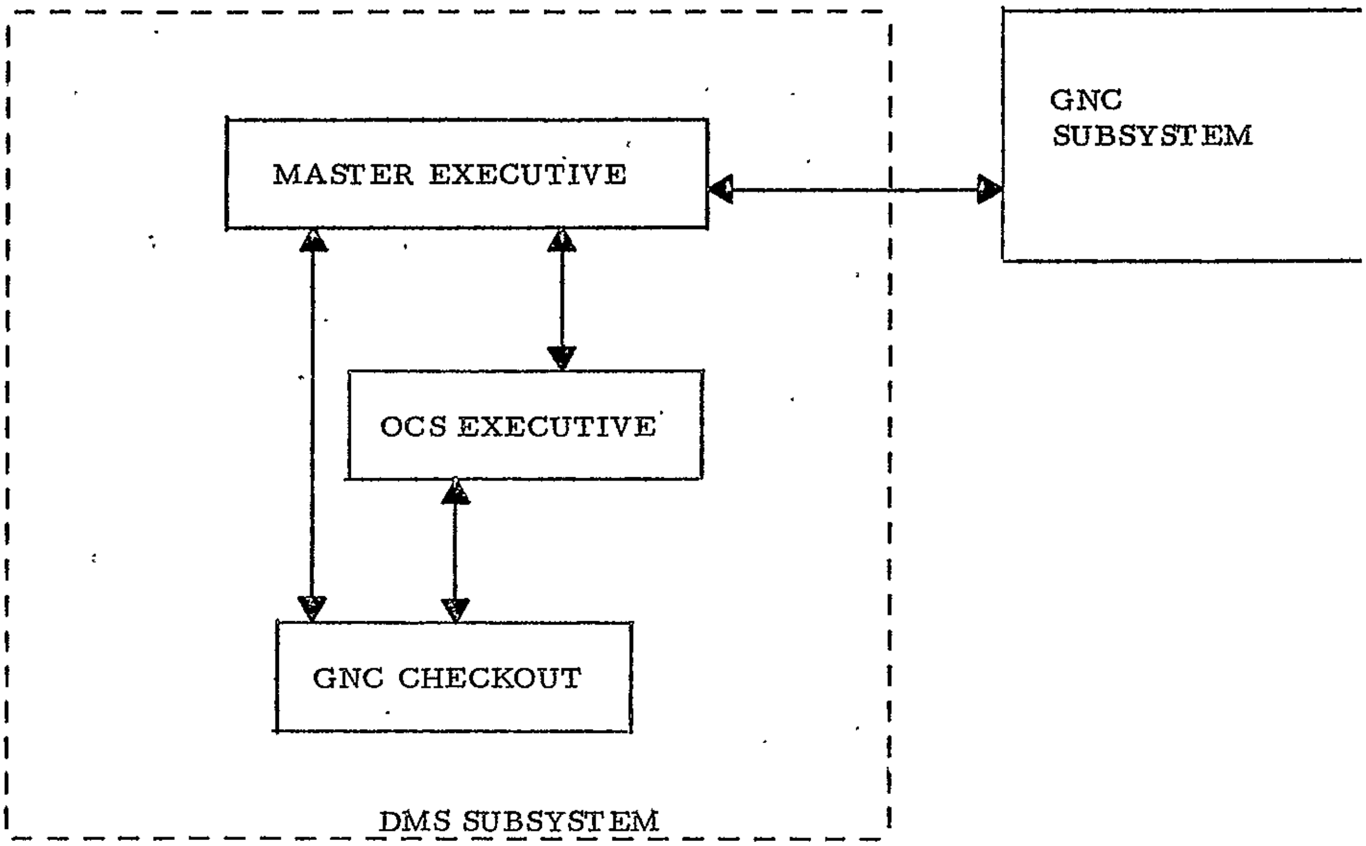


Figure 3-3. Executive and Subsystem Interfaces

3.1.2.1 Fault Detection Function

Since the fault detection function is operationally implemented in the RDAU hardware, this section discusses the control of the RDAU limit check feature. The contents of RDAU memory must be redundantly maintained in auxiliary storage so that the secondary RDAU may be initialized if the primary fails.

Input to the fault detection control function consists of the command to change RDAU limits, a mode table, and a limit table. Output consists of the mode table, limit table, the RDAU memory, and displays.

Information processing takes place in the OCS Executive, and consists of changing the RDAU channel mask to enable or disable interrupts caused by out of limit signals, changing the limits, and updating the mode and limit tables accordingly. The extensive involvement with executive table formats makes implementation as an executive service more attractive than implementation in a higher-level language.

Limit check specifications are made regarding a symbolic test point address. The fault detection control function translates this into specific RDAU memory changes for both the primary and secondary RDAU's. In doing so, it must reference the symbolic address translation, configuration, and RDAU memory tables.

A flowchart of the fault detection control function is shown in Figure 3-20.

3.1.2.2 Fault Isolation Function

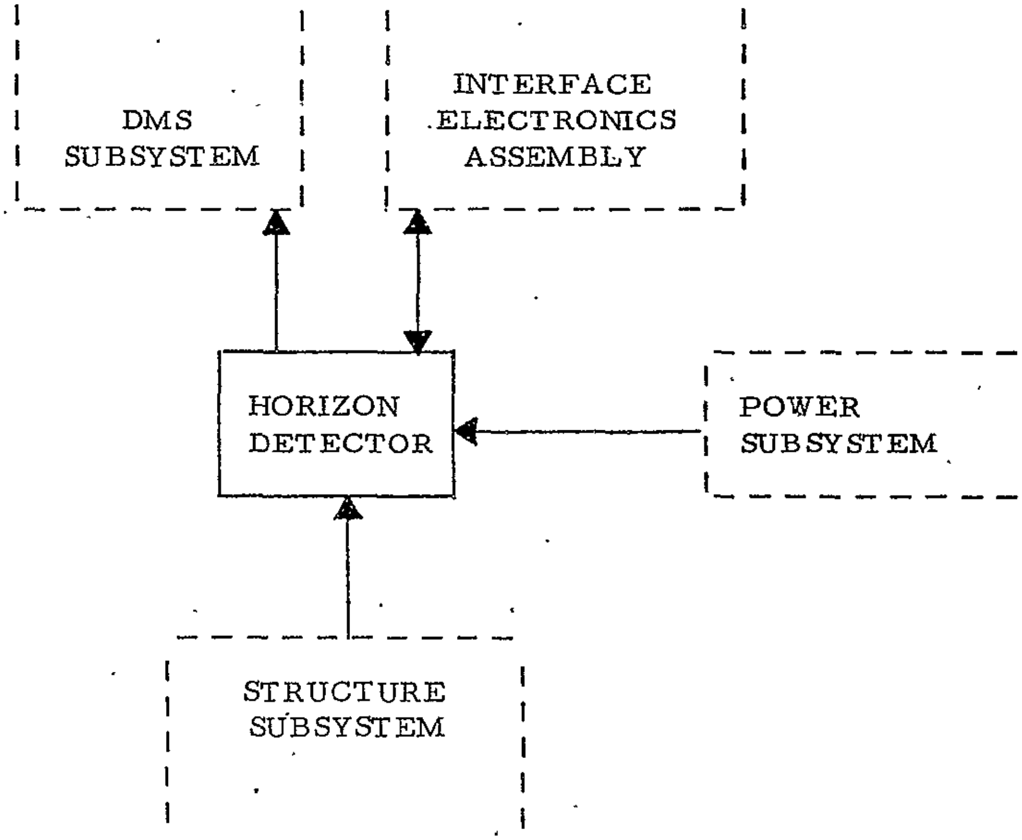
Primary logic control of isolation programs is accomplished by the language TOOL. Those services and functions which are common to fault isolation in other subystems are provided as executive services.

Input to the fault isolation function consists of RDAU interrupts, crew initiation, initiation by other programs, the symbolic address of the test points, test point measurements, data being managed by application programs, and the mode table.

Output from the fault isolation function consists of stimuli, commands, displays, mode table, and parameters for the reconfiguration module.

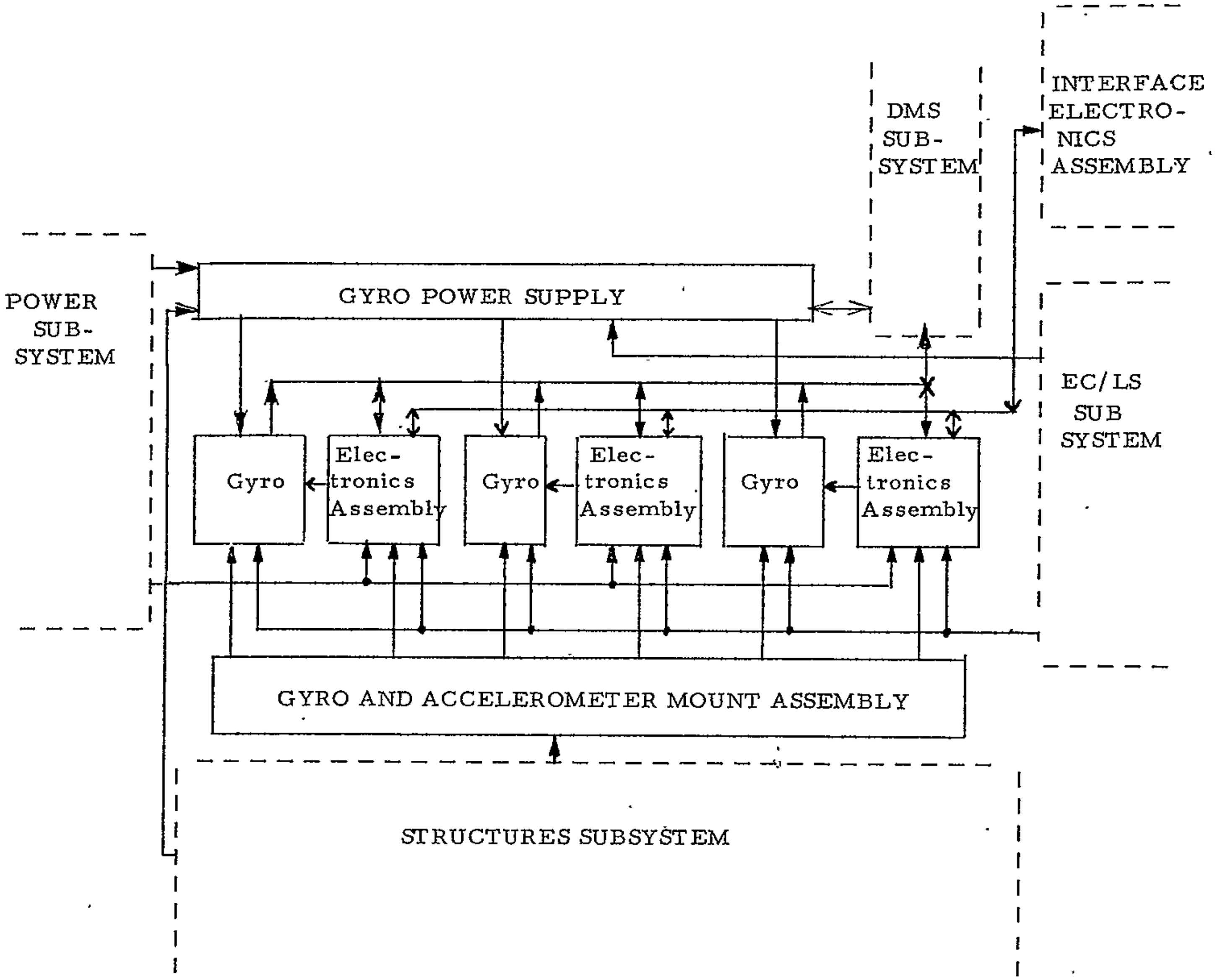
Fault isolation processing consists of determining whether the mode of the assembly allows the test to proceed, allowing a mode change if necessary, evaluating the interfaces supplied to the assembly under test, and evaluating the LRU's of the assembly. LRU evaluation involves an examination of interfaces, similar to that done for the next higher assembly; consequently, the order in which LRU's are tested is important. The modules are designed to provide verification on an as-required or periodic basis, such as just prior to artificial G mode.

Of particular importance in the isolation of failed Line Replaceable Units (LRU's) is the examination of interfaces between the assembly under analysis and other assemblies. In Figures 3-4 through 3-18, the interfaces which are important during fault isolation are shown. Prior to evaluating the performance of any assembly, it is necessary to make sure that its supporting interfaces are within tolerance. The approach required for GNC fault isolation, showing the relationship between the mode, interface, and assembly analyses appears in Figure 3-19.



Required	2
Redundant	0
Total	2

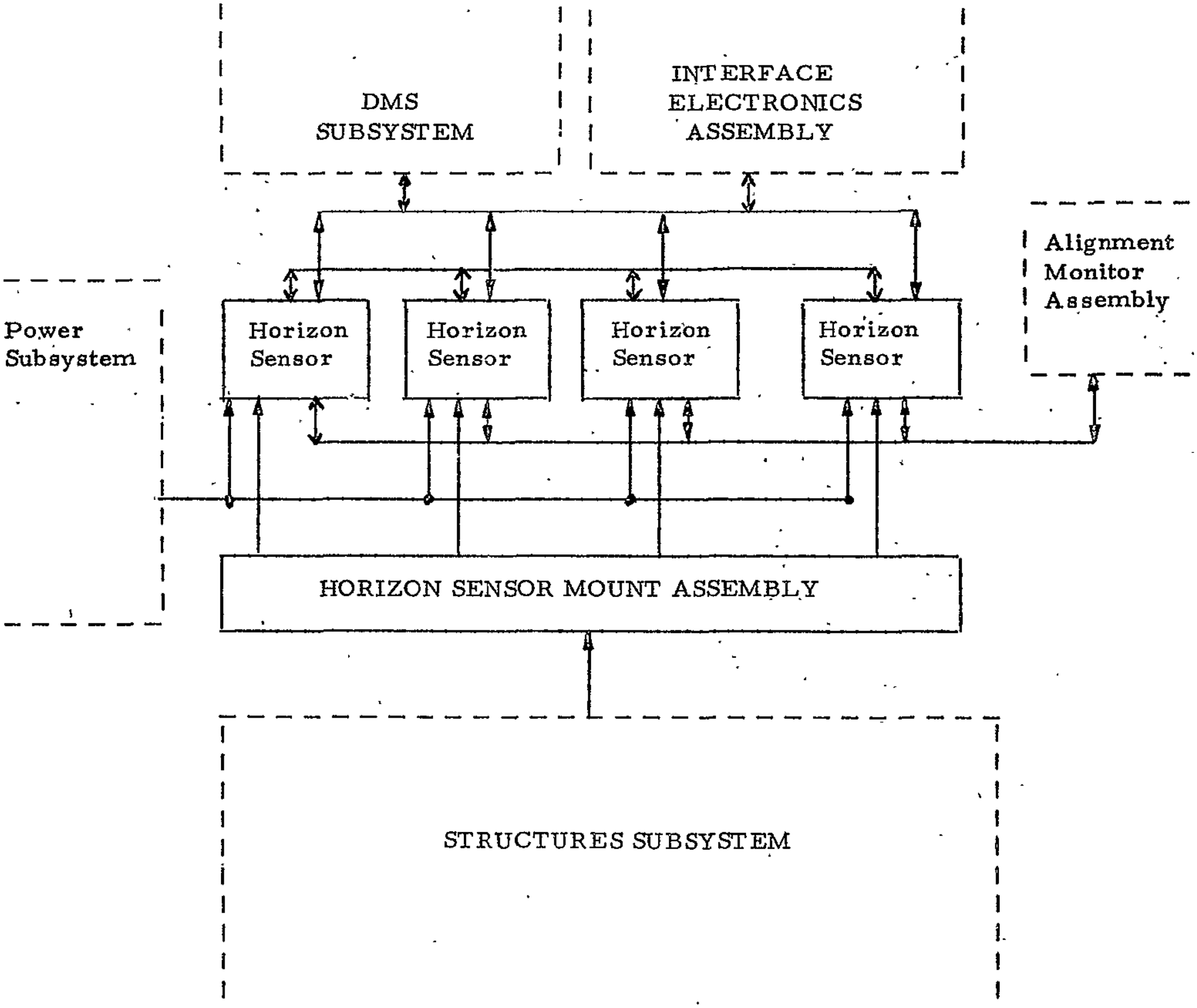
Figure 3-4. LRU Interface Diagram Horizon Detector Assembly



NOTE: One-Half of the Assembly (symmetrical) shown.

Required	1
Redundant	0
Total	1

Figure 3-5. LRU Interface Diagram, Attitude Gyro Assembly



Required	1
Redundant	0
Total	1

Figure 3-6. LRU Interface Diagram, Horizon Sensor Assembly

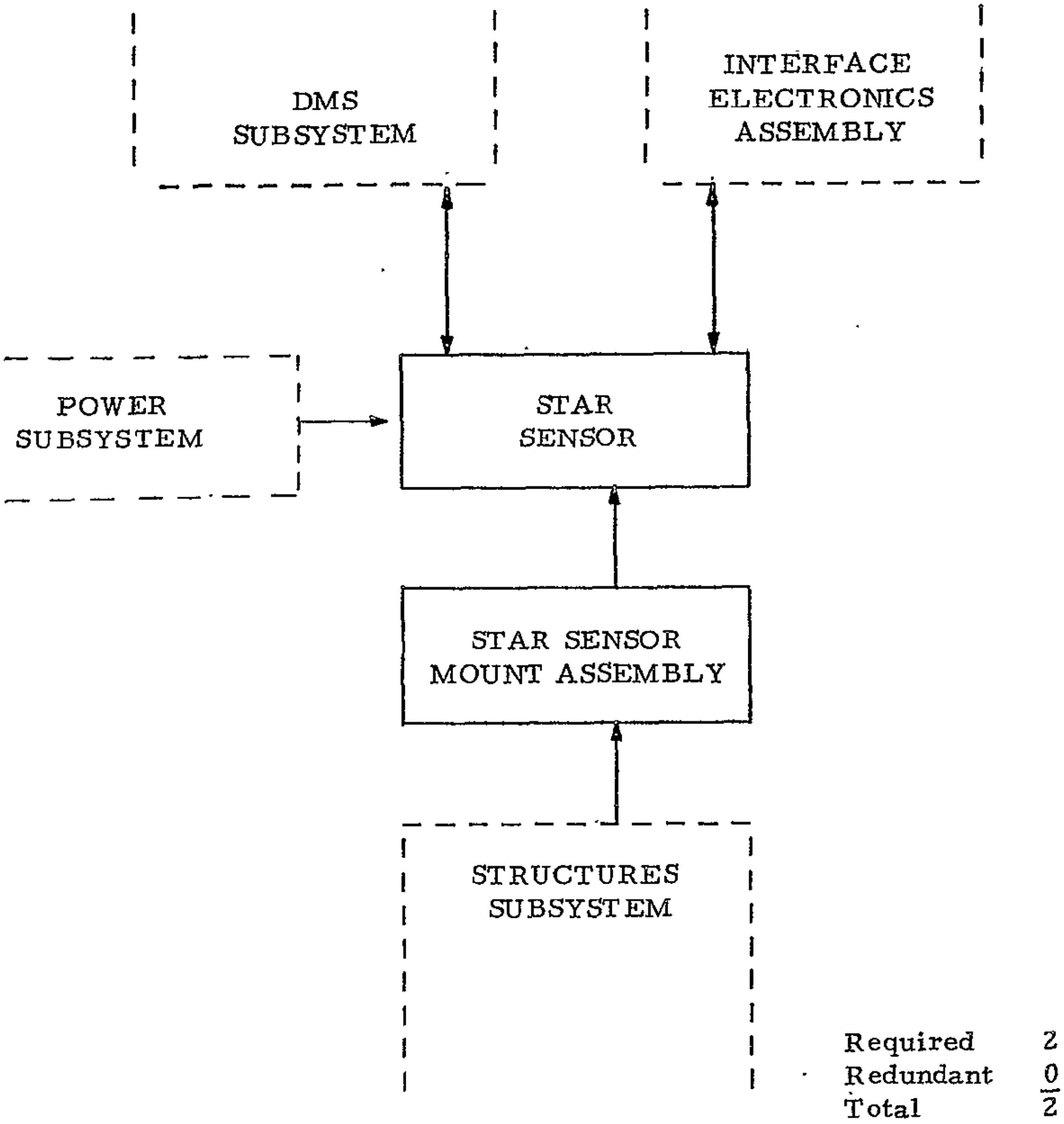


Figure 3-7. LRU Interface Diagram, Star Sensor Assembly

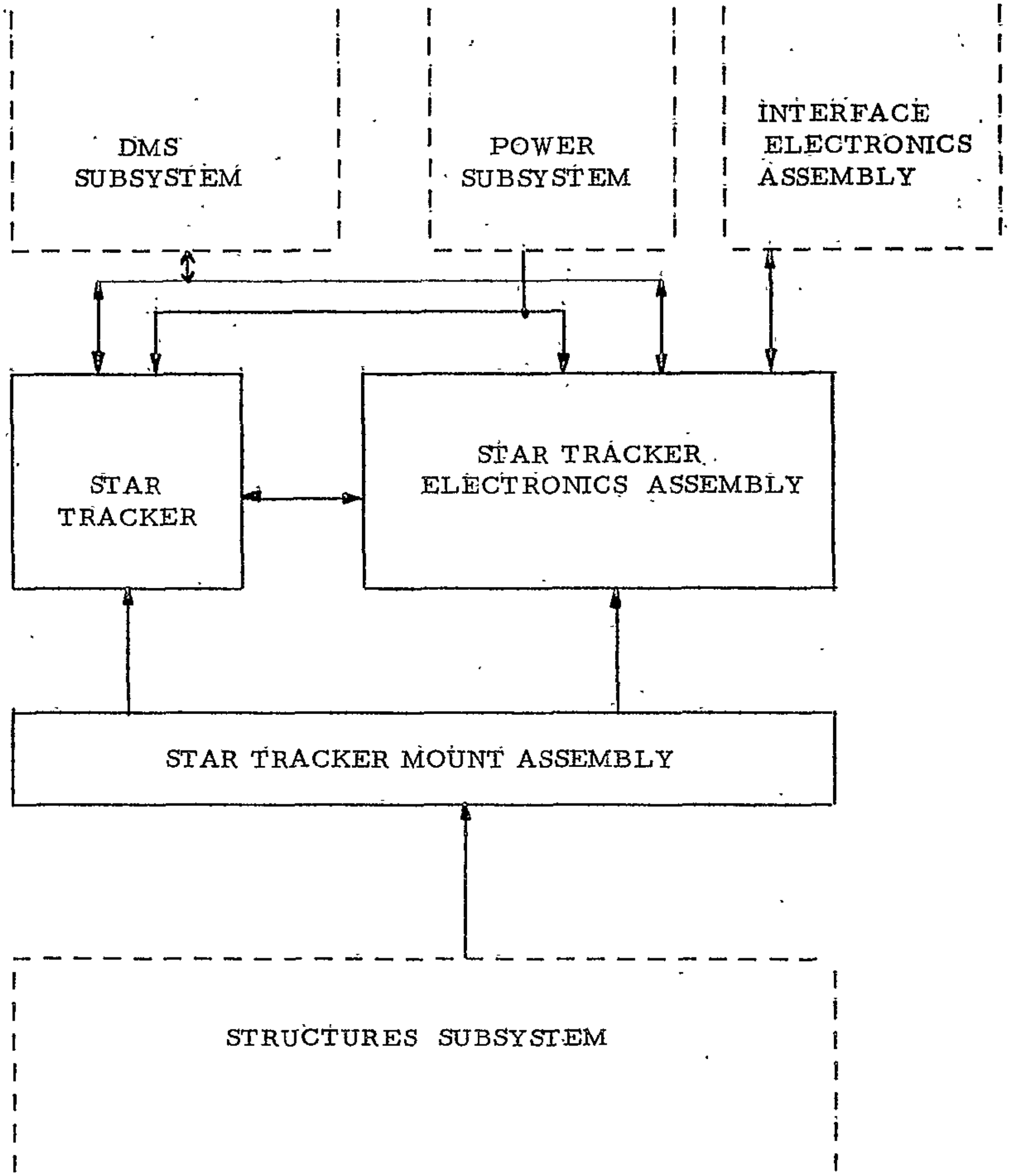
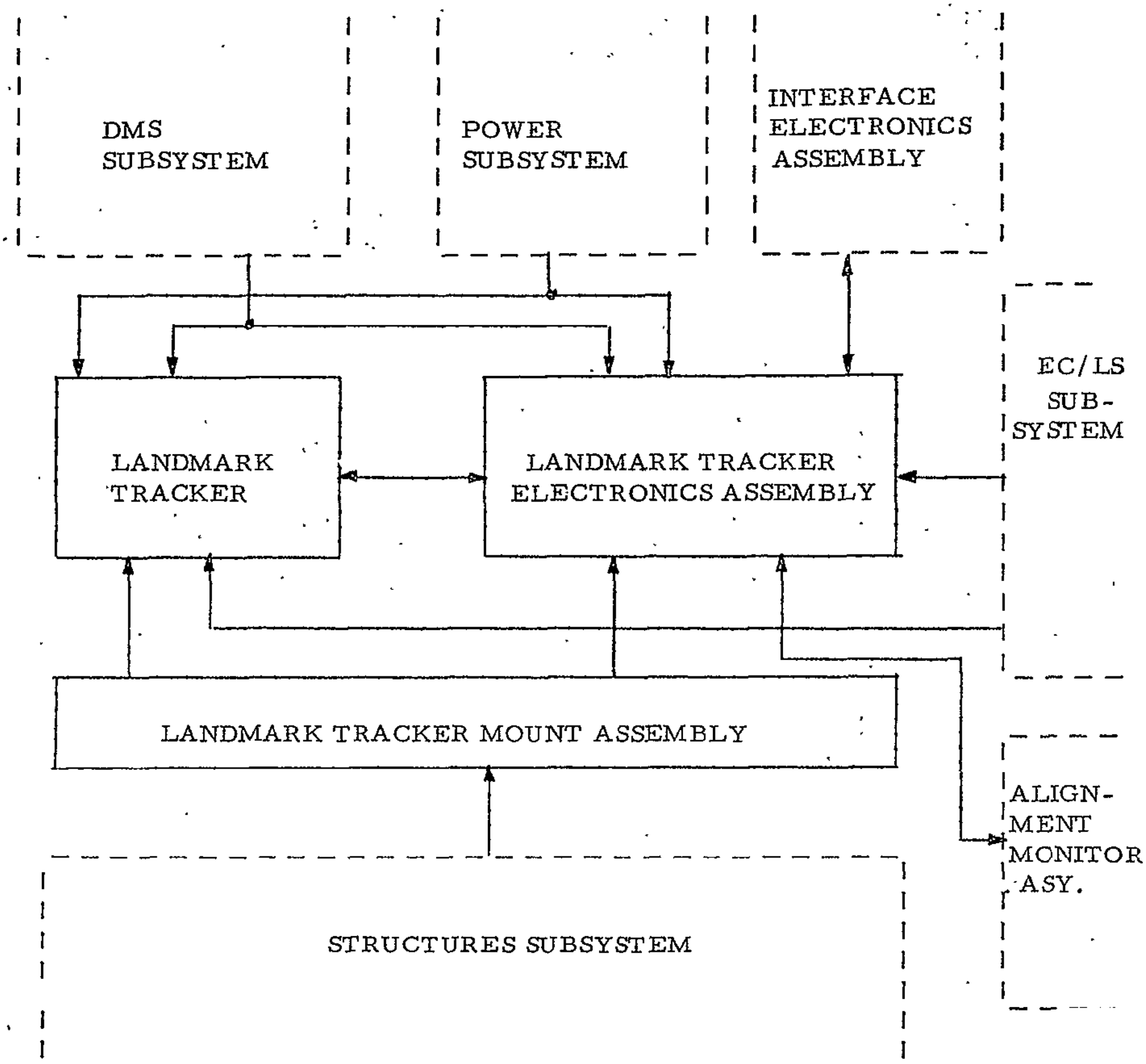
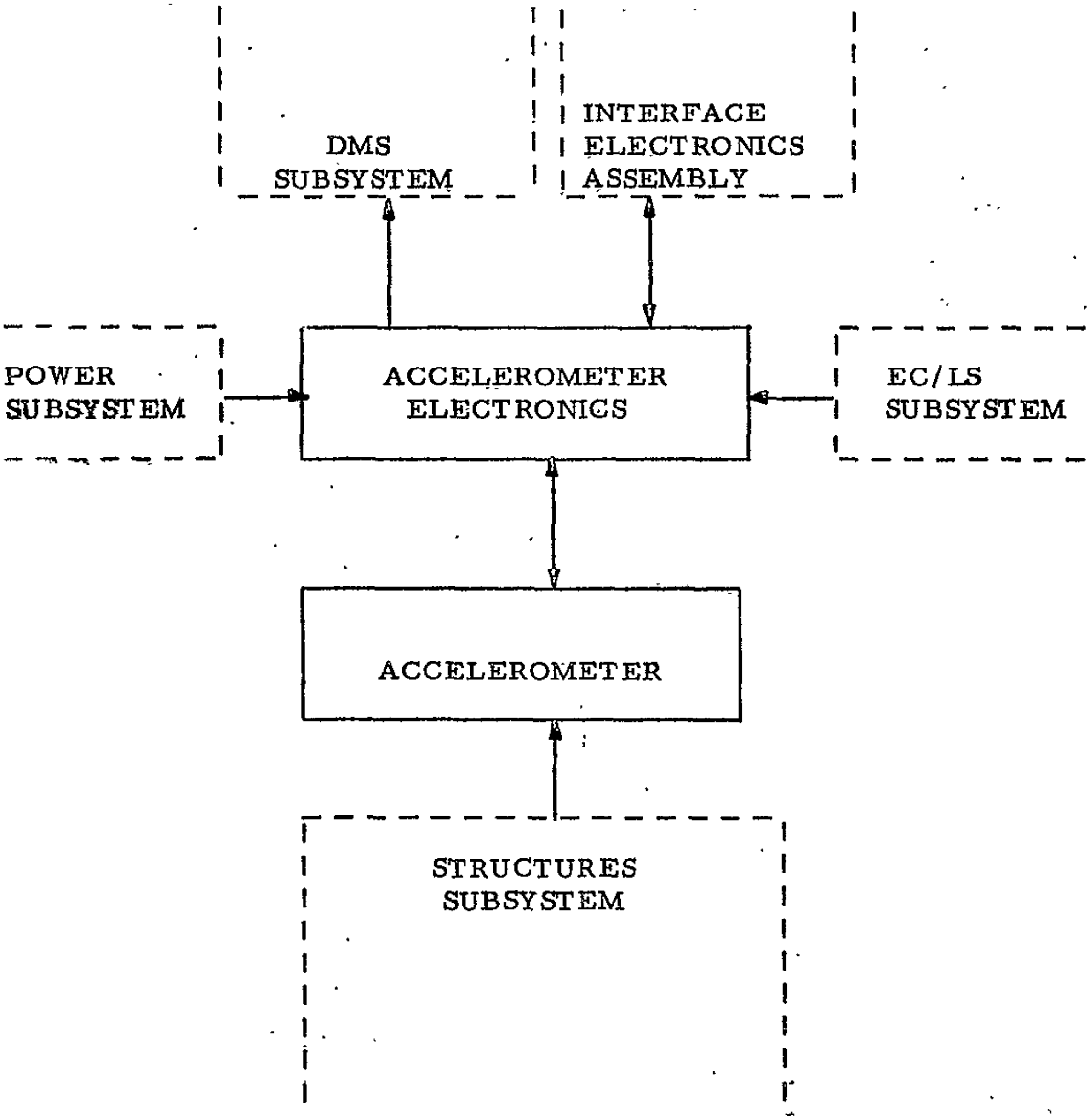


Figure 3-8. LRU Interface Diagram Star Tracker Assembly



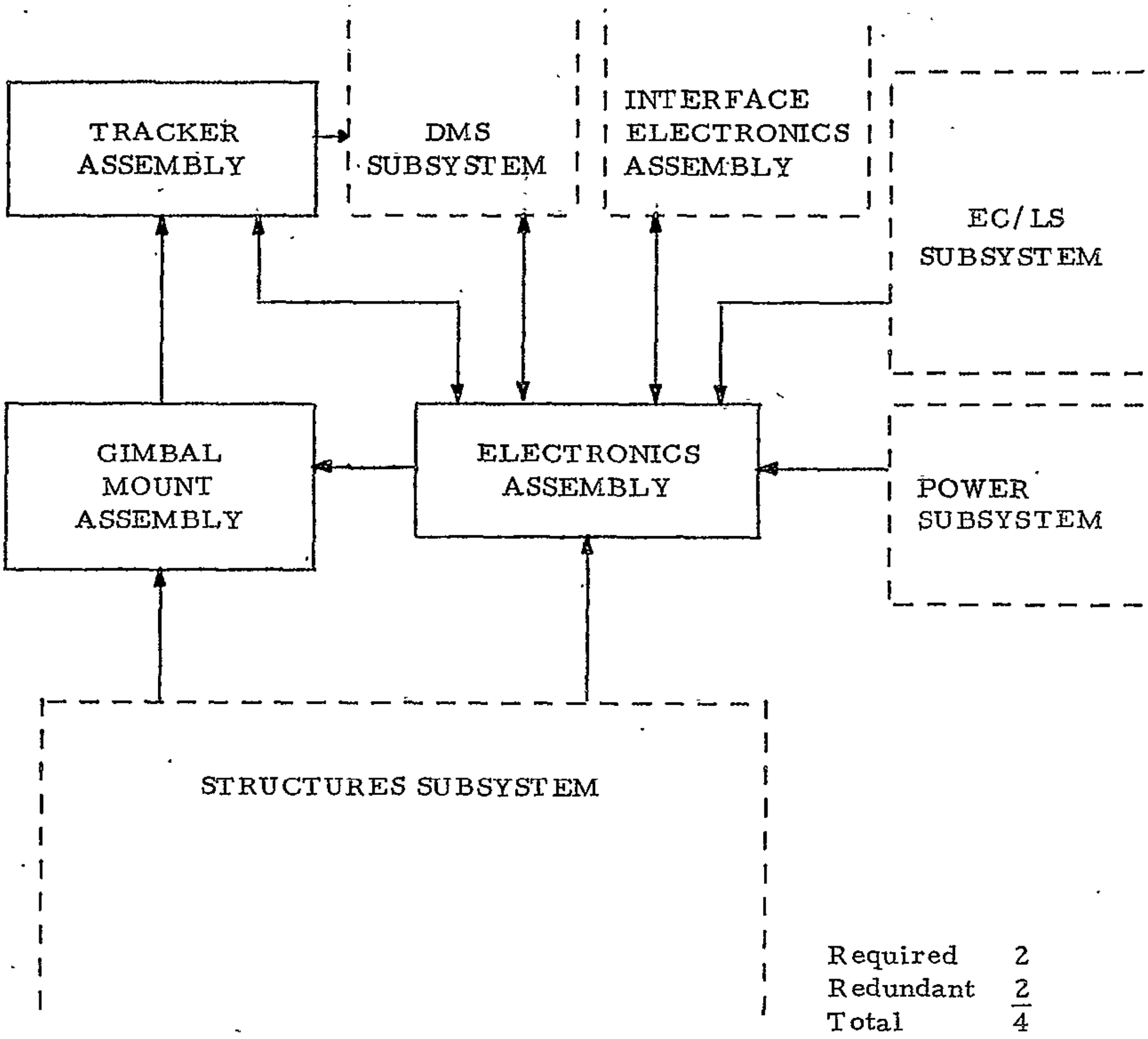
Required	1
Redundant	1
Total	2

Figure 3-9. LRU Interface Diagram Landmark Tracker Assembly



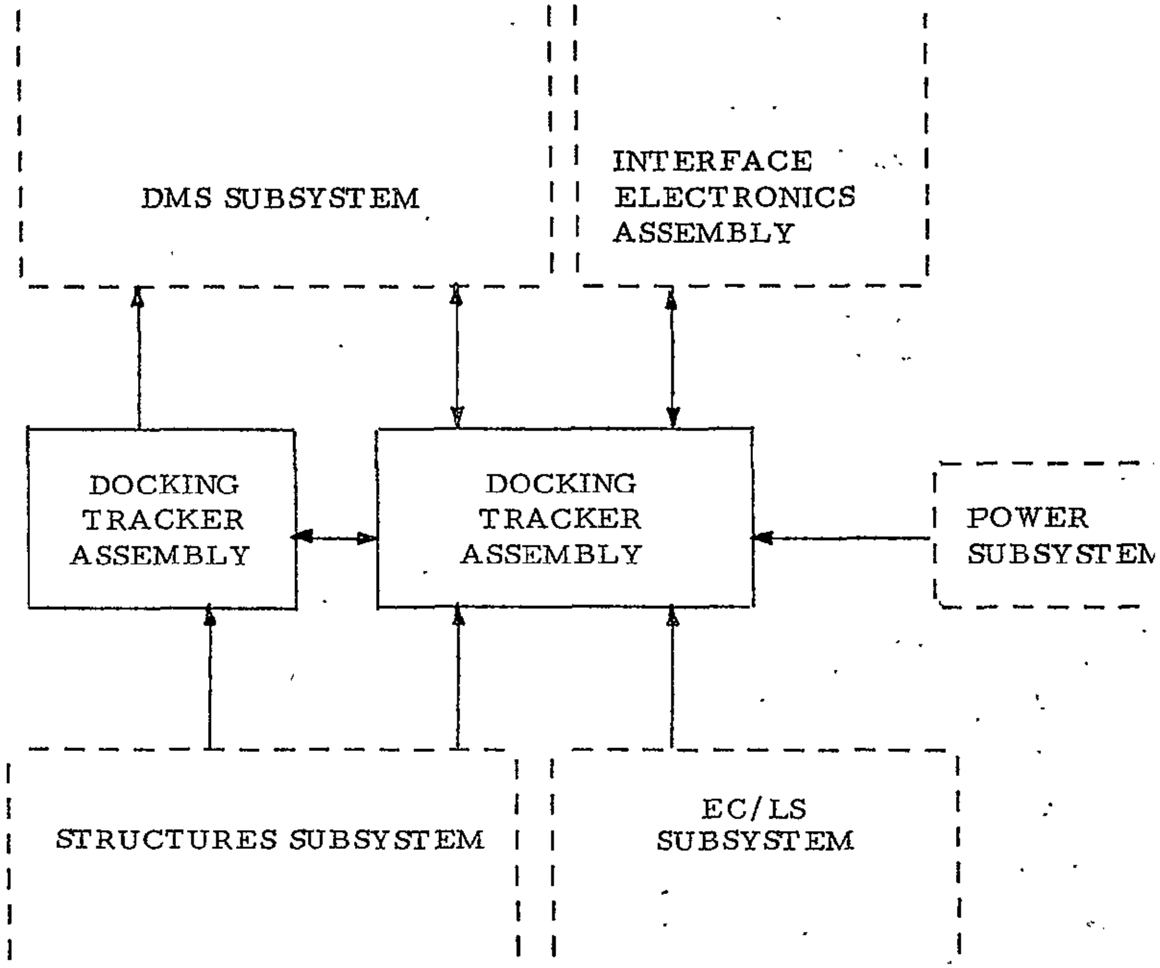
Required	1
Redundant	1
Total	2

Figure 3-10. LRU Interface Diagram Low-G Accelerometer Assembly



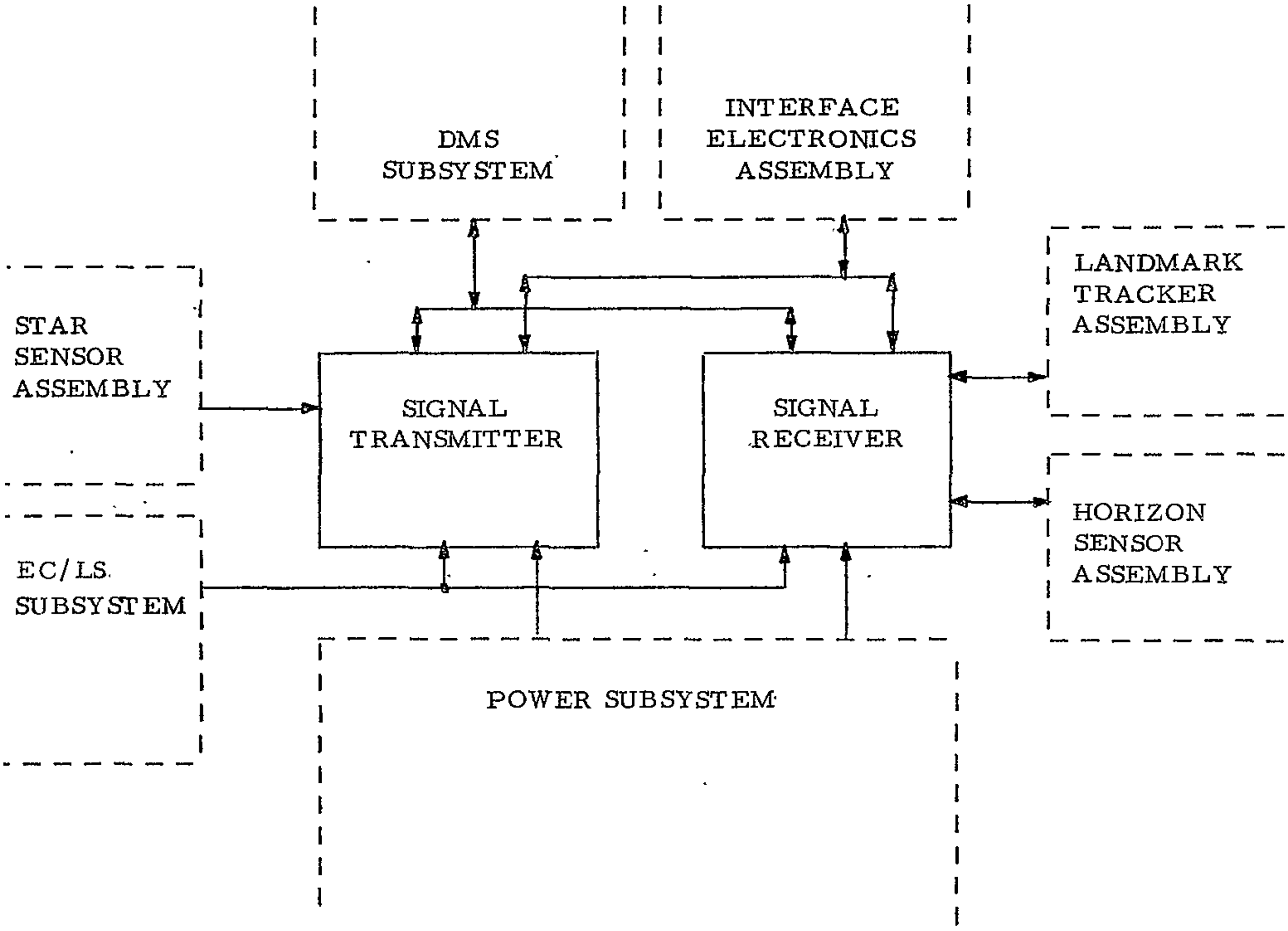
Required	2
Redundant	2
Total	4

Figure 3-11. LRU Interface Diagram Laser Rendezvous Tracker Assembly



Required	7
Redundant	0
Total	7

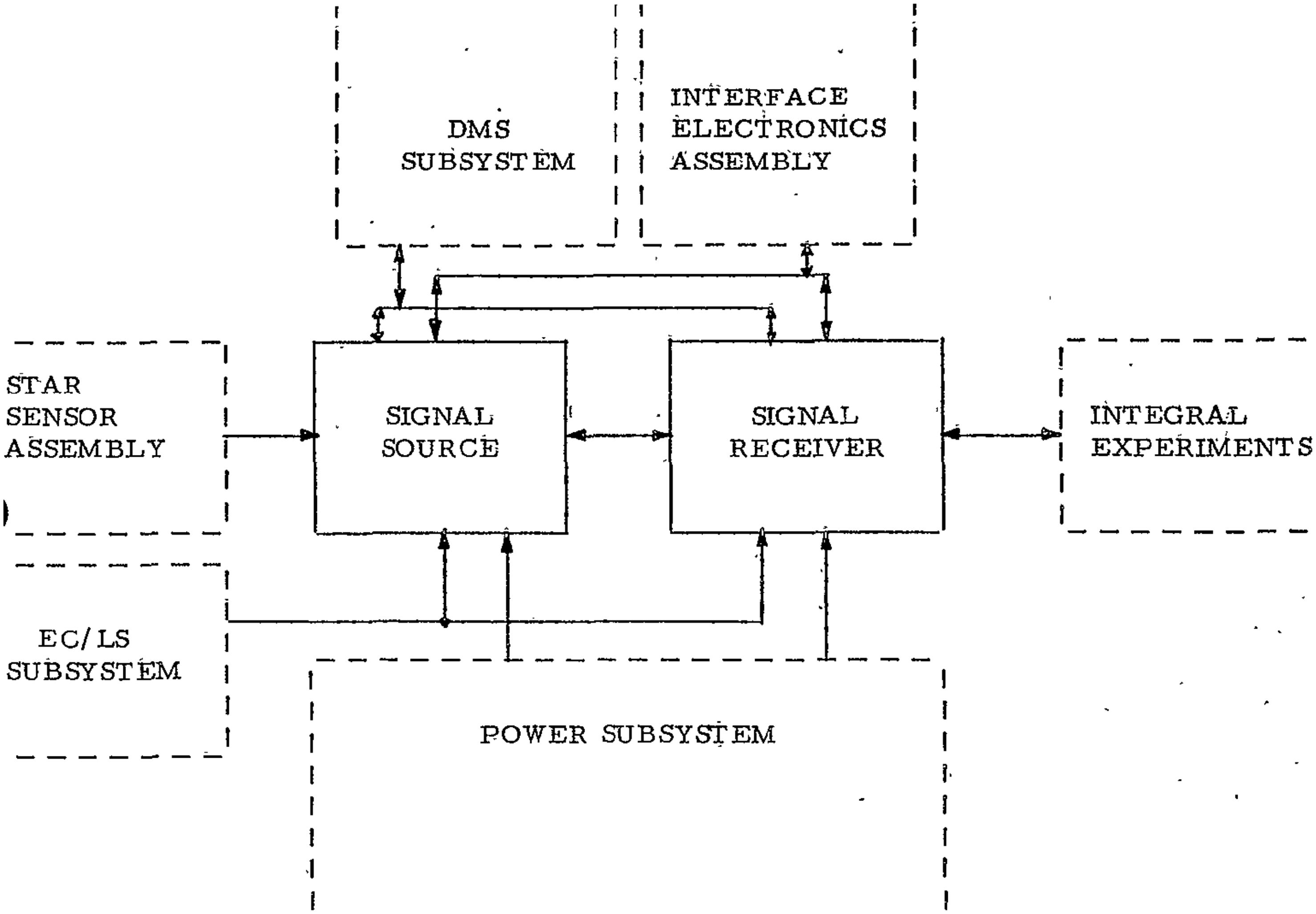
Figure 3-12. LRU Interface Diagram Laser Docking Tracker Assembly



Required	2
Redundant	0
Total	2

NOTE: Mechanical attachment to sensors being aligned is implied but not shown.

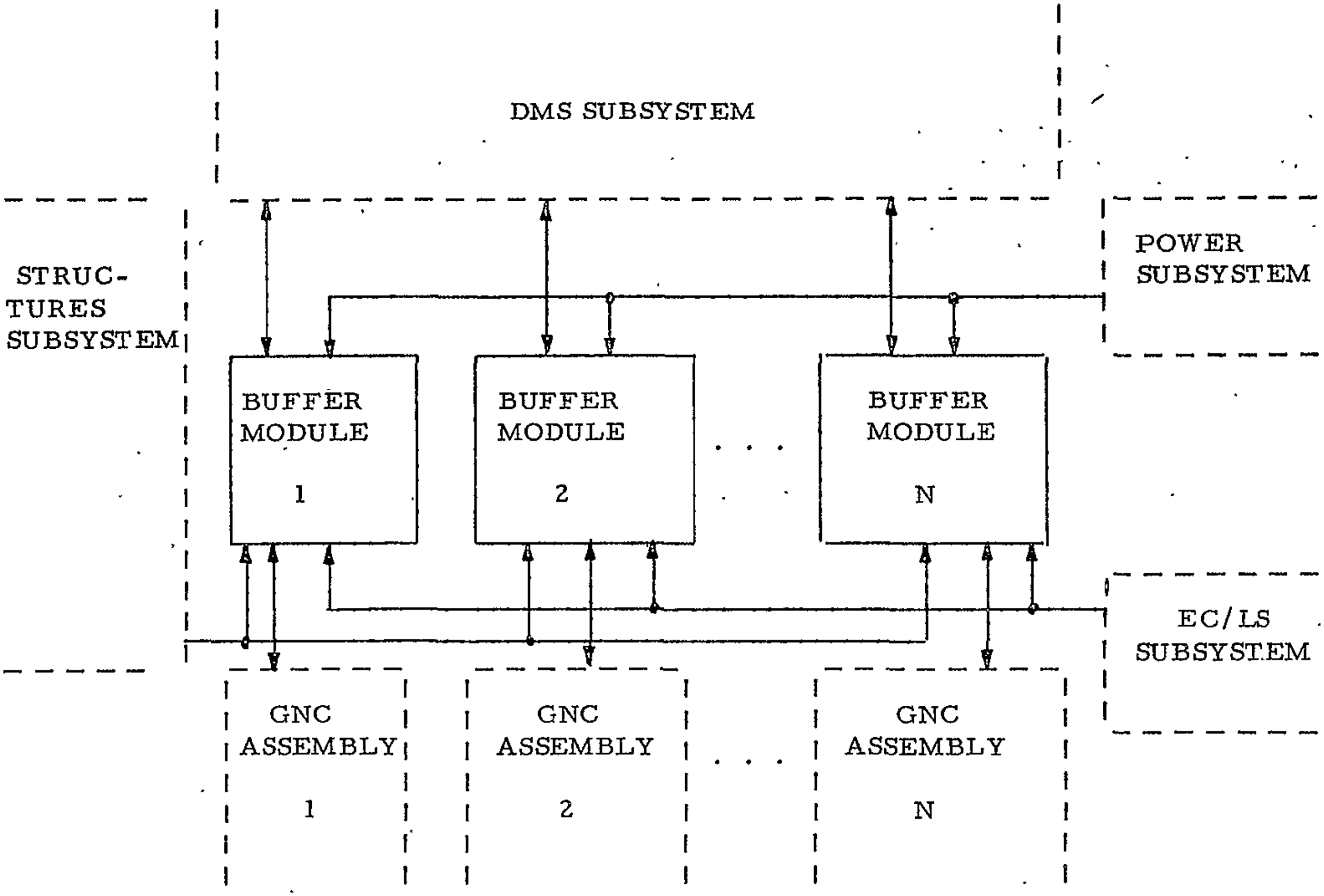
Figure 3-13. LRU Interface Diagram Sensor Alignment Monitor Assembly



NOTE: Mechanical attachment to experiments being aligned is implied, but not shown.

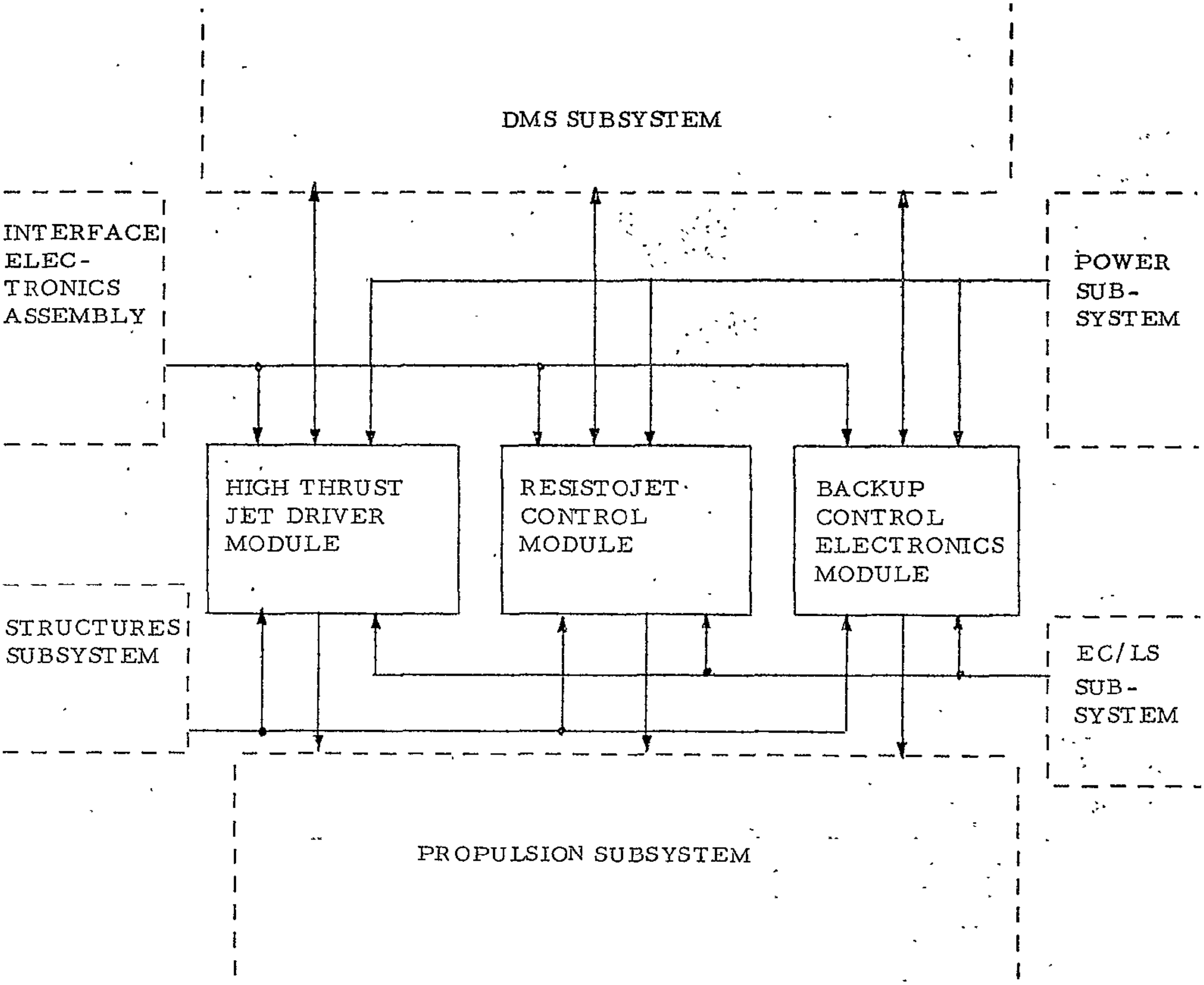
Required	2
Redundant	0
Total	2

Figure 3-14. LRU Interface Diagram Experiment Alignment Monitor Assembly



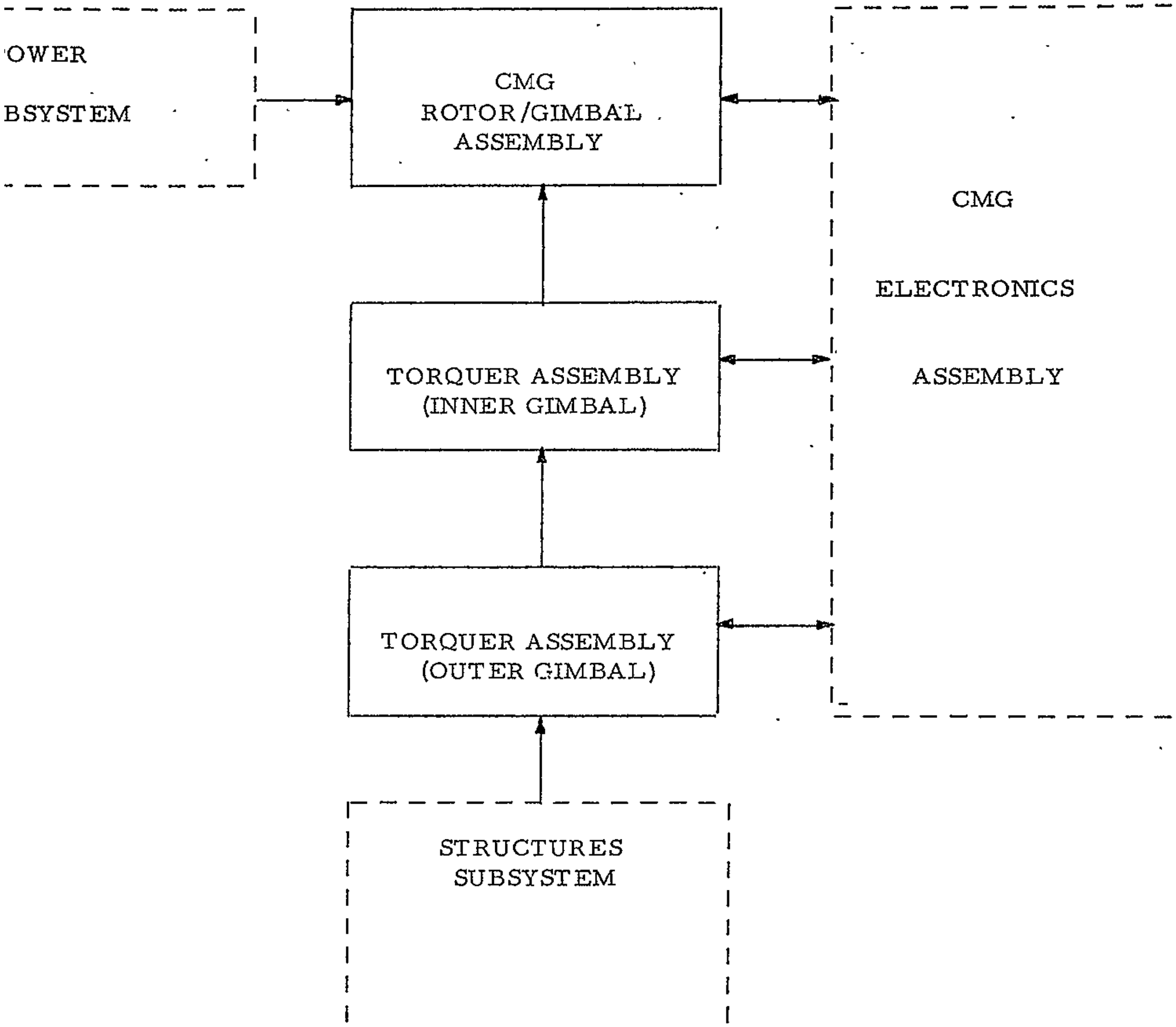
Required	1
Redundant	1
Total	2

Figure 3-15. LRU Interface Diagram Interface Electronics Assembly



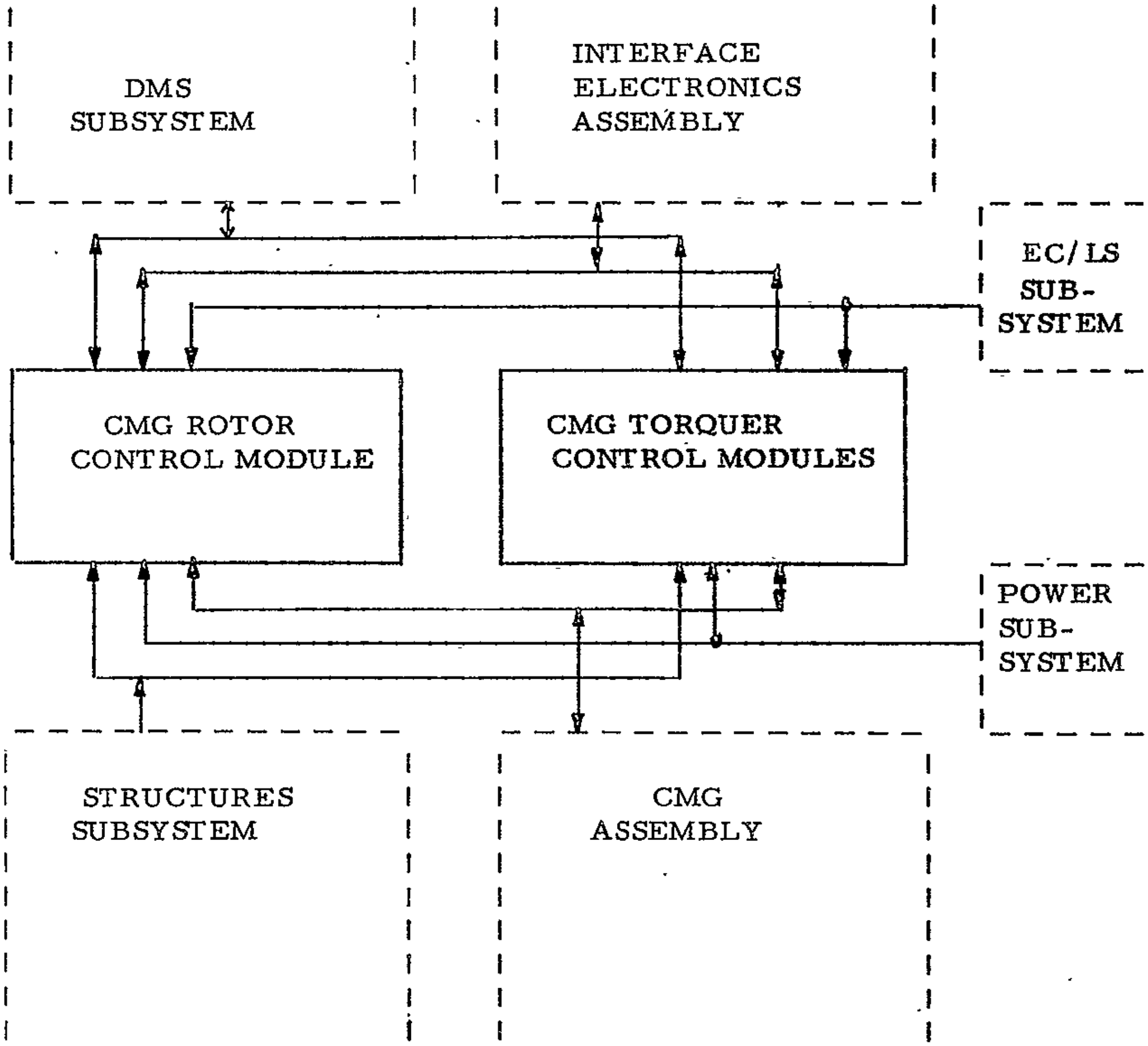
Required
Redundant
Total

Figure 3-16. LRU Interface Diagram Jet Driver Electronics Assembly



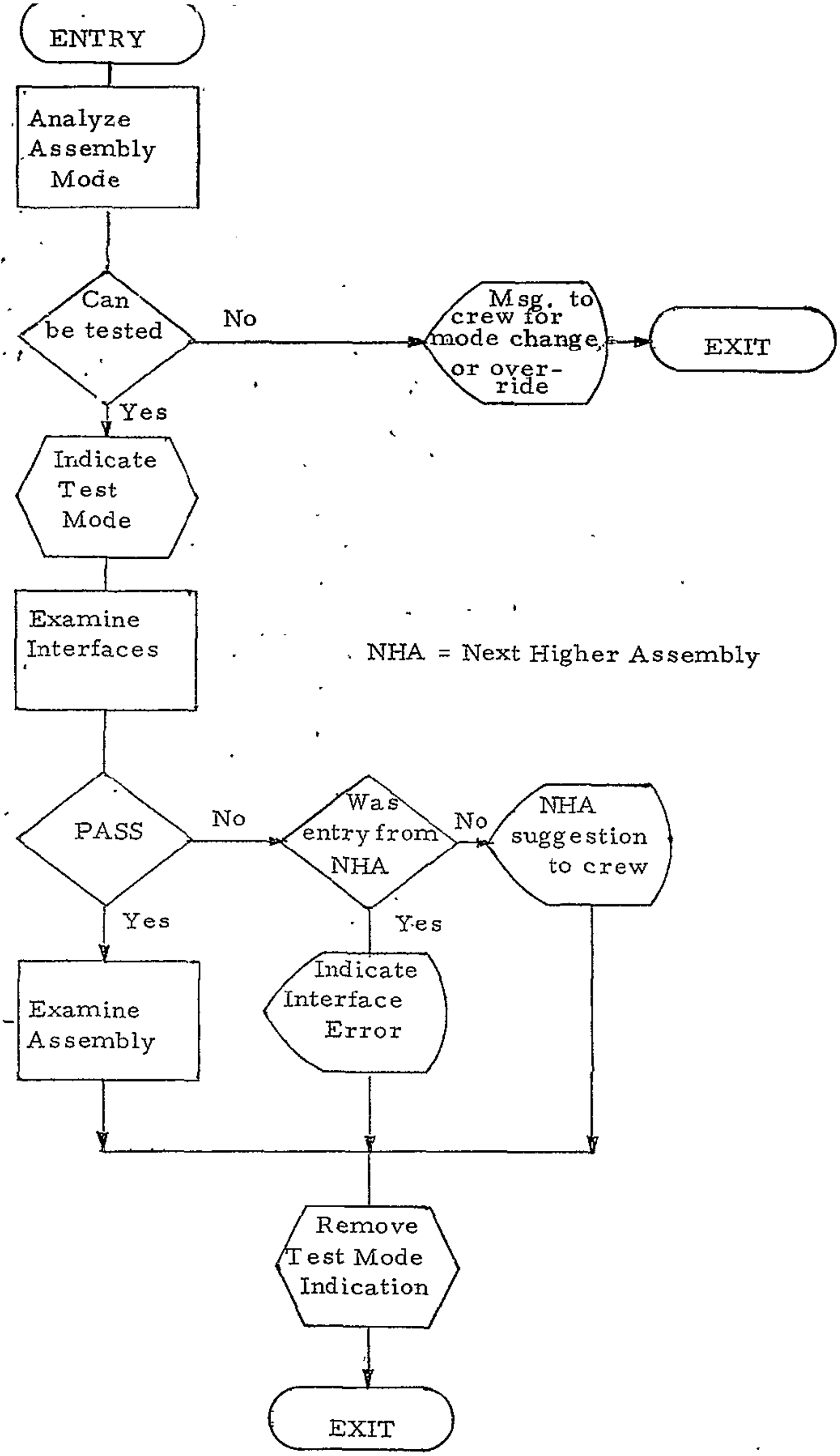
Required	4
Redundant	2
Total	6

Figure 3-17. LRU Interface Diagram CMG Assembly



Required	4
Redundant	2
Total	6

Figure 3-18. LRU Interface Diagram CMG Electronics Assembly



NHA = Next Higher Assembly

Above the LRU level, this will consist of a series of calls (executes) for lower-level modules, each having the same form as shown here.
At the LRU level, this will consist of the actual measurements and stimuli necessary to determine if an LRU failure has occurred, and if so, a message identifying the LRU.

Figure 3-19. General Fault Isolation

Information reflecting the attributes of fault isolation modules for each GNC Subsystem assembly appears in Table 3-1.

In preparing the table, the following considerations were applied to the fault isolation for each assembly:

- Is exclusive control of the assembly under analysis required, thus precluding operational use during fault isolation?
- Is the status of the assembly altered during analysis?
- Is it necessary to make use of data which is managed by application programs?
- Must the application data be altered? This consideration is mutually exclusive of the one immediately above.
- Is it necessary to make use of data maintained by the executive?
- Must the executive data be altered? (mutually exclusive)
- Is it necessary to cycle the assembly through various modes of operation during fault isolation?
- Are stimuli required during analysis?
- Are digital readings used, as contrasted with analog or bi-level?
- Is the sequence in which the LRU's of the assembly are examined important?

Fault isolation examples for the horizon detector, attitude gyro, horizon sensor, star sensor, start tracker, and rendezvous tracker are shown in Figures 3-21 through 3-30.

Table 3-1. Fault Isolation and Periodic Check Requirements

		Attributes of the F.I. Program															
		Exclusive Control	Changes Status	Acquires Apl. Data	Changes Apl. Data	Acquires Exec. Data	Changes Exec. Data	Cycles Assembly Mode	Requires Stimuli	Digital Readings	LRU Sequence Imp.						
ASSEMBLY	Horizon Detector	x	x			x			x								
	Attitude Gyro	x	x			x		x	x		x						
	Star Sensor	x	x			x		x									
	Star Tracker	x	x				x	x		x	x						
	Landmark Tracker	x	x			x		x	x	x	x						
	Low-G Accelero.	x	x	x		x					x						
	Rendezvous Tracker	x	x				x	x	x	x	x						
	Docking Tracker	x	x			x		x	x	x	x						
	Sensor Alignment Monitor	x	x	x		x					x						
	Experiment Alignment Monitor	x	x	x		x			x	x							
	Interface Electronics	x	x			x			x		x						
	Control Moment Gyro					x				x	x						
	CMG Electronics	x	x			x		x	x								

3.1.2.3 Trend Analysis Function

Trend analysis is used on selected GNC parameters for the detection of degraded performance or impending failure.

Input to the trend analysis function consists of RDAU interrupts, measurements, the Pacer, and real time. Output consists of caution and warning displays, trend table data, and fault isolation parameters.

The parameter is measured and the time of measurement is obtained. These values are combined with a pre-determined number of previous values to form a set of X-Y coordinates which could be plotted on a graph depicting parameter value versus time. An exponential smoothing of the data is performed and extrapolation estimates are calculated to determine if the trend is approaching a caution or warning condition; if prior to the next measurement cycle the parameter will be out of limits, or that a failure may occur for the LRU between resupply event i and resupply event i+1.

Trend analysis modules exist for each of the following GNC Subsystem assemblies:

- Attitude Gyro
    - gyro case temperature
    - gyro heater voltage
- Accelerometer
    - accelerometer temperature
    - accelerometer heater voltage
- Laser Rendezvous Tracker
    - tracker transmitter power monitor
    - tracker receiver energy monitor
- Laser Docking Tracker
    - tracker transmitter power monitor
    - tracker receiver energy monitor

- Jet Driver
    - driver inputs
- Control Moment Gyro
    - spin power monitor
    - vibration monitor
    - bearing temperature monitor
    - vacuum monitor

The following trend analysis methods are utilized by GNC checkout:

- Integration, with respect to time over a fixed time interval, and comparison of the integral with a fixed limit. This method is employed with the attitude gyro and accelerometer assemblies.

- An average of N samples taken during a particular phase of operation, with the average compared to that acquired previously. This method is employed with the laser rendezvous tracker and docking tracker assemblies.

- A count of the number of operations over a fixed time interval and comparison with a fixed limit. This method is employed with the jet driver assembly.

- Periodically sample for a time interval which is small compared to the period. This method is employed with the control moment gyro assembly. The samples are averaged, adjusted for trend, and used to calculate the estimated time of failure, if any.

Figures 3-31 through 3-34 contain logic flowcharts for the time integration, moving average, operations count, and periodic sample methods of trend analysis employed for the GNC Subsystem.

3.1.2.4 Reconfiguration Function

The reconfiguration function keeps track of the use of primary and redundant assemblies by using symbolic assembly identification. This implies that the application programs reference assemblies using the same symbology.

Inputs consist of the symbolic identity of the failed LRU, the configuration table, and the mode table. Identification of the LRU may come from the crew, instead of from the fault isolation function.

Outputs consist of changes to the configuration table, changes to the mode table, crew displays, mode commands, and parameters to the calibration function.

Information processing consists of changing the modes of both the replaced and the replacement assemblies, and updating the configuration table to show the relationship with the next higher and next lower assemblies. The interchanged assemblies are commanded to change modes as appropriate; and the mode table is changed to reflect the status in preparation for future fault isolation activities.

The reconfiguration function of GNC Checkout is concerned with alterations to the GNC subsystem, and the data base alterations necessary to track these changes. Therefore, a combination of mode commands and table maintenance activities are involved. The function involves extensive use of executive services for data base management, while utilizing bi-level and digital stimulus points in order to activate/de-activate the LRU's involved in reconfiguration.

If the spare is installed, reconfiguration can be accomplished automatically. If a spare is not available, the status of the containing assembly is altered to reflect the fact that it is disabled. If the spare is on board, but requires crew action, notification of a failure rate monitor may take place in order to ensure that the repair rate will exceed the failure rate.

The logic flow for the reconfiguration function is shown in Figure 3-35.

3.1.2.5 Calibration Function

Calibration may be employed periodically after repair, or as a result of replacing a failed assembly. The techniques involved are unique to the individual assemblies, and in some cases involve the acquisition of data managed by application programs.

Input to the calibration function consists of parameters supplied by the crew, information passed by the reconfiguration function, calibration tables, and operational data. Output consists of changes to the operational data tables and crew communication.

Information processing includes the employment of other functional GNC assemblies for reference purposes, and changing the calibration references for application programs. As an example, a reading of 3.27 volts may correspond to a zero degree reference for the new assembly; whereas for the assembly which failed, a reading of 2.98 volts was the zero reference for the replaced assembly.

The calibration function is concerned with the data base management involved when an LRU is replaced by crew action, as well as the stimuli and crew interaction which may be involved in actual calibration of certain GNC LRU's. The calibration function is, therefore, used during replace operations; whereas the reconfiguration function discussed above is concerned with remove operations.

The calibration function may be invoked by the crew, or automatically by the reconfiguration function for certain installed spares.

The logic flow for the calibration function is shown in Figure 3-36.

3.1.3 FLOWCHARTS

This section contains flowcharts of typical Guidance, Navigation, and Control Subsystem Checkout Programs.

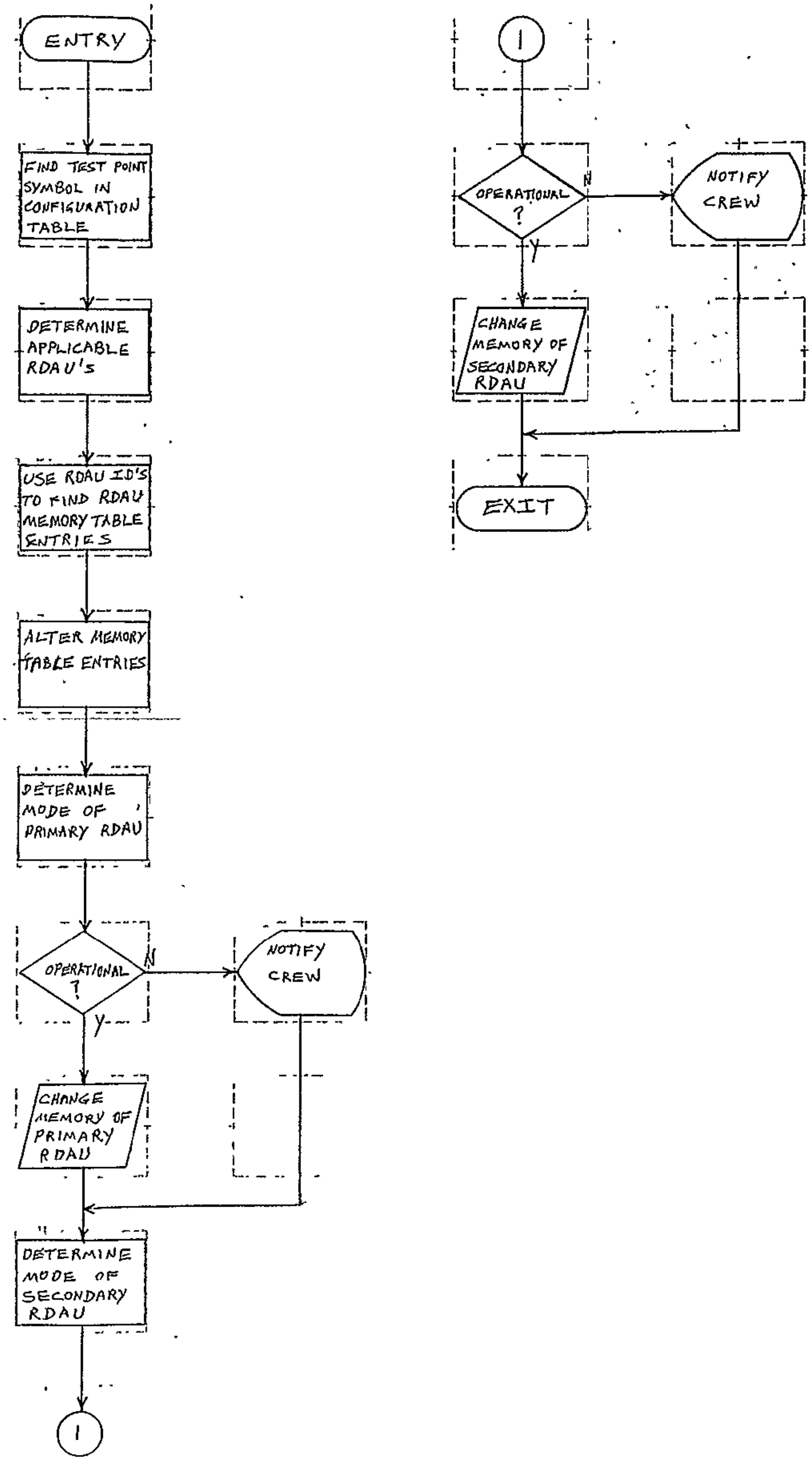


Figure 3-20. Fault Detection Control

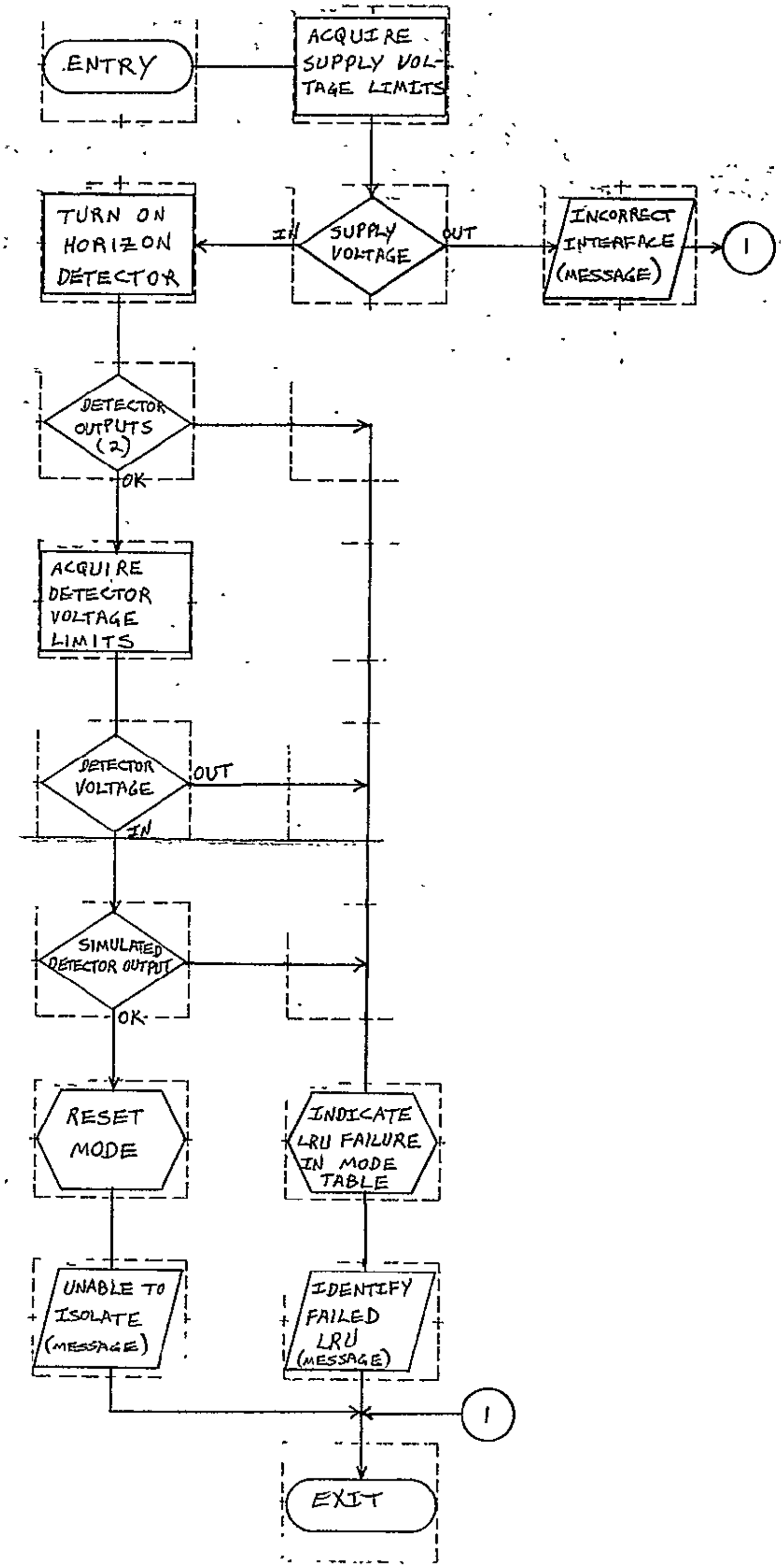
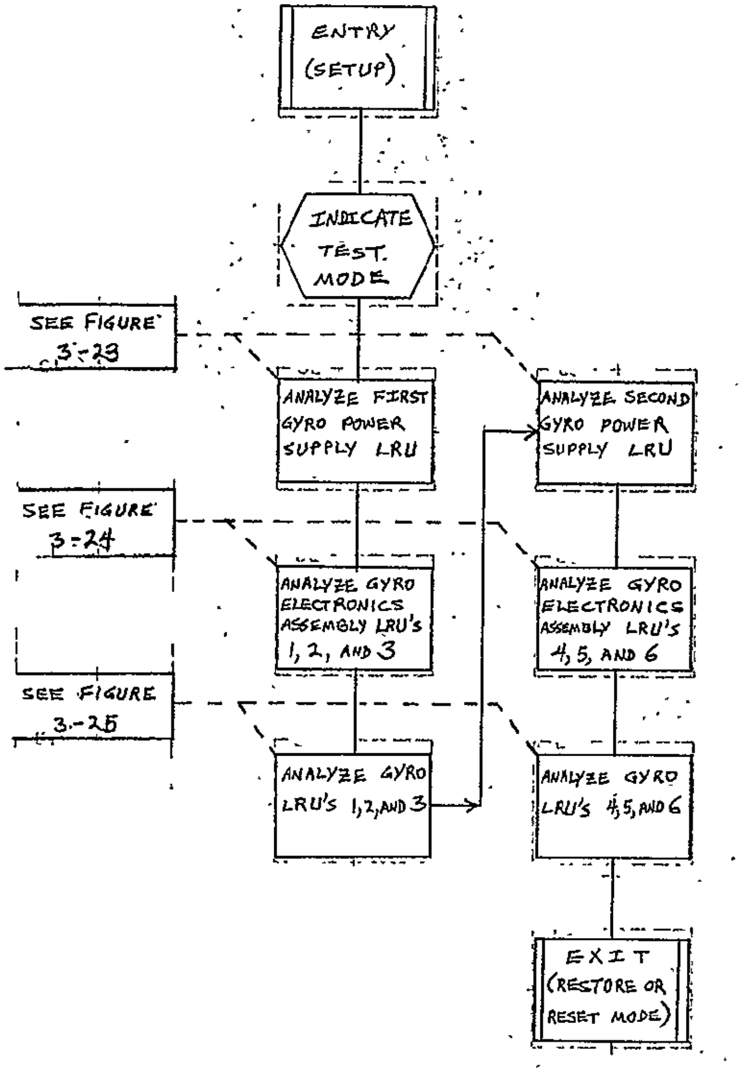


Figure 3-21. Horizon Detector F.I.



igure 3-22. Attitude Gyro Assembly F.I.

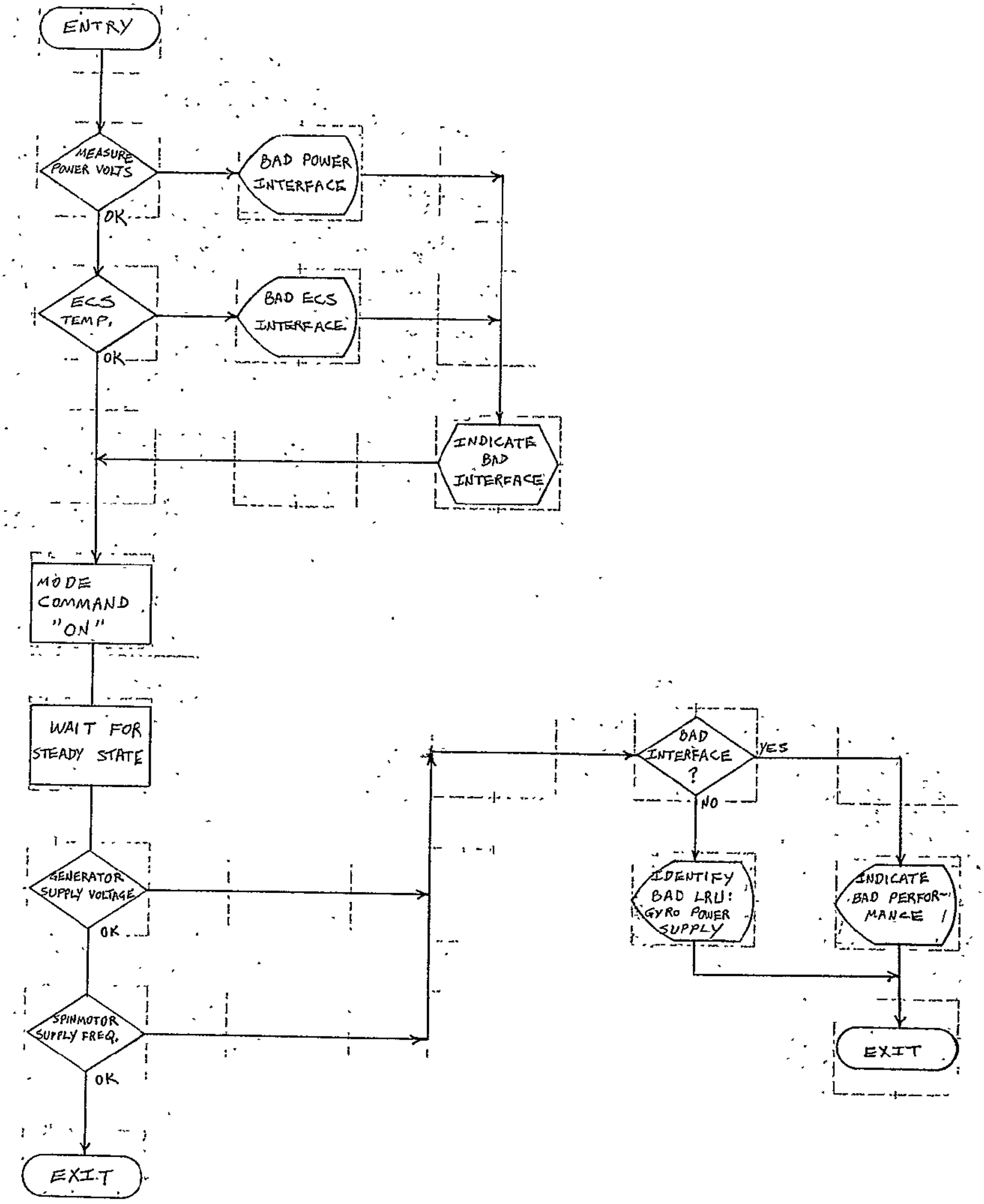


Figure 3-23. Gyro Power Supply F.I.

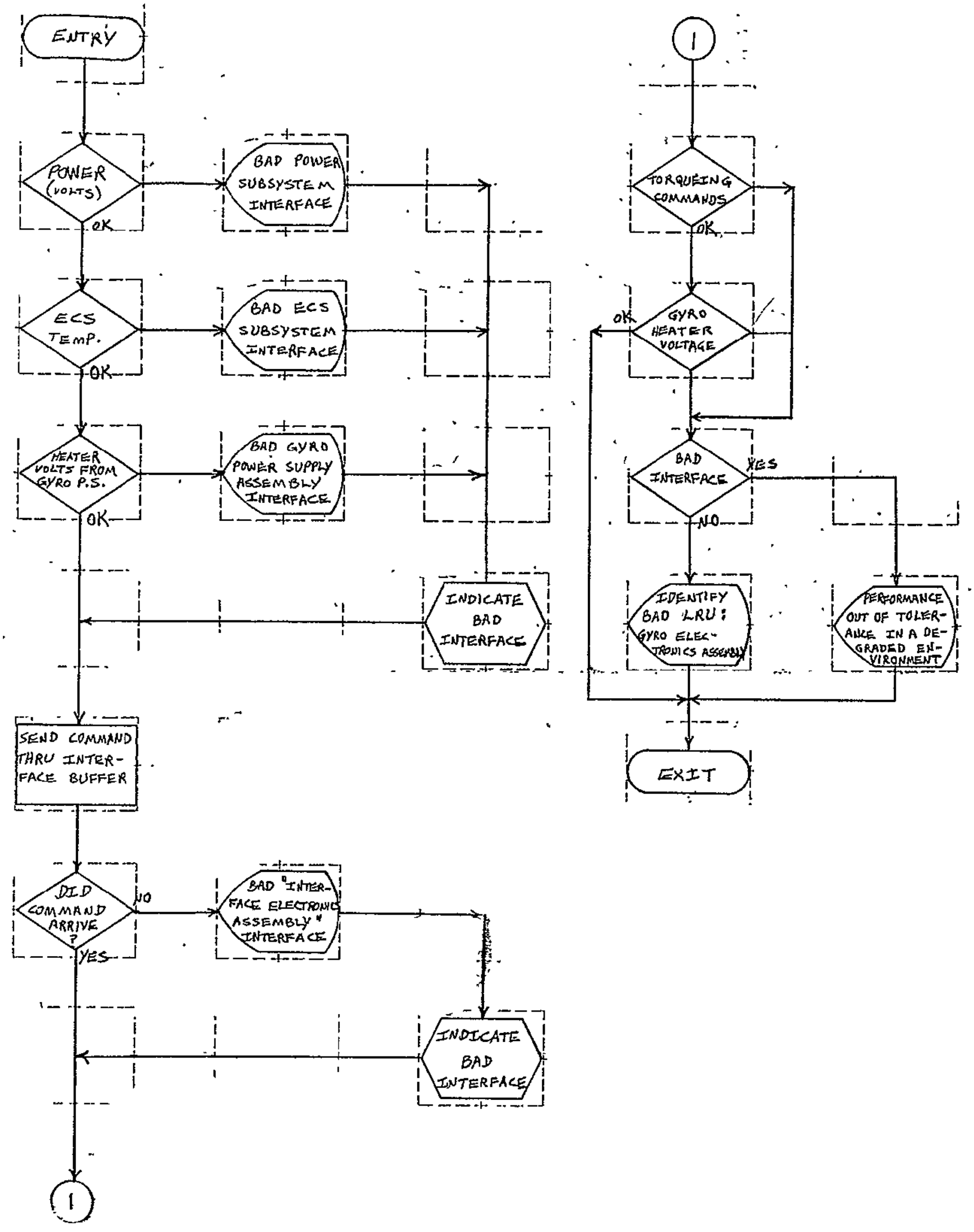


Figure 3-24. Gyro Electronics Assembly F.I.

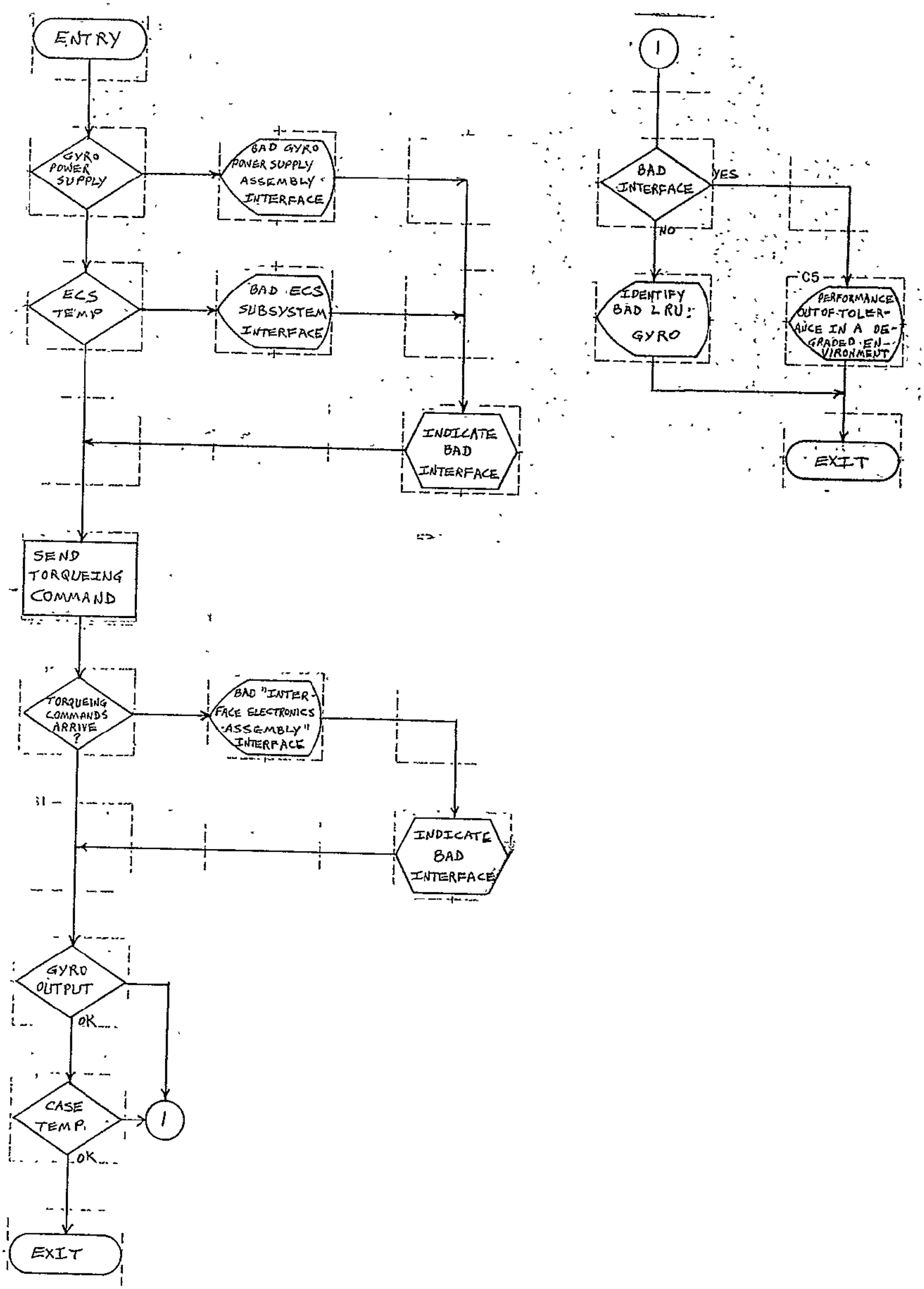
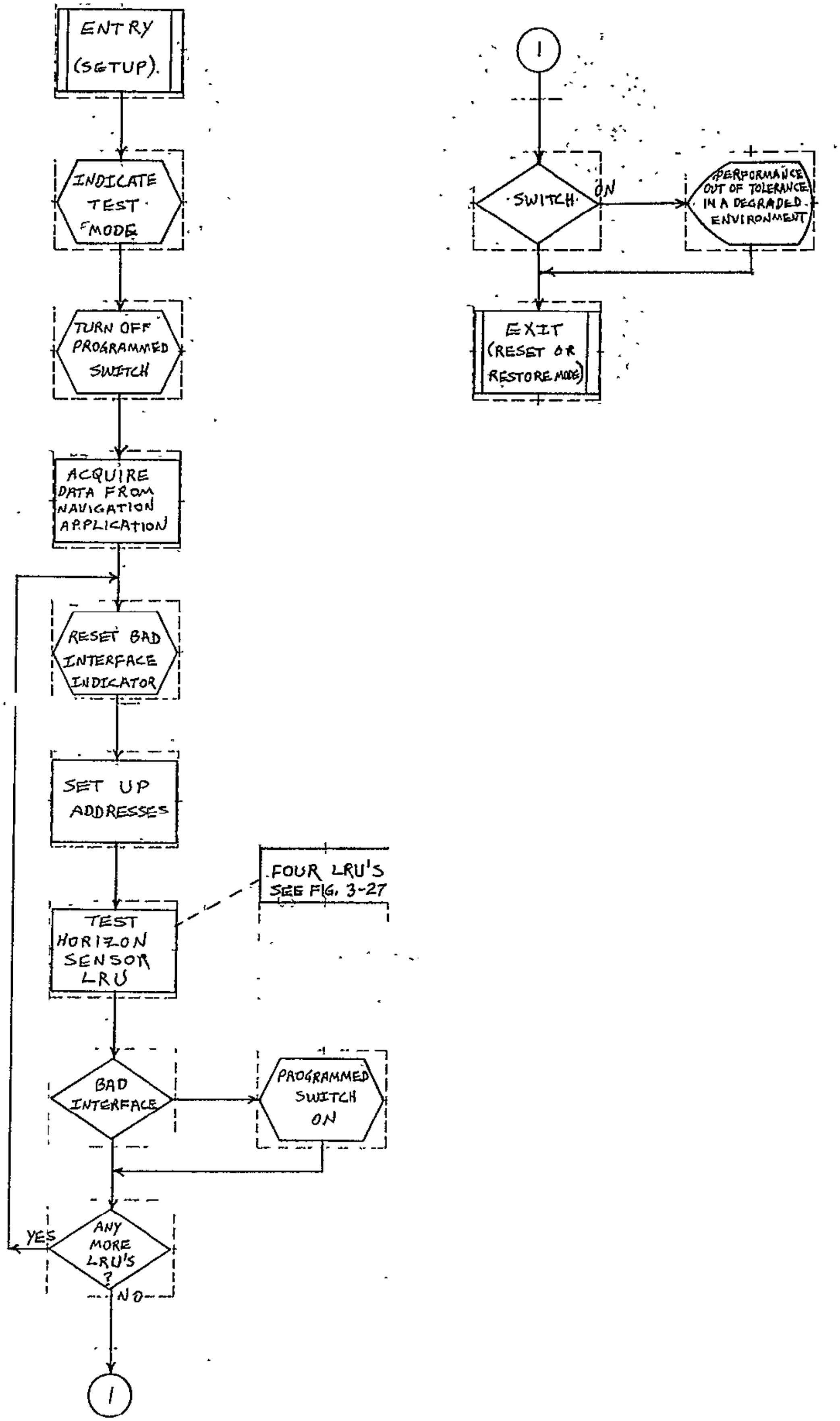


Figure 3-25. Gyro F.I.



igure 3-26. Horizon Sensor Assembly F.I.

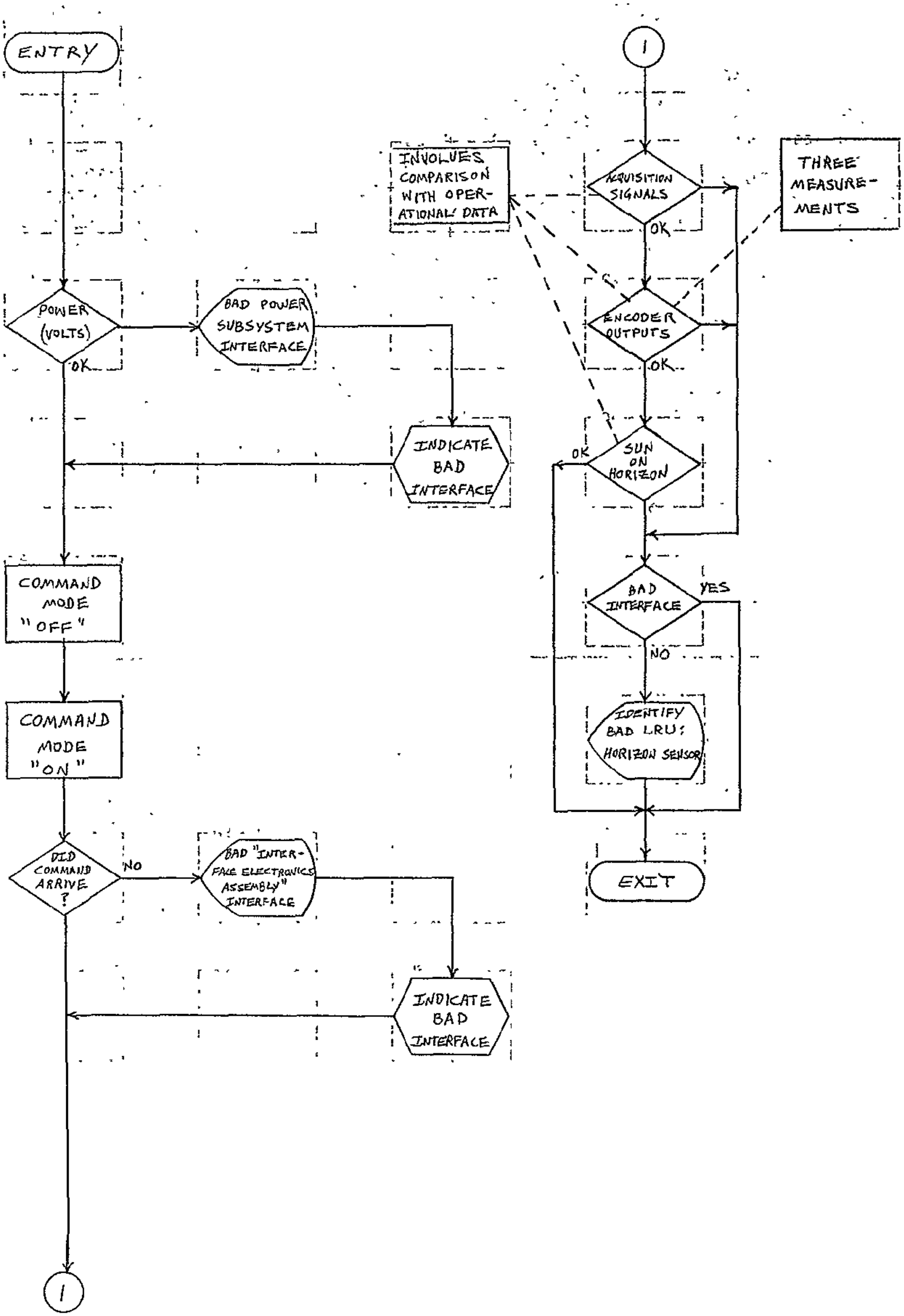


Figure 3-27. Horizon Sensor LRU F.I.

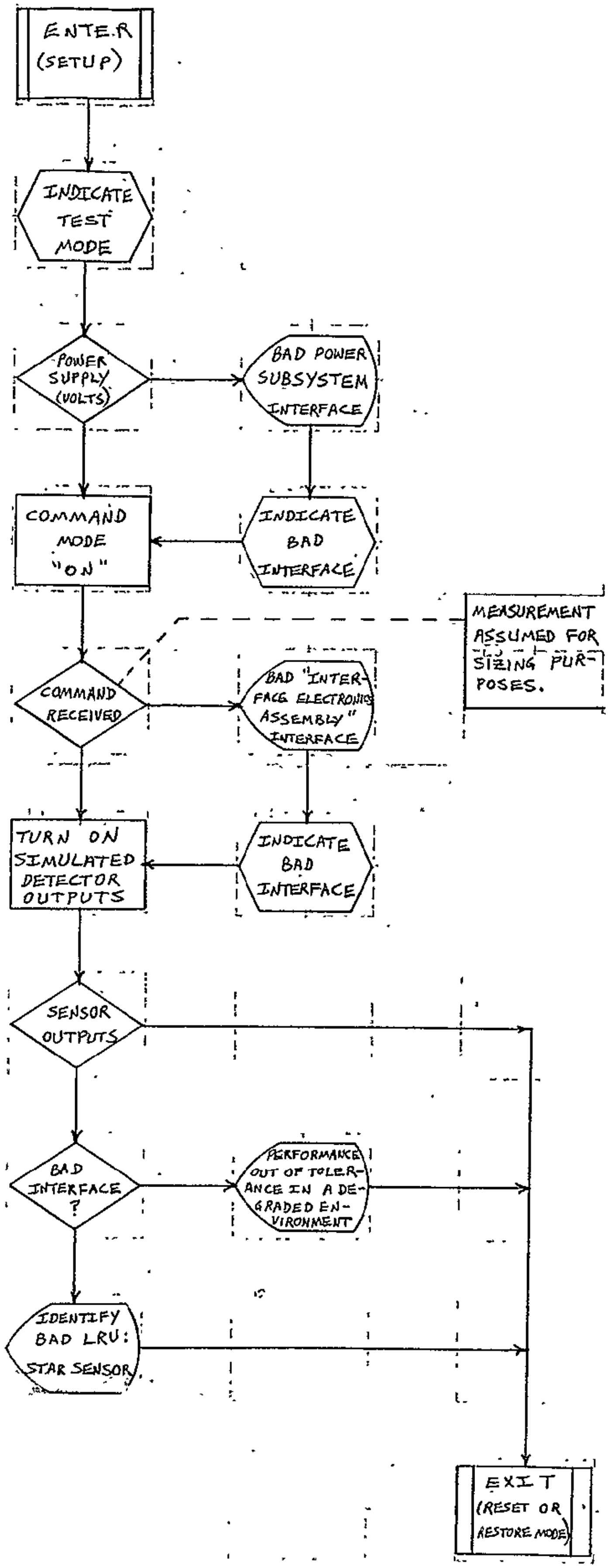


Figure 3-28. Star Sensor Assembly F.I.

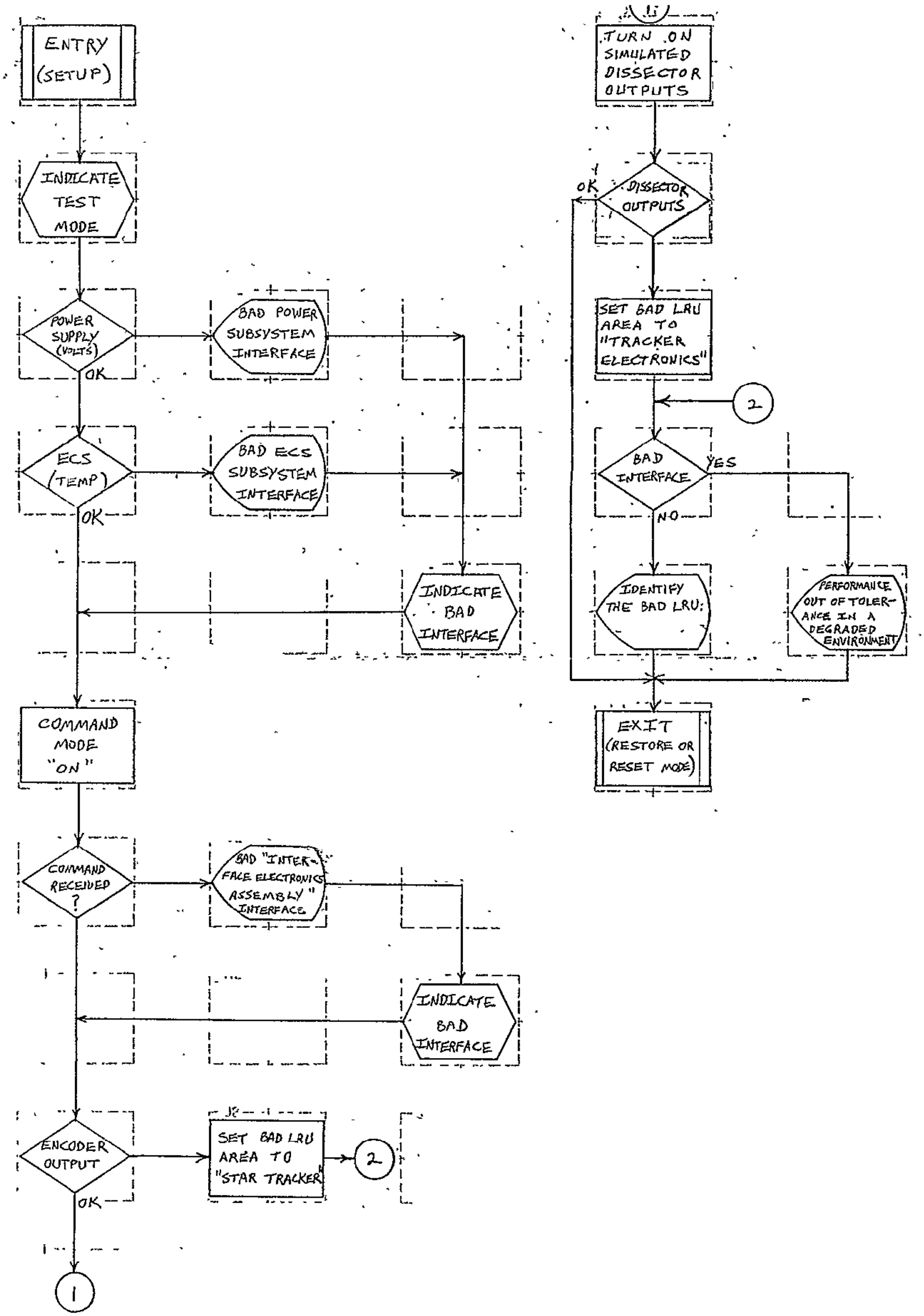


Figure 3-29. Star Tracker Assembly F.I.

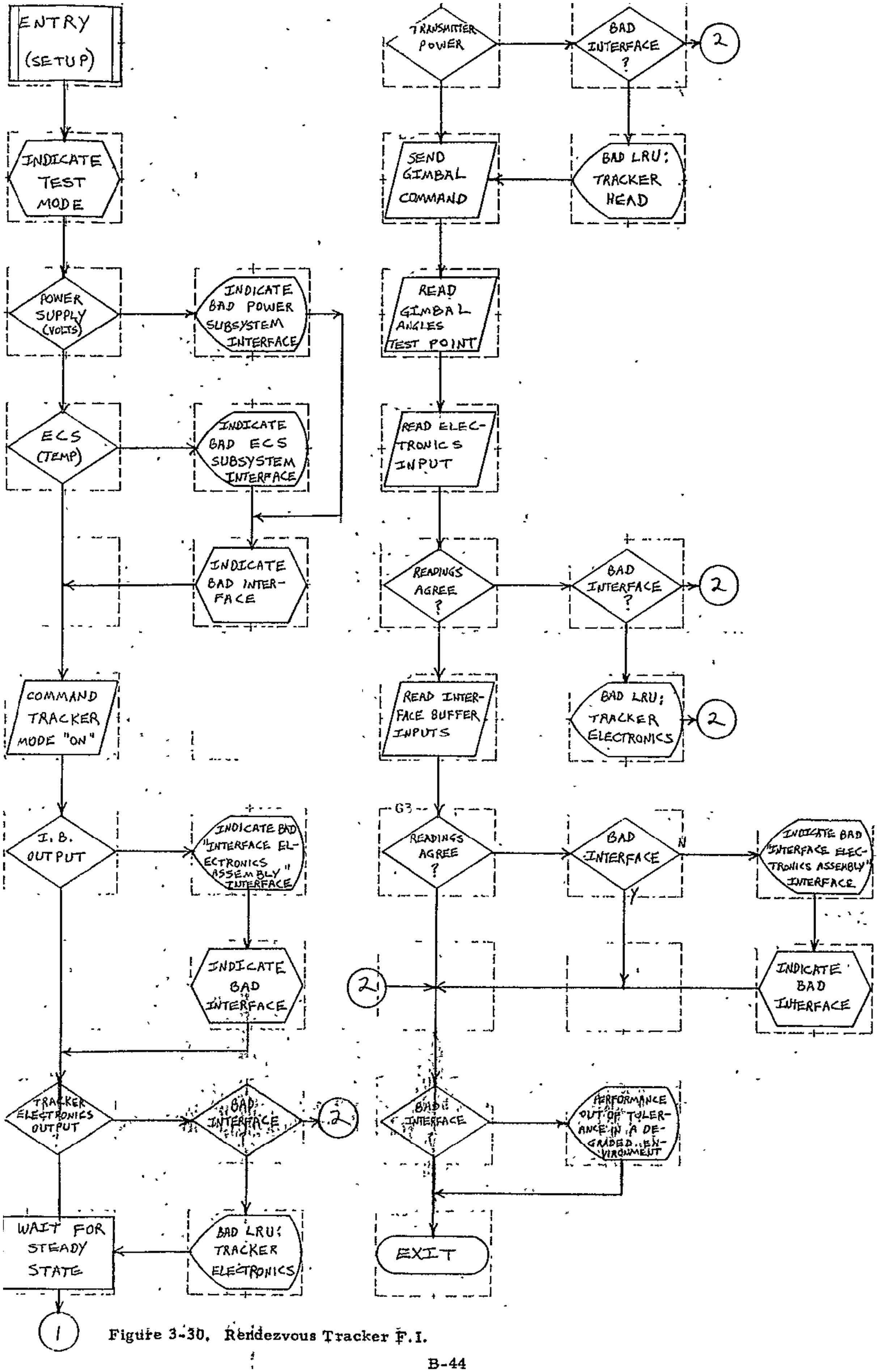


Figure 3-30. Rendezvous Tracker F.I.

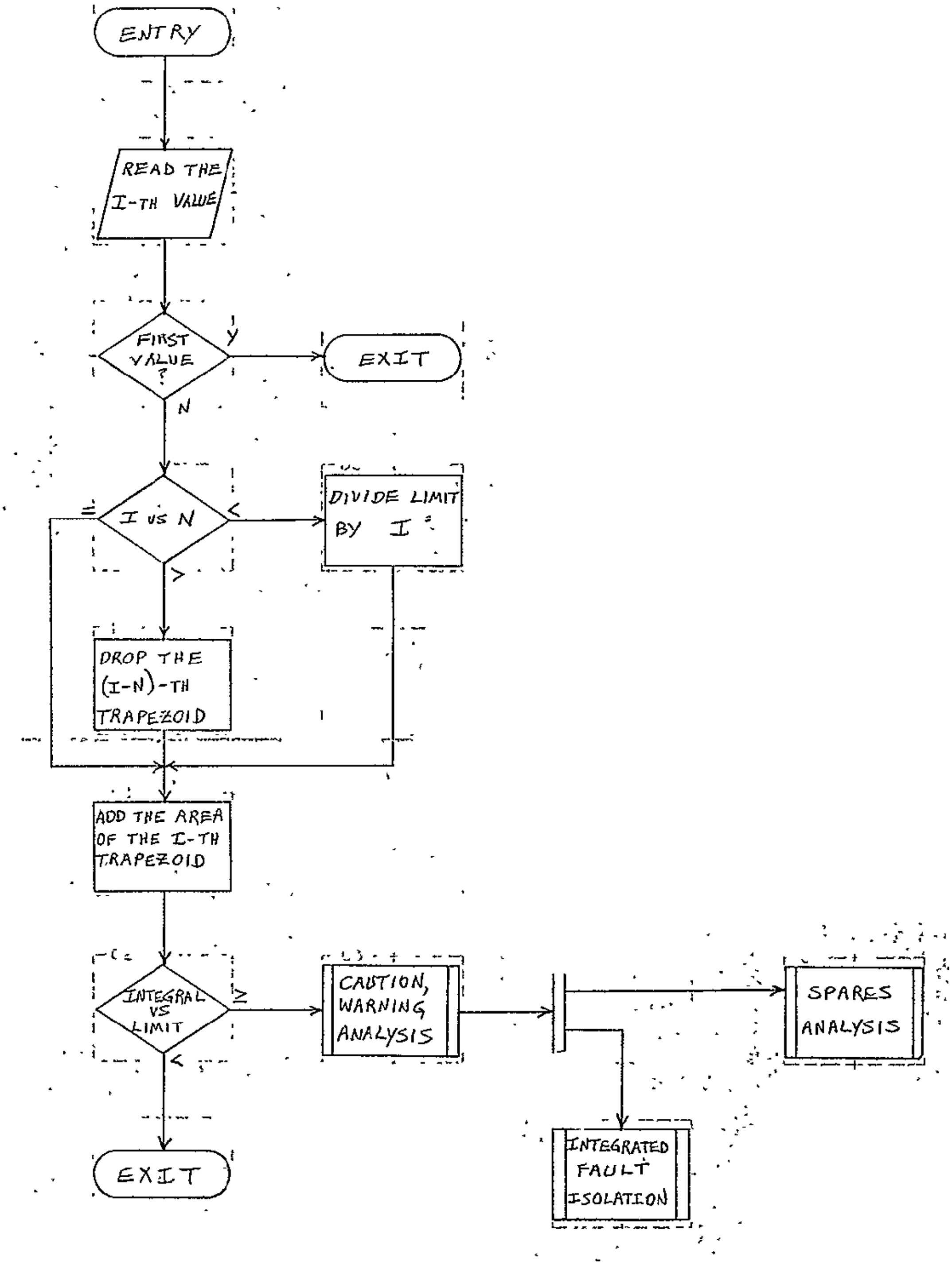


Figure 3-31. Time Integration T.A.

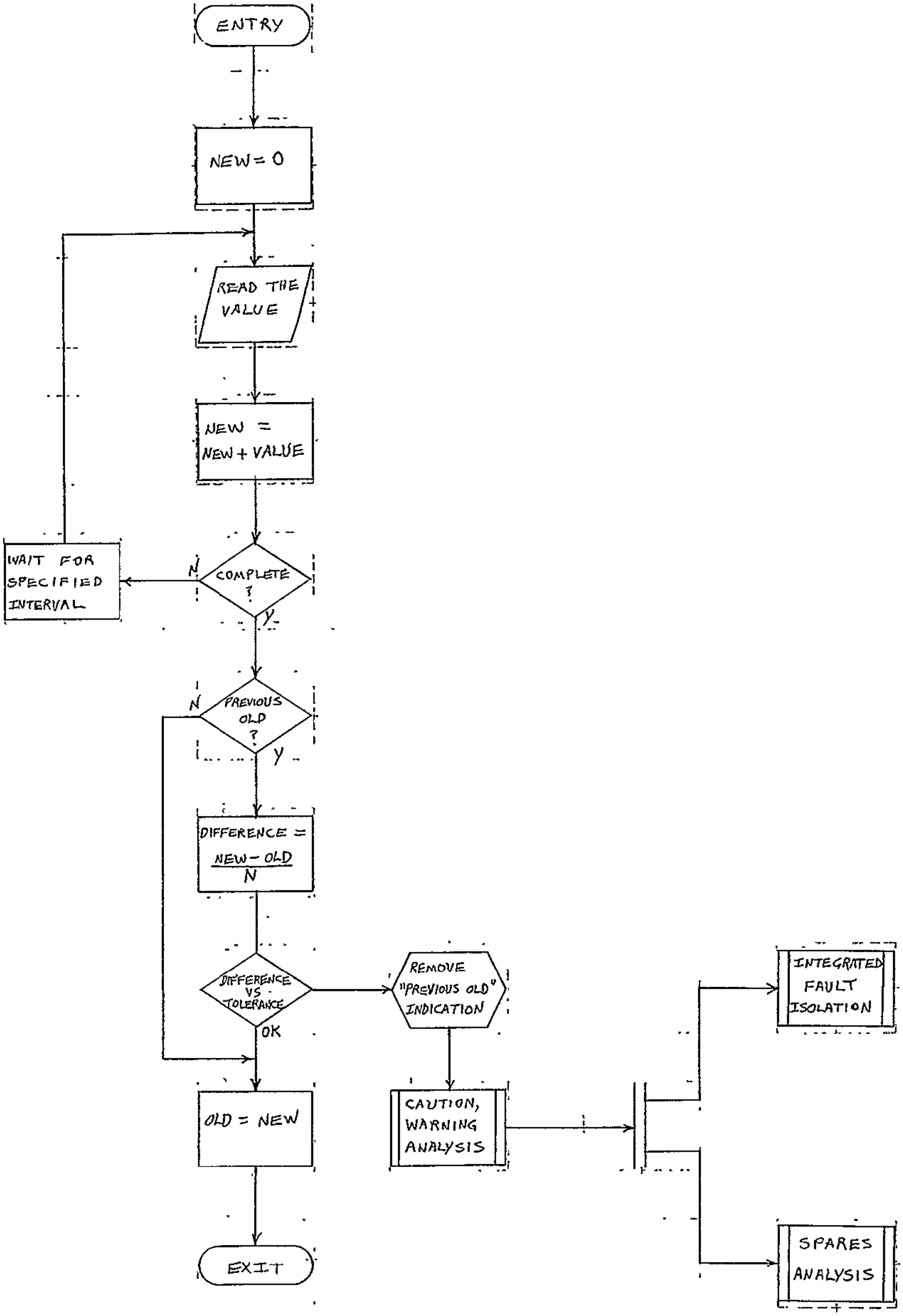


Figure 3-32. Moving Average T.A.

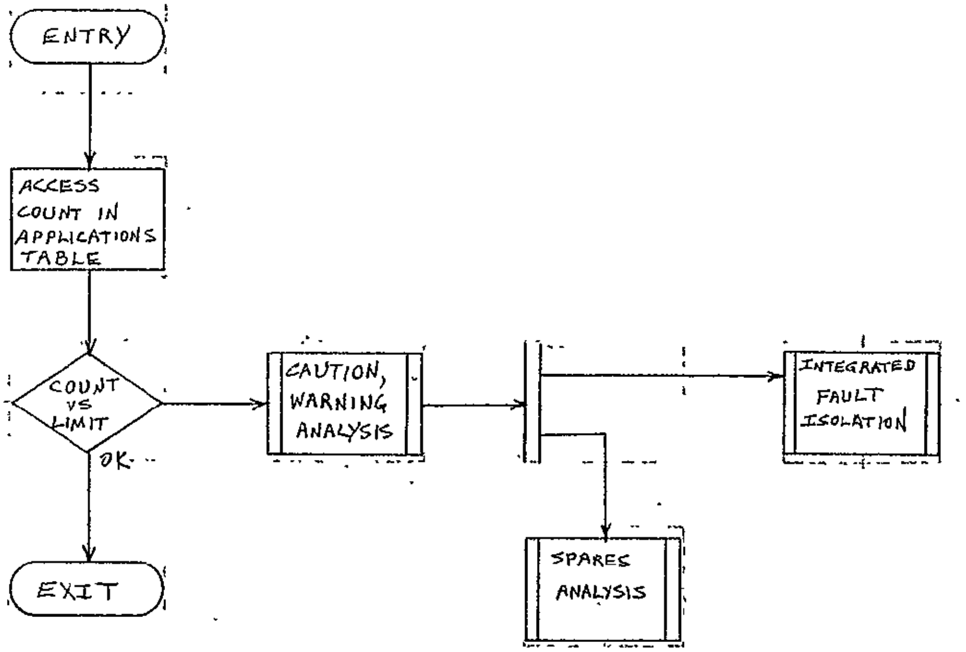


Figure 3-33. Operations Count T.A.

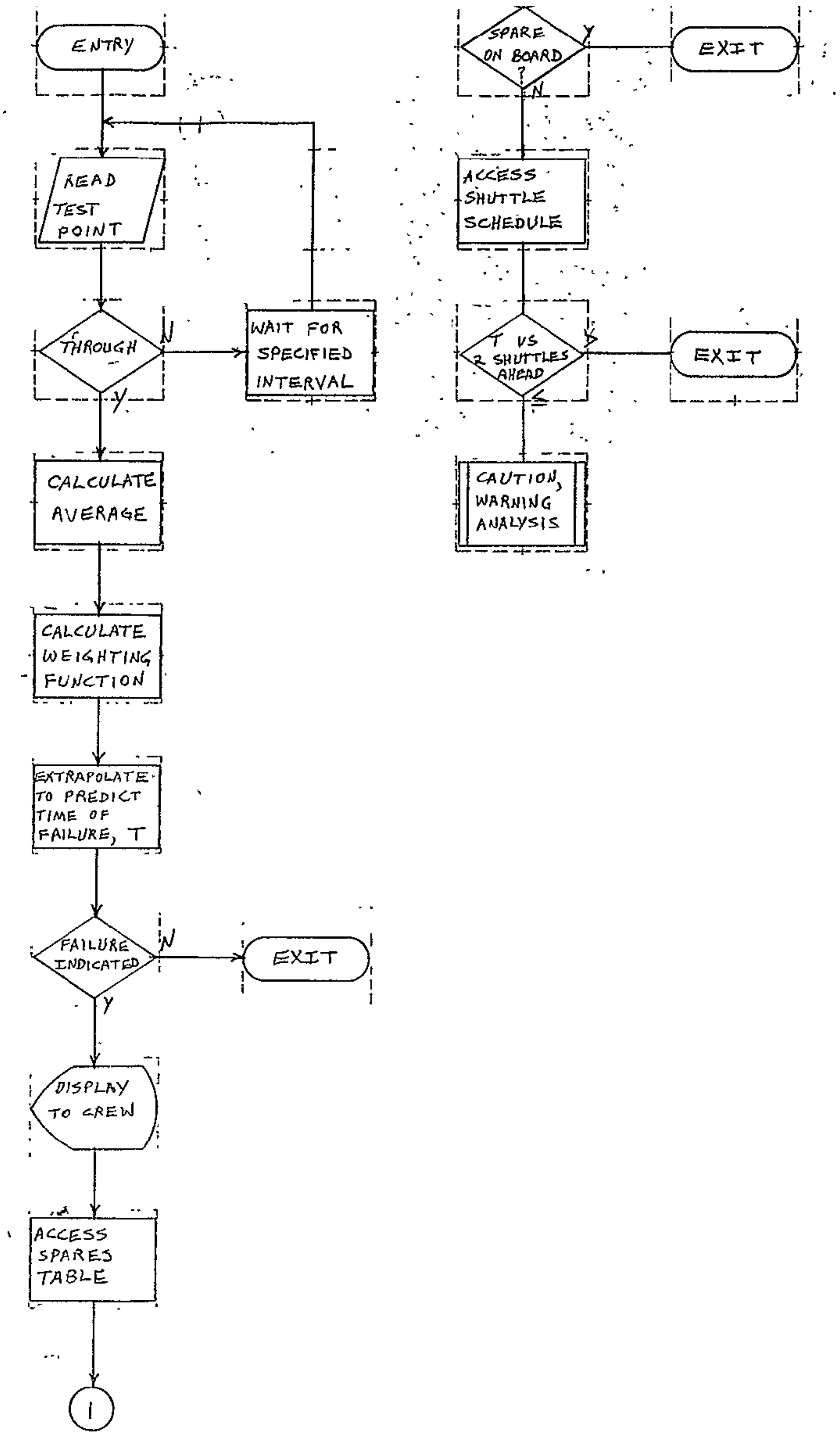


Figure 3-34. Periodic Sample T.A.

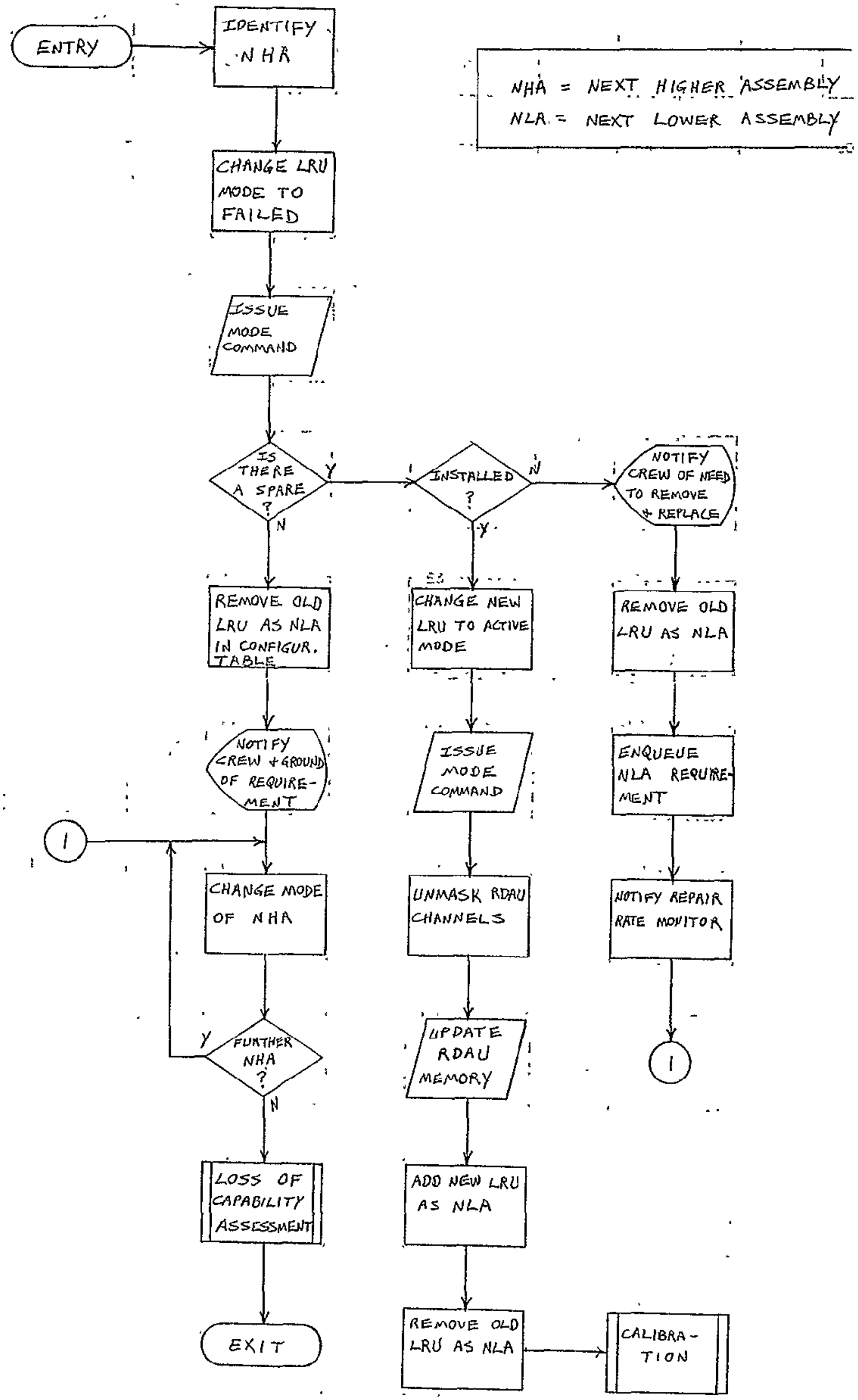


Figure 3-35. Reconfiguration

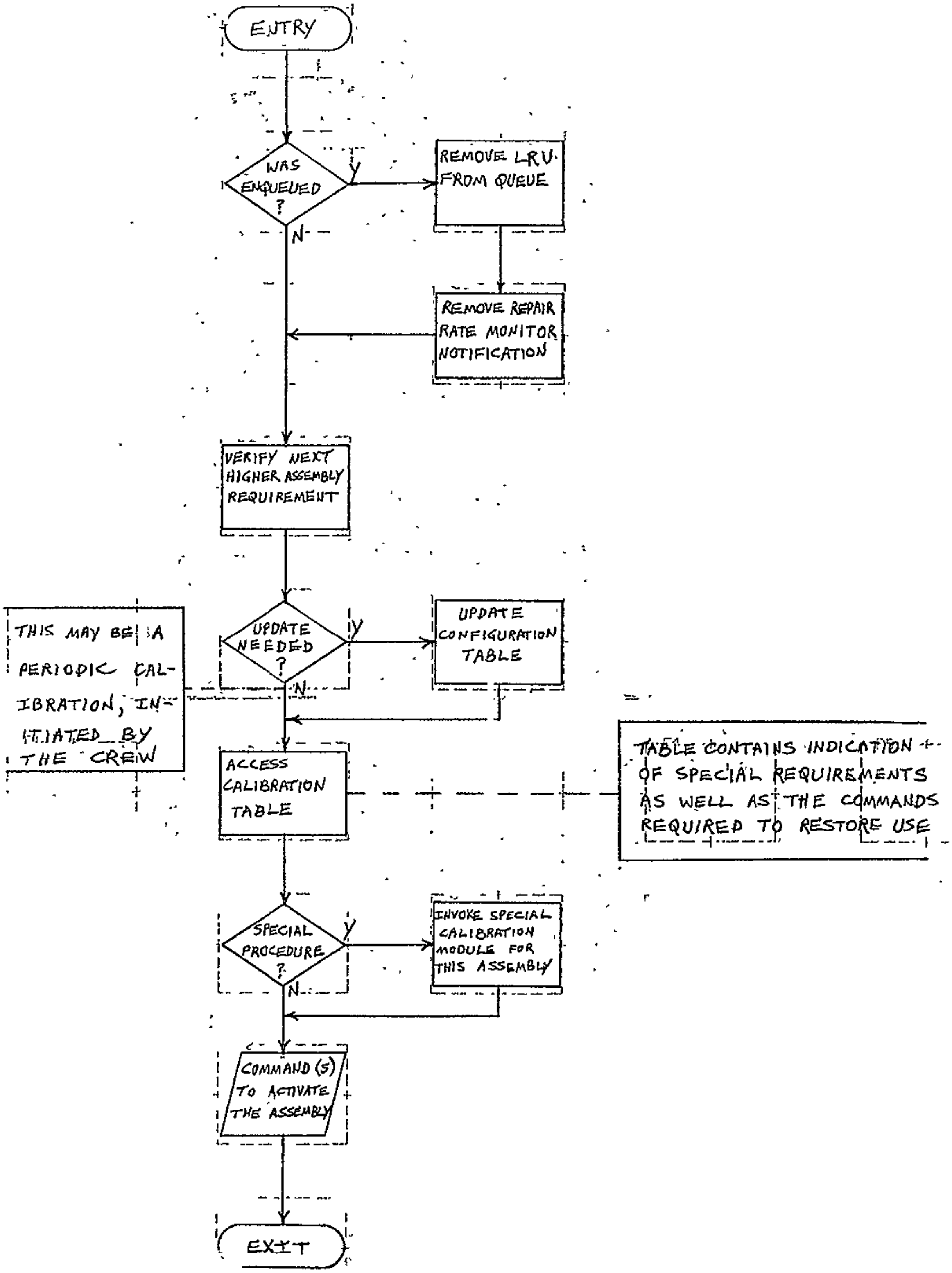


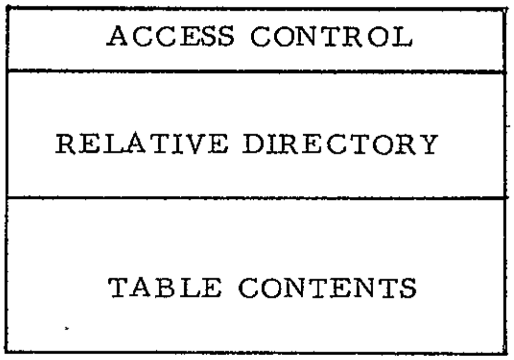
Figure 3-36. Calibration

3.1.4 DATA BASE REQUIREMENTS

Each of the following tables will be stored and updated redundantly in main memory.

- Mode
- Configuration
- Calibration
- Spares
- Trend Limit
- Operations Count
- Shuttle Schedule
- Test Point Limit
- Message Text
- Test Point Address

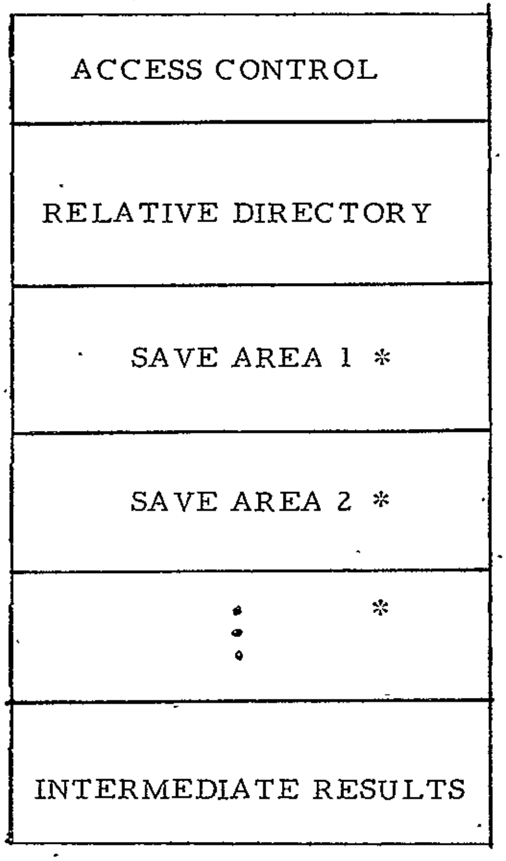
The following diagram indicates the general format of a table in the data base:



The exclusive control of the table by any task for the purpose of changing the contents is recorded in the Access Control section. The Relative Directory section contains pointers to the various types of entries, relative to the beginning of the table. A relative pointer scheme is employed to permit the table to be moved about in main memory without requiring any change to the contents.

Work space (intermediate program results) and program text is kept separate; with the result that the programs are refreshable from auxiliary storage, and checkpoint storage requirements are reduced. The amount of working storage will be pre-defined and allocated by the executive prior to starting the program. These considerations are transparent to the higher level language user.

The general layout of working storage is shown in the following diagram:



* Sufficient to accommodate the maximum module nesting.

The Access Control and Relative Directory sections are utilized in the manner described previously. The Save Area sections are employed when control is passed from one module to another to contain the status of machine registers, etc., so that restoration can be accomplished when control is returned to the originating module.

3.1.5 HUMAN PERFORMANCE

The study of the checkout requirements of the GNC Subsystem indicates that man is involved in the following ways:

- Certain test sequences will be crew-initiated.
- Crew or ground decisions may come into play when trend analysis results indicate an impending problem.
- Fault isolation results must identify the LRU and its physical location in the space station for remove-and-replace actions by the crew.
- A rate-of-execution change to the Pacer's table may be crew-or-ground-initiated.
- Fault detection results must be communicated to crew and/or ground.
- Caution signals must be activated for crew/ground cognizance.

3.2 INTERFACE REQUIREMENTS

The GNC checkout program requires services of, and is initiated by, the Executive program. Any interfaces to other programs, or data managed by other programs, is obtained through the executive. When crew or ground initiation is required, this is done with the executive serving as an interface.

3.2.1 INTERFACE DIAGRAM

The checkout function interfaces are shown in Figure 3-37. The Caution and Warning function examines a test point criticality table for each measurement detected to be out of limits, and provides required notification on the appropriate display. This function is performed by the OCS Executive which receives control from the interrupt handler of the Master Executive, from a trend analysis module, or from an application program.

The OCS Executive is also involved during fault isolation in an analysis of the mode of the assembly to be tested, and an analysis of the modes of the interfacing assemblies.

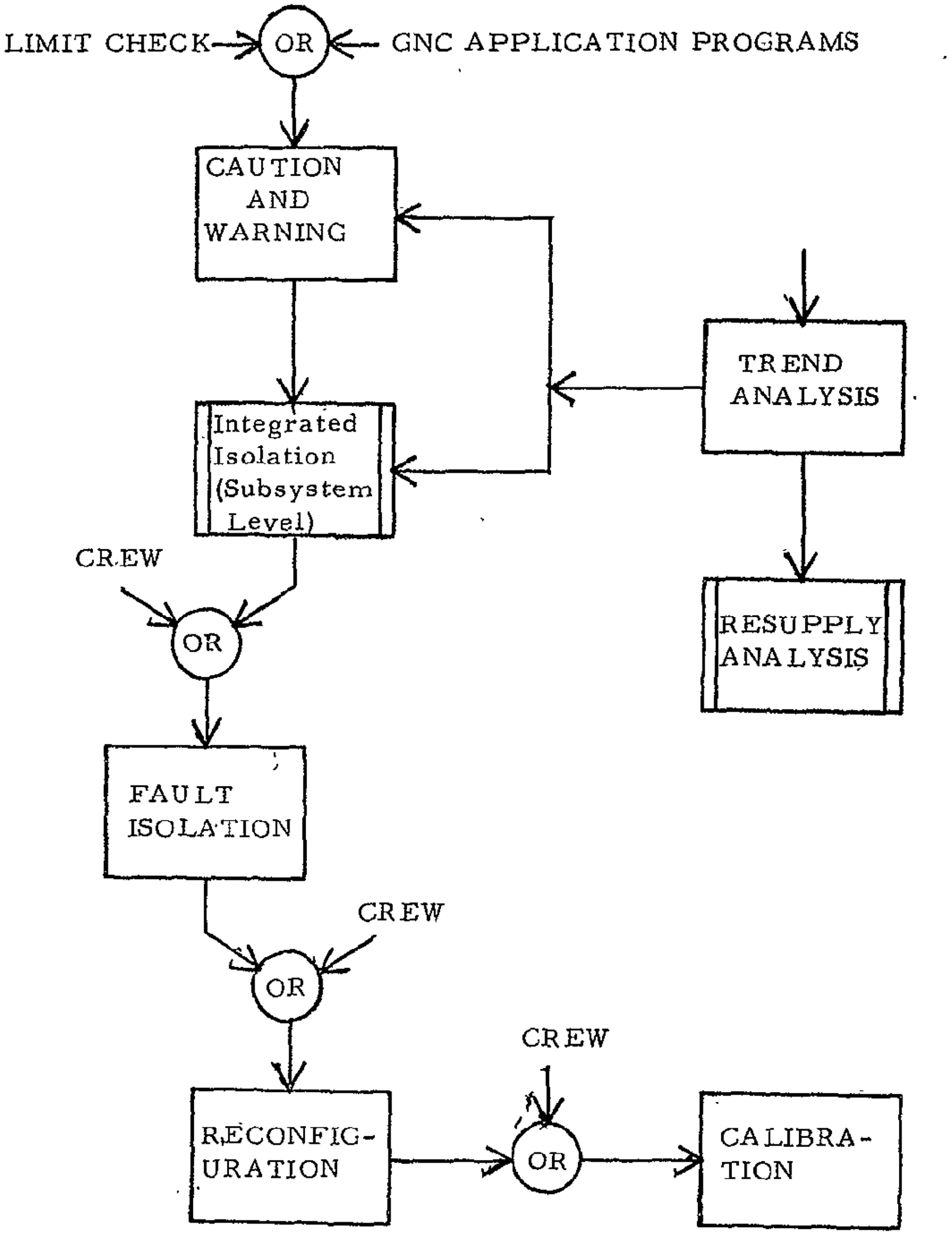


Figure 3-37. GNC Checkout Function Interfaces

3.2.2 DETAILED INTERFACE DEFINITION

The interfaces between GNC Checkout Program functions and specific Data Management Subsystem (DMS) elements and tables are shown in Figures 3-38 through 3-42.

The fault detection control function is activated by a crew or ground command.

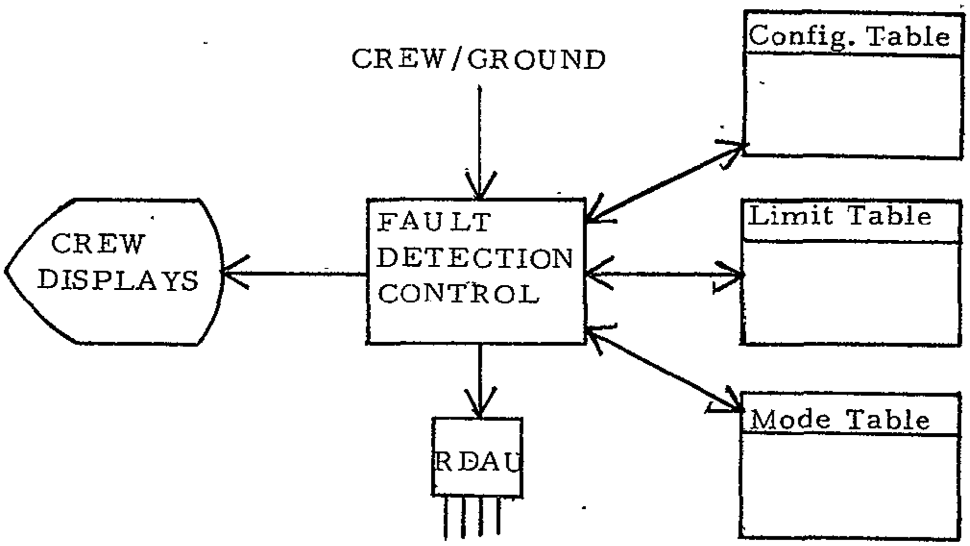


Figure 3-38. Fault Detection Control Interface

The trend analysis function receives control from an RDAU interrupt, or from the Pacer.

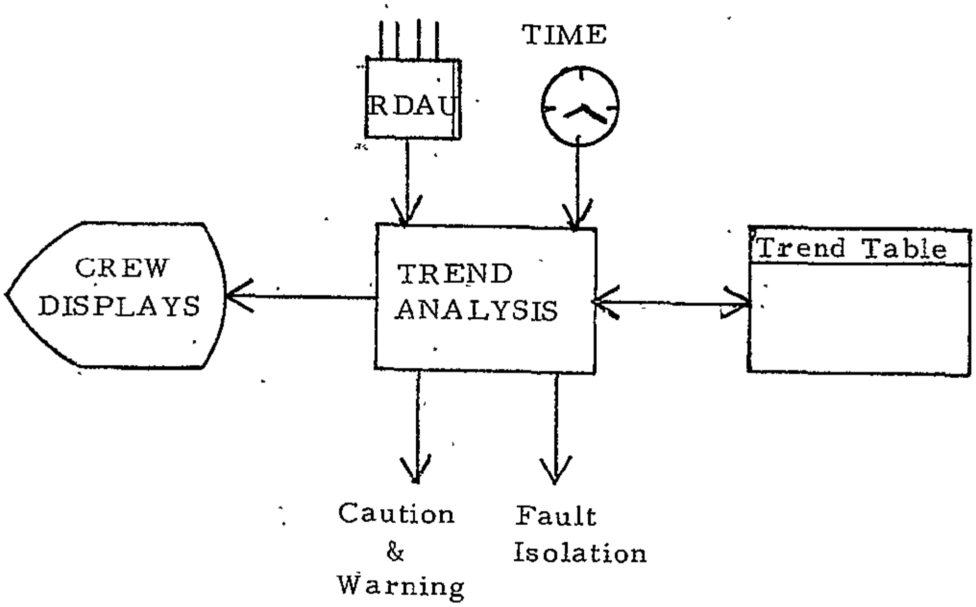


Figure 3-39. Trend Analysis Interface

Fault isolation modules for the GNC Subsystem receives control from the subsystem level fault isolation program and interface with the crew via the display units.

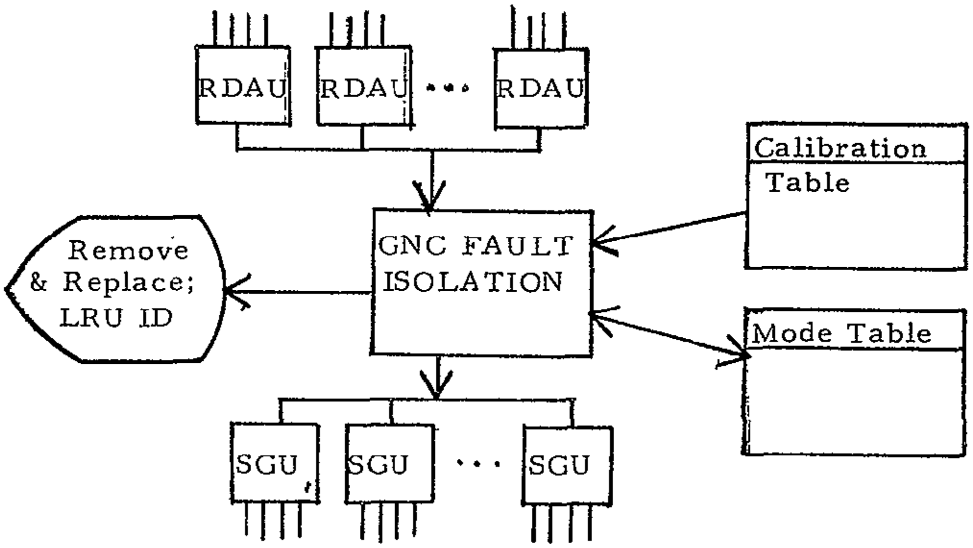


Figure 3-40. Fault Isolation Interface

The reconfiguration function receives control from a crew command or from the fault isolation modules.

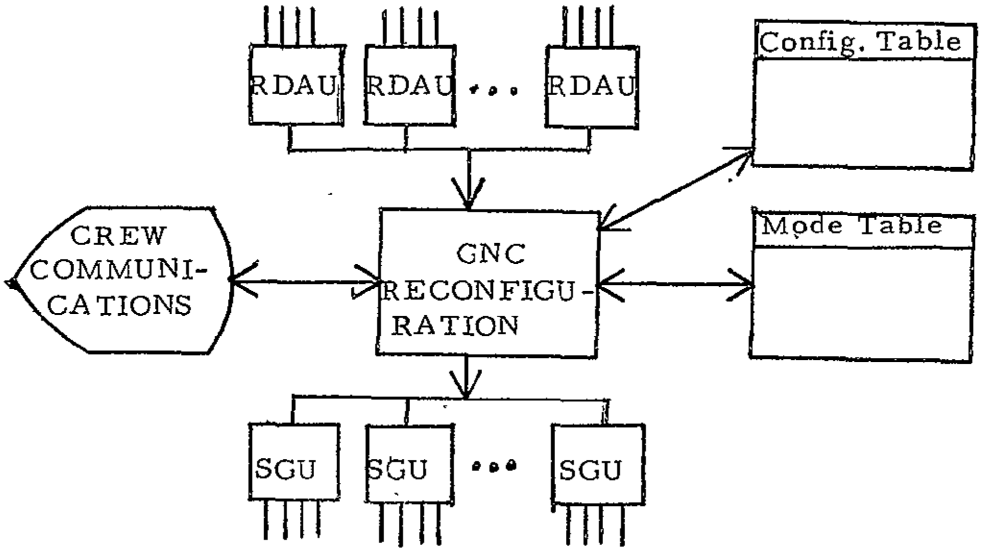


Figure 3-41. Reconfiguration Interface

The calibration function receives control from a crew command or from reconfiguration modules.

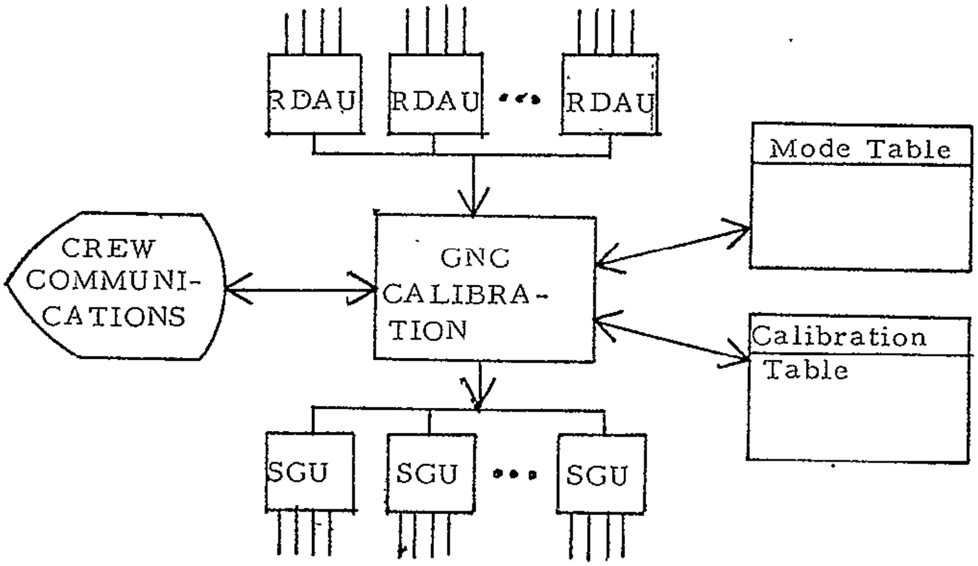


Figure 3-42. Calibration Interface

4.0 SUBSYSTEM SUMMARY

The checkout requirements study of the Guidance, Navigation, and Control Subsystem results in requirements regarding language, executive, and the Data Management Subsystem which are discussed in the following paragraphs.

4.1 LANGUAGE ANALYSIS

Because of the apparent frequency of making decisions immediately after obtaining measurements, combining the capabilities of the MEASURE and IF elements appears to be advantageous. This would be done by expanding the option list on MEASURE to include three optional branch labels for below, within, and above tolerance. The IF element would be retained for other purposes.

Table and data format compatibility between the checkout language, executive, and the other application programs is required because operational data is used by the trend, isolation, and calibration functions. In addition, an ACCESS element or an augmentation of the READ element capabilities is indicated in order for the high level language modules to acquire data from either the executive or from GNC application programs. In an analogous fashion, a STORE element or a modification of the WRITE element is indicated by the need to change the data base.

It must be possible to write a single test sequence which will test all occurrences of an LRU or other assembly, regardless of the actual test point addresses, in order to ensure that similar assemblies are treated identically. Such a sequence makes symbolic reference to test points which are resolved at execution time by a translation of a parameter list passed to the test sequence by the program which invokes it.

Modules written in the language must perform in a valid manner regardless of the manner in which they are initiated, since in many cases either crew, ground, or other programs can invoke a test module.

4.2 EXECUTIVE ANALYSIS

Maintenance and access control of the various data base tables is an important executive service implied by the GNC checkout programming requirements. The need for sharing data between application programs and checkout programs requires careful coordination in order that an error introduced in one program will not destroy a table used by other programs. By placing the actual alteration responsibility with the executive, application and checkout program changes will not disturb data base integrity. Multiprocessing makes possible the simultaneous reference to tables by several programs, thus providing rationale for having the executive serve as the controlling function for data base references or changes.

The executive must automatically initiate test sequences, trend analysis functions, and repair rate monitor displays at pre-determined time intervals; because the large number of events and wide range of time intervals make this activity unsuitable for human performance. Executive services must also be provided to allow the crew or ground to alter rates, or change entries in the rate table.

Programs must be initiated with the proper parameters by the executive. Commands for initiation will come from a variety of sources, requiring analysis to assure that the minimum parameter information is available prior to allotting DMS resources. Fault isolation modules may be initiated by the crew to verify proper operation or, for example, by the integrated fault isolation program.

The logic for translating symbolic test point addresses into actual hardware addresses will reside in the master executive since this service will be needed by any program which references measurement or stimulus points.

The OCS Executive must provide utilities and services for RDAU limit management, allowing changes to be specified by either crew or software.

4.3 PROGRAM SIZING

In order to evaluate the effect of the GNC checkout programs on the Data Management Subsystem, individual program modules are identified in Tables 4-1 through 4-4, and listed with certain characteristics which must be considered together in an overall analysis. For trend analysis modules which are implemented in machine language, the performance parameters are listed in terms of average processor time, I/O time, and storage words in Table 4-1. The characteristics of the high level language modules are tabulated according to the frequency with which the program executes the language elements. After estimating the DMS requirements of the individual language elements, the element utilization data in Tables 4-2 through 4-4 were reduced to DMS performance parameters as indicated in Tables 4-5 through 4-7.

Table 4-1. Trend Analysis Program Sizing

TREND PROGRAM	PROCESSOR TIME	I/O TIME	MAIN STORAGE	TIME IN MAIN STORAGE	AUXILIARY STORAGE	EXECUTION RATE
ATTITUDE GYRO	520±200 usec	72±8 usec	100±30 words	1240±300 usec	100±30 words	1/MIN
ACCELEROMETER	520±200 usec	72±8 usec	100±30 words	1240±300 usec	100±30 words	1/MIN
RENDEZVOUS TRACKER	3200±400 usec	720±80 usec	90±20 words	60 seconds	60±30 words	event oriented
DOCKING TRACKER	3200±400 usec	720±80 usec	90±20 words	60 seconds	60±30 words	event oriented
JET DRIVER	600±300 usec	-	100±50 words	600±300 usec	100±30 words	1/DAY
CONTROL MOMENT GYRO	2700±600 usec	360±40 usec	124±60 words	5 seconds	110±60 words	1/HOUR

Table 4-2. Fault Isolation Language Element Utilization

PROGRAM MODULES	LANGUAGE ELEMENTS												
	No. of Applications	Measure	Stimulate	If	Read	Solve	Display	Delay	Write	Call	Begin	GOTO	End
Horizon Detector	2	4	1	1	2	2	1		1		1	1	1
Attitude Gyro	6	22	6	6		4	15			6	1	6	1
Horizon Sensor	1	20	4	17	4	12	13	4		4	1	17	1
Star Sensor	2	6	7	4	2	4	4		2		1	4	1
Star Tracker	2	9	4	2		6	8			2	1	2	1
Landmark Tracker	1	14	4	14		8	12	2			1	14	1
Low-G Accelerometer	1	10	2	2	4	6	8	1		2	1	2	1
Rendezvous Tracker	4	14	4	14		8	12	2			1	14	1
Docking Tracker	7	14	4	14		8	12	2			1	14	1
Sensor Alignment Monitor	2	10	1	4	2	6	9	1	1	2	1	4	1
Experiment Alignment Monitor	2	10	1	4	4	6	9	1	1	2	1	4	1
Interface Electronics	1	26	36	8	12	12	14	12		12	1	8	1
Jet Driver Elec.	4	98	108	60		48	36	36		3	1	60	1
Control Moment Gyro	4	5	2	2		4	6	1		3	1	2	1
CMG Electronics	4	5	2	2		4	6	1		3	1	2	1

Table 4-3. Reconfiguration Language Element Usage

	▨	LANGUAGE ELEMENTS														
PROGRAM MODULES	No. of Applications	Measure	Stimulate	If	Read	Solve	Display	Delay	Write	Call	Begin	GOTO	End			
Horizon Detector	2	2	2								1		1			
Attitude Gyro	6	2	2								1		1			
Horizon Sensor	1	2	2								1		1			
Star Sensor	2	2	2								1		1			
Star Tracker	2	2	2	1	2	1	2	1	2		1	1	1			
Landmark Tracker	1	2	2	1	2	1	2	1	2		1	1	1			
Low-G Accelerometer	1	2	2	1	2	1	2	1	2		1	1	1			
Rendezvous Tracker	4	2	2	1	2	2	4	1	2		1	1	1			
Docking Tracker	7	2	2								1		1			
Sensor Alignment Monitor	2	2	2								1		1			
Experiment Align-Ment Monitor	2	2	2								1		1			
Interface Electronics	1	2	2	1	2	1	2	1	2		1	1	1			
Jet Driver Elec.	4	2	2	6	4	8	6	2	2	4	1	6	1			
Control Moment Gyro	4	2	2	1	2	1	2	1	2		1	1	1			
CMG Electronics	4	2	2	1	2	1	2	1	2		1	1	1			

Table 4-4. Calibration Language Element Usage

			LANGUAGE ELEMENTS														
		No. of Applications	*Measure*	*Stimulate*	*If*	*Read*	*Solve*	*Display*	*Delay*	*Write*	*Call*	*Begin*	GOTO	*End*			
PROGRAM MODULES	Horizon Detector	2		1	2						3	1	2	1			
	Attitude Gyro	6	2	7	4	2	4	4		2	3	1	4	1			
	Horizon Sensor	1		1	2						3	1	4	1			
	Star Sensor	2	6	8	4	2	3	4	2	1	3	1	2	1			
	Star Tracker	2	2	2	1	1	1	2	1	1	3	1	4	1			
	Landmark Tracker	1	2	2	1	1	1	2	1	1	3	1	1	1			
	Low-G Accelerometer	1		1	2						3	1	2	1			
	Rendezvous Tracker	4	2	2	1	1	1	2	1	1	3	1	1	1			
	Docking Tracker	7		1	2						3	1	2	1			
	Sensor Alignment Monitor	2		1	2						3	1	2	1			
	Experiment Alignment Monitor	2		1	2						3	1	2	1			
	Interface Electronics	1		1	2						3	1	2	1			
	Jet Driver Elec.	4		1	2						3	1	2	1			
	Control Moment Gyro	4	4	4	2	2	1	2	4	1	3	1	2	1			
	CMG Electronics	4		1	2						3	1	2	1			

The column headings in Tables 4-5 through 4-7 have the following meaning:

PROGRAM MODULE:	The name of the program.
RATE:	Frequency with which this program module will be executed for each application.
APPLICATIONS:	Number of uses for the module. As an example, if four identical assemblies required a periodic test once per hour, the number of applications would be four and the rate 1/H. Viewed from the multiprocessor, the traffic rate would be 4/H.
PROCESSOR TIME:	(Milliseconds) The time required for actual instruction execution; the average time that a processor will be busy because of the program. Note that this time is exclusive of I/O time,and that the work load for any module is shared among the processors without prejudice.
I/O TIME:	(Milliseconds) The time in which execution of the program in main storage must be suspended while a data acquisition path program is performed by a data bus controller or channel.
MAIN STORAGE:	(Words) The amount of shared main storage required by the program text (instructions), plus that required for working storage (data and constants).
TIME IN MAIN STORAGE:	The average time during which the program must occupy shared main storage. In some cases, this is more than the sum of processor and I/O times because of the need to delay briefly while the subsystem responds to stimuli.
AUXILIARY STORAGE:	(Words) The amount of auxiliary storage required for program text (instructions), tables, constants, intermediate results, results which must be retained for later use, etc.

Where encircled numbers appear in Tables 4-5 through 4-7 they have the followin meaning:

1. Rate of execution is determined by the failure rate of the LRU addressed by the program.
2. Time in main storage is the sum of processor and I/O times.
3. Auxiliary storage requirement is estimated to be the same as the main storage requirement.

Table 4-5. Reconfiguration Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec	Main Storage Words	Time in Main Storage	Auxiliary Storage
Horizon Detector	①	2	2.56	406	31	②	③
Attitude Gyro	①	6	2.56	406	31	②	③
Horizon Sensor	①	1	2.56	406	31	②	③
Star Sensor	①	2	2.56	406	31	②	③
Star Tracker	①	2	5.84	1220	147	②	③
Landmark Tracker	①	1	5.84	1220	147	②	③
Low G Accelerometer	①	1	5.84	1220	147	②	③
Rendezvous Tracker	①	4	6.56	1220	222	②	③
Docking Tracker	①	7	2.56	406	31	②	③
Sensor Align. Monitor	①	2	2.56	406	31	②	③
Experiment Align. Monitor	①	2	2.56	406	31	②	③
Interface Electronics	①	1	5.84	1220	147	②	③
Jet Driver Electronics	①	4	12.7	2440	473	②	③
Control Moment Gyro	①	4	5.84	1220	147	②	③
CMG Electronics	①	4		1220	147	②	③

Table 4-6. Fault Isolation Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Horizon Detector	①	2	5.86	1120	127	②	③
Attitude Gyro	①	6	25.2	4070	875	②	③
Horizon Sensor	①	1	29.1	4070	1006	②	③
Star Sensor	①	2	13.2	2130	320	②	③
Star Tracker	①	2	12.7	1730	463	②	③
Landmark Tracker	①	1	19.5	1840	772	②	③
Low G Accelerometer	①	1	14.3	2440	489	②	③
Rendezvous Tracker	①	4	19.5	1840	772	②	③
Docking Tracker	①	7	19.5	1840	772	②	③
Sensor Align Monitor	①	2	14.0	2140	540	②	③
Experiment Align Monitor	①	2	15.0	2540	550	②	③
Interface Electronics	①	1	57.9	11200	1230	②	③
Jet Driver Electronics	①	4	165	21600	3559	②	③
Control Moment Gyro	①	4	8.57	1320	356	②	③
CMG Electronics	①	4	8.57	1320	356	②	③

Table 4-7. Calibration Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Horizon Detection	①	2	2.38	710	75	②	③
Attitude Gyro	①	6	11.9	2340	328	②	③
Horizon Sensor	①	1	2.38	710	75	②	③
Star Sensor	①	2	14.3	2640	348	②	③
Star Tracker	①	2	5.89	1420	173	②	③
Landmark Tracker	①	1	5.89	1420	173	②	③
Low G Accelerometer	①	1	2.38	710	75	②	③
Rendezvous Tracker	①	4	5.89	1420	173	②	③
Docking Tracker	①	7	2.38	710	75	②	③
Sensor Align. Monitor	①	2	2.38	710	75	②	③
Experiment Align. Monitor	①	2	2.38	710	75	②	③
Interface Electronics	①	1	2.38	710	75	②	③
Jet Driver Electronics	①	4	2.38	710	75	②	③
Control Moment Gyro	①	4	9.33	2030	218	②	③
CMG Electronics	①	4	2.38	710	75	②	③

APPENDIX C

EPS SUBSYSTEM CHECKOUT PROGRAM DESIGN

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

1.0 INTRODUCTION

The purpose of this specification is to provide performance and design requirements for both the Isotope/Brayton and the Solar Array Electrical Power Subsystem checkout programs to be used for the Space Station during pre-flight testing and while in orbit. Of necessity, the detail employed herein is a function of the engineering design details available in the Phase B Space Station studies.

In comparison, the Isotope/Brayton design places greater arithmetic requirements on the high-level checkout language, while the Solar Array design employs substantially more test points, and requires more storage for trend analysis data. Other differences are discussed in the paragraphs below.

The checkout functions identified are those for trend analysis, status monitoring periodic checkout, fault isolation, and reconfiguration. Fault detection is feasible with the RDAU preprocessor, under control of the checkout executive program; and is therefore not identified as a checkout application program function.

2.0 APPLICABLE DOCUMENTS

- "Electrical Power", Space Station, MSFC-DRL-160, Line Item 13, Preliminary Systems Design Data, Volume I, Space Station Preliminary Design, Book 1, Contract NAS8-25140, July 1970.
- "Electrical Power Subsystem, Environmental Control and Life Support Subsystem", Solar-Powered Space Station, Preliminary Design, Volume II, DRL NO: MSC T-575, Line Item 13, Contract NAS9-9953, July 1970.
- "Performance/Design and Product Configuration Requirements for Space Station Module, Space Station Test Articles, and Nose Cone System", Space Station, MSFC-DRL-160, Line Item 19, Preliminary Performance Specifications, Volume I, Contract NAS8-25140, July 1970.
- "Baseline Subsystem Descriptions", Interim Report RA-1, Contract NAS9-11189, August 1970.
- "Failure Mode and Effects Analysis", Interim Report RA-4, Contract NAS9-11189, September 1970.
- "Subsystem Checkout Strategy", Interim Report RA-5, Contract NAS9-11189, September 1970.
- "Line Replaceable Unit Definition", Interim Report RA-6, Contract NAS9-11189, September 1970.
- "Subsystem Test Definition and Measurement/Stimulus List", Interim Report RA-8, Contract NAS9-11189, November 1970.
- "Solar Array Checkout Analysis", Interim Report RA-13, Contract NAS9-11189, November 1970.
- "Structure Subsystem Checkout Program", Interim Report OCS-2-2021, Contract NAS9-11189, November 1970.
- "GNC Subsystem Checkout Program", Interim Report OCS-2-2022, Contract NAS9-11189, November 1970.

3.0 REQUIREMENTS

In this section, the technical aspects of the Isotope/Brayton (I/Br) and the Solar Array (SA) checkout programs are described. The computer program components are identified, and their functions, structure, processing, input, output, and data base requirements are discussed.

3.1 PERFORMANCE

Both the Isotope/Brayton and the Solar Array configurations of the Space Station Electrical Power Subsystem (EPS) require the same general checkout program functions. Trend analysis is required for assessment of a series of measurements. Status monitoring is required to smooth the effects of transients. Fault isolation is required to locate the failed assembly and identify the LRU. Reconfiguration involves the recovery from failure. Periodic checkout exercises all modes of certain assemblies to verify proper operation. As discussed below, individual functions differ in their details depending on whether the I/Br or SA design is used.

Block diagrams of EPS in the Isotope/Brayton and Solar Array configurations are shown in Figure 3-1 and Figure 3-2, respectively.

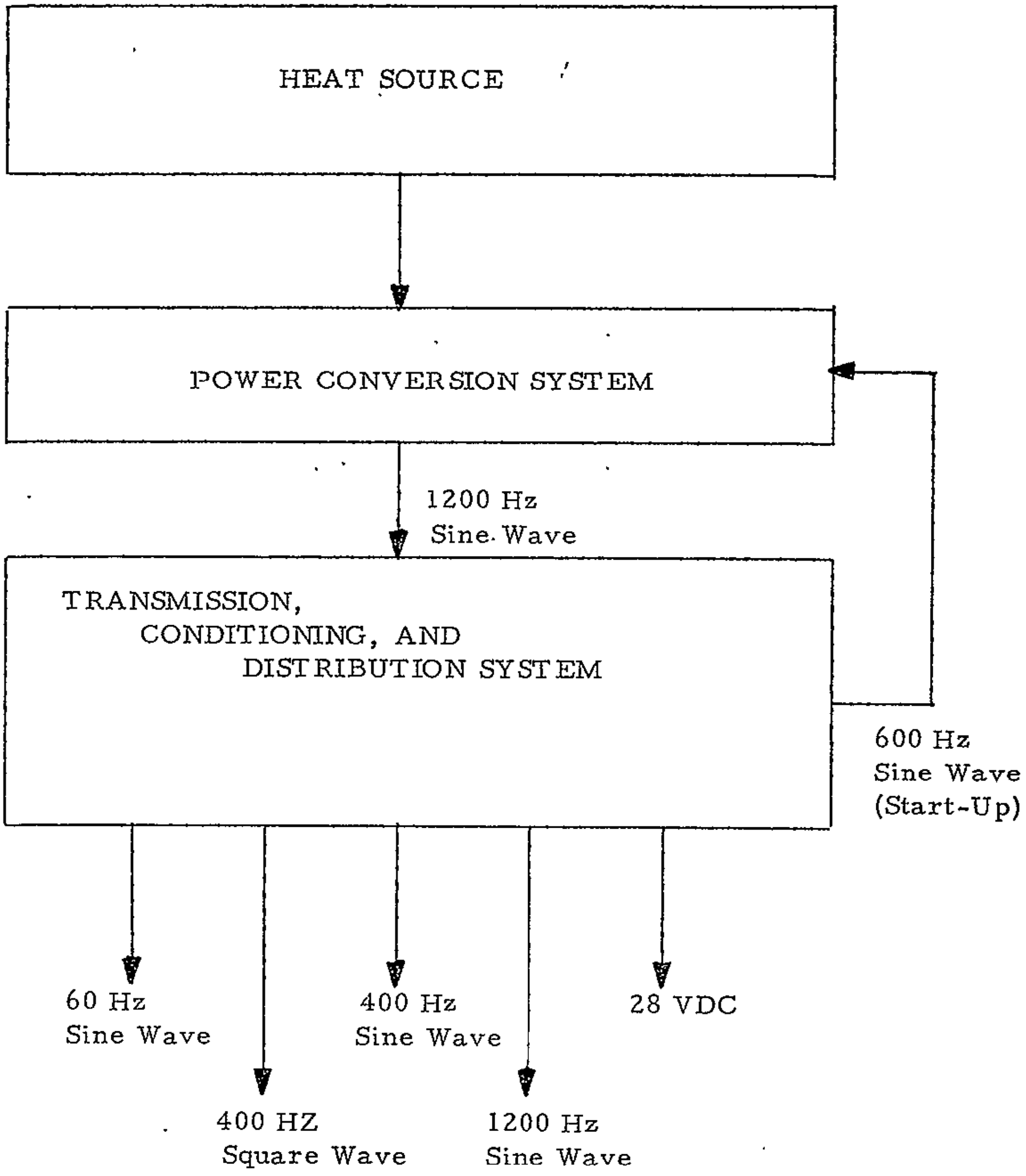


Figure 3-1. Electrical Power Subsystem Diagram
Isotope/Brayton Configuration

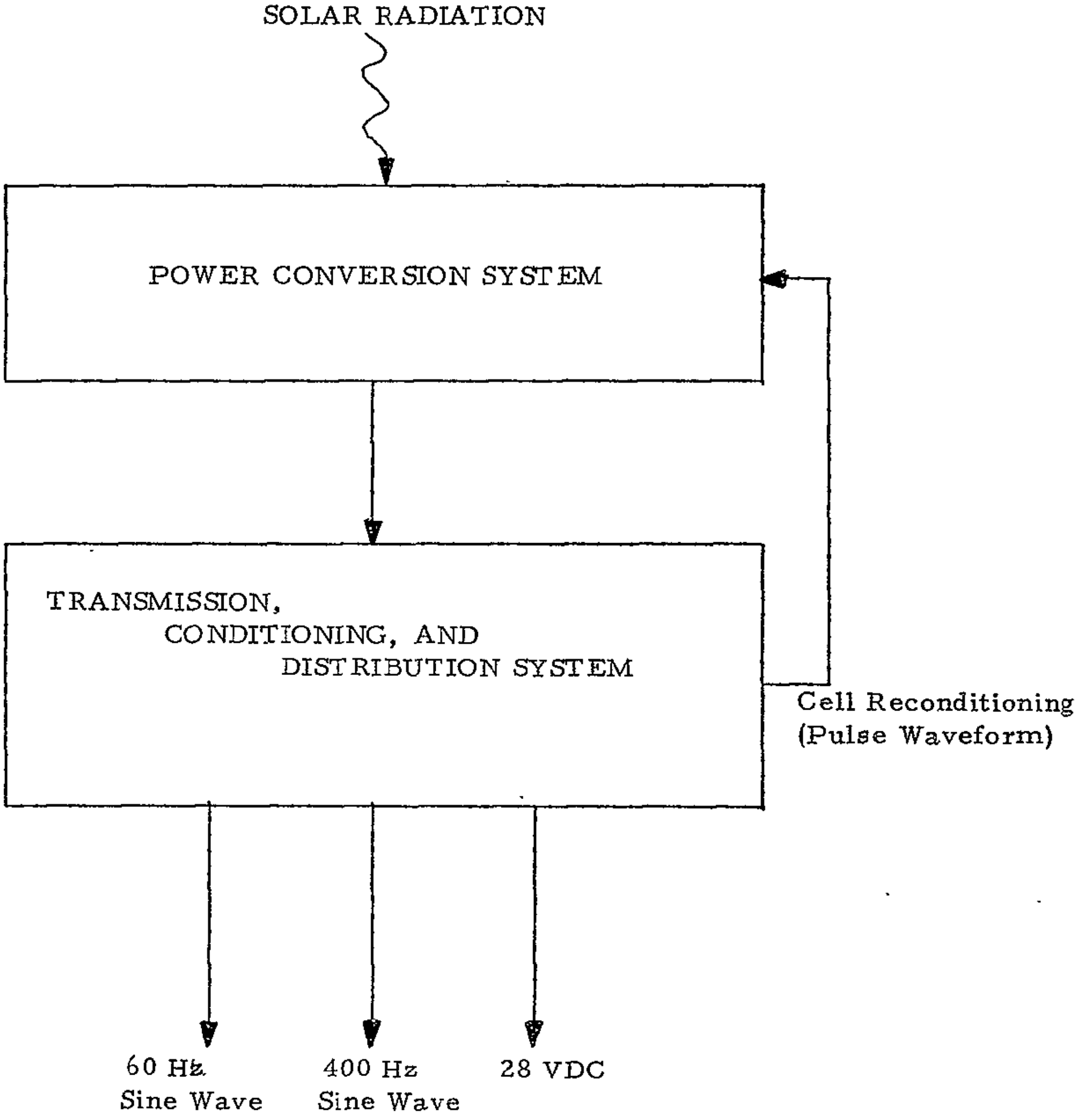


Figure 3-2 . Electrical Power Subsystem Diagram
Solar Array Configuration

3.1.1 SYSTEM REQUIREMENTS

Both EPS designs require that the physical laws concerning the power sources be employed in checkout. This reduces the number of test points while increasing the checkout program complexity.

Because the power supplied to other subsystems is vital to their performance, certain measurements must be made at periods of less than one second, which may provide the upper bound for DMS response, both from the software and the hardware standpoint.

The Electrical Power Subsystem is an essentially serial hierarchy of assemblies, compared with other subsystems of the Space Station; consequently, the modularity of EPS checkout programs is influenced by extensive interface between modules.

Transients can momentarily cause test points to exceed their limits and then return to within limits without adversely affecting the load. Therefore, the RDAU limit checking capability must be augmented by successive test point sampling at specified intervals in order to distinguish between real out-of-tolerance conditions and temporary ones.

3.1.2 OPERATIONAL REQUIREMENTS

The EPS checkout programs are required to perform trend analysis, status monitoring, fault isolation, reconfiguration and periodic checkout.

3.1.2.1 Trend Analysis

Trend analysis is performed on selected parameters of the EPS for performance evaluation and the detection of an impending failure prior to the time when an actual out-of-limit measurement is obtained.

Input to the trend analysis function consists of control from the executive at regular intervals, status of the assembly with which the parameter is associated, and a reading of the parameter itself. The requirement to sample certain solar array parameters at a rate of once per orbit indicates a need for specifying the time at which sampling should begin, as well as the rate at which the executive will give control to the trend analysis function. The measurements previously obtained are also required as input to the trend analysis function.

Output of the trend analysis function consists of the measurement collection for storage until the next sample, displays of exceptional conditions to the crew and/or ground, and the initiation of other checkout functions such as caution and warning analysis, and fault isolation.

The trend analysis functions for the Isotope/Brayton design are concerned with maintaining a fixed history of measurements for display upon request, and with using collections of measurements to predict an impending failure. These two methods of trend analysis are described as follows:

- Data Collection - Gather raw data from selected test points and store on an as-requested basis. The quantity of data retained is limited to a pre-set value, selected for each particular test point. The oldest measurements are dropped as new ones are obtained.
- Extrapolation - Gather data as above and perform exponential smoothing, adjustment for trend, and extrapolation after each measurement to determine if an impending out-of-limit condition exists. This function makes use of the checkpoint and restart services of the executive, as depicted in Figure 3-13.

In addition to the two trend analysis functions described above, the Solar Array design requires the following:

- Telemetry - Measurements obtained by this function are telemetered directly to the ground. Provision is made for temporary onboard storage, in case the telemetry link is unavailable.

The on-off cycling of certain assemblies requires that a status check be performed prior to measuring the parameter. If the assembly is inactive, the measurement is not made.

In Table 3-1 and Table 3-2, the Isotope/Brayton and Solar Array trend analysis requirements are summarized. An estimate of the auxiliary storage requirements for intermediate results may be estimated from the number of test points and the number of retained measurements. Assuming 8 bits per measurement and 32 bits per word, the I/Br auxiliary storage requirement for trend analysis is approximately 66K words compared with an SA requirement of 1476K words. These estimates include provisions for three checkpoints in the checkpoint log.

3.1.2.2 Status Monitoring

The status monitoring functions augment the continuous monitoring functions provided in hardware form by the RDAU preprocessor. Control is passed to the status monitoring functions when an RDAU limit check occurs for selected parameters.

Input consists of test point readings and their associated limits. Output consists of crew displays and fault detection indications if the status monitoring function can confirm an error detected by an RDAU. If no confirmation is obtained, a crew display indicating the fact that status monitoring was involved is provided.

Information processing for both the Isotope/Brayton and the Solar Array transmission/conditioning/distribution assembly consists of successive measurements after an out-of-limit condition has been detected by an RDAU, to determine if the parameter will remain out of limits during a pre-set number of consecutive readings. This technique is applied to most bus voltages. The flow chart in Figure 3-14 depicts a module which can be used for any parameter requiring this type of status monitoring. The delay between measurements is adjustable to meet the successive sampling rates required by each application.

For selected I/Br parameters such as compressor inlet temperature and fuel capsule temperatures, measurement of parallel redundant parameters is required to distinguish between a defective transducer LRU and a true out-of-limit condition.

In addition, it is required to raise the limit checking threshold prior to passing control to the caution and warning analysis module of the checkout executive. This is done for the following in the I/Br power subsystem:

- Heat Source
    - Fuel Capsule Temperature
    - BeO Heat Sink Temperature
- Control and Monitoring
    - Speed Control Signals

Table 3-1. Isotope/Brayton Trend Analysis

TEST POINT	NUMBER OF MEASUREMENTS	TREND METHOD	MEASUREMENT RATE	MEASUREMENTS RETAINED PER TEST POINT
SHIELD:				
Shield Drive Motor Torque	2	Data Collection	See Note 1	TBD
POWER CONVERSION SYSTEM:				
HRHX Coolant Inlet Temperature	2	Extrapolation	1/mo.	TBD
Recuperator Outlet Temperature	2	Extrapolation	1/mo.	TBD
Gas Loop Flow Rate	2	Extrapolation	1/week	TBD
GAS MANAGEMENT SYSTEM:				
Gas Storage Pressure	2	Data Collection	1/week	TBD
HEAT REJECTION SYSTEM:				
Pump Motor Current In	2	Extrapolation	1/week	52
Pump Motor Pressure Out	2	Extrapolation	1/week	52
Radiator Coolant Discharge	2	Extrapolation	1/week	52
HEAT SOURCE:				
Fuel Capsule Temperature	1	Extrapolation	1/week	52
TRANSMISSION/CONDITIONING/DISTRIBUTION:				
Alternator Feeder Currents	6	Data Collection	4/hour	1344
Source Bus Voltage	12	Extrapolation	4/day	84
Main 28 VDC Distributor Bus Voltage	4	Extrapolation	4/day	84
28 VDC Bus Tie Cable Current	1	Data Collection	4/hour	1344
28 VDC Load Bus Voltage	12	Data Collection	4/hour	1344
260 VDC Link Bus Voltage	2	Data Collection	4/hour	1344
400 Hz Square Wave Distributor Bus Voltage	6	Data Collection	4/hour	1344

Table 3-1. Isotope/Brayton Trend Analysis (Continued)

TEST POINT	NUMBER OF MEASUREMENTS	TREND METHOD	MEASUREMENT RATE	MEASUREMENTS RETAINED PER TEST POINT
400 Hz Sine Wave Distributor Bus Voltage	6	Data Collection	4/hour	1344
Regulated Transformer-Rectifier Output Current	5	Data Collection	4/hour	1344
Regulated Transformer-Rectifier Temperature	5	Data Collection	4/day	56
High Voltage Rectifier Output Current	4	Extrapolation	4/day	64
High Voltage Rectifier Regulator Temperature	4	Extrapolation	4/day	64
400 Hz Square Wave Inverter Temperature	2	Extrapolation	4/day	64
400 Hz Sine Wave Inverter Temperature	2	Extrapolation	4/day	64
60 Hz Sine Wave Inverter Temperature	2	Extrapolation	4/day	64
Battery Charger Regulator Output Current	10	Data Collection	4/hour	1344
Battery Charger Regulator Temperature	10	Extrapolation	4/day	64
Battery Charger Regulator Rate Mode	10	Data Collection	4/hour	1344
Battery Buck Regulator Temperature	10	Extrapolation	4/day	64
Battery Terminal Voltage	10	Extrapolation	4/day	64
Battery Monitor Voltage	10	Extrapolation	4/day	64
Battery Temperature	10	Extrapolation	4/day	64

NOTE 1: The measurement rate (TBD) will apply only during periodic checkout of the shield motor.

Table 3-2. Solar Array Trend Analysis

TEST POINT	NUMBER OF MEASUREMENTS	TREND METHOD	MEASUREMENT RATE	MEASUREMENTS RETAINED PER TEST POINT
BATTERIES:				
Battery Voltage	12	Data Collection	4/day	84
ARRAY:				
Circuit Voltage	160	Telemetry	Varies	2
Circuit Current	160	Telemetry	Varies	2
TRANSMISSION/CONDITIONING/DISTRIBUTION				
Core & Boom Inverter Power Output (3ϕ)	8	Data Collection	4/hour	1344
Core & Boom Inverter Temperature	8	Extrapolation	4/day	84
Inverter Feeder Current	12	Data Collection	4/hour	1344
Primary Bus Voltage	12	Data Collection	4/minute	78720
Primary Bus Tie Cable Current	12	Data Collection	4/minute	78720
Battery Charger Temperature	12	Extrapolation	4/day	84
Autotransformer Temperature	4	Extrapolation	4/day	84
Secondary Bus Structure Coolant Temperature(in)	4	Extrapolation	4/day	84
Secondary Bus Structure Coolant Temperature(out)	4	Extrapolation	4/day	84
Secondary Bus Structure DC Bus Voltage	4	Data Collection	4/hour	1344
Secondary Bus Structure AC Bus Voltage	12	Extrapolation	4/hour	2016
60 Hz Inverter Temperature	2	Extrapolation	4/day	84
Rectifier-Filter Temperature	4	Extrapolation	4/day	84
Rectifier-Filter Input Voltage	12	Data Collection	4/hour	1344
Rectifier-Filter Output Current	4	Data Collection	4/hour	1344

3.1.2.3 Periodic Checkout

Peridic checkout functions are required to supplement the continuous monitoring performed by the RDAU hardware in order to make a quantitative evaluation of operating characteristics, and to verify the operation of inactive or standby systems.

Input consists of test point measurements, mode/status indications, the configuration table, and interactions with the crew. Output consists of stimuli, mode/status changes, configuration changes, and crew displays.

Information processing involves a variety of techniques, ranging in complexity from verifying that parameters are within limits to cycling a standby assembly through its various modes, and using it to replace an operational assembly of the same type. Limit check verification is performed as an executive service. Other periodic tests are indicated for the Isotope/Brayton power subsystem in Table 3-3, and for the Solar Array power subsystem in Table 3-4. An example of periodic checkout flow is shown in Figure 3-15.

3.1.2.4 Fault Isolation

The fault isolation function locates the source of error which has been suggested by fault detection, status monitoring, crew/ground, periodic checkout, or trend analysis. It is a goal of this function to isolate to the failed Line Replaceable Unit (LRU). The modular design of this function follows the design of the EPS itself. In Figure 3-3 through Figure 3-9, two levels of detail are presented for the Isotope/Brayton hierarchy, while the Solar Array design is shown in Figure 3-10 through Figure 3-12.

Input consists of information from configuration and mode/status tables, measurements, and crew interaction. Output consists of stimuli, commands through operational interfaces, and displays.

Information processing consists of determining if interfaces to the subsystem or assembly are being properly supplied, followed by an evaluation of the output of the assembly. If the supplied interfaces from other assemblies are within tolerance and the output is bad, the assumption is made that the fault lies within the assembly, and further analysis is made using the test points and operational interfaces associated with the assembly.

Some special fault isolation considerations which arise for the Electrical Power Subsystem are outlined as follows:

- Considered as a single assembly, the interfaces supplied to EPS are principally structural, with comparatively minor interfaces with EC/LS and DMS.

TEST NAME	NO. OF TEST APPLICATIONS	MEASUREMENTS	STIMULI	FREQUENCY
Drive Mechanisms	2	2	1	4/year
IRV System	2	6	-	4/year
Inverters	1	82	30	1/week
Battery Chargers	10	9	4	1/week
Selector Switches.	TBD	TBD	TBD	1/month
Motor Generators	2	22	8	1/month
Transformer-Rectifiers	5	5	1	1/month
High-Voltage Rectifiers	4	12	2	1/month
Buck Regulators	10	6	4	1/month
Circuit Breakers	1	208	208	4/year
Contactors	TBD	TBD	TBD	4/year
Switches	1	81	81	4/year
Differential Relays	1	6	6	2/year
Reverse-Current Relays	1	18	18	2/year

TEST NAME	NO. OF TEST APPLICATIONS	MEASUREMENTS	STIMULI	FREQUENCY
60 Hz Inverters	2	7	3	1/week
Battery Chargers	18	11	2	1/week
Core & Boom Inverters	1	80	36	1/month
Rectifier-Filters	4	5	3	1/month
Power Contactors	1	88	88	4/year
Differential Protection	1	52	52	2/year
Array Circuit I-V Test	1	400	-	1/day
Battery Controls & Indicators	1	12	12	1/week
Battery Cell Recondition Signals	1	72	36	1/week
Battery Cell By-Pass	1	1800	900	1/month

- Some of the EPS assemblies, such as the primary buses in the I/Br design, are connected together at the same hierarchical level.
- Assemblies of EPS tend to be serially interrelated, rather than parallel, as in other subsystems such as GNC.
- In the I/Br design, transducers are specified as LRU's.

Because of the simplicity of incoming interfaces, particularly at the power conversion system level, the Electrical Power Subsystem may be used as a beginning point for integrated fault isolation at the subsystem level.

Fault isolation for assemblies which operate in closed loops may involve an intermediate interface evaluation after supplied interfaces are examined, in order to evaluate tie connections at the same assembly level. In some cases, opening the loop may be required for additional analysis. Modular concepts are also affected by closed loop operation, since a single fault isolation program module which addresses all the assemblies may be required, as opposed to a module which evaluates one assembly and is used multiple times.

The serial nature of EPS requires more extensive interface between fault isolation modules, and a deeper module nesting than would be the case for a more parallel assembly.

The specification of transducers as LRU's implies the use of calculations involving alternate measurements to ascertain whether the transducer indication is accurate. In the I/Br design, energy balance equations are employed which make use of temperature, pressure, and fluid flow to corroborate measurements obtained through transducers which are themselves line replaceable units.

3.1.2.5 Reconfiguration

The reconfiguration function maintains the portion of the configuration table as it applies to EPS. This function becomes active as a result of the removal of an assembly containing a failed LRU, or the addition of an assembly after repair.

Input consists of status and configuration data from tables and symbolic identities of the assemblies to be reconfigured. Output consists of table updates, stimuli, and commands necessary to connect or disconnect the assembly. Measurements are made to assure that the stimuli and commands have taken place.

Information processing includes the logic necessary to effect remove/replace activities with EPS assemblies, and to record the result in the configuration and status tables. Interface with operational programs such as start-up and shut-down functions are required during processing associated with the I/Br combined rotating unit. In both the SA and I/Br transmission, conditioning, and distribution modules, interface with the power management operational module is required for load balancing.

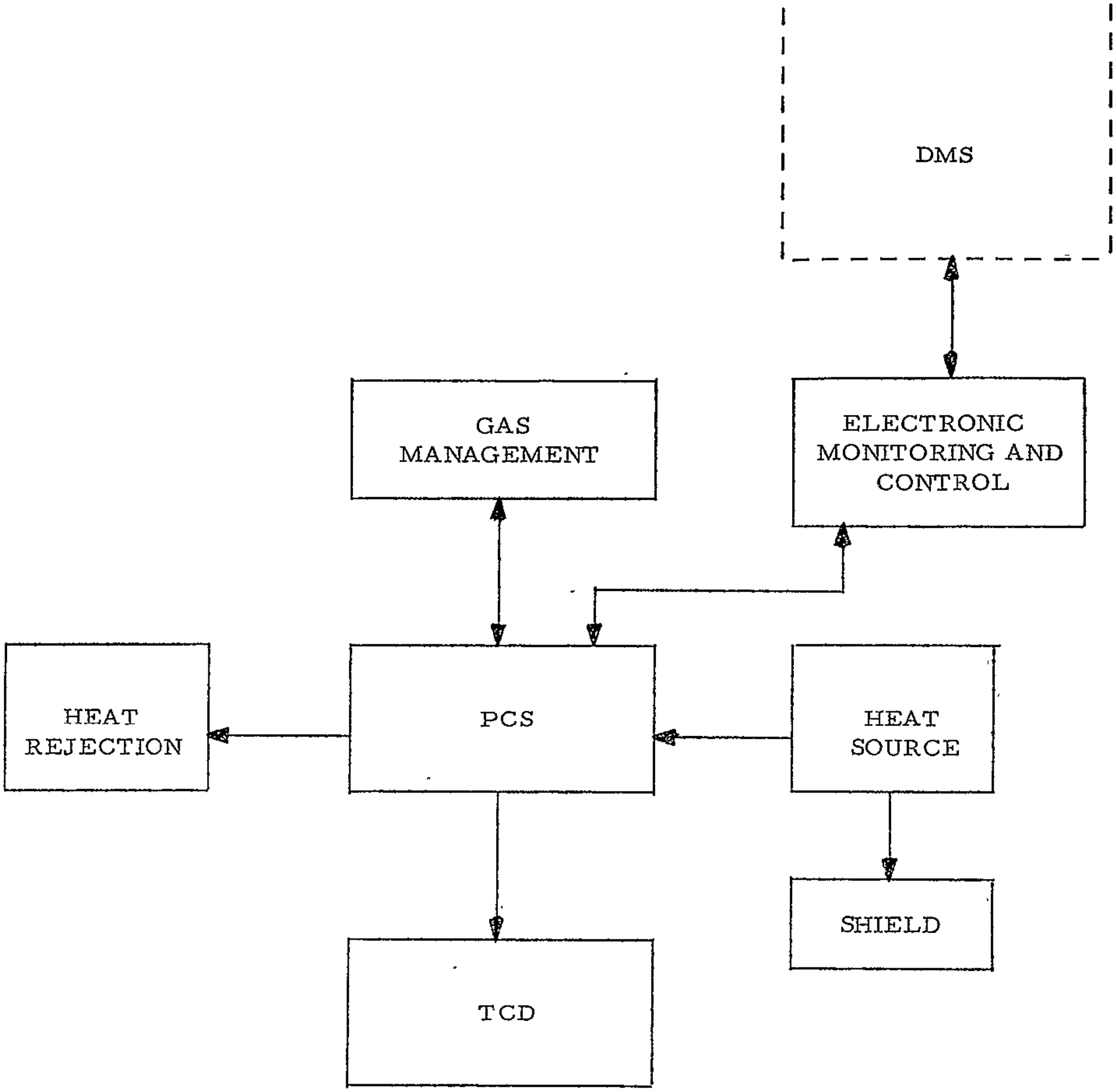
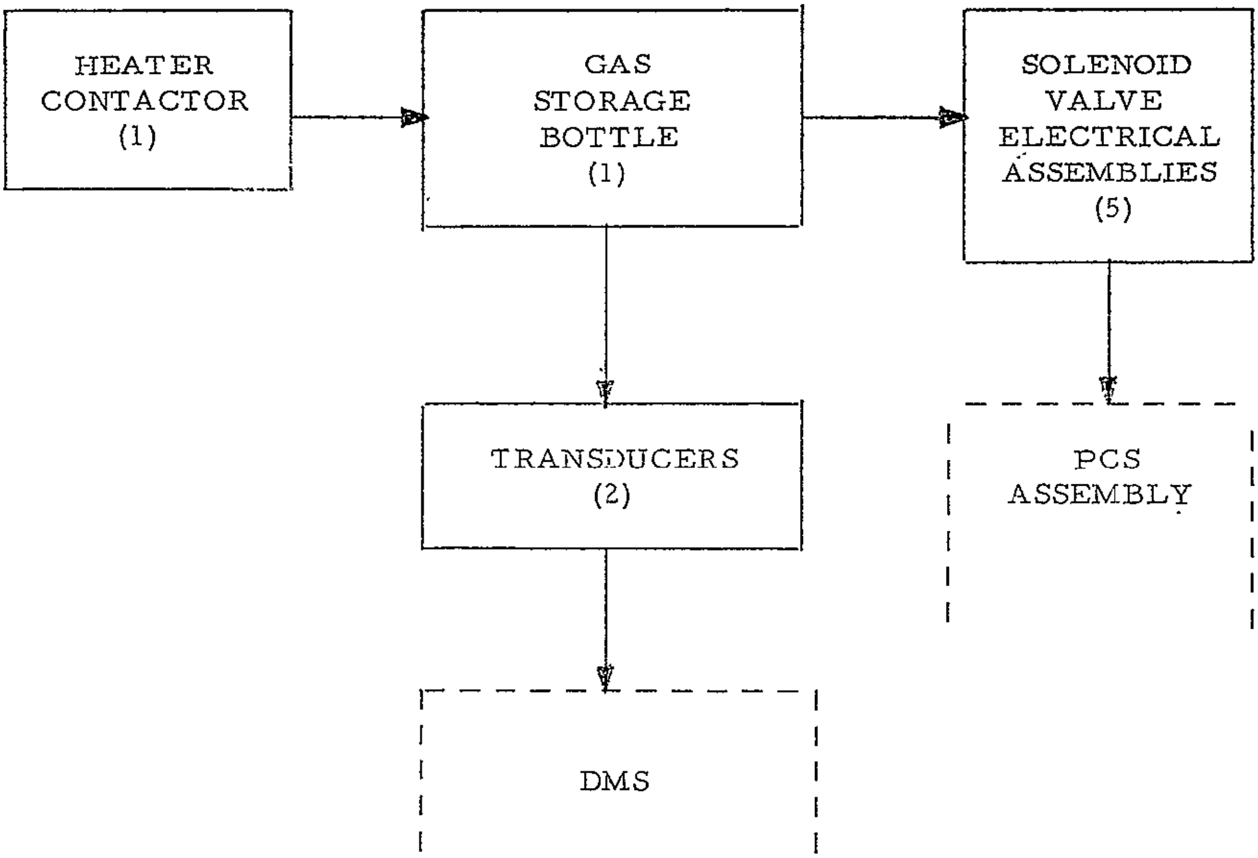


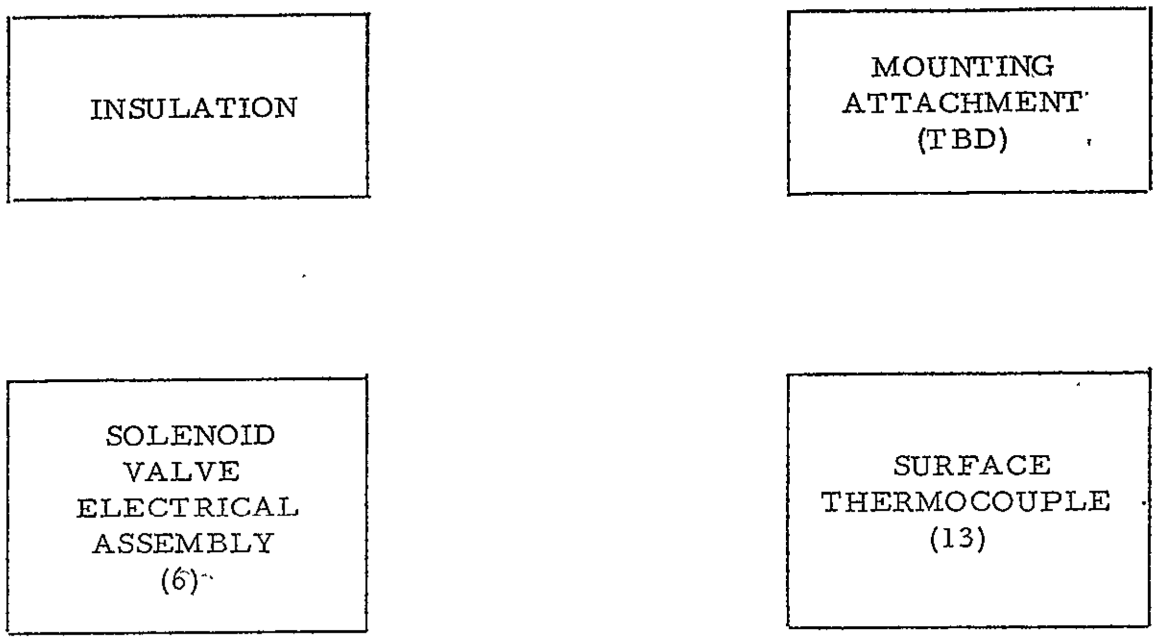
Figure 3-3 EPS Assembly Relationships, I/Br Configuration



Measurements	9	Per Assembly
Stimuli	0	
LRUs	9	

Required	2
Redundant	0
Total	2

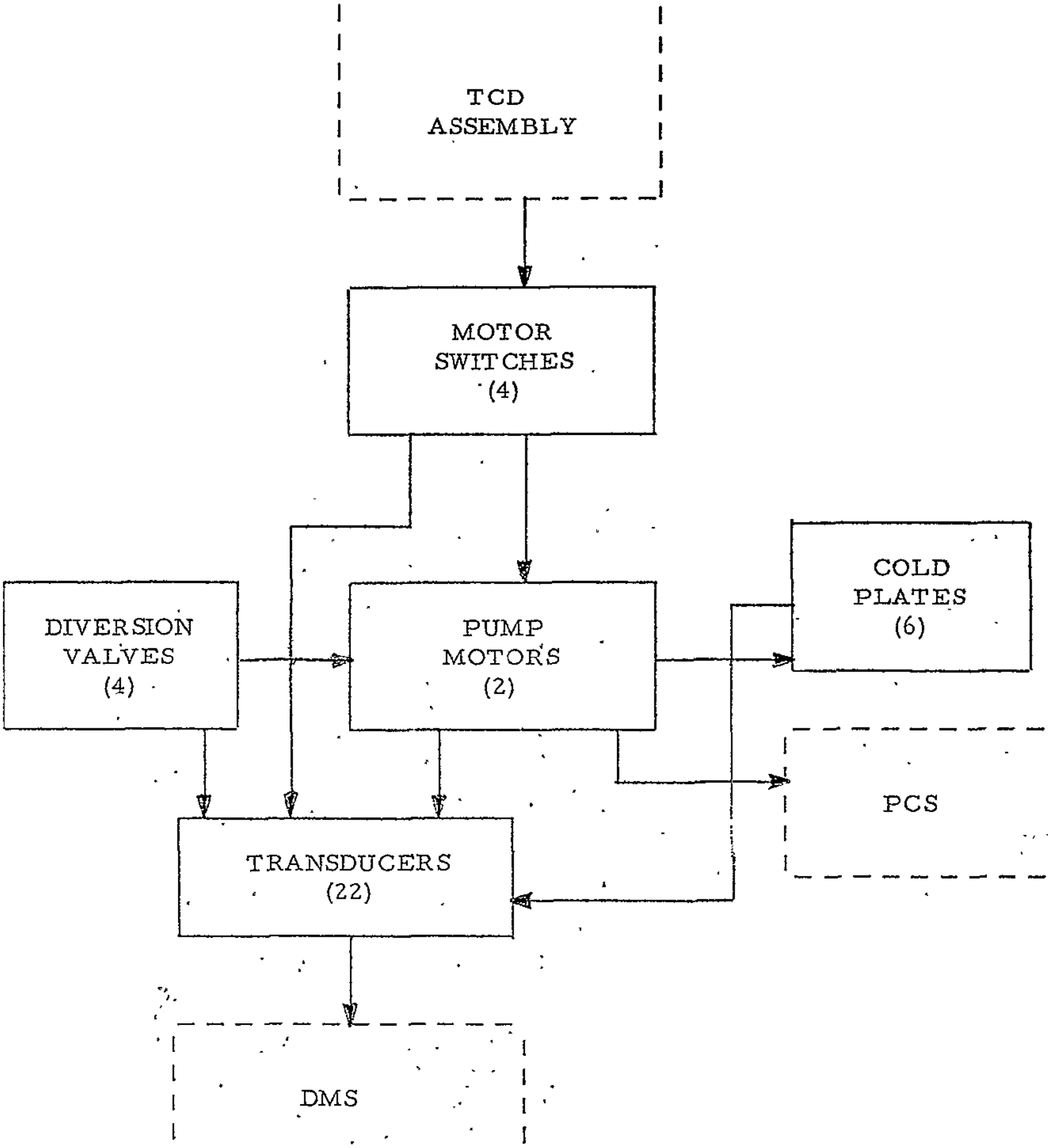
Figure 3-4 . LRU Interface Diagram, Gas Management Assembly



Measurements	20	Per Assembly
Stimuli	0	
LRUs	19+TBD	

Required	2
Standby	1
Total	3

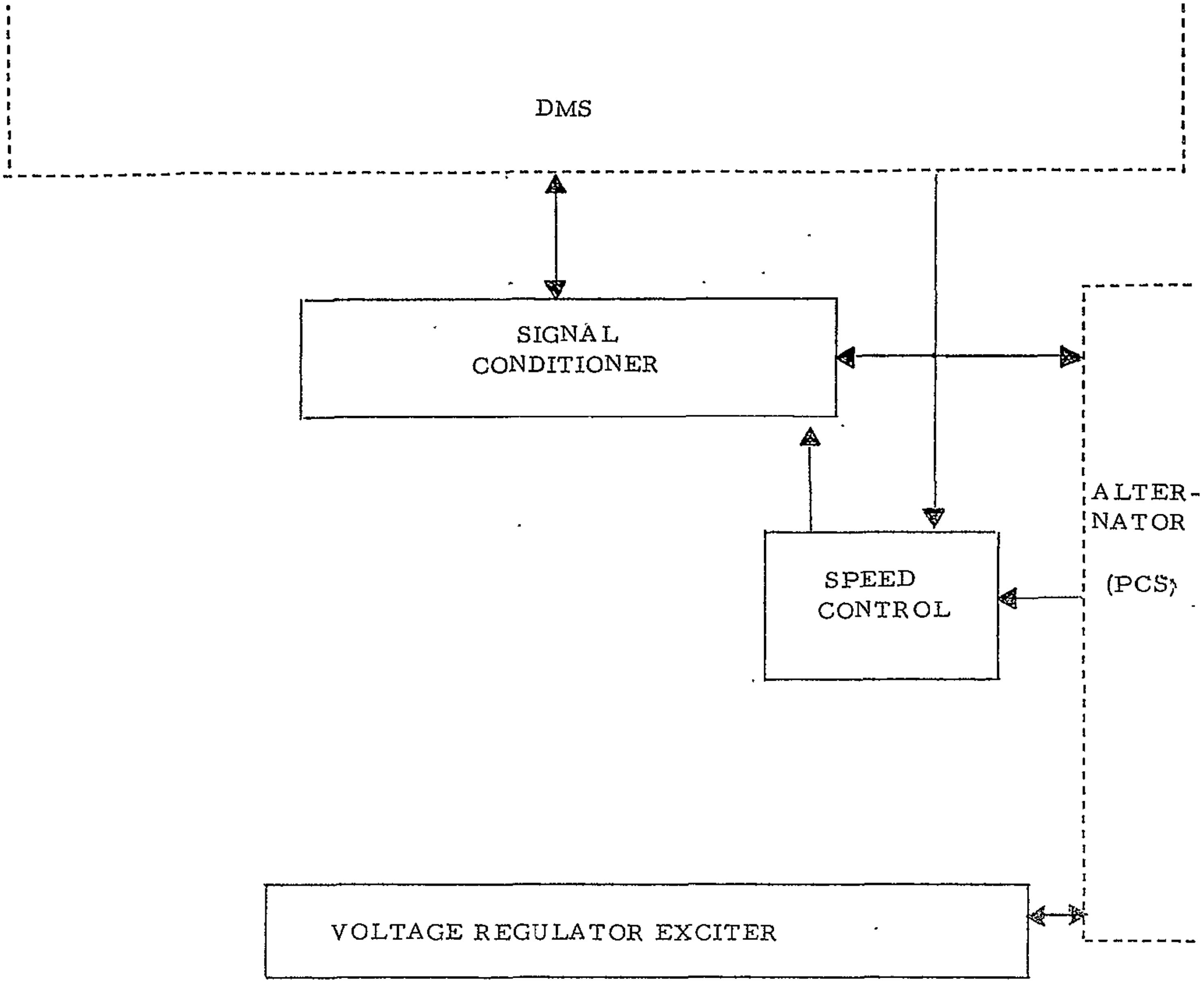
Figure 3-5 . LRU Interface Diagram, Power Conversion Subsystem



Measurements	26	Per Assembly
Stimuli	0	
LRUs	38	

Required	1
Redundant	1
Total	2

Figure 3-6. LRU Interface Diagram, Heat Rejection System Assembly



Measurements	15	Per Assembly	Required	1
Stimuli	3		Redundant	1
LRU's	5		Total	2

Figure 3-7. LRU Interface Diagram, Electronic Monitoring and Control Assembly

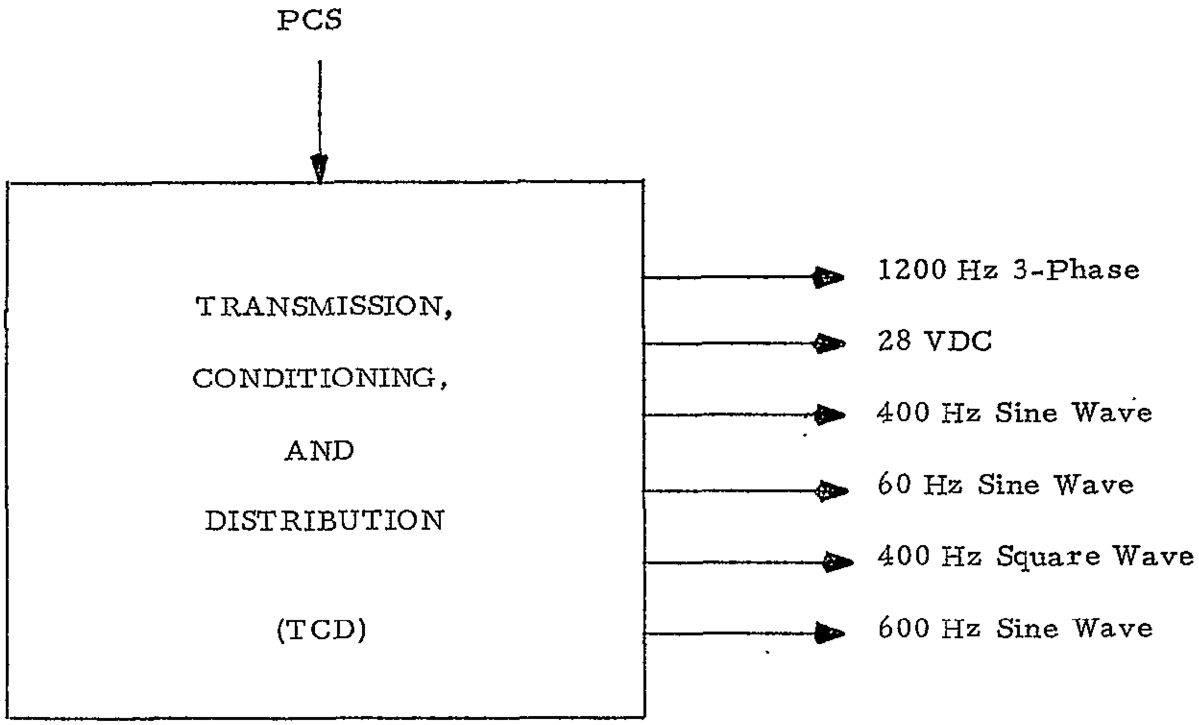
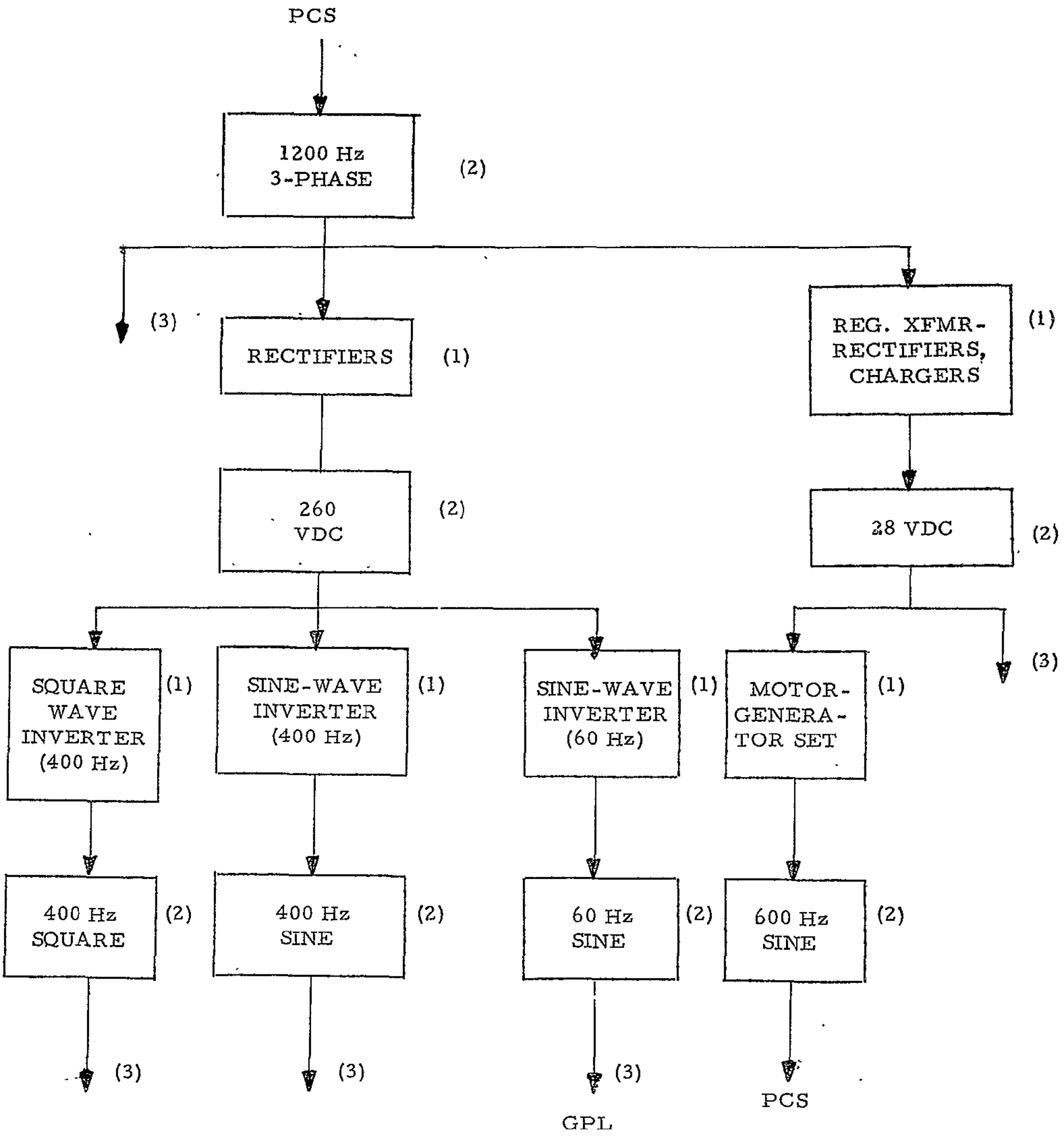


Figure 3-8. T/C/D (I/Br Configuration)



NOTES:
(1) Power Conditioning Equipment
(2) Distribution and Transmission Equipment (Buses, Breakers, etc.)
(3) To the Load

Figure 3-9 . Power TCD (Isotope/Brayton)

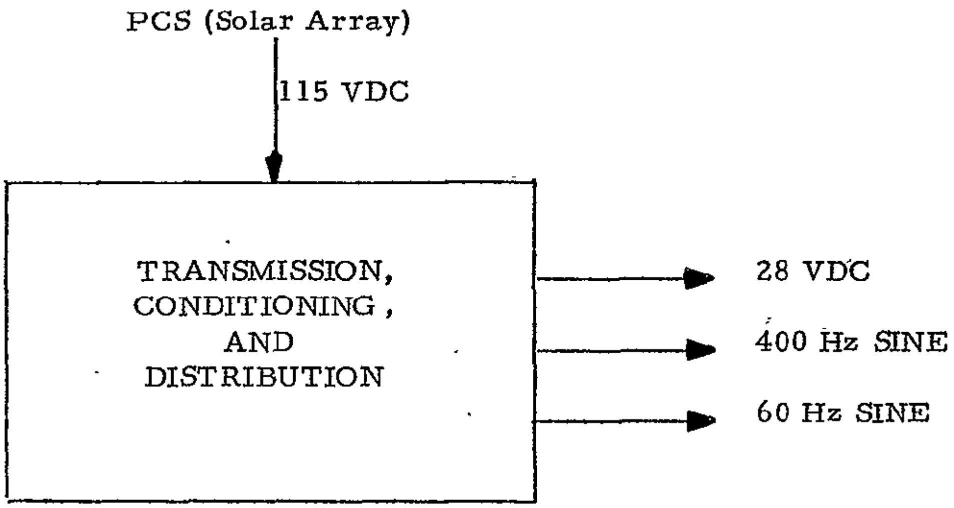


Figure 3-10 . T/C/D (Solar Array Configuration)

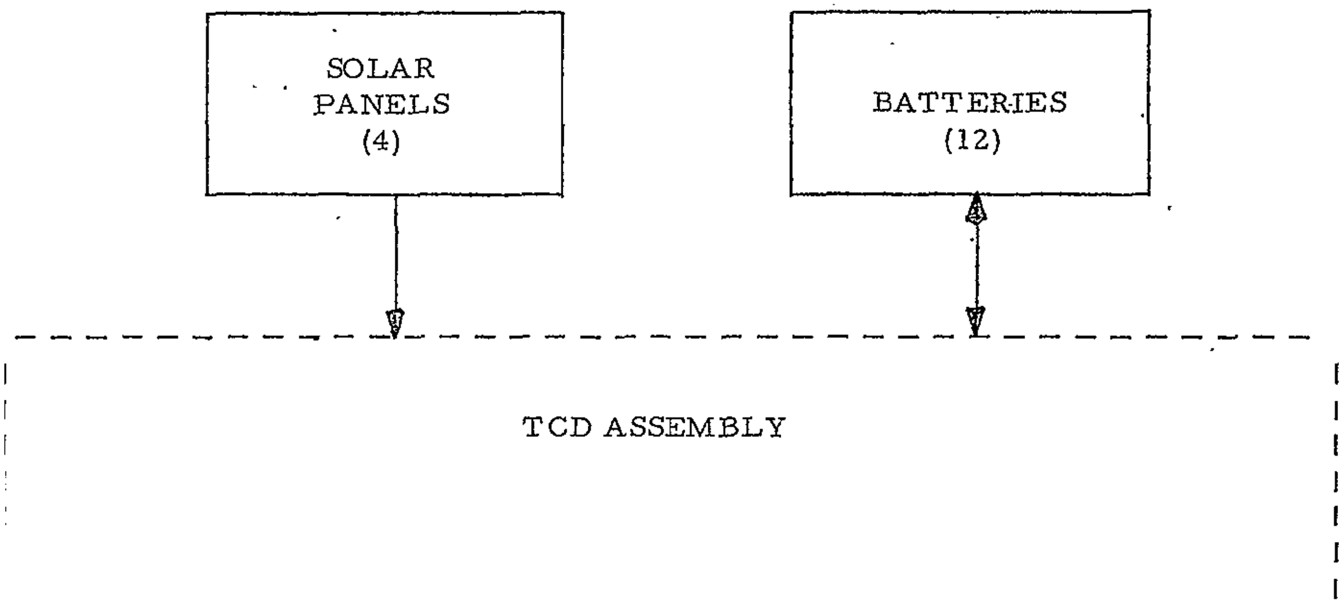


Figure 3-11. Power Source (Solar Array Configuration)

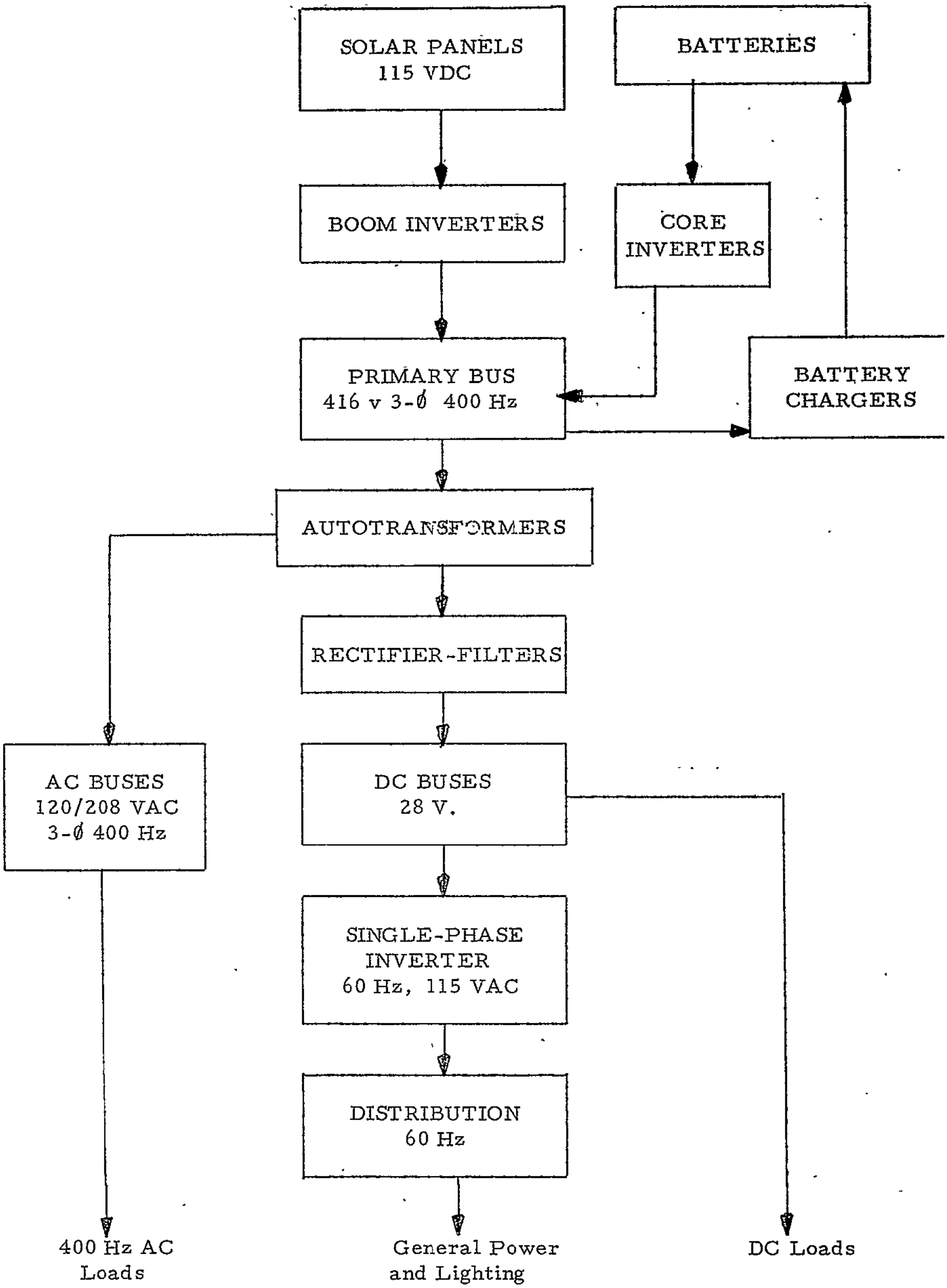


Figure 3-12. Assembly Diagram, Electrical Power Subsystem (Solar Array Configuration)

3.1.3 FLOWCHARTS

This section contains flowcharts of typical Electrical Power Subsystem Checkout Programs.

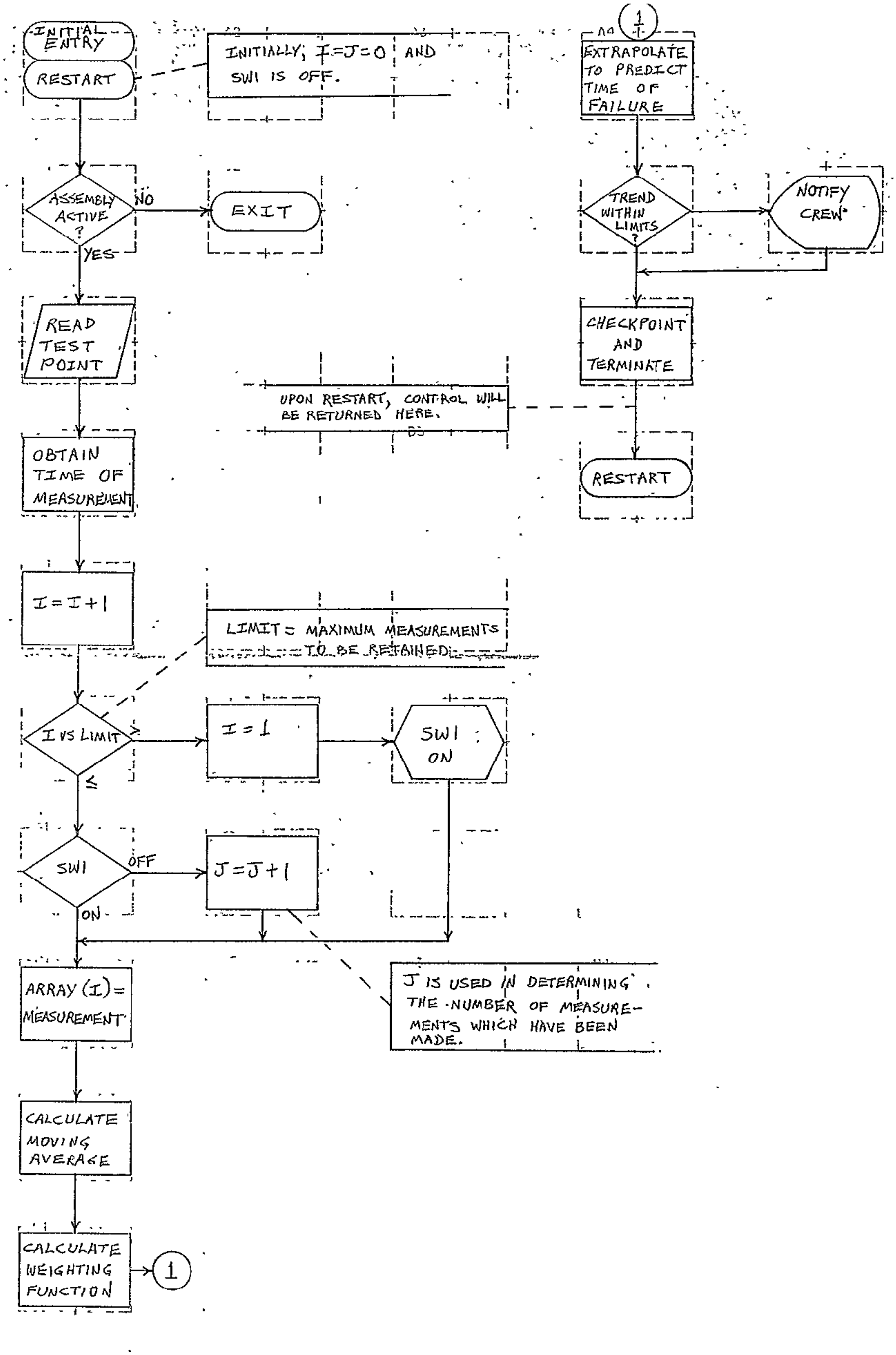


Figure 3-13. Extrapolation Trend Analysis

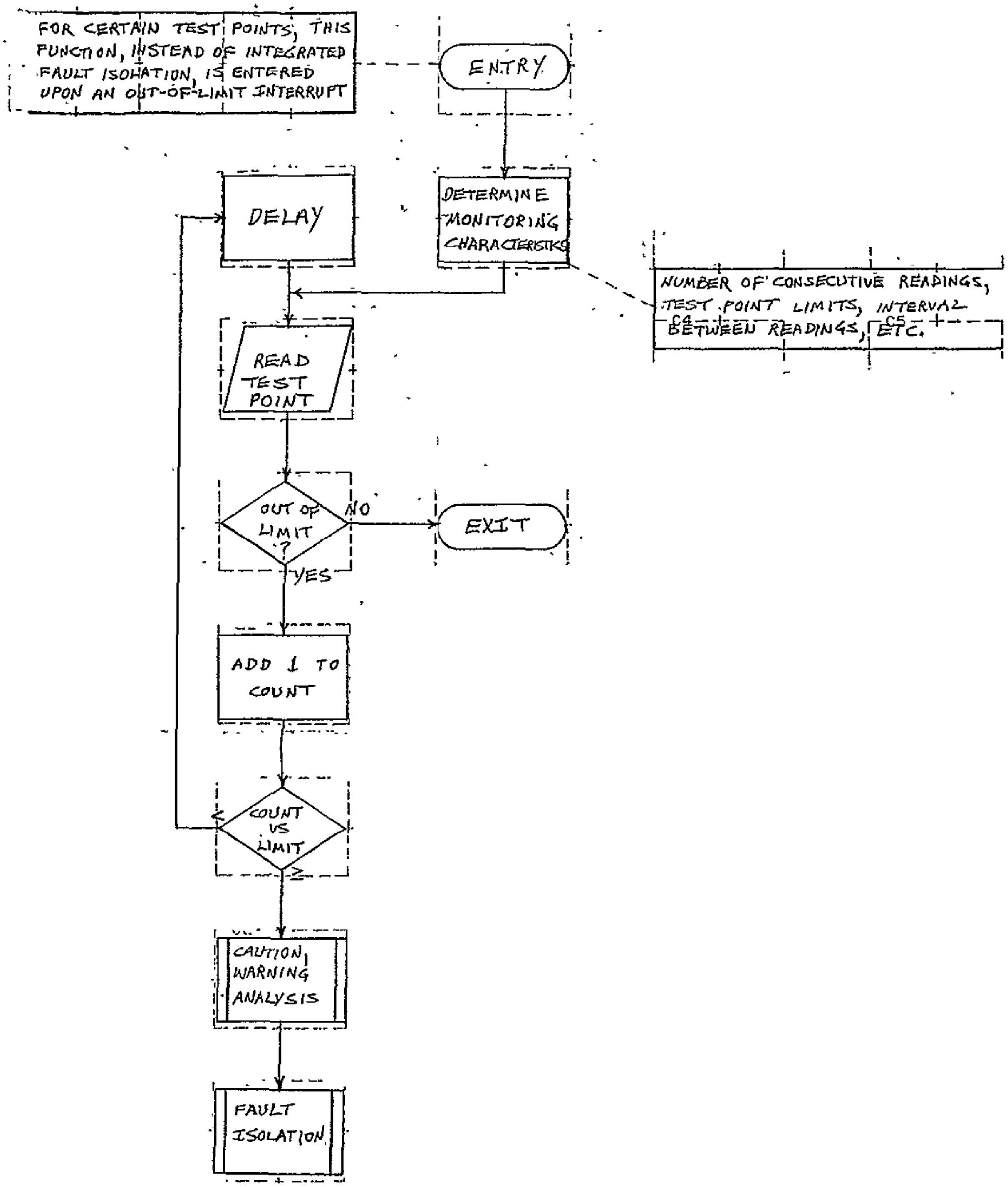


Figure 3-14. Status Monitoring

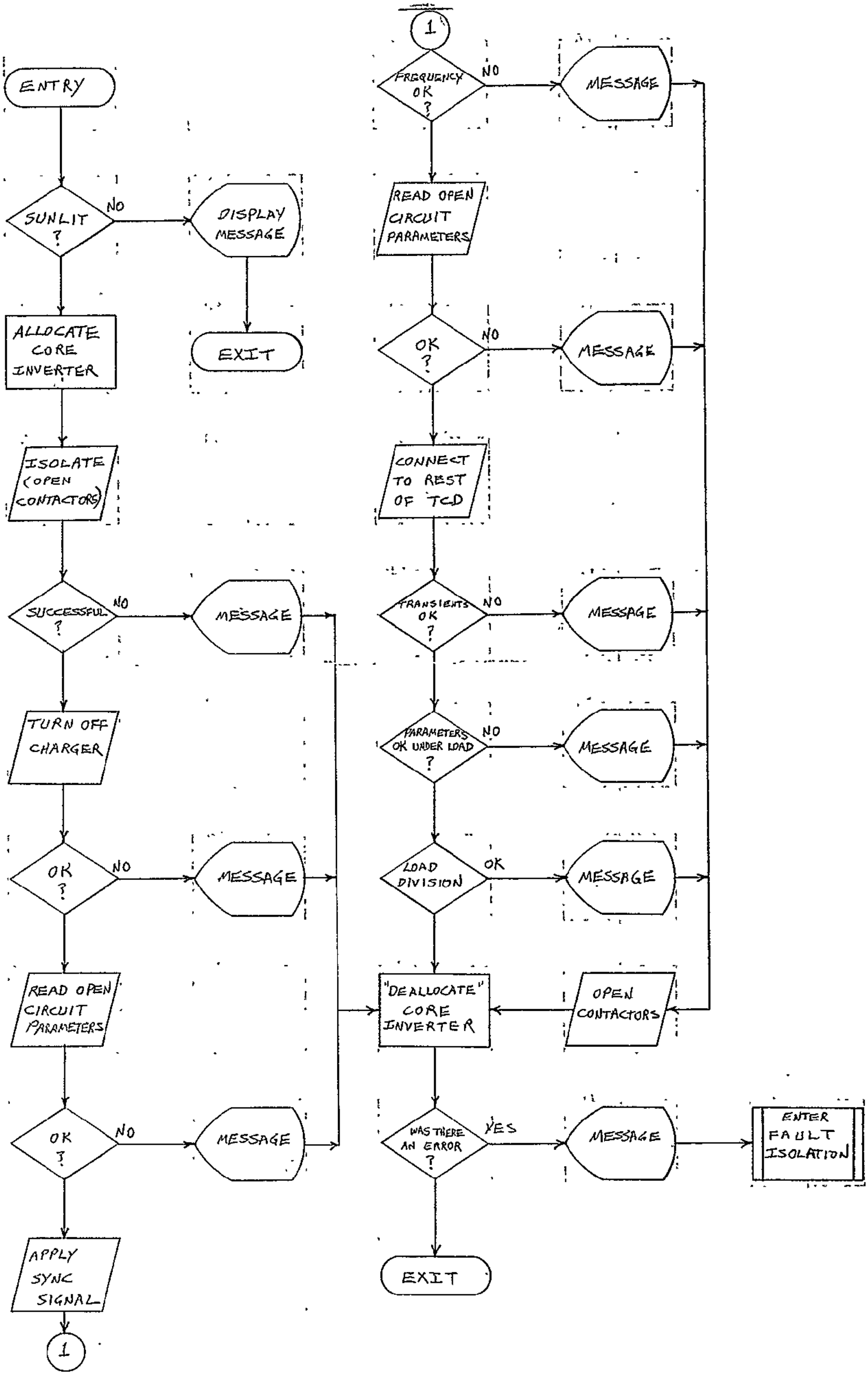


Figure 3-15. Core Inverter Periodic Test

3.1.4 DATA BASE REQUIREMENTS

Trend analysis programs of the EP Subsystem utilize the checkpoint/restart services of the executive; therefore requiring auxiliary storage for checkpoints and for data logging.

Because certain test points require special processing prior to fault isolation, use of a Limit Interrupt Table is indicated. This table contains the test point symbolic ID, and the name of any special routine such as a status monitoring test, to successively measure the test point, which is given control prior to fault isolation. Test points are entered in the table on an exception basis; fault isolation is entered directly if the ID is not found in the table.

A program parameter table to contain constants required by programs initiated automatically by the executive is required to contain such items as the number of trend analysis measurements to be retained, delay between measurements, etc.

The hierarchical complexity of the EPS increases the importance of the configuration table. In this table will be recorded the changes to the subsystem as a result of reconfiguration.

3.1.5 HUMAN PERFORMANCE

As described above, fault detection provides the indication of a problem and fault isolation locates it. The design of fault isolation modules is such that they will perform the same analysis if indication of a problem is received from the crew or ground. Similarly, the reconfiguration modules and the status monitoring modules may receive crew or ground initiation.

3.2 INTERFACE REQUIREMENTS

Although the checkout programs, language, and executive are designed to operate in a multiprocessor, there is no restriction as to the number of processors which must be available. In fact, a uniprocessor would be sufficient, provided enough main storage is available to contain the executive, program text, and data. The minimum Data Management Subsystem (DMS) configuration required for an EPS checkout function is as follows:

1 - Auxiliary Storage LRU
1 - Processor LRU
3 - Memory LRU's
1 - Data Bus Controller LRU
TBD - Data Bus Terminal LRU's
TBD - Remote Data Acquisition Unit LRU's
TBD - Stimulus Generation LRU's

This minimum configuration does not accomodate DMS failures.

The three memory LRU's are assumed to be utilized in the following manner: one for executive text, one for program text, and one for executive tables and program data. The number of RDAU's and Stimulus Generation Units (SGU's) will be determined by the function and the design details of the EPS. The number of data bus terminals is determined from the number of RDAU's and SGU's.

3.2.1 INTERFACE DIAGRAM

The relationship among the various EPS checkout functions and their means of initiation are shown in Figure 3-16.

3.2.2 DETAILED INTERFACE DEFINITION

Figures 3-17 through 3-21 indicate the interface requirements for the individual functions.

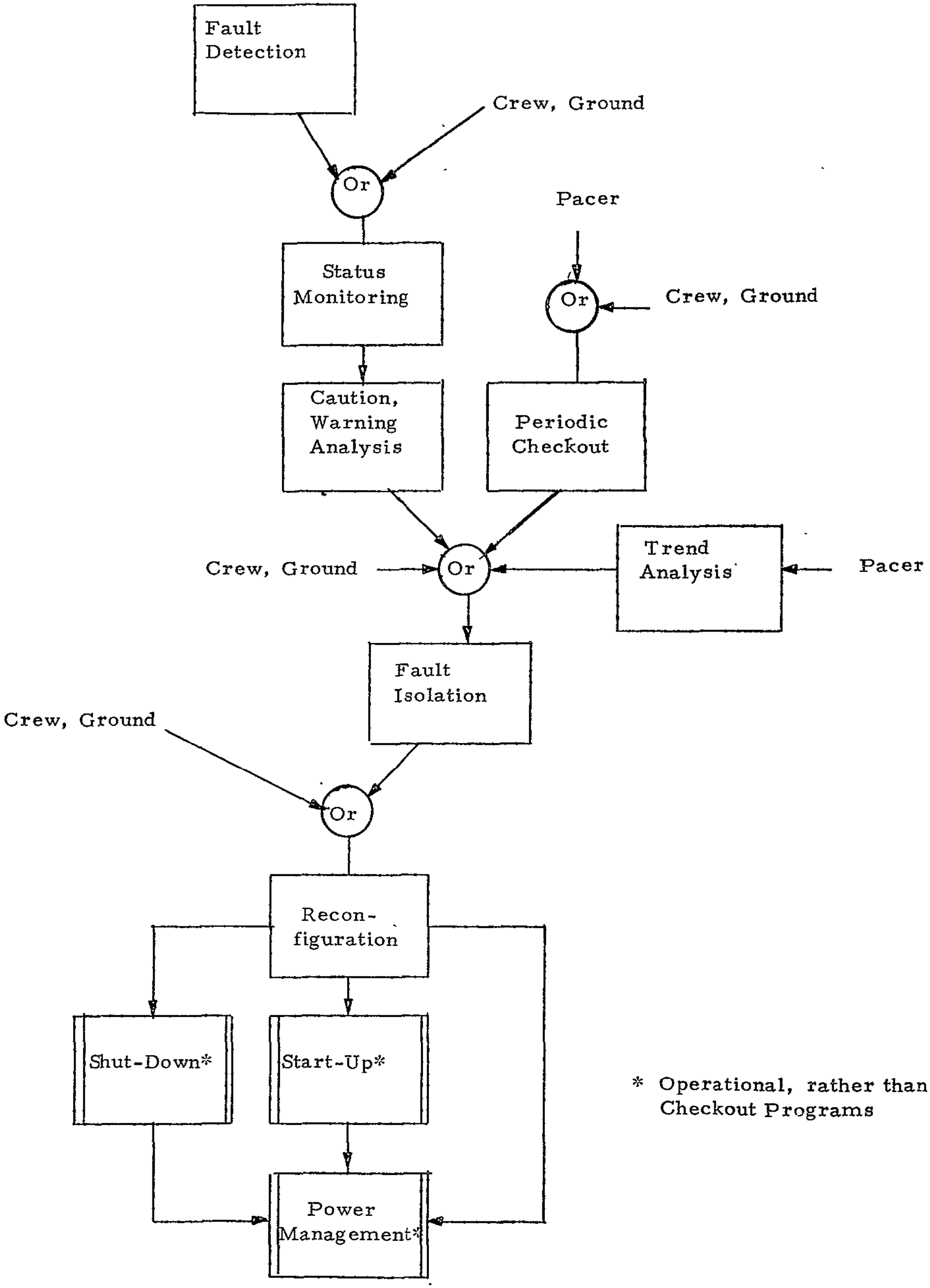


Figure 3-16 . General Function Interface

(The trend analysis function receives control from an RDAU interrupt or from the Pacer.)

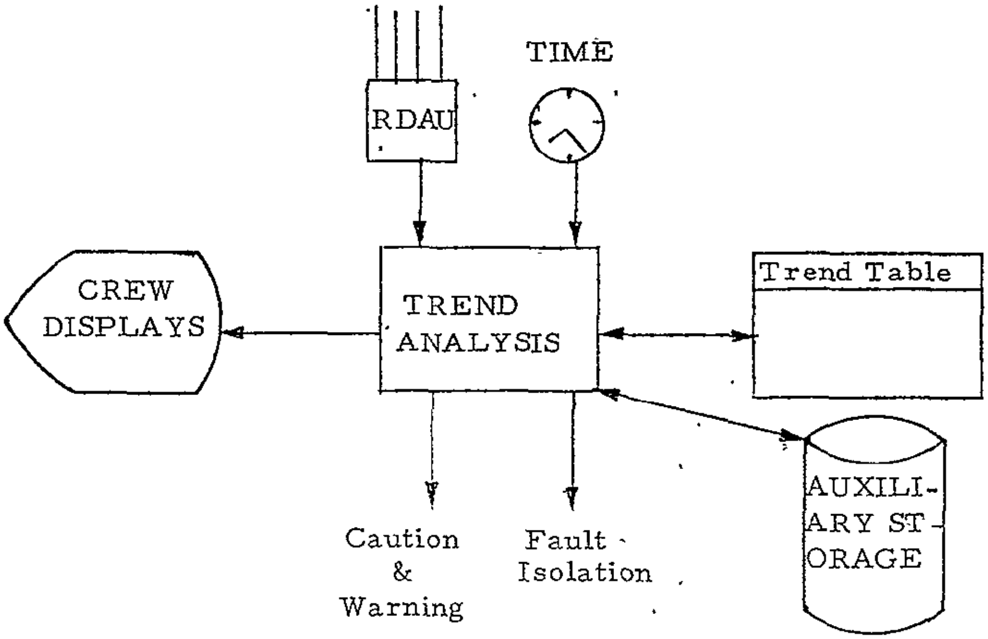


Figure 3-17. Trend Analysis Interface

(The Status Monitoring function is normally initiated by the executive as a result of an out-of-limit condition detected by an RDAU for certain selected test points. The function can also be initiated by a crew or ground).

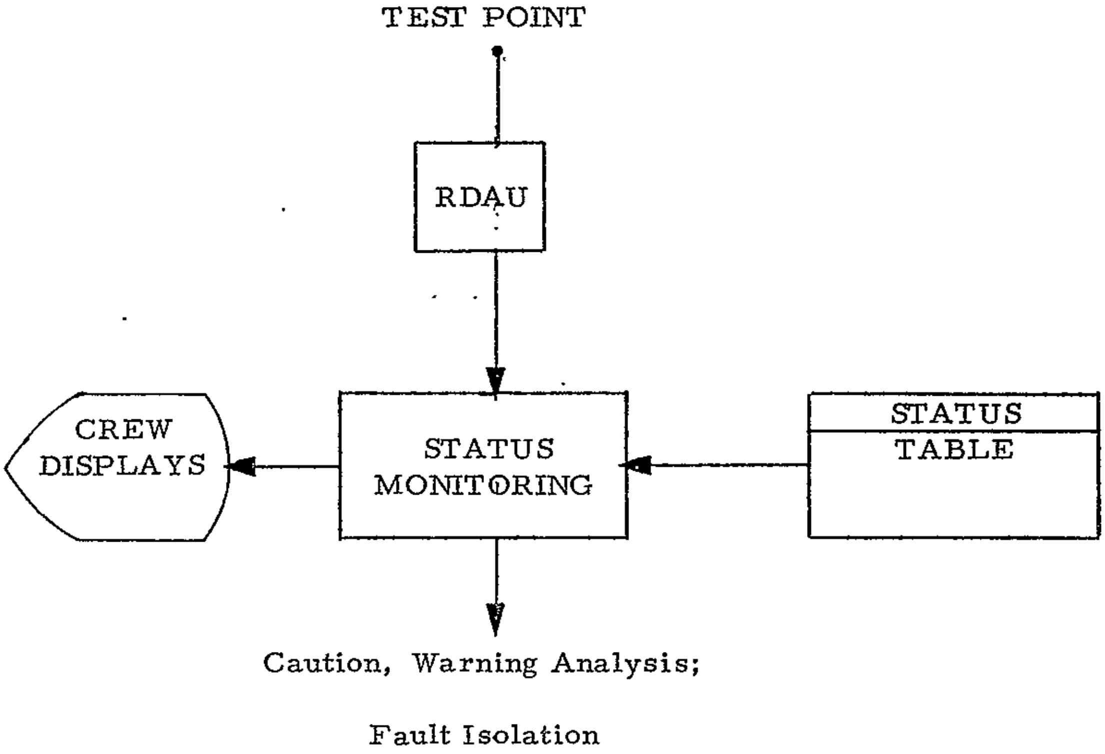


Figure 3-18. Status Monitoring Interface

(The periodic checkout function is normally initiated by the crew; however, it is possible to schedule the test automatically by utilizing the pacer (an executive service which utilizes the interval timer and a table of events)).

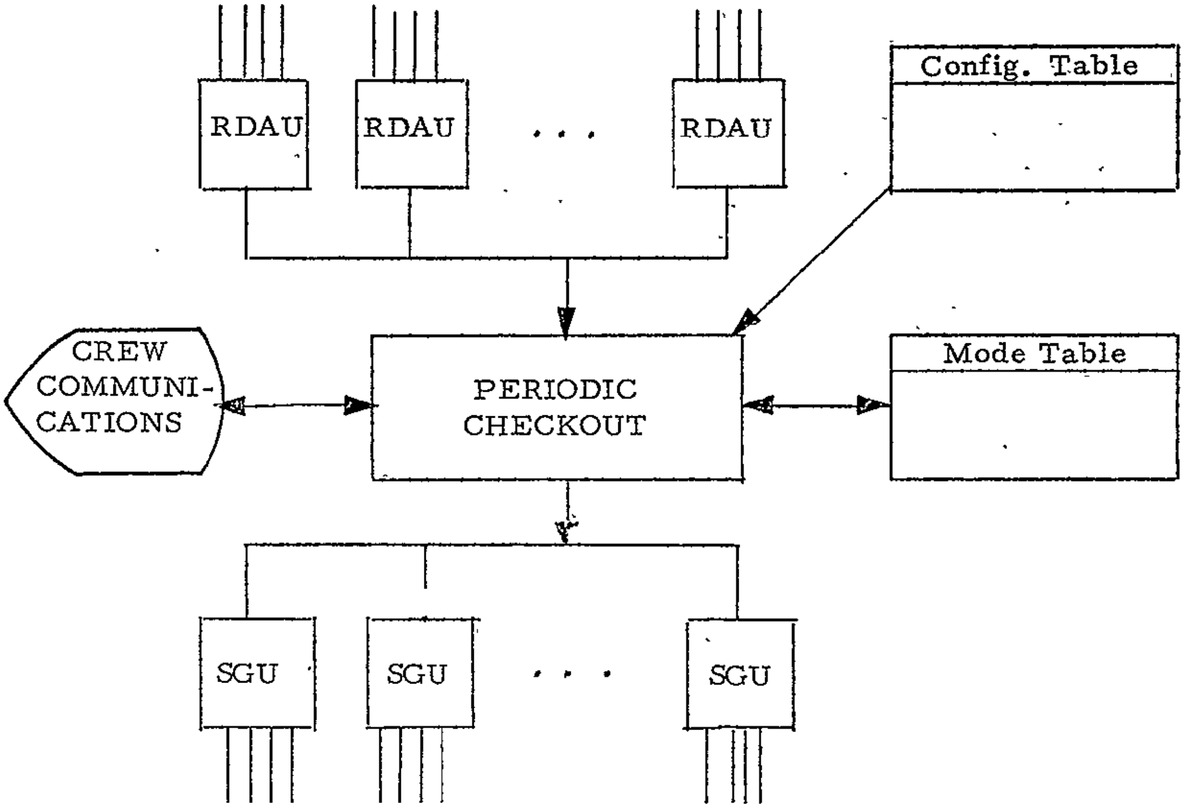


Figure 3-19. Periodic Checkout Interface

(Fault isolation modules for the EPS Subsystem receive control from the subsystem level fault isolation program and interface with the crew via the display units).

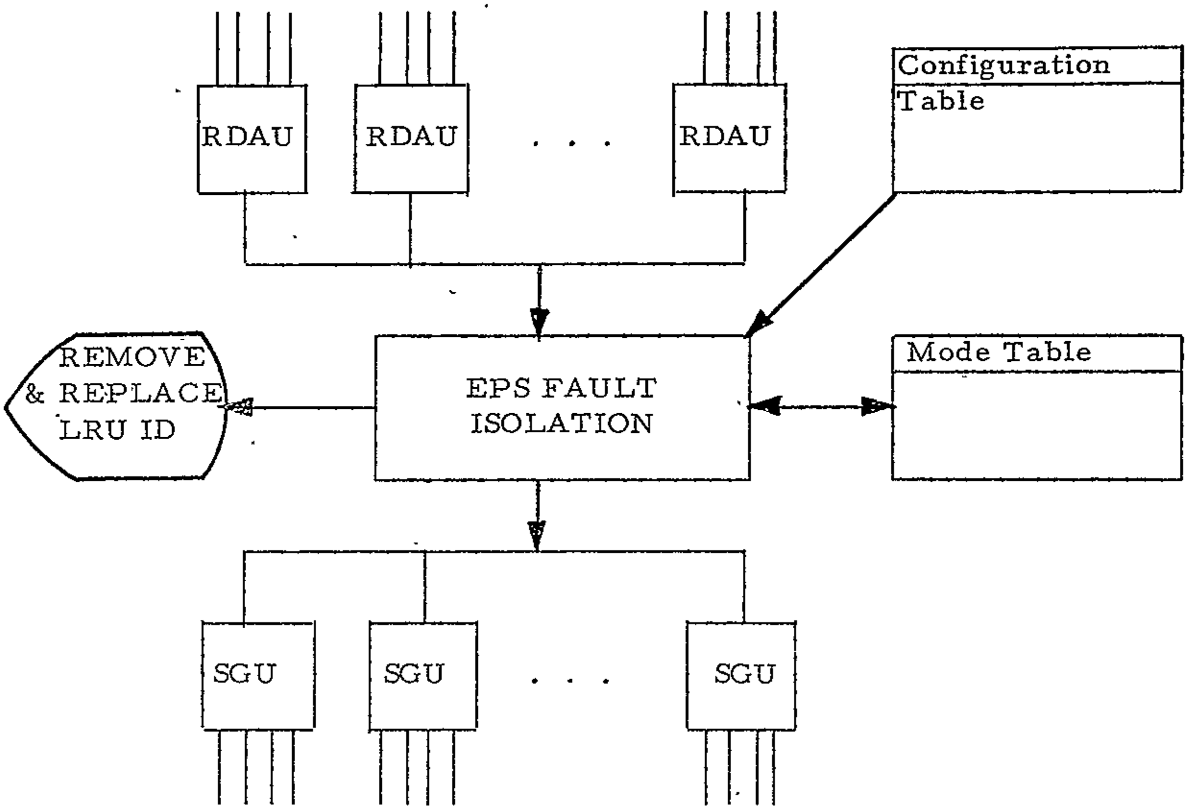


Figure 3-20. Fault Isilation Interface

(The reconfiguration function receives control from a crew command, or from the fault isolation modules).

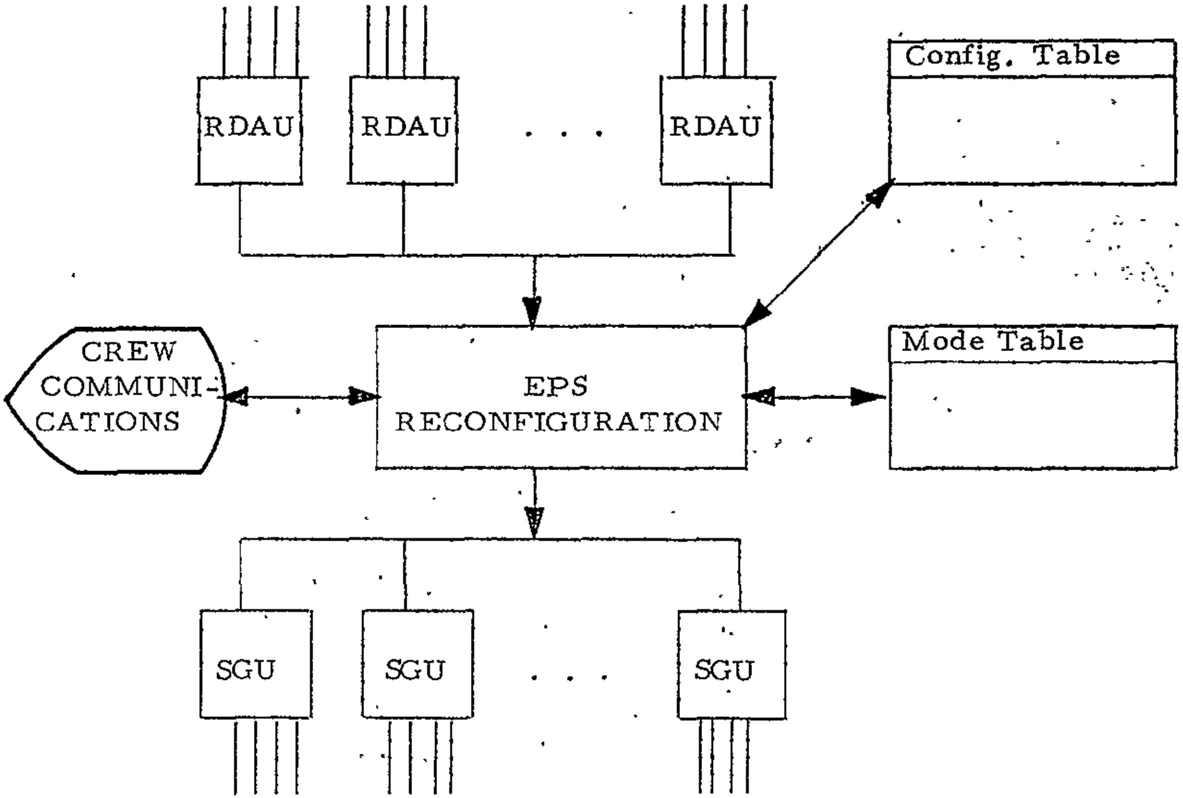


Figure 3-21. Reconfiguration Interface

4.0 SUBSYSTEM SUMMARY

The checkout requirements study of the Electrical Power Subsystem results in requirements regarding language, executive, and the Data Management Subsystem which are discussed in the following paragraphs.

4.1 LANGUAGE ANALYSIS

In addition to the language requirements identified in the interim reports for the Structure and GNC Subsystems, the Electrical Power Subsystems require extensive arithmetic capability in the language in order to incorporate energy balance equations in fault isolation and periodic checkout functions. The I/Br design requires this to a greater extent than does the SA design.

The language usage and checkout module identification is indicated in Table 4-1 through 4-4 for the Isotope/Brayton power system, and in Table 4-5 through Table 4-8 for the Solar Array power system.

4.2 EXECUTIVE ANALYSIS

Most executive services identified for the Structure and GNC Subsystems are applicable to the Electrical Power Subsystem in both the Isotope/Brayton and the Solar Array designs. These services include:

- Data base access control
- Initiation of checkout programs at specified time intervals
- Pre-initiation parameter analysis
- Symbolic test point translation
- I/O services
- Task management

Additional executive requirements are discussed in the following paragraphs.

Parameter Assembly - When a program is initiated automatically by the executive, it is necessary to assemble a parameter list required by the program. Such lists may be stored and updated in the parameter table discussed in Section 3.1.4 by other programs, including the executive. A trend analysis program is an example of a program requiring parameter assembly. The trend program may use parameters to indicate the number of measurements to be retained, whether a stimulus is required, trend equation constants, etc.

Special Limit Programs - Certain test points may require special routines, prior to entering fault isolation, such as status monitoring. Thus, prior to giving control to fault isolation when an RDAU limit check interrupt is received and verified, the test point ID must be used as an augment to a table containing indications of special routines. If such an indication is affirmative, the special routine is given control.

Trend Sunchronization - Because certain functions are performed once per orbit for the Solar Array design, there is a requirement to be able to specify the time at which trend analysis is to begin, as well as the rate at which it is to proceed. This capability is provided by the executive utility which updates the rate table.

Trend Checkpoint - The trend analysis functions for both the I/Br and the SA design utilizes the checkpoint/restart services of the executive. After measurement and associated processing has been completed, a checkpoint is requested by the trend program. When the next trend cycle is required, restart is issued to retrieve the intermediate results stored previously.

4.3 I/BR AND SA COMPARISON

The checkout program function categories of trend analysis, status monitoring, periodic checkout, fault isolation, and reconfiguration are applicable to both the Isotope/Brayton and the Solar Array designs. No calibration requirements have been identified.

The trend analysis functions required for I/Br are basically data collection and extrapolation techniques. The SA trend analysis functions employ a telemetering method, in addition to the same techniques identified for I/Br. A quantitative comparison between trend analysis requirements for the two designs is as follows!

	Measurement Categories	Number of Test Points	Storage Requirements	Maximum Rate
Isotope/Brayton	31	160	66K	4/hour
Solar Array	18	446	147K	4/minute

It is feasible for trend analysis to be performed by general-purpose executive utilities, written in machine language. However, with array manipulation capabilities, the trend functions can be implemented as high-level language modules. A trade study is appropriate to consider justification for adding array-handling facilities to the language solely for trend analysis.

A status monitoring function common to both I/Br and SA is the successive measuring technique. This is the sole method required for SA. In addition, I/Br requires a voting technique for parallel redundant parameters, and a multi-level limit setting method for parameters which are classified first as caution, and then as warning indicators.

Certain periodic checkout functions for I/Br are more complex than for SA because of the greater complexity of the power source.

The fault isolation functions for I/Br require arithmetic capabilities in the language because of the extensive use of energy balance equations, to distinguish between failure of a tranducer LRU and a true out-of-tolerance condition. Only rudimentary arithmetic requirements are indicated for SA fault isolation. Both designs exhibit complex interrelationships among their various assemblies, implying the need for careful design and maintenance of the configuration, mode, and status portions of the data base.

The reconfiguration functions of I/Br require interface with start-up and shut-down functions when applied to the combined rotating unit and other assemblies of the power conversion system. The SA design requires similar interfaces to comparatively less complex functions concerned with boom and panel deployment. Both designs require interface between the reconfiguration and power management functions.

4.4 PROGRAM SIZING

In order to evaluate the effect of the Electrical Power checkout programs on the Data Management Subsystem, individual program modules are identified in Tables 4-1 through 4-10, and listed with certain characteristics which must be considered together in an overall analysis. For trend analysis modules which are implemented in machine language, the performance parameters are listed in terms of average processor time, I/O time, and storage words in Tables 4-1 and 4-2. The characteristics of the high level language modules are tabulated according to the frequency with which the program executes the language elements. After estimating the DMS requirements of the individual language elements, the element utilization data in Tables 4-2 through 4-4 were reduced to DMS performance parameters as indicated in Tables 4-5 through 4-7.

Table 4-1. I/Br Trend Analysis Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Data Collection	1/W	4	1.25	186	180	187	1188
Data Collection	4/D	5	1.25	186	180	187	1188
Data Collection	4/H	58	1.25	186	180	187	1188
Extrapolation	1/M	4	3.10	186	230	189	300
Extrapolation	1/W	9	3.10	186	230	189	300
Extrapolation	4/D	80	3.10	186	230	189	300

Table 4-2. SA Trend Analysis Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Telemetry	Var.	320	0.550	110	120	111	1000
Data Collection	4/H	36	1.25	186	180	187	1188
Data Collection	4/M	28	1.25	186	180	187	59220
Extrapolation	4/D	38	3.10	186	230	187	300
Extrapolation	4/H	12	3.10	186	230	187	300

Table 4-3. I/Br Status Monitoring Language Usage Estimates

			LANGUAGE ELEMENTS														
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	Successive Tests	118	5			3	2	1			1		1	3	1		
	Redundant Compare	13	2			3	2	1			1		1	3	1		
	Threshold Adjust	3	1			3	2	1		2	1	2	1	3	1		

Table 4-4. I/Br Periodic Checkout Language Usage Estimates

			LANGUAGE ELEMENTS													
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	Drive Mechanisms	2	2	1	2	1		1					1	1	1		
	IRV System	2	6		2	2		3				1	1	2	1		
	Inverters	1	82	36	20	60	30	10	36			4	1	60	1		
	Battery Charters	10	9	4	2	6	4	4	2			1	1	6	1		
	Selector Switches	TBD	---	---	---	---	---	---	---	---	---	---	---	---	TBD		
	Motor-Generators	2	22	8	2	16	8	6	4	2	2	3	1	16	1		
	Transformer-Rectif.	5	5	1	2	2		2					1	2	1		
	Buck Regulators	10	6	4	2	2	2	2	1			1	1	2	1		
	Circuit Breakers	1	208	208	416	100		50			16	4	1	100	1		
	Contactors	TBD	---	---	---	---	---	---	---	---	---	---	---	---	TBD		
	Switches	1	81	81	162	40		6			4	2	1	40	1		
	Differential Relays	1	6	6	12	4	3	6	2		5	6	1	4	1		
	Reverse-Curnt. Relay	1	18	18	36	9		3	18		4	2	1	9	1		

Table 4-5. I/Br Fault Isolation Language Usage Estimates

			LANGUAGE ELEMENTS														
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	Power Conversion	3	30		2	15	18	23		4	4	6	1	15	1		
	Gas Management	2	14		2	12	6	12	5	3	4	2	1	12	1		
	Heat Rejection	2	35		4	20	20	40	18	6	7	4	1	20	1		
	Heat Source	2	24		2	18	10	4				1	1	18	1		
	1200 Hz Bus	1	161	115	8	100	50	20		4	20	6	1	100	1		
	260 VDC System	1	61	34	24	30	10	40	10	2	6	8	1	30	1		
	Square Wave	1	144	40	20	30	30	25	4	4	10	6	1	30	1		
	Sine Wave	1	165	60	20	30	30	25	4	4	10	6	1	30	1		
	GPL	1	50	10	4	25	8	12		2	4	3	1	25	1		
	28 VDC System	1	338	177	40	100	130	60	30	40	30	10	1	100	1		
	Motor-Generator	1	72	44	8	40	10	6	10	4	6		1	40	1		
	Monitor & Control	2	20	8	2	10	4	6	3	2	5		1	10	1		

Table 4-6. I/Br Reconfiguration Language Usage Estimates

			LANGUAGE ELEMENTS													
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	Power Conversion	1	24	12	6	10	8	4	8	6	4	8	1	10	1		
	1200 Hz Network	1	30	30	30	15		8		4	6	10	1	15	1		
	H-V Rectifiers	1	16	4	4	6		4	4	4	2	6	1	6	1		
	260 VDC Network	1	16	16	4	4		4		4	4	4	1	4	1		
	Regulator-Transfor.	1	12	12	12	6		2	6	6	4	6	1	6	1		
	Battery Charger	1	30	10	10	5		2	10	2		4	1	5	1		
	Buck Regulator	1	20	10	10			2	1	2		2	1		1		
	Square Wave Invert.	1	4	4	4			2	1	4			1		1		
	Sine Wave Inverters	1	4	4	4			2	1	4			1		1		
	60 Hz Inverters	1	4	4	4			2	1	4			1		1		
	28 VDC Network	1	40	40	20	10		2		20	12	6	1	10	1		
	Square Wave Network	1	20	20	10	5		2		10		4	1	5	1		
	Sine Wave Network	1	20	20	10	5		2		10		4	1	5	1		
	60 Hz Network	1	20	20	10	5		2		10		4	1	5	1		
	Batteries	1	10	20	10			2		10			1		1		
	600 Hz Network	1	20	20	12	6		2		4			1	6	1		

Table 4-7. SA Status Monitoring Language Usage Estimates

		▨	LANGUAGE ELEMENTS														
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	Successive Tests		5			3	2	1			1						

Table 4-8. SA Periodic Checkout Language Usage Estimates

		LANGUAGE ELEMENTS													
	Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
60 Hz Inverters	2	7	3	2	4	3	2	1		2	1	1	4	1		
Battery Chargers	18	11	2	2	5	4	3	2			1	1	5	1		
Core & Boom Invert.	1	80	36	72	36	12	4	36	12	10	6	1	36	1		
Rectifier-Filters	4	5	3	2	2	1	2	2	1		2	1	2	1		
Power Contactors	1	88	88	88	44		4					1	44	1		
Differential Protec.	1	52	52	52	52		6		4	5	2	1	52	1		
Array Circuit I-V	1	400			400	400	4			10		1	400	1		
Battery Controls	1	12	12	12	6	6	4			4	2	1	6	1		
Cell Recondition	1	72	36		72		3					1	72	1		
Cell By-Pass	1	1800	900		900		6		4			1	900	1		

Table 4-9. SA Fault Isolation Language Usage Estimates

		LANGUAGE ELEMENTS															
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	Solar Panel	4	100	1	8	8		4		4	4	4	1	8	1		
	Boom Inverter	1	48	24	4	12		6			6	2	1	12	1		
	Battery Charger	1	100	50	26	13	13	8	13	4	10	3	1	13	1		
	Core Inverter	1	24	12	2	6		3		2	3	2	1	6	1		
	Primary Bus	1	44	32	12	12	12	6				2	1	12	1		
	400 Hz Network	1	24	12	12	6	4	4		4	2	1	1	6	1		
	Autotransformer	4	18	5	2	4	4	4		2	3	2	1	4	1		
	Rectifier-Filter	1	20		4	4		2		1	2	1	1	4	1		
	28 VDC Network	1	300	200	40	100	130	60	30	40	30	10	1	100	1		
	60 Hz Inverter	2	8	4	2	4		2	2	2	2	1	1	4	1		
	60 Hz Network	1	50	10	4	25	8	12		2	4	3	1	25	1		
	Source, Integrated	1	80			40					12		1	40	1		
	TCD, Integrated	1	80			40					12		1	40	1		
	Batteries	1	24	12	12	6		4		12	4	12	1	6	1		

Table 4-10. SA Reconfiguration Language Usage Estimates

			LANGUAGE ELEMENTS														
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	Solar Panel	1	8	8	4	6	4	2		8	4	1	1	6	1		
	Inverters	1	12	12	6	8	5	4	2	4	4	2	1	8	1		
	Chargers	1	8	4	4	6		2	4	2	6	1	1	6	1		
	Primary Bus	1	10	10	10	10		8		4	6	6	1	10	1		
	400 Hz Network	1	30	30	20	20	10	10		10	12	6	1	20	1		
	Autotransformer	1	6	6	2	4	4	4		2	3	2	1	4	1		
	Rectifier-Filter	1	5	4	2	2		2	1	4	2	1	1	2	1		
	28 VDC Network	1	20	20	20	20		4		5	6	3	1	20	1		
	60 Hz Inverter	1	4	4	4			2	1	4			1		1		
	60 Hz Network	1	20	20	10	5		2		10		4	1	5	1		
	Batteries	1	10	20	8	1	1	2		2	3	1	1	1	1		

The column headings in Tables 4-11 through 4-18 have the following meanings:

PROGRAM MODULE:	The name of the program.
RATE:	Frequency with which this program module will be executed for each application.
APPLICATIONS:	Number of uses for the module. As an example, if four identical assemblies required a periodic test once per hour, the number of applications would be four and the rate 1/H. Viewed from the multiprocessor, the traffic rate would be 4/H.
PROCESSOR TIME:	(Milliseconds) The time required for actual instruction execution; the average time that a processor will be busy because of the program. Note that this time is exclusive of I/O time and that the work load for any module is shared among the processors without prejudice.
I/O TIME:	(Milliseconds) The time in which execution of the program in main storage must be suspended while a data acquisition path program is performed by a data bus controller or channel.
MAIN STORAGE:	(Words) The amount of shared main storage required by the program text (instructions) plus that required for working storage (data and constants).
TIME IN MAIN STORAGE:	The average time during which the program must occupy shared main storage. In some cases, this is more than the sum of processor and I/O times because of the need to delay briefly while the subsystem responds to stimuli.
AUXILIARY STORAGE:	(Words) The amount of auxiliary storage required for program text (instructions), tables, constants, intermediate results, results which must be retained for later use, etc.

Where encircled numbers appear in Tables 4-11 through 4-18, they have the following meaning:

1. Rate of execution is determined by the failure rate of the LRUs addressed by the program.

2. Time in main storage is the sum of processor and I/O times.

3. Auxiliary storage requirement is estimated to be the same as the main storage requirement.

Table 4-11. I/Br Status Monitoring Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Successive Tests	①	118	5.10	711	151	②	③
Redundant Compare	①	13	3.27	407	127	②	③
Treshold Adjust	①	3	4.90	1120	146	②	③

Table 4-12. I/Br Periodic Checkout Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Drive Mechanisms	4/Yr.	2	3.55	508	83	②	③
ICV System	4/Yr.	2	6.56	1020	194	②	③
Inverters	1/W	1	114	14800	2301	②	③
Battery Chargers	1/W	10	13.3	1730	354	②	③
Selector Switches	TBD	TBD	TBD	TBD	TBD	TBD	TBD
Motor Generators	1/MO	2	30.8	4670	746	②	③
Transformer Rectifiers	1/MO	5	5.82	813	151	②	③
Buck Regulators	1/MO	10	9.43	1429	195	②	③
Circuit Breakers	4/Yr.	1	531	88500	7361	②	③
Contactors	TBD	TBD	TBD	TBD	TBD	TBD	TBD
Switches	4/Yr.	1	204	34100	2421	②	③
Differential Relays	2/Yr.	1	22.6	4670	510	②	③
Reverse-Current Relays	2/Yr.	1	49.4	8530	697	②	③

Table 4-13. I/Br Fault Isolation Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.).	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Power Conversion	①	3	40.4	6100	1749	②	③
Gas Management	①	2	21.8	3460	822	②	③
Heat Rejection	①	2	52.8	7430	2252	②	③
Heat Source	①	2	24.5	2850	658	②	③
1200 HZ Bus	①	1	228	34900	4269	②	③
260 VDC System	①	1	98.4	15400	2616	②	③
Square Wave	①	1	154	24800	3020	②	③
Sine Wave	①	1	179	29000	3236	②	③
GPL	①	1	53.9	8330	1296	②	③
28 VDC System	①	1	453	72600	8619	②	③
Motor Generator	①	1	94.2	14600	1630	②	③
Monitor and Control	①	2	26.5	4500	640	②	③

Table 4-14. I/Br Reconfiguration Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Power Conversion	①	1	41.0	7920	751	②	③
1200 HZ Network	①	1	68.6	13200	1075	②	③
H-V Rectifiers	①	1	23.5	4870	465	②	③
260 VDC Network	①	1	29.5	6090	478	②	③
Regular Transformer	①	1	32.1	6900	478	②	③
Battery Charger	①	1	35.9	6300	508	②	③
Break Regulator	①	1	26.8	4870	338	②	③
Square Wave Inverters	①	1	10.2	2030	172	②	③
Sine Wave Inverters	①	1	10.2	2030	172	②	③
60 HZ Inverters	①	1	10.2	2030	172	②	③
28VDC Network	①	1	82.4	1056	1056	②	③
Square Wave Network	①	1	39.7	7920	493	②	③
Sine Wave Network	①	1	39.7	7920	493	②	③
60 HZ Network	①	1	39.7	7920	493	②	③
Batteries	①	1	30.3	6090	335	②	③
600 HZ Network	①	1	35.4	6090	447	②	③

Table 4-15. SA Status Monitoring Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Successive Tests	①	118	4.82	711	143	②	③

Table 4-16. SA Reconfiguration Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Solar Panel	①	1	22.0	4670	411	②	③
Inverters	①	1	27.5	5080	541	②	③
Chargers	①	1	15.9	3450	348	②	③
Primary Bus	①	1	29.8	6300	684	②	③
400 Hz Network	①	1	71.4	13800	1355	②	③
Autotransformer	①	1	14.9	2840	367	②	③
Rectifier Filter	①	1	11.4	2540	224	②	③
28 VDC Network	①	1	48.2	8930	819	②	③
60 Hz Inverter	①	1	10.2	2030	172	②	③
60 Hz Network	①	1	39.7	7920	493	②	③
Batteries	①	1	26.6	5080	329	②	③

Table 4-17. SA Periodic Checkout Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
60 Hz Inverters	1/W	2	11.0	1830	260	②	③
Battery Charger	1/W	18	12.9	1730	320	②	③
Core & Boom Inverters	1/Mo	1	142	24800	2127	②	③
Rectifier-Filters	1/Mo	4	9.13	1630	190	②	③
Power Contractors	4/Yr	1	168	26800	2028	②	③
Differential Protection	2/Yr	1	116	18100	1732	②	③
Array Circuit I-V	1/D	1	464	42700	12230	②	③
Battery Controls	1/W	1	28.1	4870	523	②	③
Cell Recondition	1/W	1	83.1	11000	1606	②	③
Cell By-Pass	1/Mo	1	1240	173000	19347	②	③

Table 4-18. SA Fault Isolation Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec).	Main Storage Words	Time in Main Storage	Auxiliary Storage
Solar Panel	①	4	74.9	13500	1162	②	③
Boom Inverter	①	1	53.5	9350	892	②	③
Battery Charges	①	1	123	21300	1814	②	③
Core Inverter	①	1	28.5	5280	472	②	③
Primary Bus	①	1	61.7	9350	970	②	③
400 HZ Network	①	1	35.9	6300	595	②	③
Autotransformer	①	4	21.6	3960	458	②	③
Rectifier/Filter	①	1	17.9	3250	338	②	③
28 VDC Network	①	1	444	71100	8382	②	③
60 HZ Inverter	①	2	12.6	2440	1257	②	③
60 HZ Network	①	1	54.0	8330	1296	②	③
Source Integrated	①	1	63.0	10600	1258	②	③
TCD, Integrated	①	1	63.0	10600	1258	②	③
Batteries	①	1	46.0	10600	699	②	③

APPENDIX D

ENVIRONMENTAL CONTROL AND LIFE SUPPORT SUBSYSTEM CHECKOUT PROGRAM DESIGN

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF ILLUSTRATIONS

LIST OF TABLES

1.0 SCOPE

The Environment Control and Life Support Subsystem (EC&LS) provides cabin atmosphere control and purification, water and waste management, pressure suit support, and thermal control for the entire space station.

The fault detection function required for the EC&LS subsystem is accomplished by tables which are monitored by the OCS executive program. The tables contain the parameters which must be monitored to assure subsystem performance. These tables are transferred to the Remote Data Acquisition Unit (RDAU) via the master executive program and exception monitoring is accomplished. Figure 1-1 provides a graphic description of this function.

Table 1-1 has been provided to indicate the extent of the overall fault detection requirements.

The program described by this document is required for periodic checkout and fault isolation.

Initiation of the periodic checkout function is accomplished as the result of a keyboard entry by a crew member. It is anticipated that periodic checkout will be accomplished just prior to resupply, so that required replacements can be included in the resupply provisions.

The fault isolation utilizes the same software modules as the periodic checkout; however it is anticipated that analysis of the detected error will permit selection of the appropriate module to begin the required fault isolation. If the error is not detected in the selected assembly, the program provides this information and recommends that the periodic test be executed.

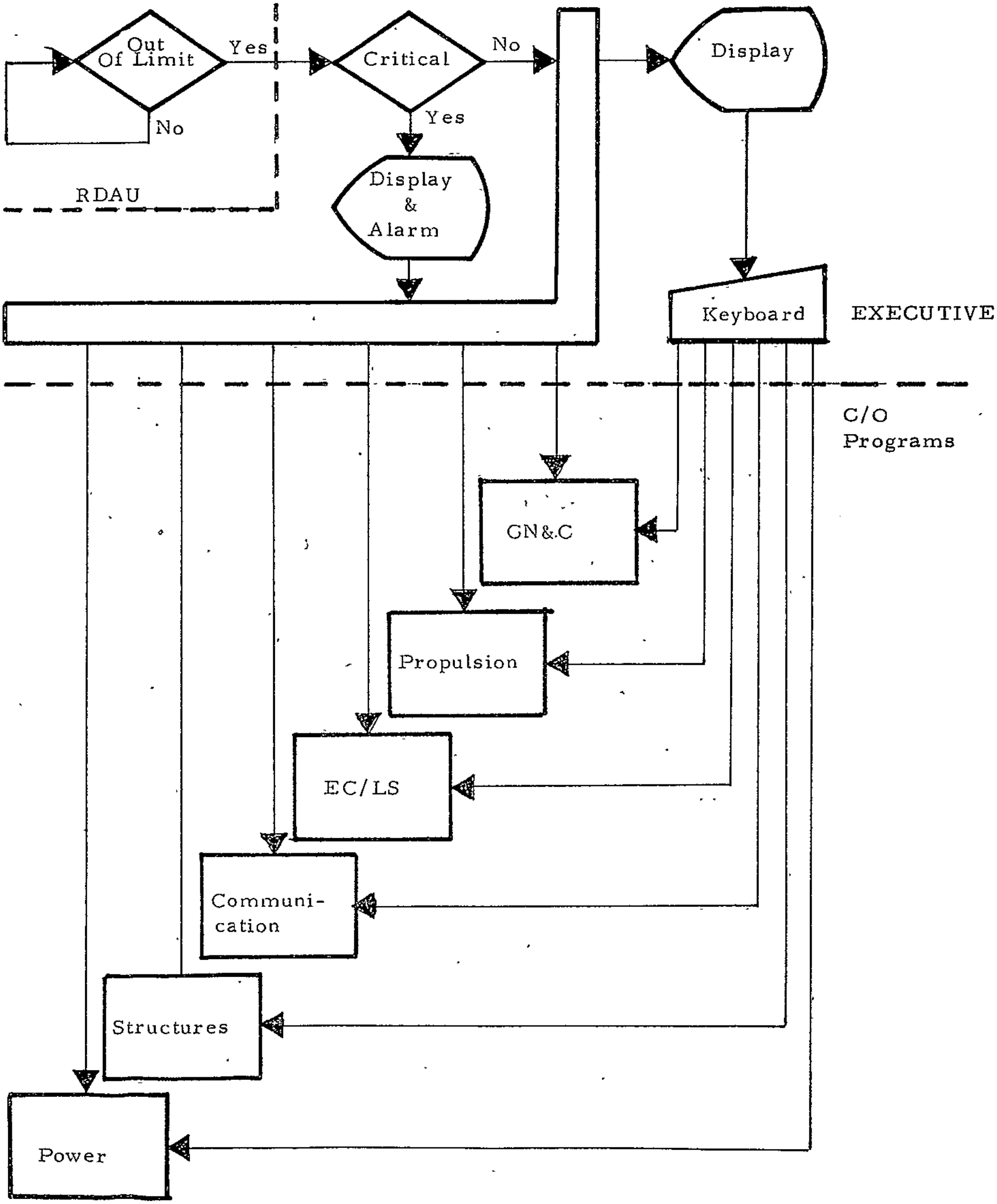


Figure 1-1. Fault Detection Logic

Table 1-1. EC&LS Fault Detection Summary

Sample Rate Assembly Group	1/SEC	1/MIN	1/HOUR
ATMOSPHERE SUPPLY	50		
WATER MANAGEMENT	22	8	
THERMAL CONTROL	68		
WASTE MANAGEMENT	8		
IVA/EVA SUPPORT*	120		
ATMOSPHERE RECONDITIONING	60		
PER SECOND	228		
PER MINUTE	13600	8	
PER HOUR	820, 800	480	
PER DAY	20, 699, 200	10, 520	
TOTAL PER DAY	20, 709, 720		

* Only during IVA/EVA Activity

2.0 APPLICABLE DOCUMENTS

- Space Station MSFC-DRL-160 Line Item 8, Volume V Book Mechanical, McDonnell-Douglas Astronautics Co.
- Baseline Subsystem Descriptions, Interim Report RA-1, August 1970, Contract NAS9-11189, IBM.
- Line Replaceable Unit Definition, Interim Report RA-6, September 1970, Contract NAS9-11189, IBM.
- Subsystem Test Descriptions and Measurement Stimulus List, Interim Report RA-8, November 1970, Contract NAS9-11189, IBM.
- Failure Mode and Effects Analysis, Interim Report RA-4, September 1970, Contract NAS9-11189, IBM.
- Subsystem Checkout Strategy, Interim Report RA-5, September 1970, Contract NAS9-11189, IBM.
- Space Station MSFC-DRL-160 Line Item 19, Volume I, Performance/ Design and Product Configuration Requirements, July 1970, Contract NAS8-25140, McDonnell-Douglas Astronautics Co.

3.0 REQUIREMENTS

This program meets the periodic testing and fault isolation requirements for the Environmental Control and Life Support (EC&LS) Subsystem.

3.1 PERFORMANCE

Six specific functional areas of this subsystem require automated checkout, which are:

- Thermal Control
- Atmosphere Supply and Control
- Atmosphere Reconditioning
- Water Management
- Waste Management
- IVA/EVA

Figure 3-1 provides a block diagram of the functional areas of this subsystem.

3.1.1 SYSTEM REQUIREMENTS

This section describes system constraints and/or requirements which have influenced design.

3.1.1.1 Subsystem Definition

This program specification is based upon the subsystem definition which is available as a result of this study contract. Some test points in this subsystem are currently defined at the assembly level, and consequently every failure which is detected cannot currently be identified with a Line Replacement Unit (LRU). Also, the correlation between the assembly test points identified in the "Subsystem Test Descriptions and Measurement Stimulus List" (RA-8) and the LRU's identified in the "Line Replaceable Units Definition" is not always apparent.

3.1.1.2 Thermal Control Assembly Group

There are no test points defined to permit exchange of active and redundant elements under control of the computer. The existence of such test stimuli is required for execution of the periodic test.

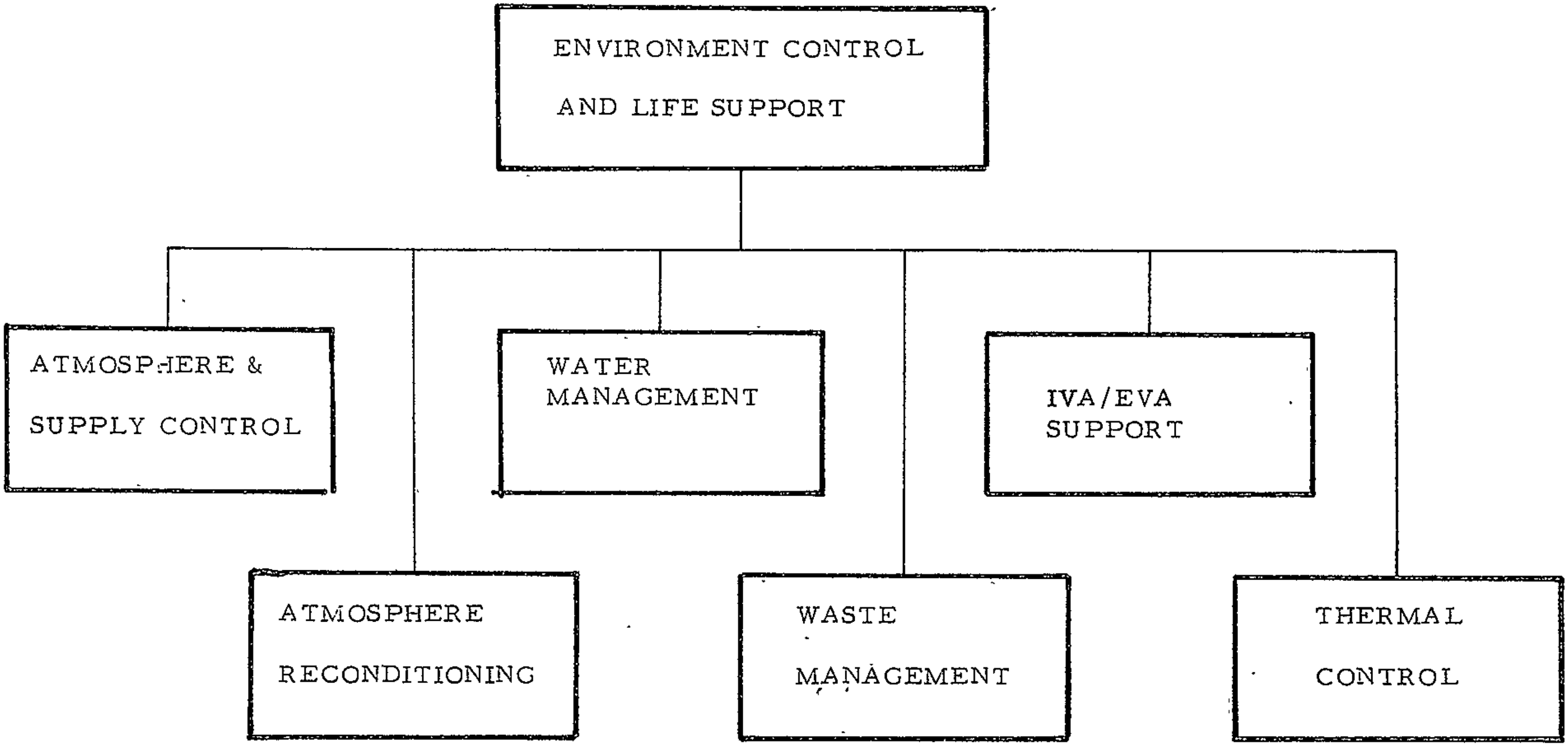


Figure 3-1. EC&LS Subsystem Functional Areas

3.1.1.3 Water Management Group

The Urine Recovery assembly in the Water Management assembly group requires a test point to open and close the solenoid valves which are associated with the chemical injector.

3.1.1.4 Waste Management Assembly Group

A test point is required to determine the fecal collection seal cover position. The measurement stimulus list has been modified to omit status monitoring of the fecal container pressure.

A test point is required to provide the position of the selector valve.

A test point is also required to determine the position of the handle which indicates the operate and process phases.

3.1.1.5 IVA/EVA Assembly Group

Flowcharts were not developed for this subsystem.

3.1.1.6 Trend Analysis

Trend analysis is utilized for functions which are subject to performance degradation of known and measurable characteristics. These include electrolysis cells, reverse osmosis membranes, absorption beds, and evaporator wicks. By observing the change in the major performance parameters, component replacement can be scheduled at a convenient time for the crew. Hazardous conditions can be avoided by trend analysis prediction of out-of-tolerance conditions. Trend analysis is also used to monitor expendable use rates. This pinpoints locations of excessive expendable use rates indicative of possible leakage or other failures, and also provides a basis for resources management and resupply planning activities. An example of this application is the use of nitrogen repressurization history to detect abnormal cabin repressurization rates, which may be indicative of a leak in the vehicle pressure shell.

Although the measurement and stimulus list has identified those points which require trend analysis, the required algorithms have not been specified. Consequently, the trend analysis requirements could significantly impact the estimates which are based on a least squares technique.

Table 3-1 has been included to provide an overview of the amount of trend analysis which must be accomplished.

3.1.2 OPERATIONAL REQUIREMENTS

This program specification defines specific operational requirements for

Table 3-1. EC&LS Trend Analysis Summary

Sample Rate / Assembly Group	1/MIN	1/HOUR	1/DAY
Atmosphere and Control	4	24	
Water Management	8	48	4
Thermal Control		8	
Waste Management			
IVA/EVA			
Atmosphere Recondition	7	32	
Data Item/Min			19
Data Item/Hour			1,252
Data Item/Day			30,052

the checkout of the space station Environmental Control and Life Support Subsystem.

The general checkout sequence addresses the least dependent functional group first. The thermal control equipment is checked out first because its operation does not depend on other functional groups. Many other assemblies depend upon thermal control equipment outputs. By verifying thermal control, deficiencies due to inadequate heating and cooling are eliminated as possible causes of problems in the EC/LS equipment. The checkout of functional groups follows the sequence below.

- Thermal Control
- Atmosphere Supply
- Atmosphere Reconditioning
- Water Management
- Waste Management
- IVA/EVA

Sequencing within an assembly group follows the same general procedure; when the assemblies and LRU's which are least dependent can be identified, they are checked first.

3.1.2.1 Thermal Control Module

The function of the thermal control assembly group is to collect, transport, distribute, and reject space station heat such that the crew and equipment are maintained within the required temperature limits. This program module issues stimulus and monitors test points to assure the effectiveness of this operation.

3.1.2.1.1 Source and Type of Inputs

The inputs to this module are the test points associated with specific assemblies. The assemblies which are examined to assure the quality of thermal operations include:

- Heating water circuit control assembly (ISOTOPE/BRAYTON only)
- Heating water recirculation assembly (ISOTOPE/BRAYTON only)
- Coolant water control assembly
- Coolant water recirculation assembly
- Radiator control assembly
- Radiator recirculation assembly

3.1.2.1.2 Destination and Types of Outputs

The outputs from this module are the normal operational messages indicating the out-of-tolerance situations and progress of the testing.

3.1.2.1.3 Information Processing

The condition of the thermal control assembly group is evaluated by software examination of the temperature and flow rates of coolant water and Freon.* If these measurements are within limits, the remainder of the tests are bypassed for purposes of fault isolation.

The periodic test performs the following tests on each assembly group in sequence.

1. The Heating Water Circuit Control Assembly controls the passage of water through the heat exchangers. This program module is capable of isolating faults which occur in the heat exchangers and relief valves.
2. The Checkout Requirements for the Heating Water Recirculation Assembly are essentially identical to those for the radiator recirculation assembly.
3. The Coolant Water Control Assembly is used to determine the required flow rate based on heat loads within the circuit. Fault isolation to the temperature sensors and the controller can be accomplished by this program module.
4. The Coolant Water Recirculation Assembly is used to maintain the flow and pressure rates within the system. The software module which checks out this assembly is capable of isolating problems in the pump and accumulator.
5. The Radiator Control Assembly includes the valves and associated controls for radiator selection and isolation purposes. The software module associated with this assembly is capable of detecting faulty

---

* Trademark of the Dupont Co.

operation of the isolation and flow reversal valves. There are two assemblies per module, and two complete circuits within each assembly. This configuration provides the capacity to maintain two circuits in an active state, and two circuits in a redundant status.

6. The Radiator Recirculation Assembly performs a function similar to the coolant water recirculation assembly for the Freon in the system. The basic difference is that the pump rate is not controlled by the radiator control assembly. Two additional test points have been provided to permit the sequence to check the Freon temperature and flow rates. This checkout program module is capable of isolating problems in the pump and accumulator.

Trend analysis is used to evaluate trends which are developing based upon the water and Freon temperature. The required algorithms to accomplish this analysis are currently undefined.

The replacement of the Isotope/Brayton power system with a solar array power system impacts the thermal control assembly group in that the heating water recirculation assembly and heating water control assembly are no longer included.

3.1.2.2 Atmosphere Supply and Control Module

The major functions of the atmosphere supply and control assembly group are:

- Provide oxygen and nitrogen
- Maintain atmosphere pressure and composition control
- Provide for compartmental pressurization and depressurization

3.1.2.2.1 Source and Types of Inputs

The inputs to this module are the test points associated with specific assemblies. The assemblies which are examined to assure the quality of the atmosphere supply and control assembly group performance are:

- Dump and Relief valve assembly
- Oxygen gas storage assembly
- Nitrogen gas storage assembly
- Pressure Reduction assembly
- Airlock pump assembly

- Pumpdown accumulator assembly
- Airlock pumpdown pressure control assembly
- Docking port pumpdown pressure control assembly

3.1.2.2.2 Destination and Type of Outputs

The outputs from this module are the normal operational messages indicating out-of-tolerance conditions, failing LRU's, and the progress of the listing.

3.1.2.2.3 Information Processing

The checkout of the atmosphere supply and control assembly group is accomplished by a group of program modules which meet the requirements for both periodic checkout and fault isolation. The assemblies were divided into groups of associated assemblies as depicted in Table 3-2. The pressure group is primarily responsible for maintaining cabin pressure. Two systems supply the requirements for the entire space station. The pump group interfaces with areas which require repressurization and depressurization.

The checkout of both groups is accomplished on line by allocation of specific elements.

If an error is detected in either of the areas which cannot be isolated to an assembly, the failure is assumed to have occurred in the plumbing.

This program module begins execution by checking the Dump and Relief Valve Assembly. This assembly prevents excess pressure being built up in a compartment, and provides the capability to manually purge the atmosphere. The dump and relief valve position is examined. An open status or excess pressure reading is used as an indication of a pressure problem. The program begins to examine the assemblies which are in the pressure group.

The Oxygen gas storage assembly is used to store the oxygen used for the compartmental atmosphere. For fault isolation the tank pressure and temperature are limit checked, and if in tolerance, the fault isolation logic proceeds to the next assembly. If a test point is detected out-of-limits or the periodic test is being executed, further analysis is performed.

The Nitrogen gas storage assembly accomplishes the same function for the nitrogen gas as the oxygen gas storage assembly performs for the oxygen. Consequently, the required software is identical to that required for the Oxygen gas storage assembly.

Table 3-2. Atmosphere Supply and Control Assembly Grouping

Pressure Group

- Oxygen gas storage assembly
- Nitrogen gas storage assembly
- Pressure reduction assembly
- Pressure control assembly
- Dump and Relief valve assembly

Pump Group

- Airlock pump assembly
- pumpdown accumulator assembly
- Docking Port pumpdown pressure control
- Airlock pumpdown pressure control

The pressure reduction assembly is used to reduce the pressure of the oxygen and nitrogen which is being taken from storage. The fault isolation portion of this module limit checks the upstream and downstream pressures for both oxygen and nitrogen. If these are within limits, this assembly is considered operational. If a test point is detected out-of-limits, or the periodic test is being executed, the shutoff valves, diverter valves, and heaters are examined.

The pressure control assembly controls the supply of nitrogen to the cabin and the pressure in the tunnel. This assembly requires that gas use rates be available for display to the operator upon demand. In addition, the number of actuations which are accomplished on the solenoid valve must be maintained for purposes of trend analysis.

The pressure in the oxygen supply and cabin pressure are limit checked to assure proper operation of this assembly. If either exceed limits or if a periodic test is being conducted, the solenoid valves and cabin pressure control are also checked out.

The airlock pump assembly is used to reclaim atmosphere from areas which are operationally pressurized and depressurized. This reclaimed air is then pumped to the pumpdown accumulator assemblies.

The repressurization line pressure is checked for both periodic testing and fault isolation. If this test point is in limit, the fault isolation test will proceed to the next assembly. If an error is detected or the periodic test is being executed, the solenoid valves and reciprocating compressor are examined.

The docking port and airlock pumpdown pressure control assemblies are used to control the rate of pressurization and depressurization of the respective areas. This program module checks each docking port and airlock. The operational software has the responsibility for assuring that the proper limits are maintained in the RDAU limit table, based upon the pressurization status of the specific areas.

The pumpdown accumulator assemblies are used to store air until it is needed for repressurization. Each assembly is equipped with a shutoff valve to isolate the equipment in the event of a failure.

The periodic and fault isolation tests both check the pressure and temperature in each assembly.

Trend analysis requirements for this module indicate that the executive must collect tank temperature and pressure from the N_2 and O_2 supply tanks on an hourly basis. In addition, the executive must maintain a count of the number of solenoid valve actuations in the pressure control assembly.

3.1.2.3 Atmosphere Reconditioning Assembly Group

The functions of the atmosphere reconditioning assembly group are as follows:

- Control atmosphere, temperature, humidity and air circulation
- Remove and collect CO_2 from the atmosphere
- Regenerate oxygen from the CO_2
- Control and monitor atmosphere trace contaminants
- Collect and transfer biowaste gasses to the propulsion subsystem

3.1.2.3.1 Source and Types of Inputs

The inputs for this module are the test points associated with specific assemblies. The assemblies which are examined to assure the quality of the atmosphere reconditioning assembly group are:

Temperature and Humidity Control Assembly

Trace Contaminant Control Assembly

CO_2 Control Assembly

CO_2 Conversion Assembly

Water Electrolosis Assembly

Ventilation Assembly

Bio-science Lab Atmosphere Purification Assembly

Dispensary Purification Assembly

3.1.2.3.2 Destination and Types of Outputs

The outputs of this module are normal opeational messages indicating out-of-tolerance conditions, failing LRU's, and the progress of the testing.

3.1.2.3.3 Information Processing

The checkout of the atmosphere reconditioning assembly group is accomplished by a group of program modules which meet the requirements for both periodic checkout and fault isolation. A unique program module is defined for each assembly.

The Temperature and Humidity Control Assembly uses a condenser/heat exchanger to control atmosphere temperature and humidity. Excess moisture is removed from the returned air and pumped to the wash water condensate recovery assembly. The fault isolation test exercises the temperature sensors when they are suspected to be in error, while the periodic test always exercises them. In all other respects the fault isolation and periodic test modules are identical.

The Trace Contaminant Control Assembly utilizes a catalytic oxidizer and sorbent beds to control atmospheric trace contaminants. The program module which checks out this assembly checks the bed temperature and valve position. If these are in tolerance, the fault isolation routine exits. If an error is detected or the periodic test is being executed, the performance of the heater and fan is also evaluated.

The CO_2 Control Assembly controls the level of CO_2 in the atmosphere by using molecular sieves. Air is received directly from the temperature and humidity control assembly. After the removal of CO_2, it is returned to the cabin. The air inlet CO_2 concentration test point, used by the controller to determine which assembly is currently in use, is located in the temperature and humidity assembly, and is checked during the execution of this module.

The operation of this assembly is dependent upon whether an Isotope/Brayton or Solar Array power source is used. If a Solar Array is used, it is not necessary to test the circulation of hot fluids within the system.

The requirements for periodic checkout and fault isolation program modules are identical for this assembly.

The CO_2 Conversion Assembly uses a sabatier reactor to convert carbon dioxide into water and methane. The periodic test and fault isolation tests for this assembly are identical, except that the fault isolation test only examines the controller in the event that an error is detected.

The Water Electrolysis Assembly is used to convert water into hydrogen and oxygen. The periodic test and fault isolation program module requirements for this assembly are identical.

The Ventilation Assembly provides ventilation within the space station, except for the bio-science and dispensary area. The fault isolation and periodic test program module requirements are identical for this assembly.

The Trace Contaminant Monitoring Assembly uses a hybrid chromatograph/ mass spectrometer (not yet developed) to assess gas components, both qualitatively and quantitatively. In the event that the fault detection techniques detect an error in this assembly, the temperature and humidity control and trace contaminant control assemblies are examined. If there is no failure detected in these areas, the monitoring assembly is considered to be in error.

The periodic test isolates to this assembly by examining the temperature and humidity control and trace contaminant control assembly first.

The Bio-Science Laboratory is an area in the space station where experiments concerning plants, micro-biology, and invertebrates are performed. The Bio-Science Laboratory Atmosphere Purification Assembly is used to permit isolating this area from the rest of the space station. One program module is needed to meet the requirements for both periodic testing and fault isolation.

The dispensary is an area isolated atmospherically from the rest of the space station so that communicable diseases cannot be transferred. The Dispensary Atmosphere Purification Assembly provides the ventilation and bacteria filtering necessary. One program module is needed to meet the requirements for both periodic testing and fault isolation.

The carbon dioxide control assembly is the only part of this assembly group which is directly impacted by the use of a solar array as a power source.

3.1.2.4 Water Management

The major functions of the water management assembly group are:

- Conversion of urine, wash water, and condensate into potable water
- Distribution of the water

3.1.2.4.1 Source and Types of Inputs

The inputs to this module are the test points associated with the specific assemblies. The assemblies which are examined to assure the quality of the water management assembly group are:

- Urine Recovery Assembly
- Wash Water and Condensate Recovery Assembly
- Water Storage Assembly

3.1.2.4.2 Destination and Type of Outputs

The outputs from this module are the normal operational messages indicating out-of-tolerance conditions, failing LRUs, and the progress of the testing.

3.1.2.4.3 Information Processing

The checkout of the water management assembly group is accomplished by a program module which meets the requirements for both periodic checkout and fault isolation.

The Urine Recovery Assembly uses air evaporation to extract water from wicks which have been saturated with waste water.

The Wash Water and Condensate Recovery Assembly purifies the water by a reverse osmosis process separating the high solids content fluid from the water.

The Water Storage Assembly accomplishes the storage of both potable and wash water.

3.1.2.5 Waste Management Module

The major functions of the waste management assembly group include:

- Collect and transfer urine
- Collect, process, and store fecal waste

3.1.2.5.1 Source and Type of Inputs

The inputs to this module are test points associated with the specific assemblies. The assemblies which are examined to assure the quality of the waste management assembly group performance are:

- Urine collection assembly
- Fecal waste collection assembly

3.1.2.5.2 Destination and Type of Outputs

The outputs from this module are the normal operational messages indicating out-of-tolerance situations, failing LRU's, and the progress of testing.

3.1.2.5.3 Information Processing

The checkout of the waste management assembly group is accomplished by a program module which meets the requirements for both periodic checkout and fault isolation.

The program module begins by examining air outlet moisture, which is the only test point associated with the Urine Collection Assembly. This assembly uses flush water from the water storage assembly. The required power is obtained from the associated fecal waste collection assembly.

The Fecal Waste Collection Assembly is then examined to assure proper operation in both the operate and process phases.

The only difference between the fault isolation and periodic tests is that the periodic test examines the ten urine collection and four fecal collection assemblies, while the fault isolation routine only examines those where an error has been detected.

3.1.2.6 IVA/EVA Assembly Group Module

The function of the IVA/EVA assembly group is to provide support to a suited crewman during intravehicular (IVA) or extravehicular (EVA) activities.

3.1.2.6.1 Source and Types of Inputs

The inputs to this module are the test points associated with specific assemblies. The assemblies which are examined to assure the quality of the IVA/EVA assembly group are:

- Portable life support system assemblies (PLSS)
- Umbilical Life Support Assemblies (ULSA)
- IVA Support Assemblies

3.1.2.6.2 Destination and Types of Output

The outputs from the module are the normal operational messages indicating out-of-tolerance conditions, failing LRU's, and the progress of the testing.

3.1.2.6.3 Information Processing

The checkout of the IVA/EVA assembly group is accomplished by a program module which meets the requirements for both periodic checkout and fault isolation. An option has been incorporated in this program to permit selection of the IVA/EVA checkout module only.

Four Portable Life Support System Assemblies (PLSS) provide self-contained life support, voice communications, and telemetry for EVA. The PLSS is a back pack which is identical to the one which was developed for the Apollo program. The test points are examined only when the PLSS is connected to the pressure suit.

The Umbilical Life Support Assemblies (ULSA) provide closed water cooling and open circuit oxygen to the pressure suits when the umbilicals are connected to the IVA support assemblies. This assembly is examined when connected to the pressure suits.

The IVA Support Assemblies distribute oxygen from the Atmosphere Supply and Control Assembly Group, and coolant water from the Thermal Control Subsystem Group to the umbilical life support assemblies within the Space Station.

The executive system is required to recognize the inclusion of a pressure suit in the system; and activate the RDAU test points to assure the proper operation of this assembly.

3.1.2.7 Termination Routine

This module is included to assure the orderly return of control to the executive at the completion of execution.

3.1.2.7.1 Source and Type of Inputs

The inputs to this module indicate a successful or unsuccessful completion of program execution. In the event termination is required due to operator intervention, or a detected failure, additional inputs as to the status of the associated components are required.

3.1.2.7.2 Destination and Type of Outputs

None

3.1.2.7.3 Information Processing

This module must accomplish the functions required to assure the subsystem is properly configured before control is returned to the executive.

3.1.2.8 Group Logic Module

This software module is used to select the appropriate sequence of program modules to be executed in the event an error is detected in this subsystem. It also provides the sequencing required for periodic tests.

3.1.2.8.1 Source and Type of Inputs

This module provides the entry points for the periodic and fault isolation testing. In the event that a fault is detected, the failing test point will be identified to this module.

3.1.2.8.2 Destination and Types of Outputs

The only outputs from this module are the displays associated with the progress of the testing.

3.1.2.8.3 Information Processing

This module determines whether fault isolation or periodic testing is to be accomplished. In the event that fault isolation is required, the detected error is isolated to an assembly. If the program is unable to isolate an error in the selected and associated assemblies, a message is presented to the operator recommending execution of the periodic test.

3.1.3 PROGRAM FLOWCHARTS

Program flowcharts are provided in this section. The chart ID position on a diagram is used to distinguish between the levels of detail in the flowcharts. A chart ID of A is used to depict the overall information flow of the program, and is considered functional. A chart ID of B or C is intended to provide more detailed information.

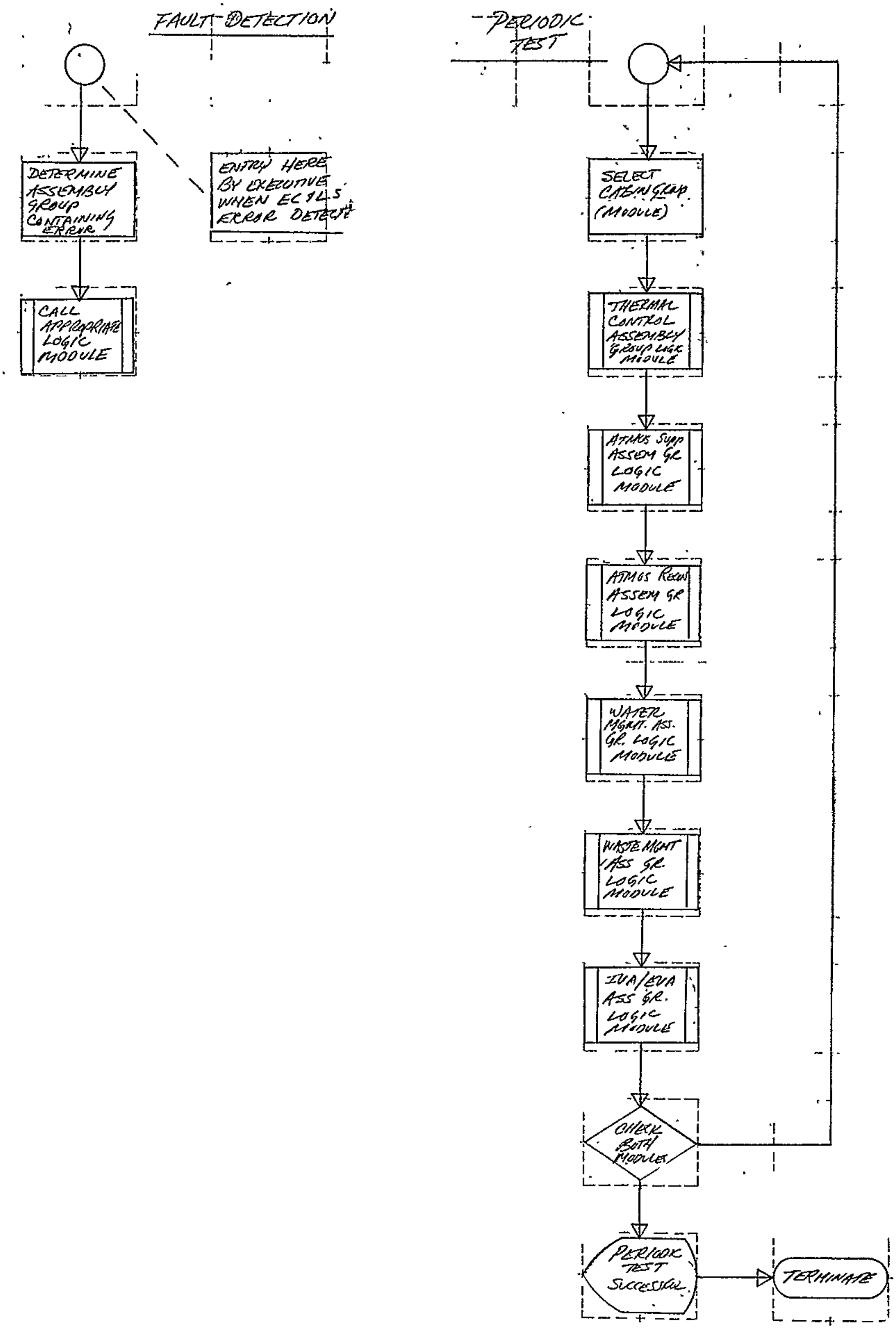


Figure 3-2. Fault Detection and Periodic Test

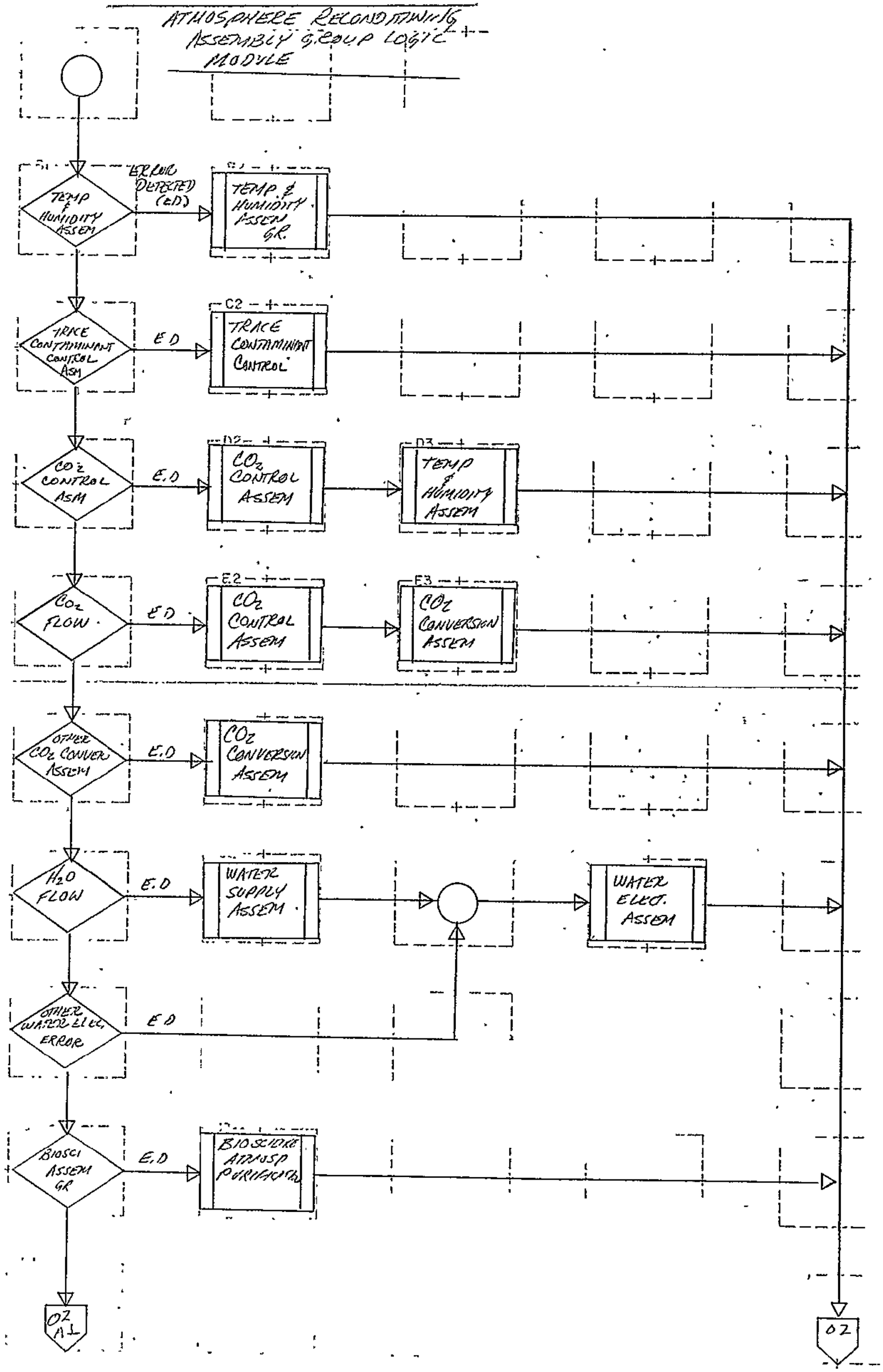


Figure 3-3. Atmosphere Reconditioning Assembly Group Logic Module

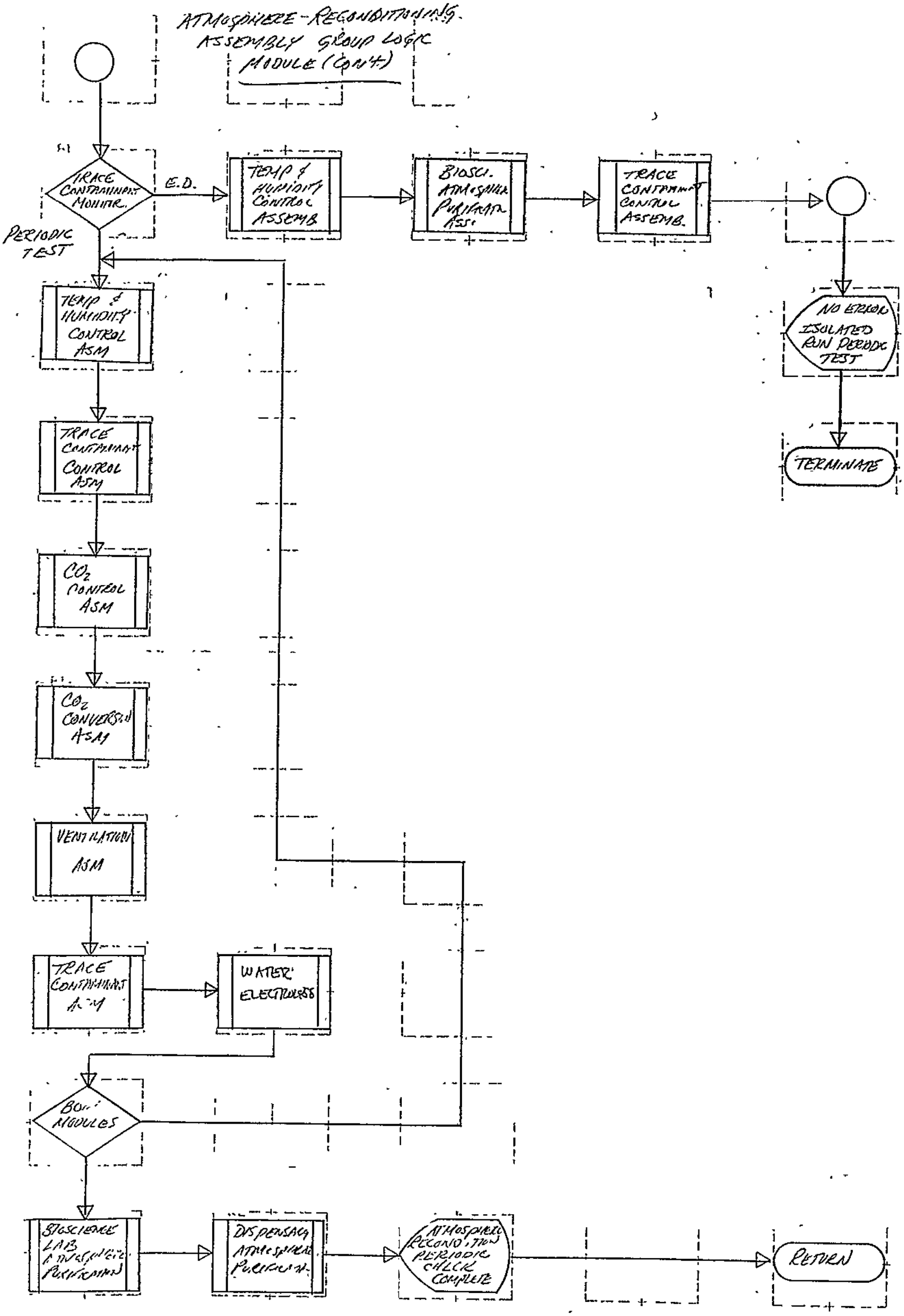


Figure 3-3. Atmosphere Reconditioning Assembly Group Logic Module (Sheet 2 of 2)

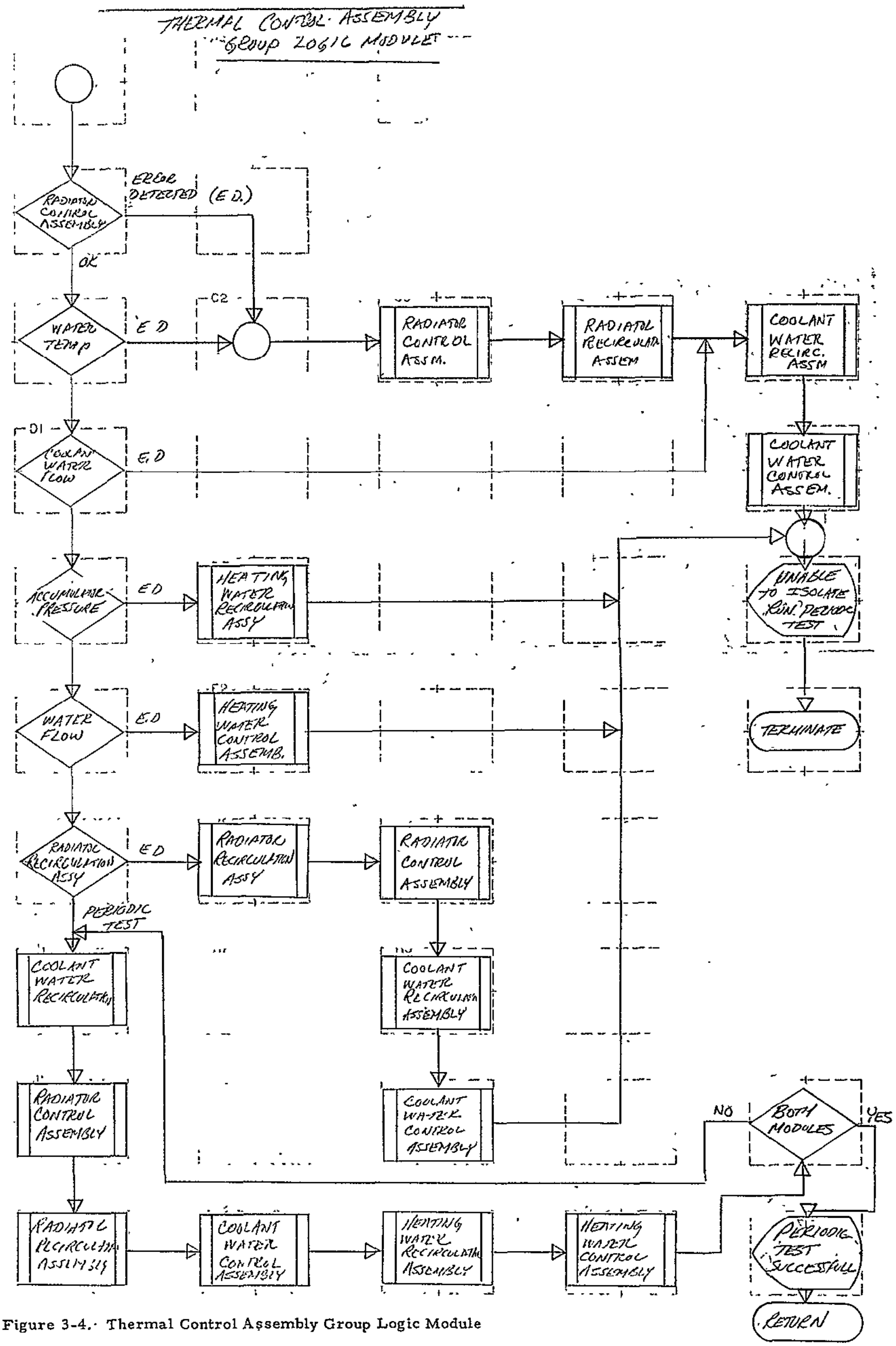


Figure 3-4. Thermal Control Assembly Group Logic Module

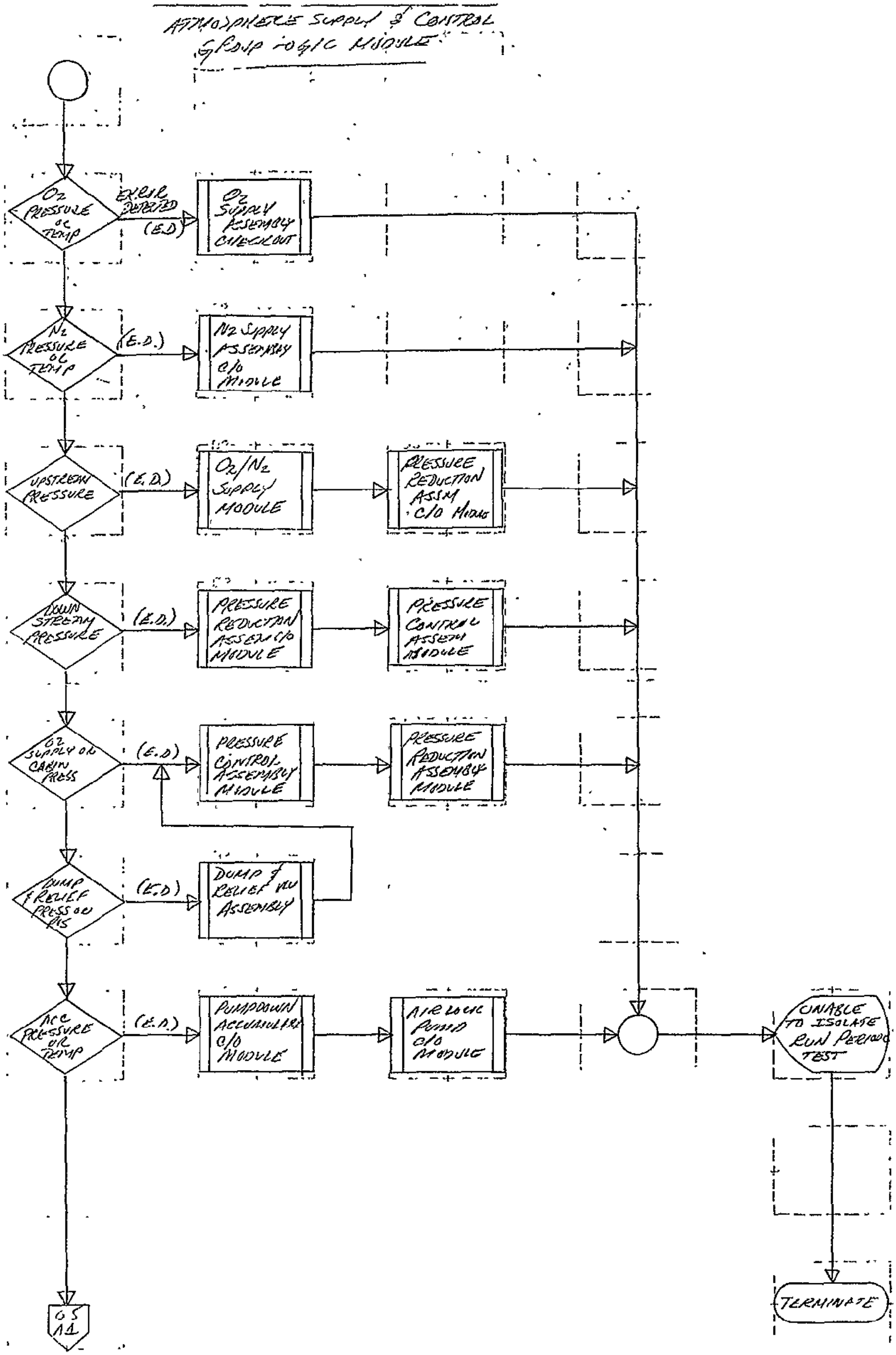


Figure 3-5. Atmosphere Supply and Control Group Logic Module (Sheet 1 of 2)

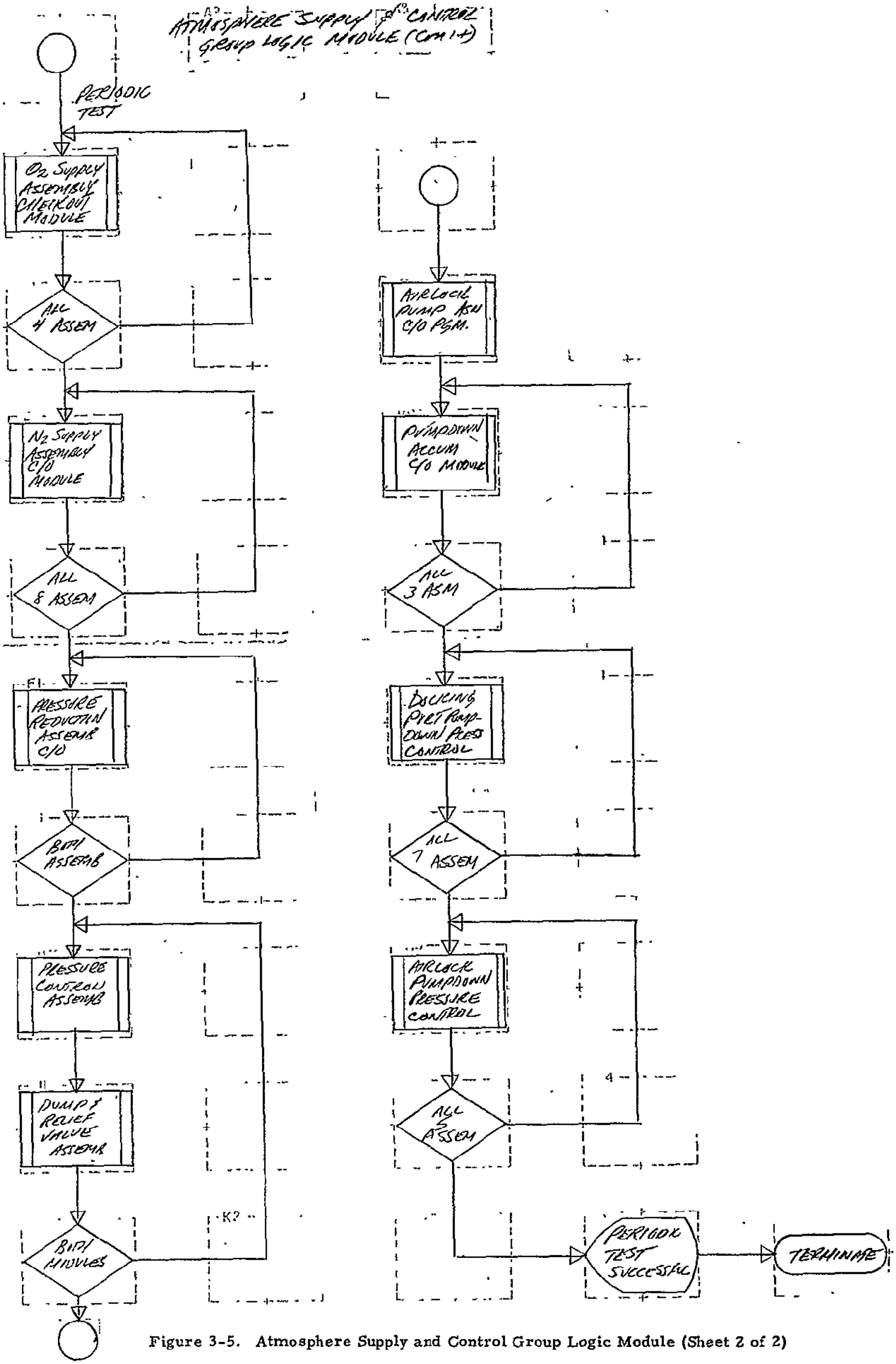


Figure 3-5. Atmosphere Supply and Control Group Logic Module (Sheet 2 of 2)

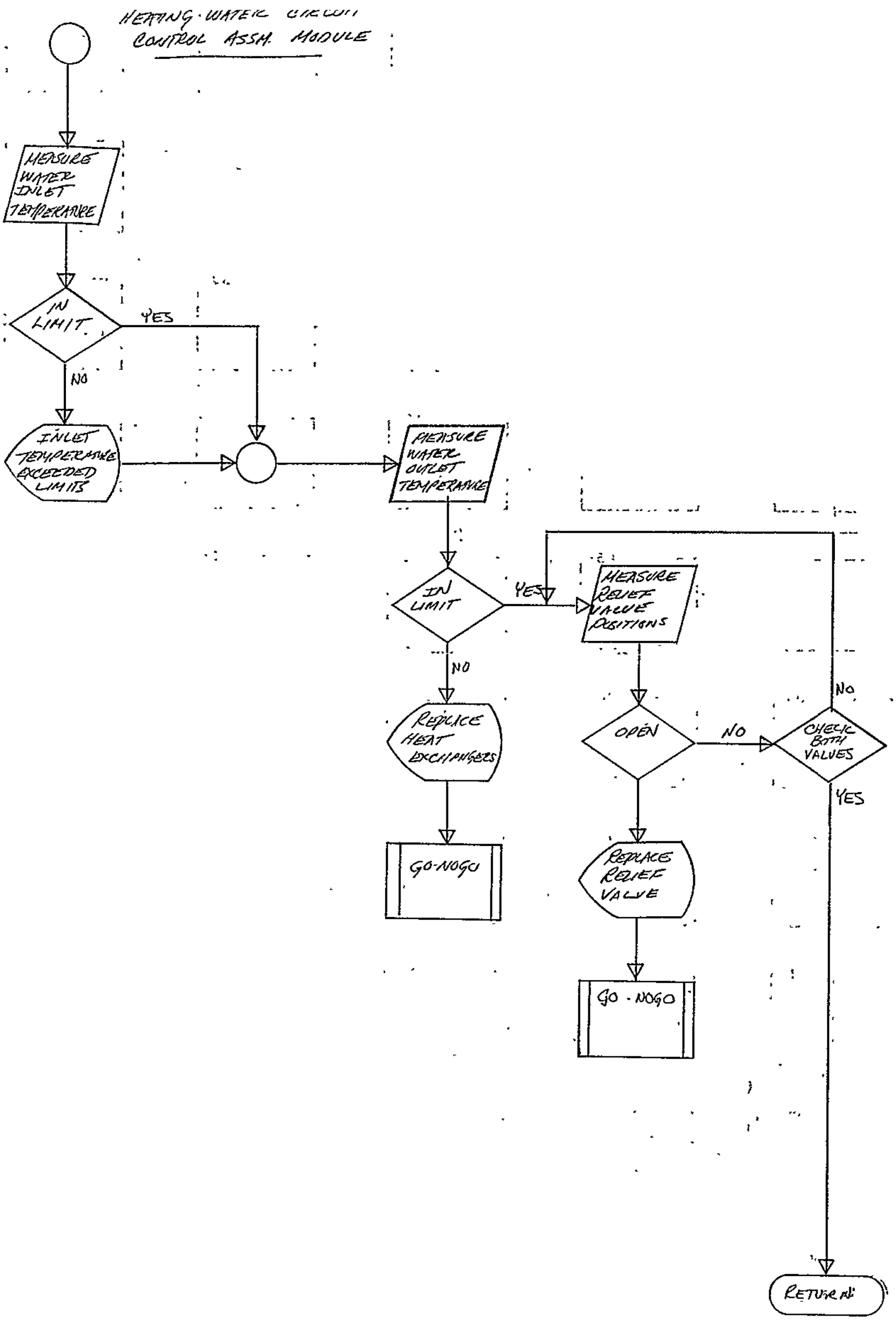


Figure 3-6. Heating Water ~~Circuit~~ Control Assembly Module

(P.2 BLANK)

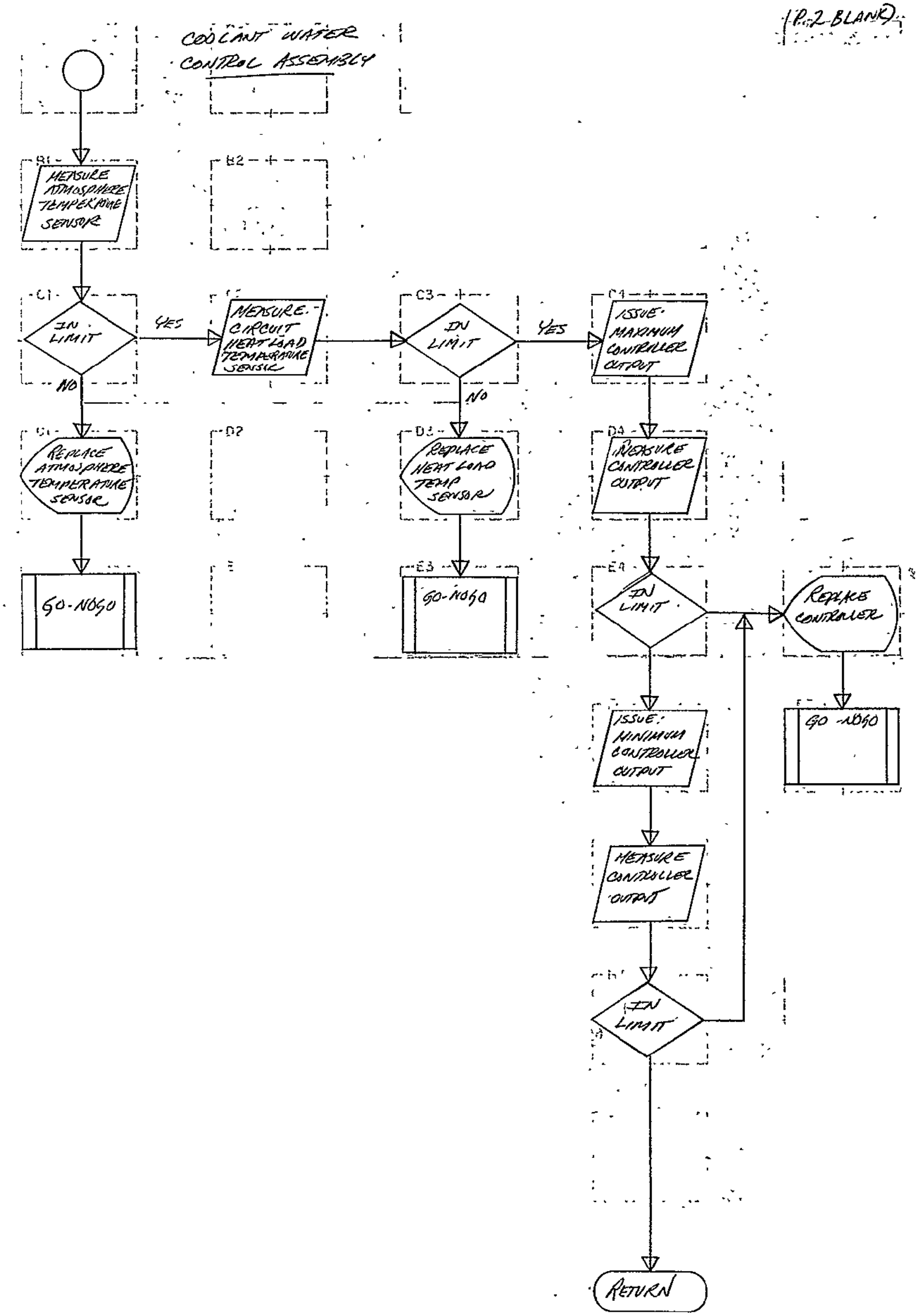


Figure 3-8. Checkout of Coolant Water Recirculation Assembly

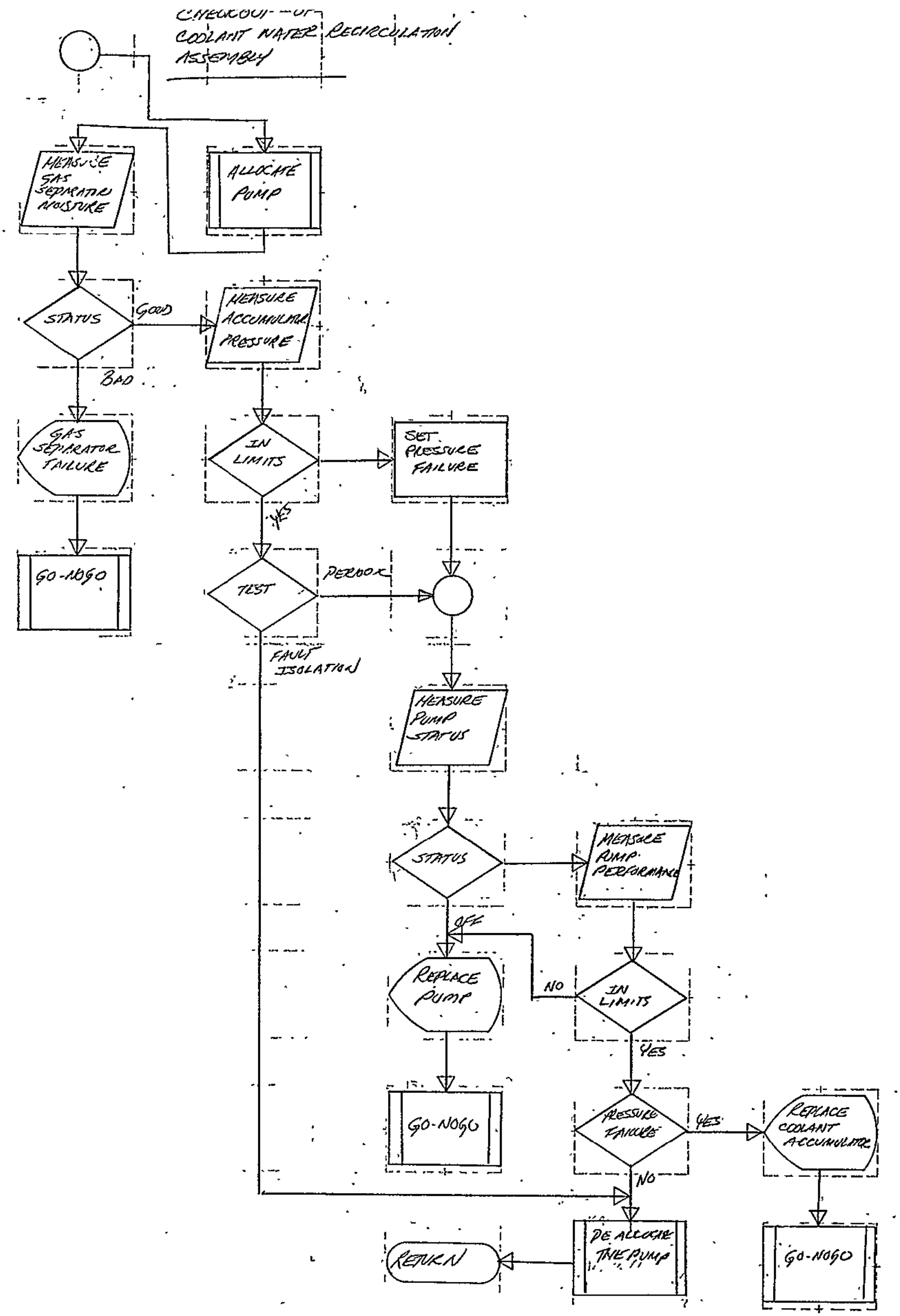


Figure 3-9. Radiator Control Assembly (Sheet 1 of 2)

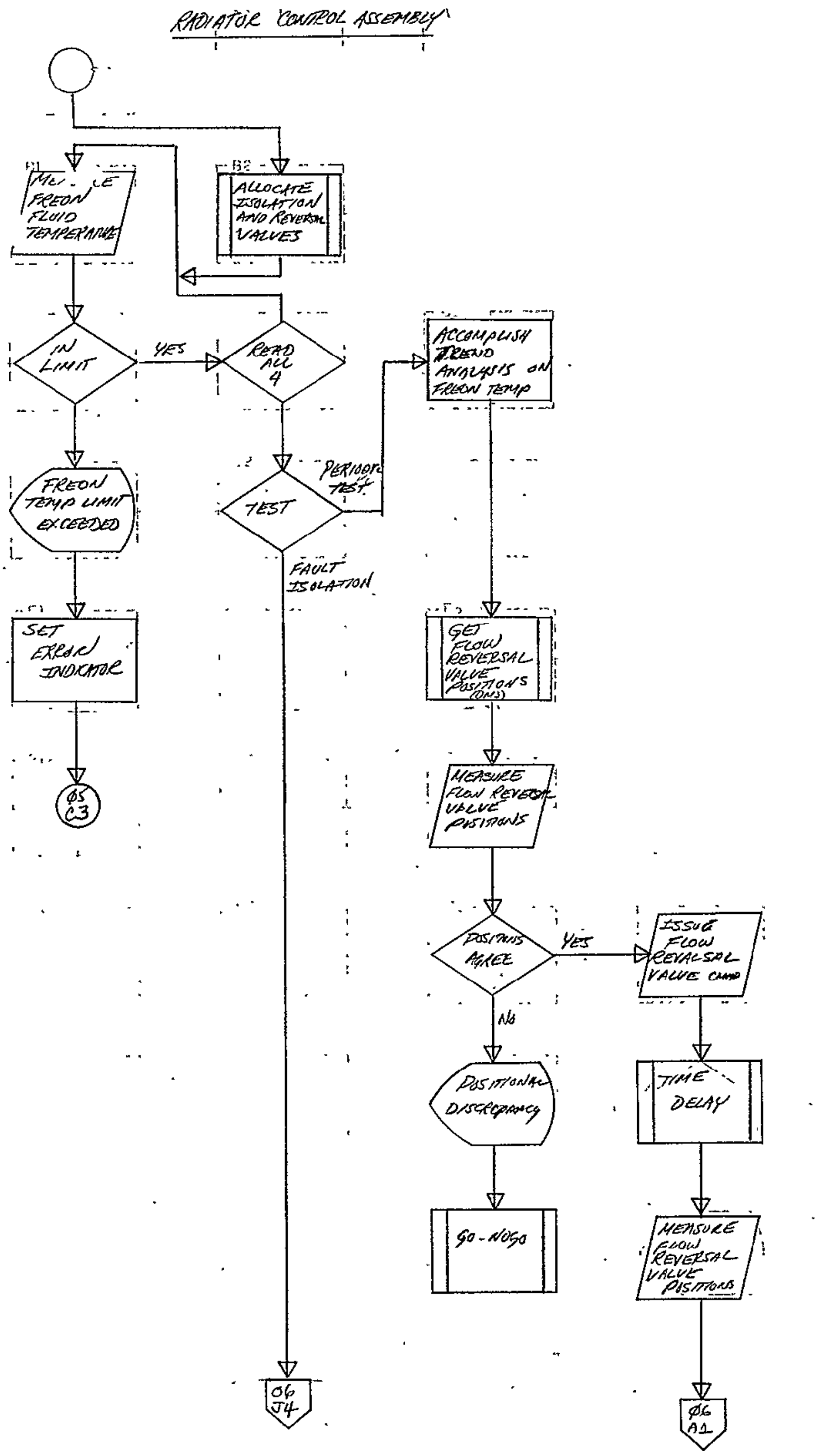


Figure 3-9. Radiator Control Assembly (Sheet 2 of 2)

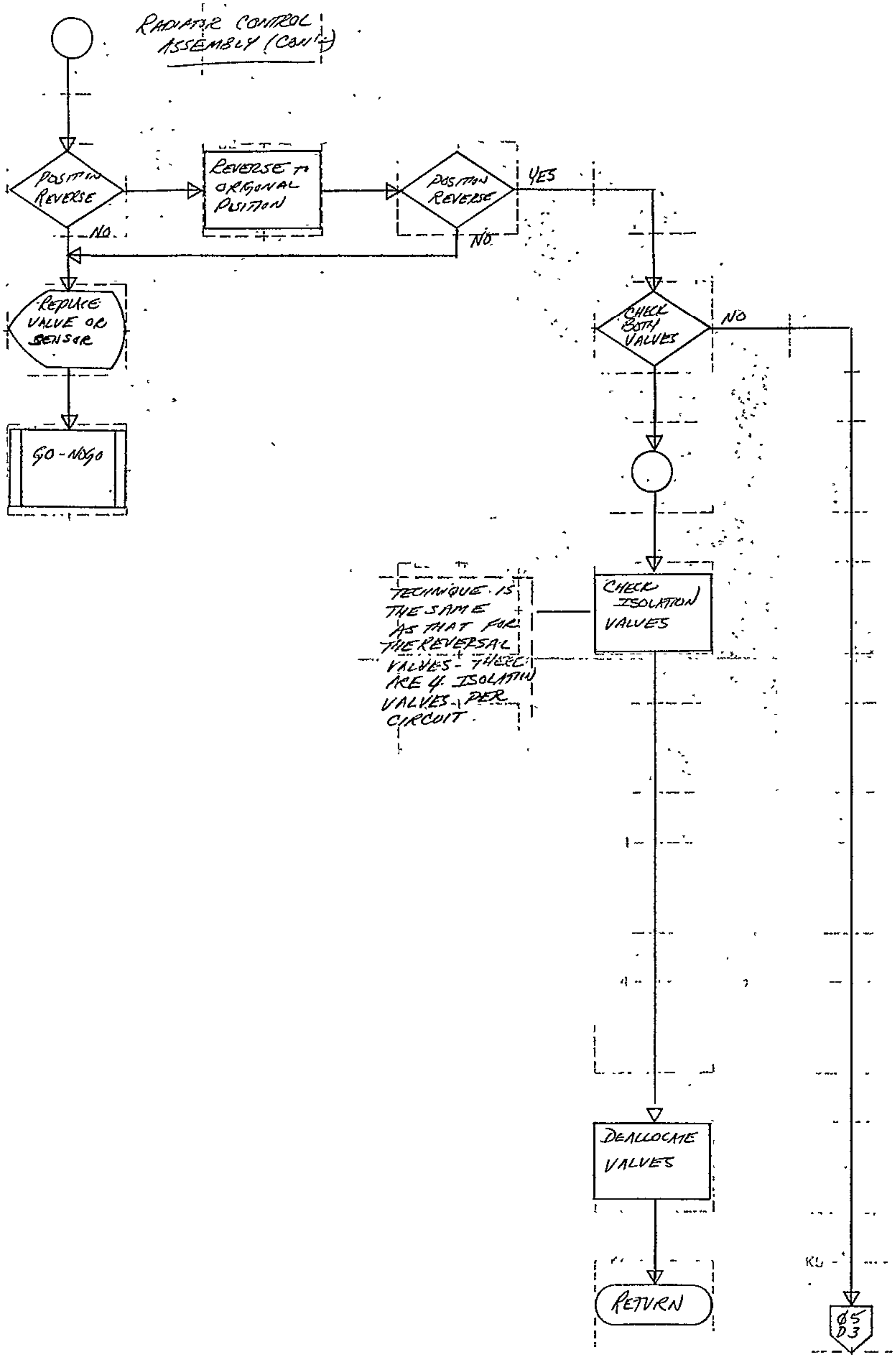


Figure 3-10. Radiator Recirculation Assembly

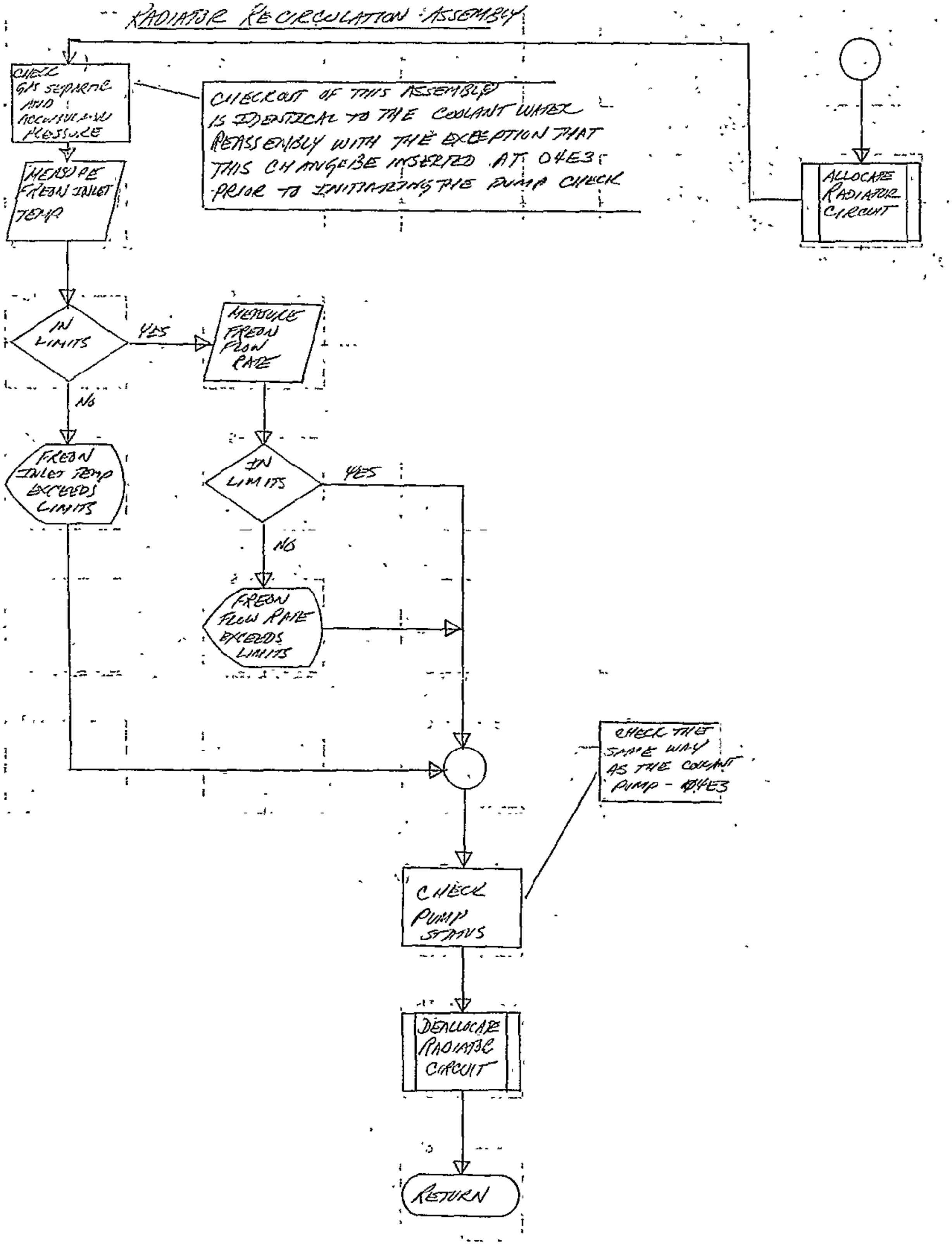


Figure 3-11. Oxygen Supply Assembly Checkout Module (Sheet 1 of 2)

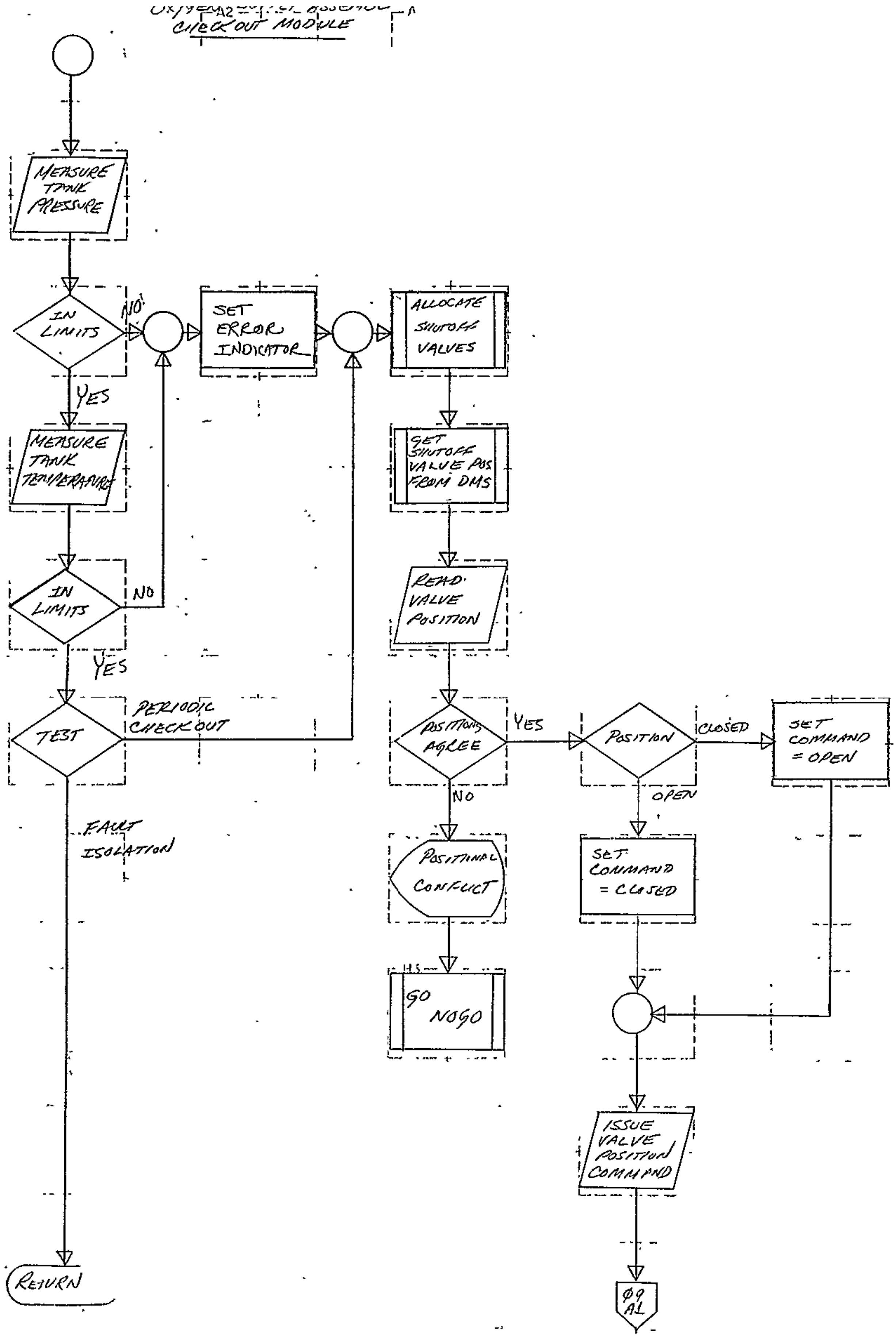


Figure 3-11. Oxygen Supply Assembly Checkout Module (Sheet 1 of 2)

OXYGEN SUPPLY ASSEMBLY
CHECKOUT MODULE (CONT'D)

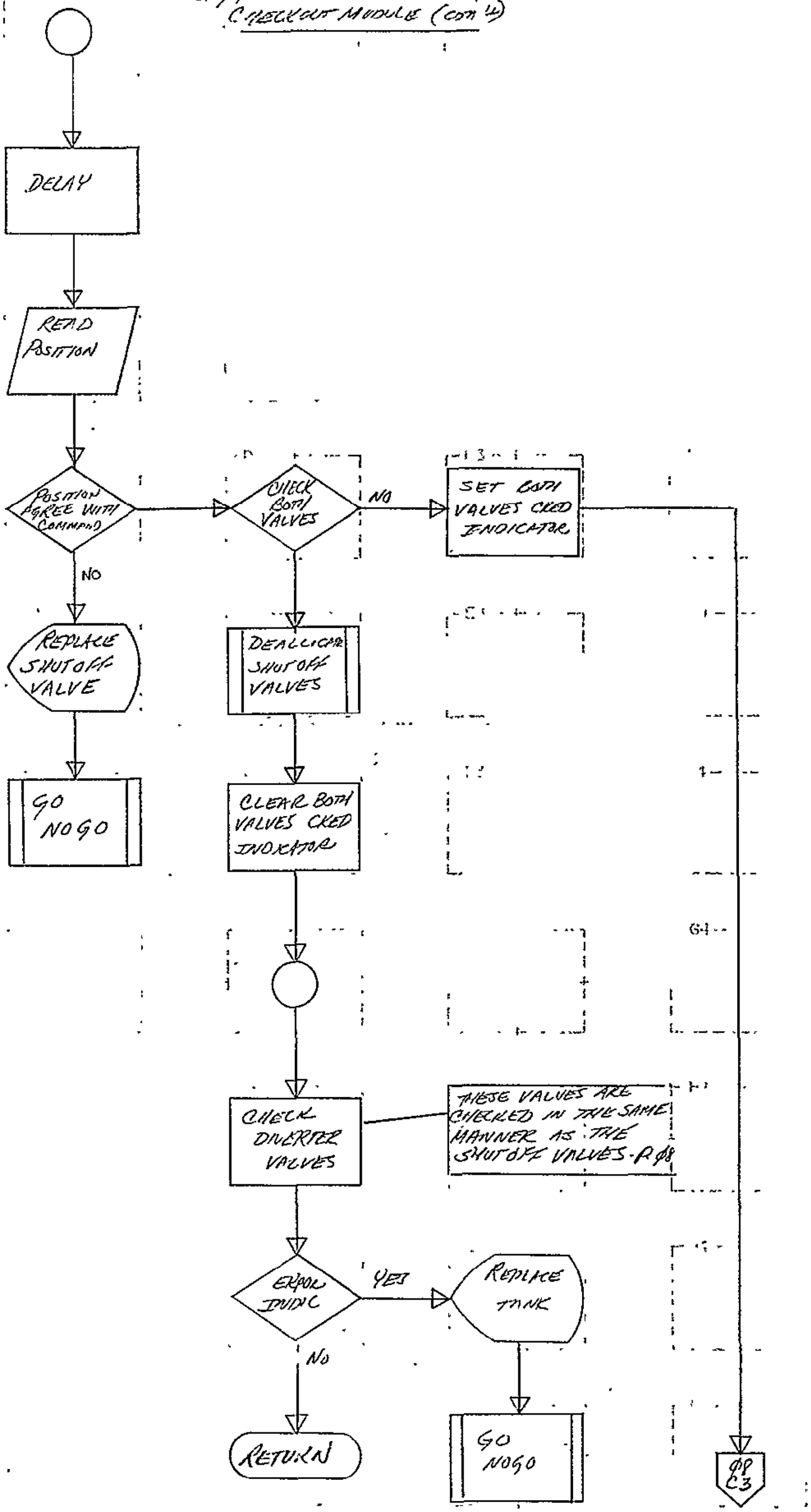


Figure 3-11. Oxygen Supply Assembly Checkout Module (Sheet 2 of 2)

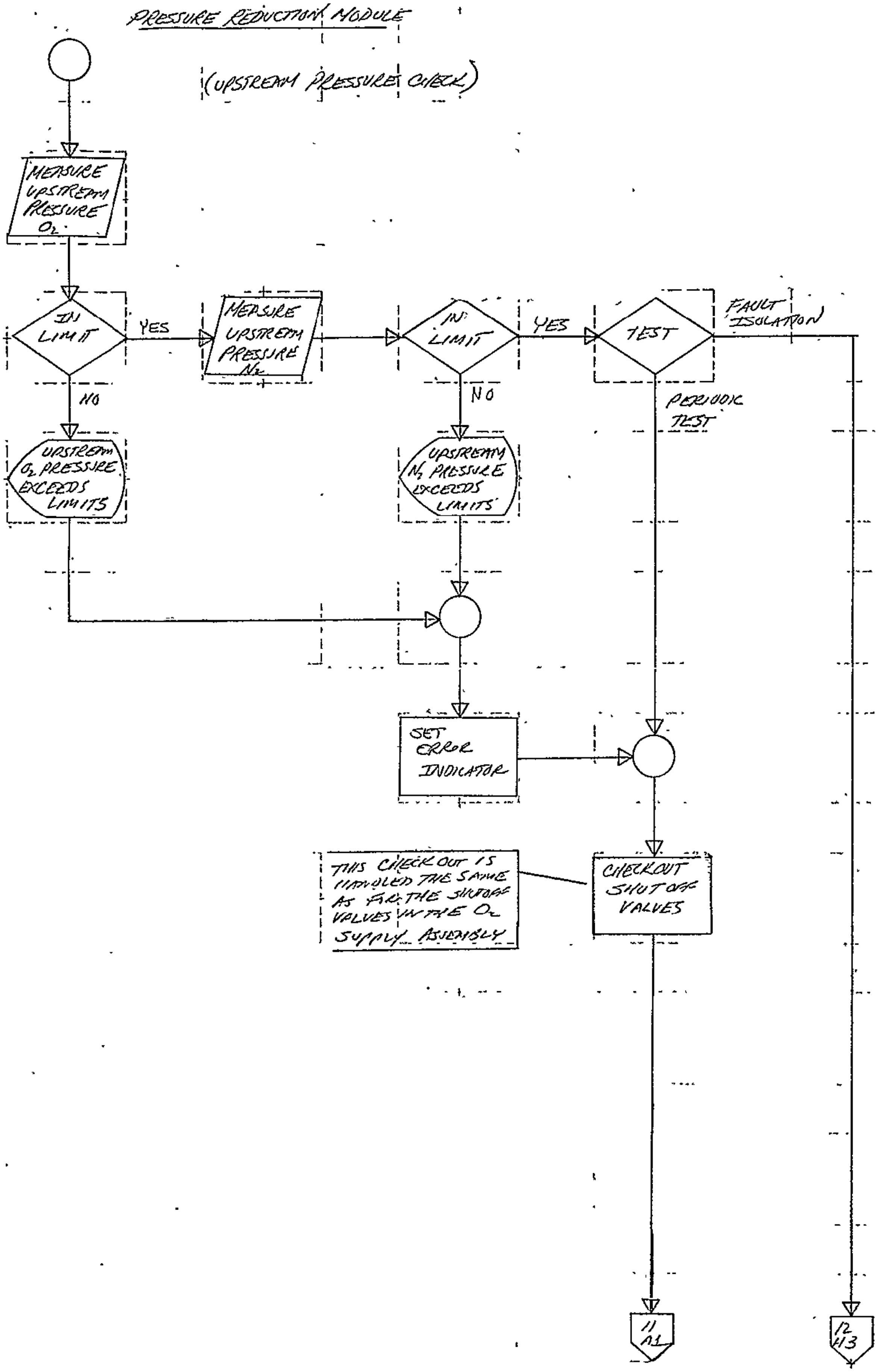


Figure 3-12. Pressure Reduction Module (Sheet 1 of 4)

PRESSURE REDUCTION MODULE (CONT'D)

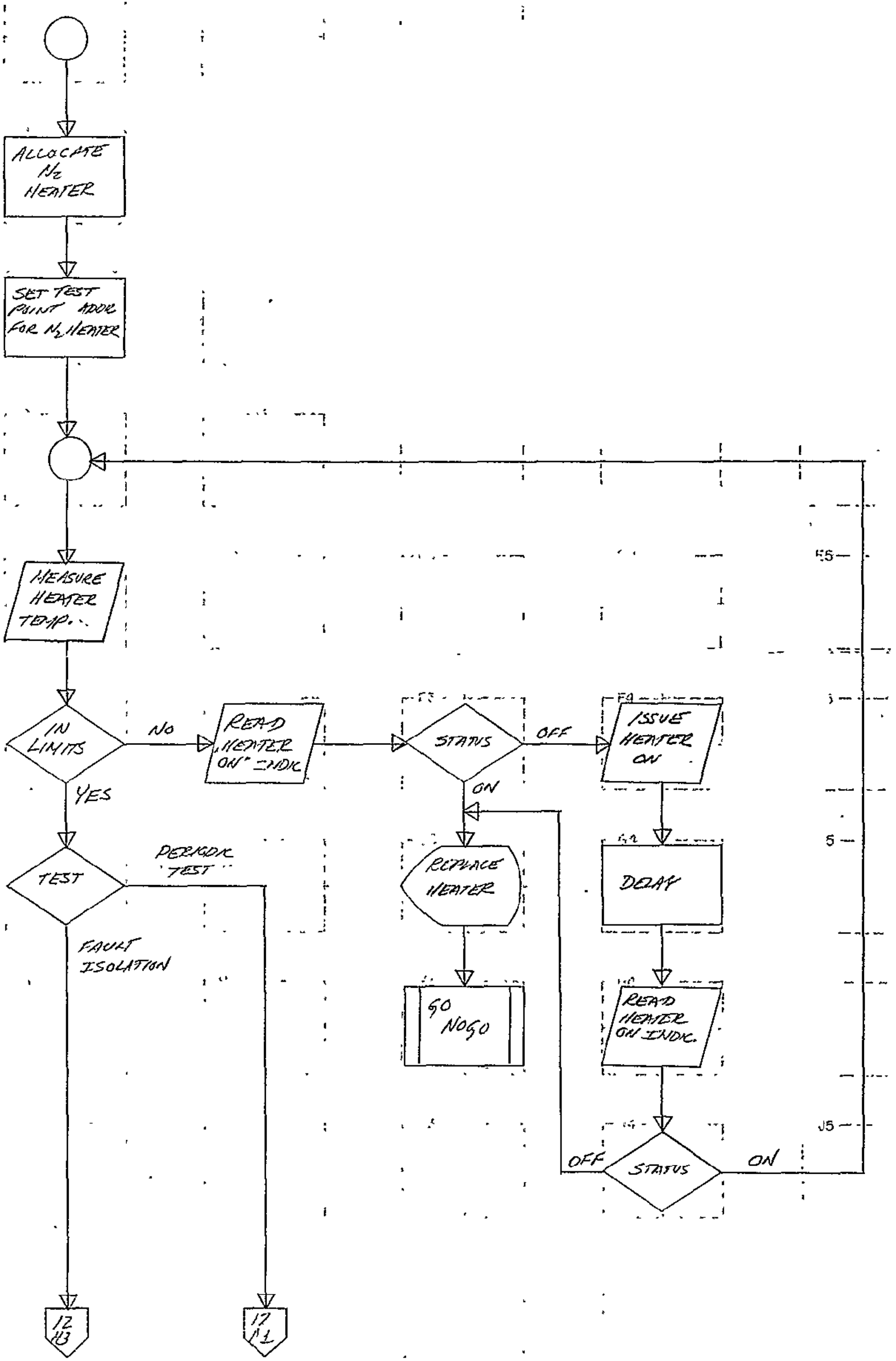


Figure 3-12. Pressure Reduction Module (Sheet 2 of 4)

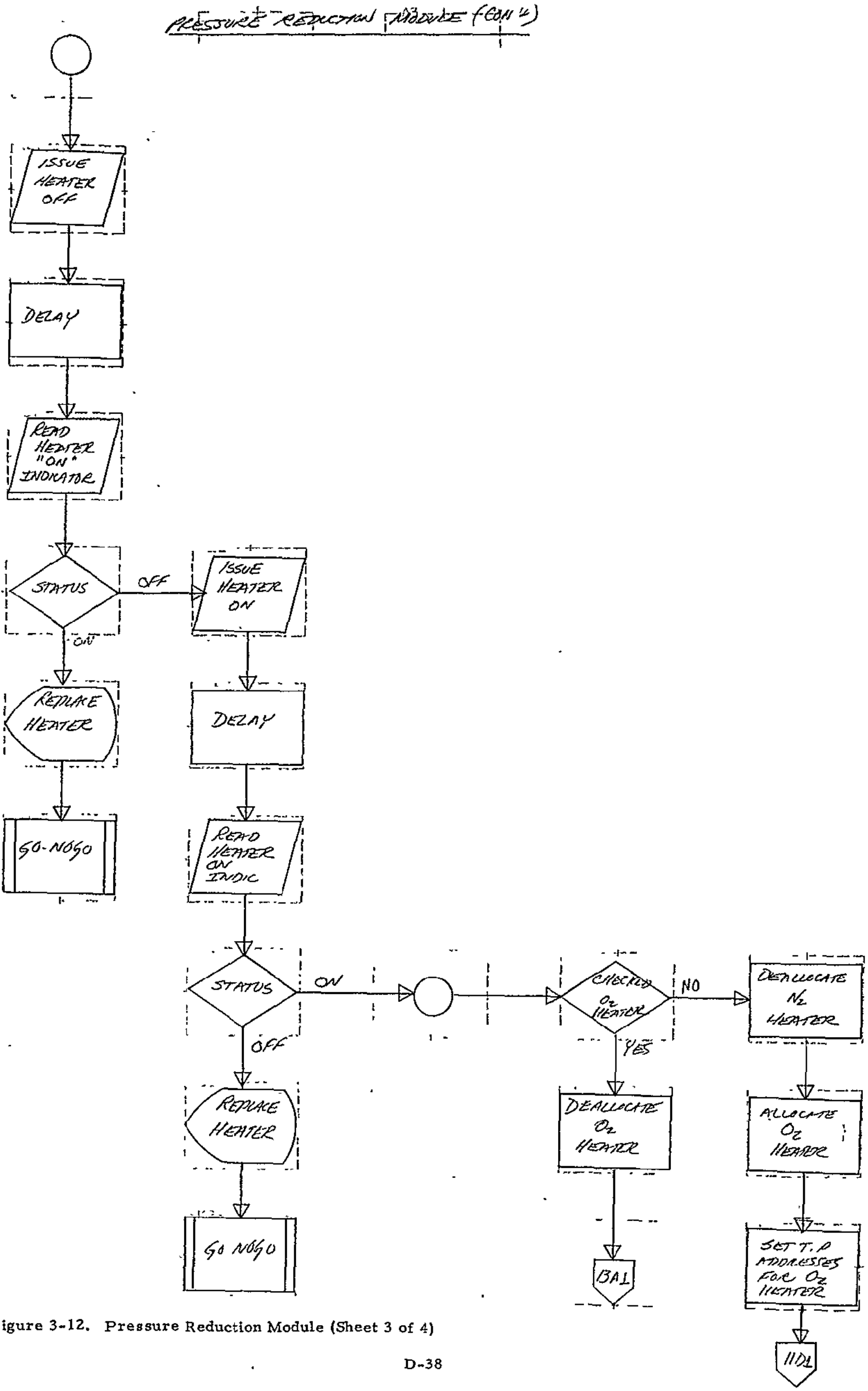


Figure 3-12. Pressure Reduction Module (Sheet 3 of 4)

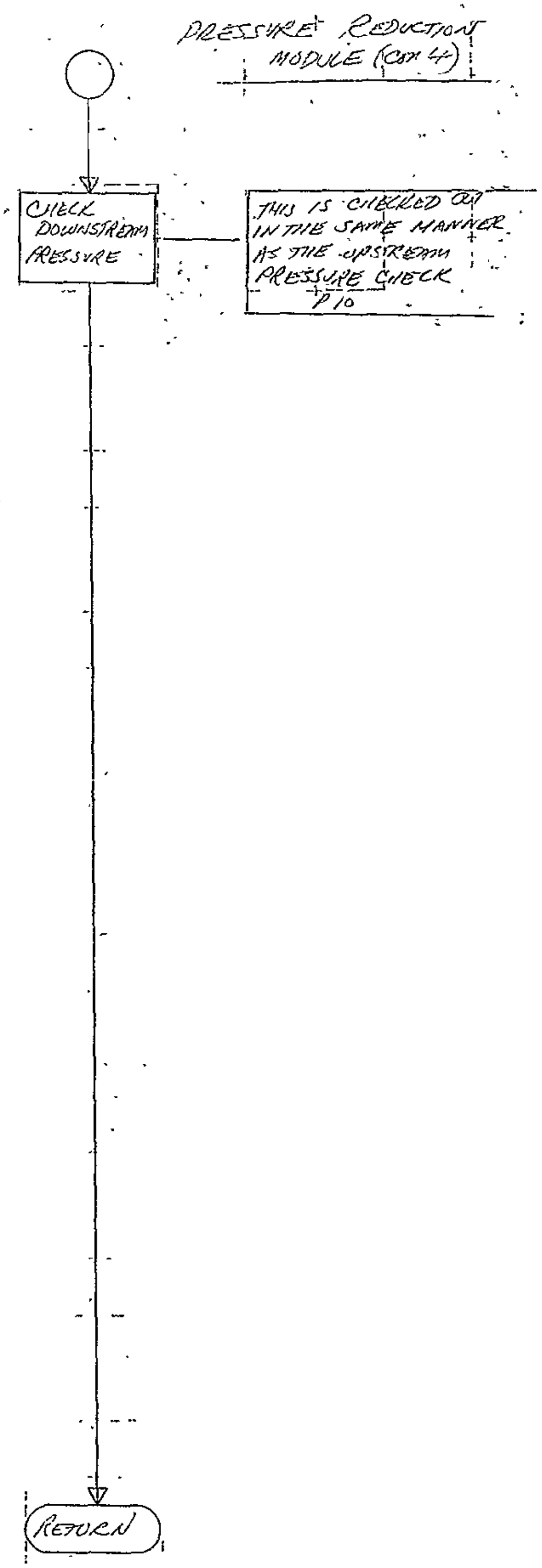


Figure 3-12. Pressure Reduction Module (Sheet 4 of 4)

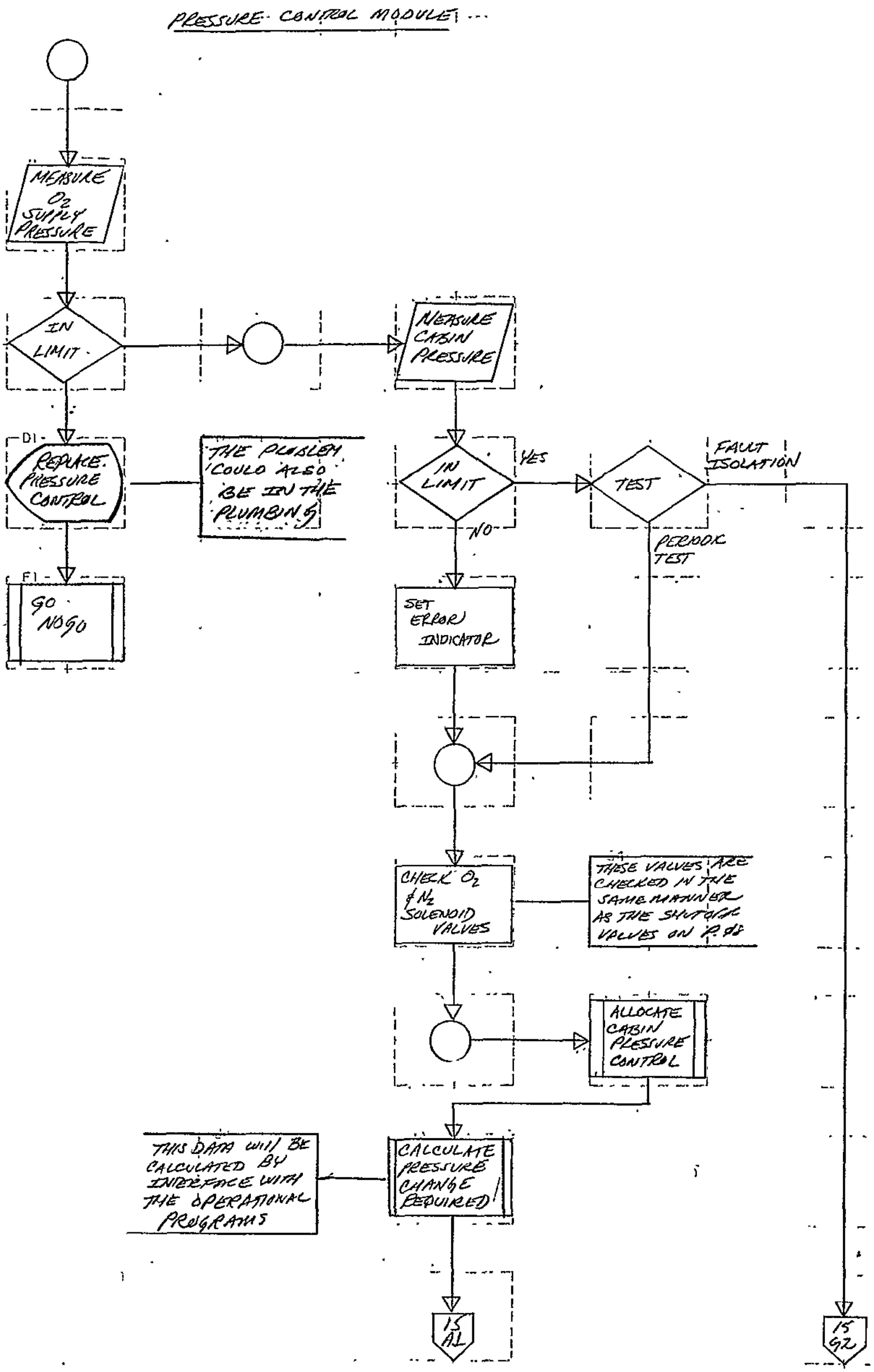


Figure 3-13. Pressure Control Module (Sheet 1 of 2)

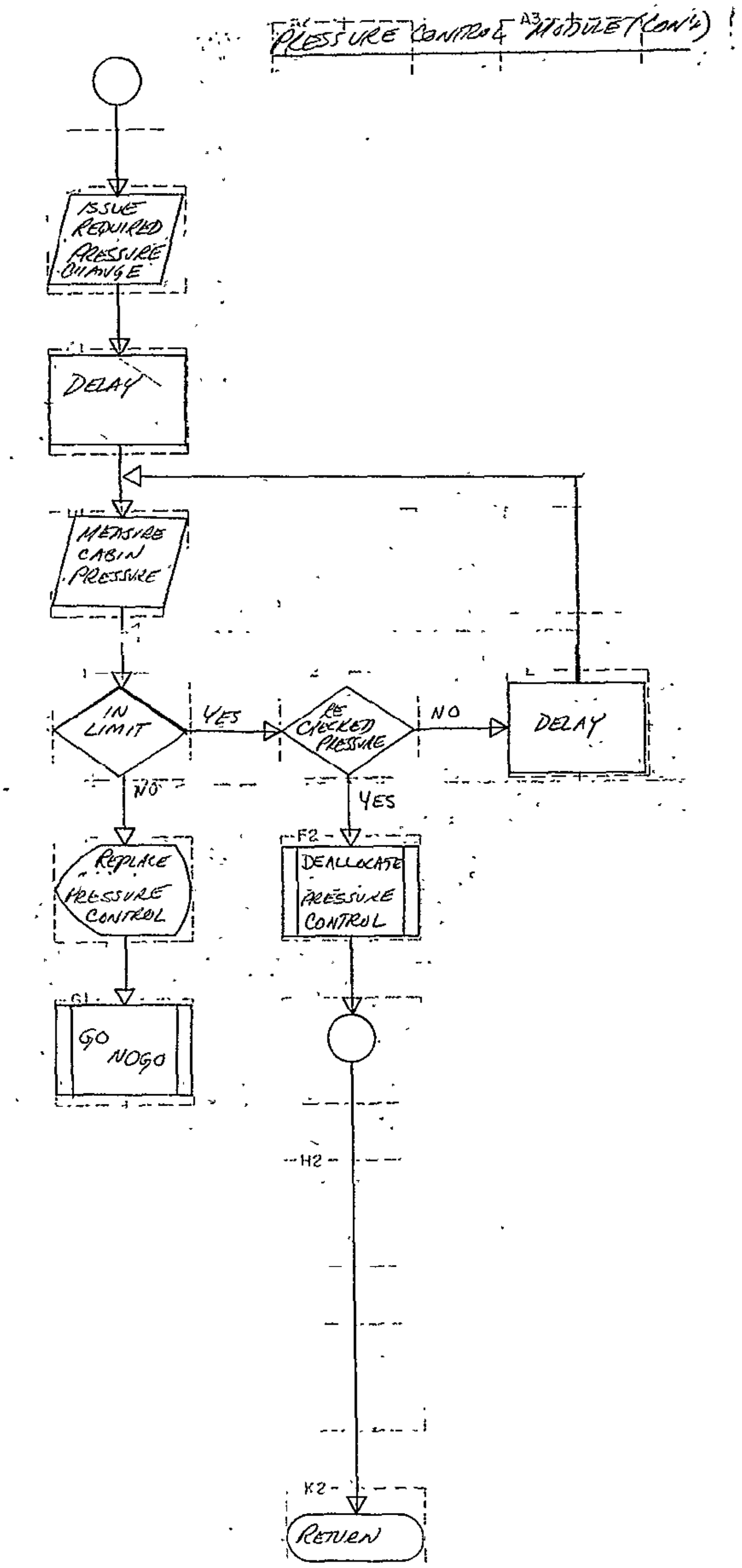


Figure 3-13. Pressure Control Module (Sheet 2 of 2)

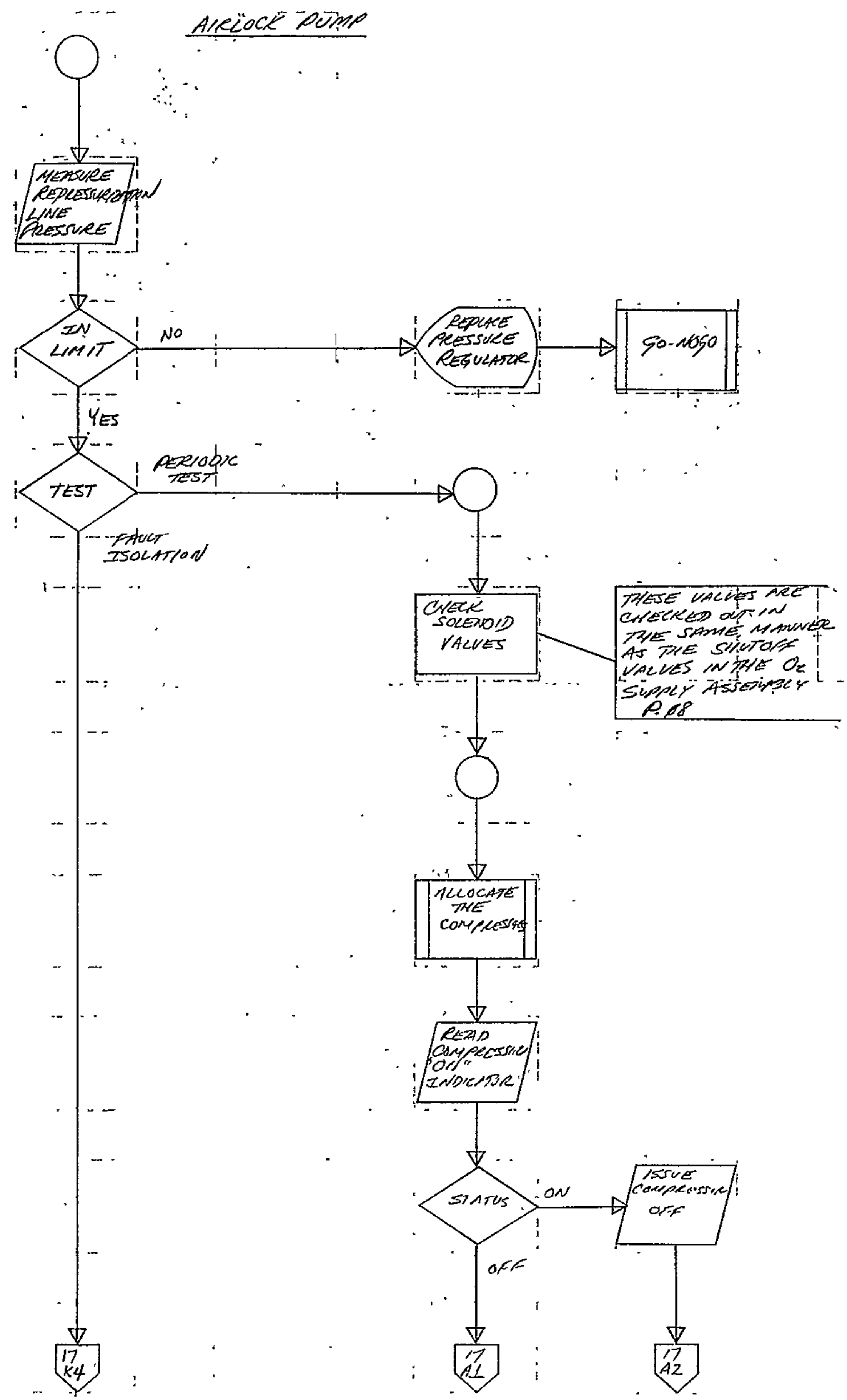


Figure 3-14. Airlock Pump (Sheet 1 of 2)

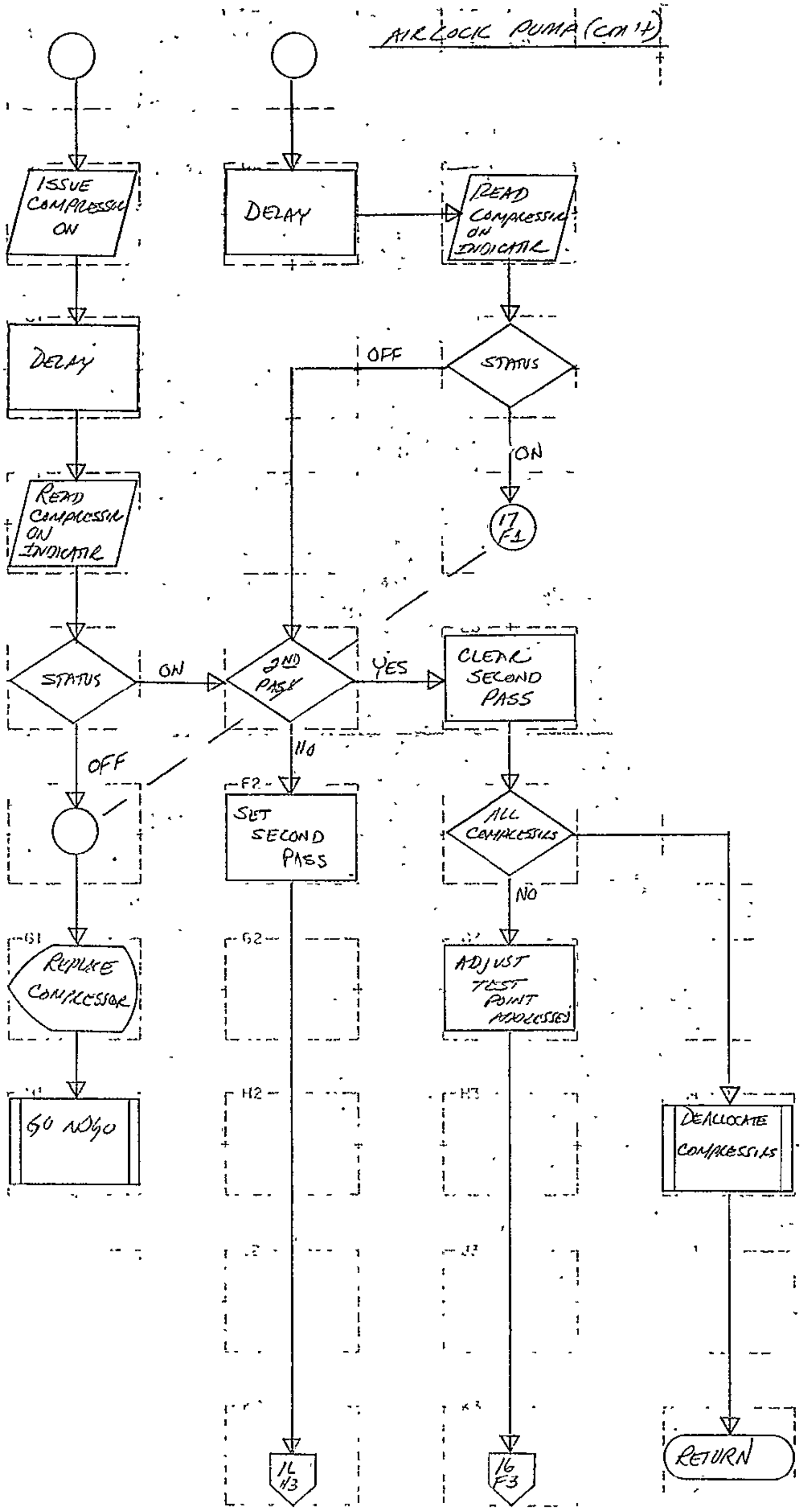


Figure 3-14. Airlock Pump (Sheet 2 of 2)

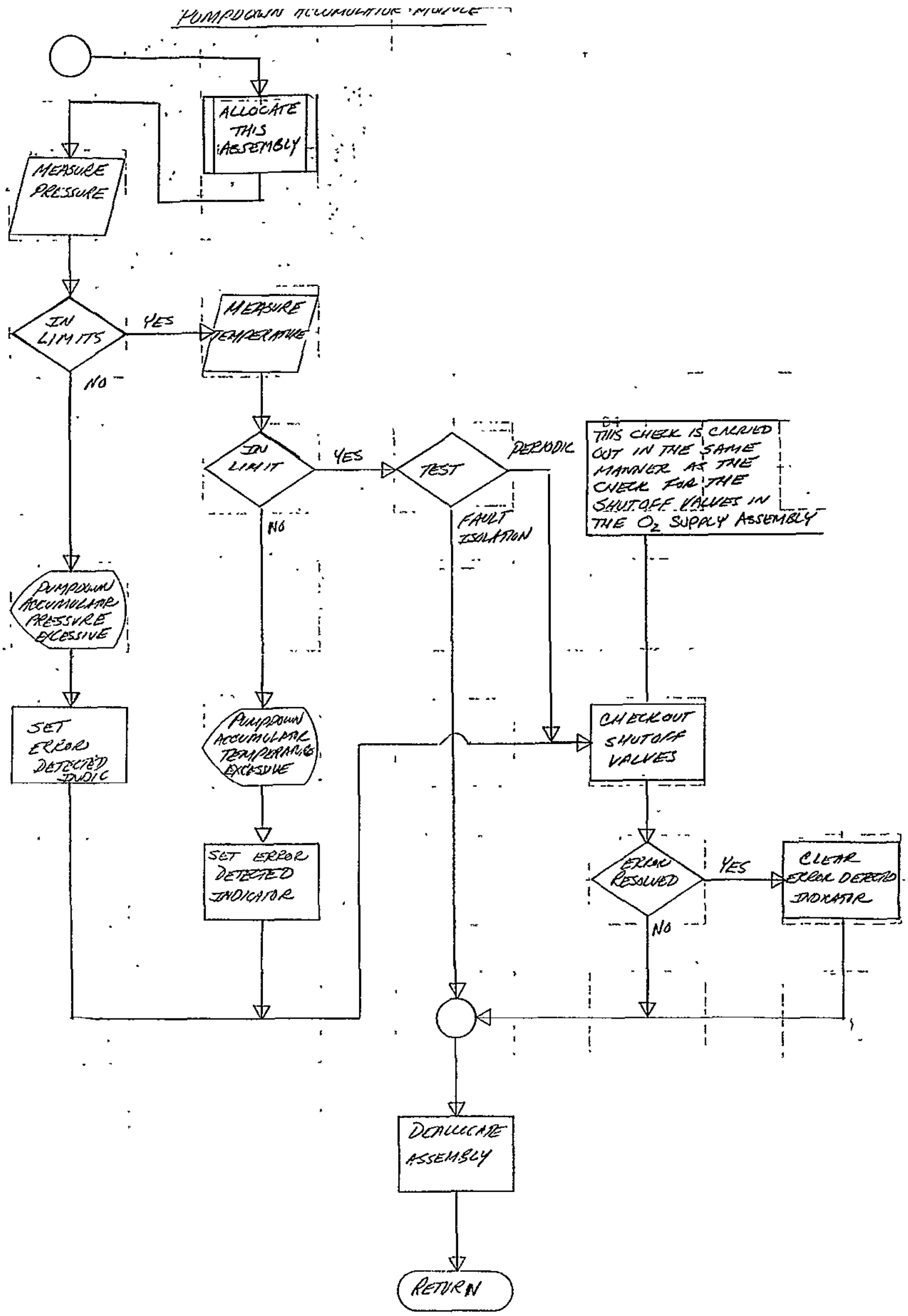


Figure 3-15. Pumpdown Accumulator Module

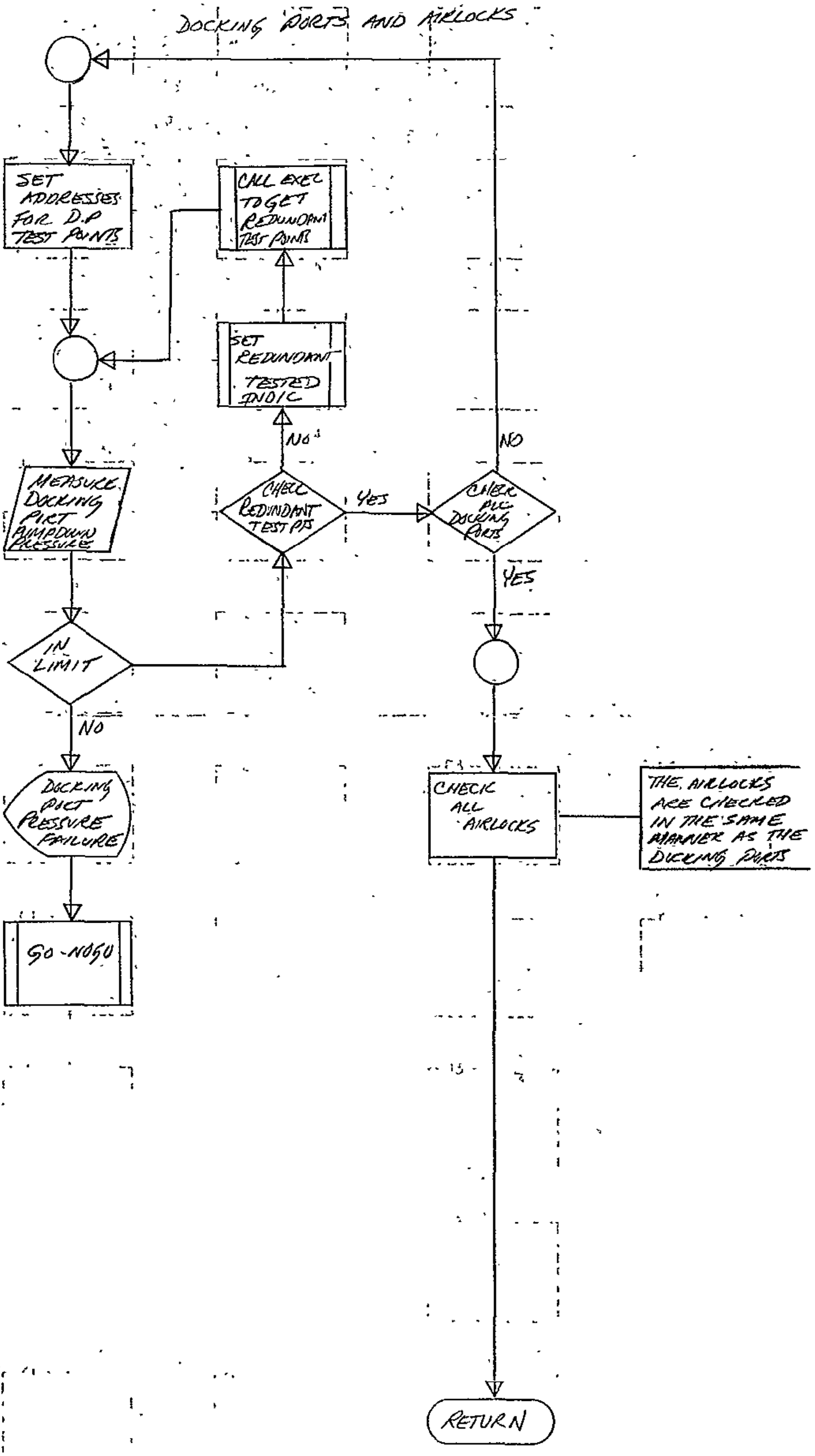


Figure 3-16. Docking Ports and Airlocks

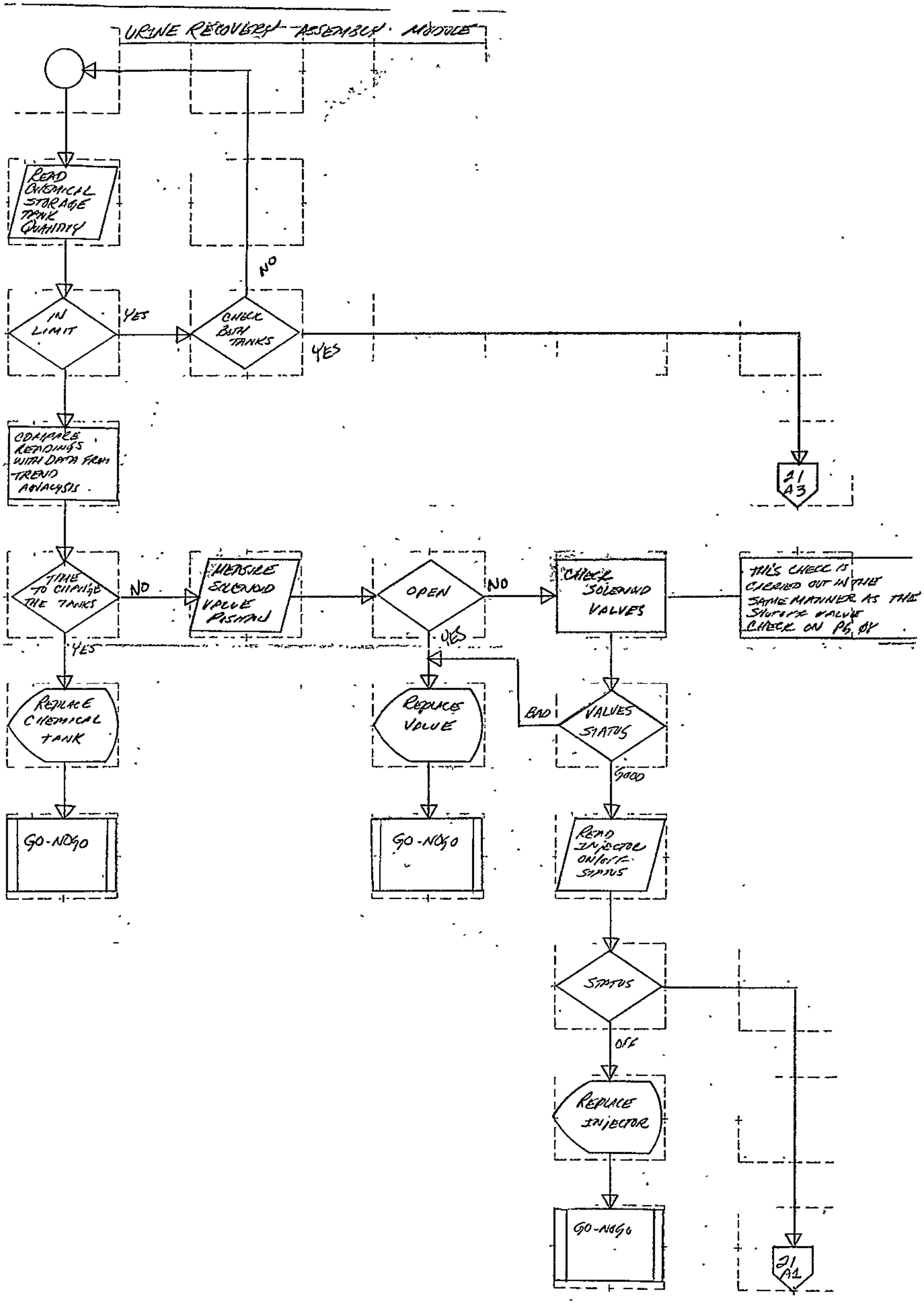


Figure 3-17. Urine Recovery Assembly Module (Sheet 1 of 6)

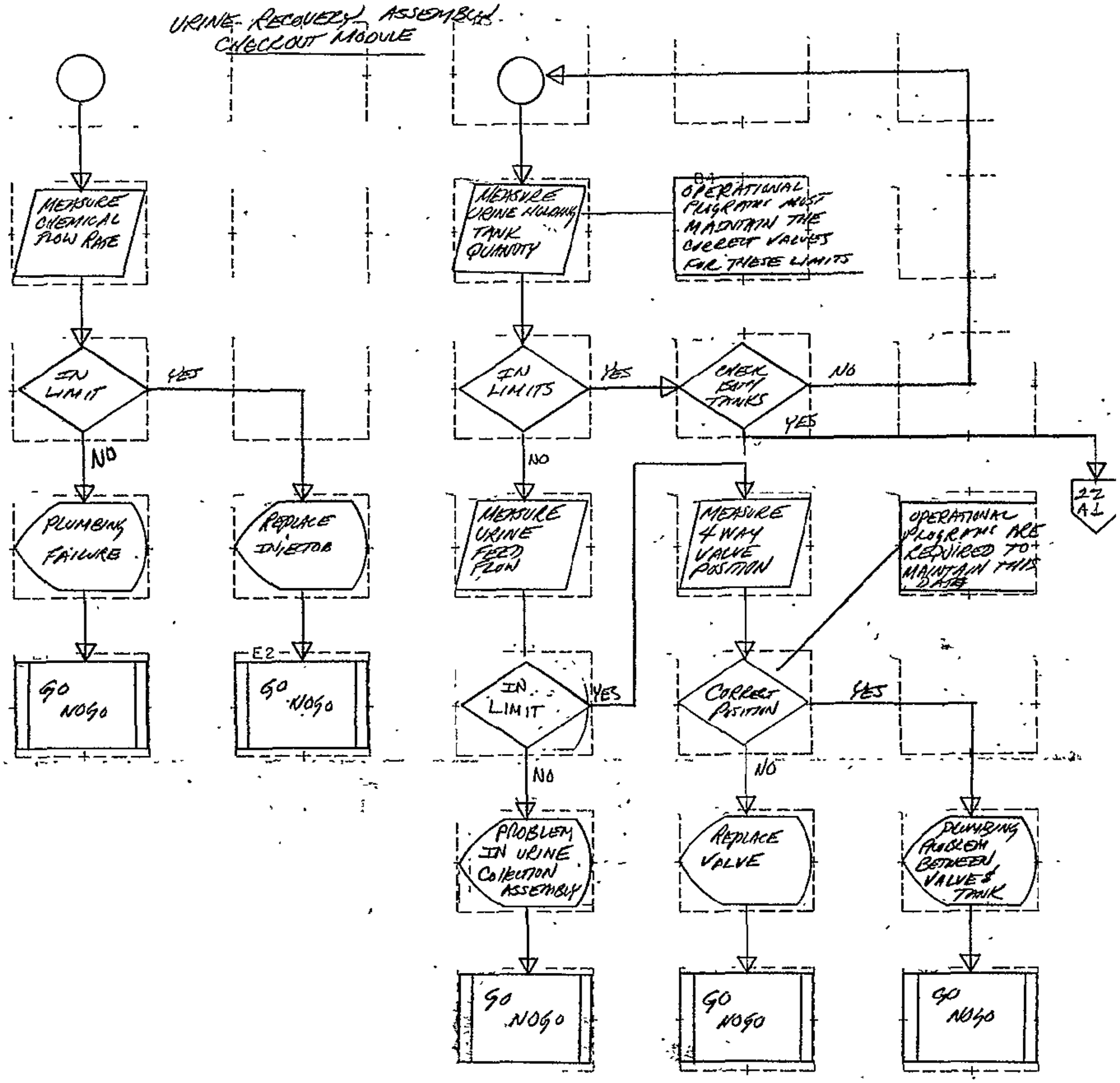


Figure 3-17. Urine Recovery Assembly Checkout Module (Sheet 2 of 6)

URINE RECOVERY ASSEMBLY
CHECKOUT MODULE

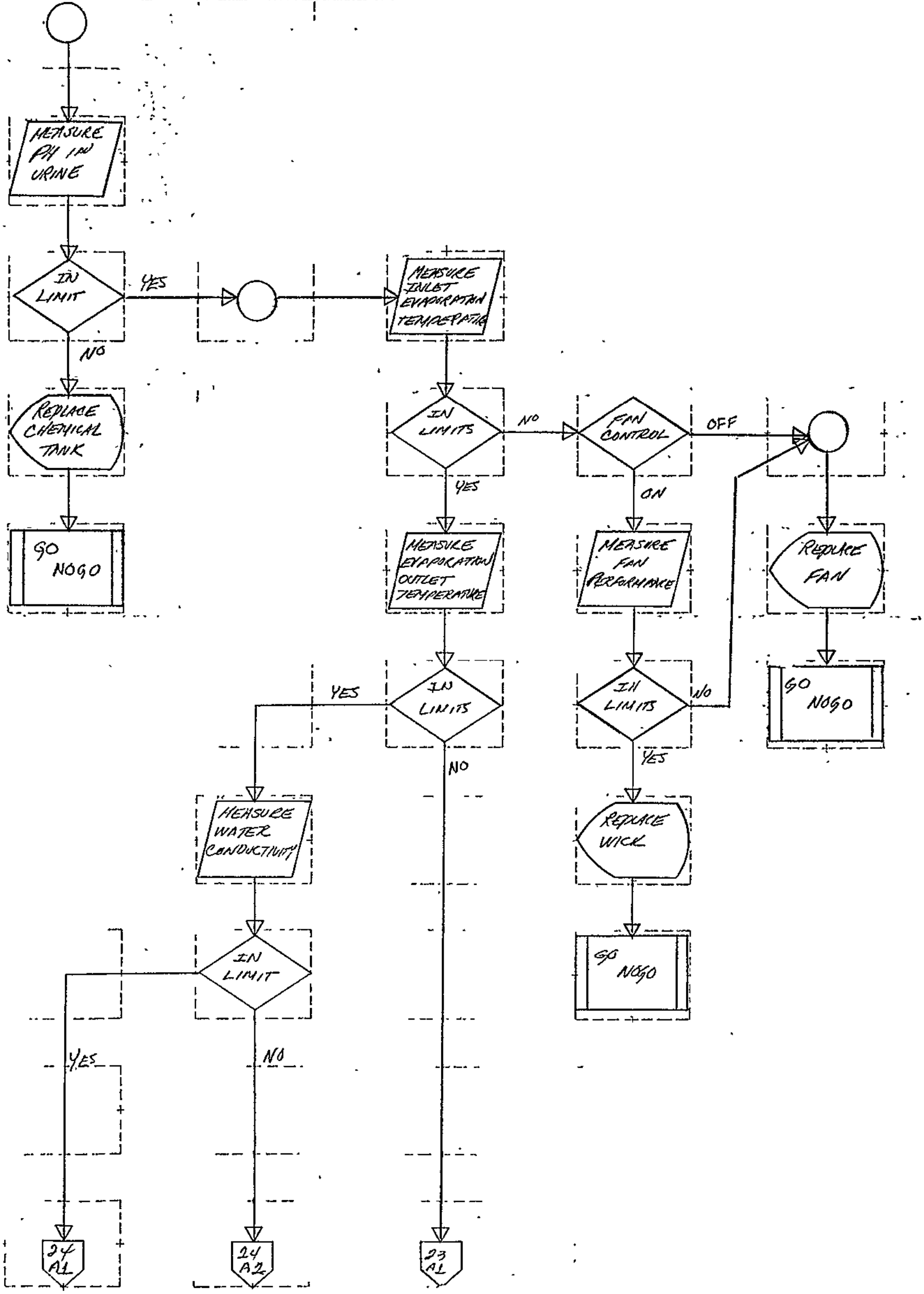


Figure 3-17. Urine Recovery Assembly Checkout Module (Sheet 3 of 6)

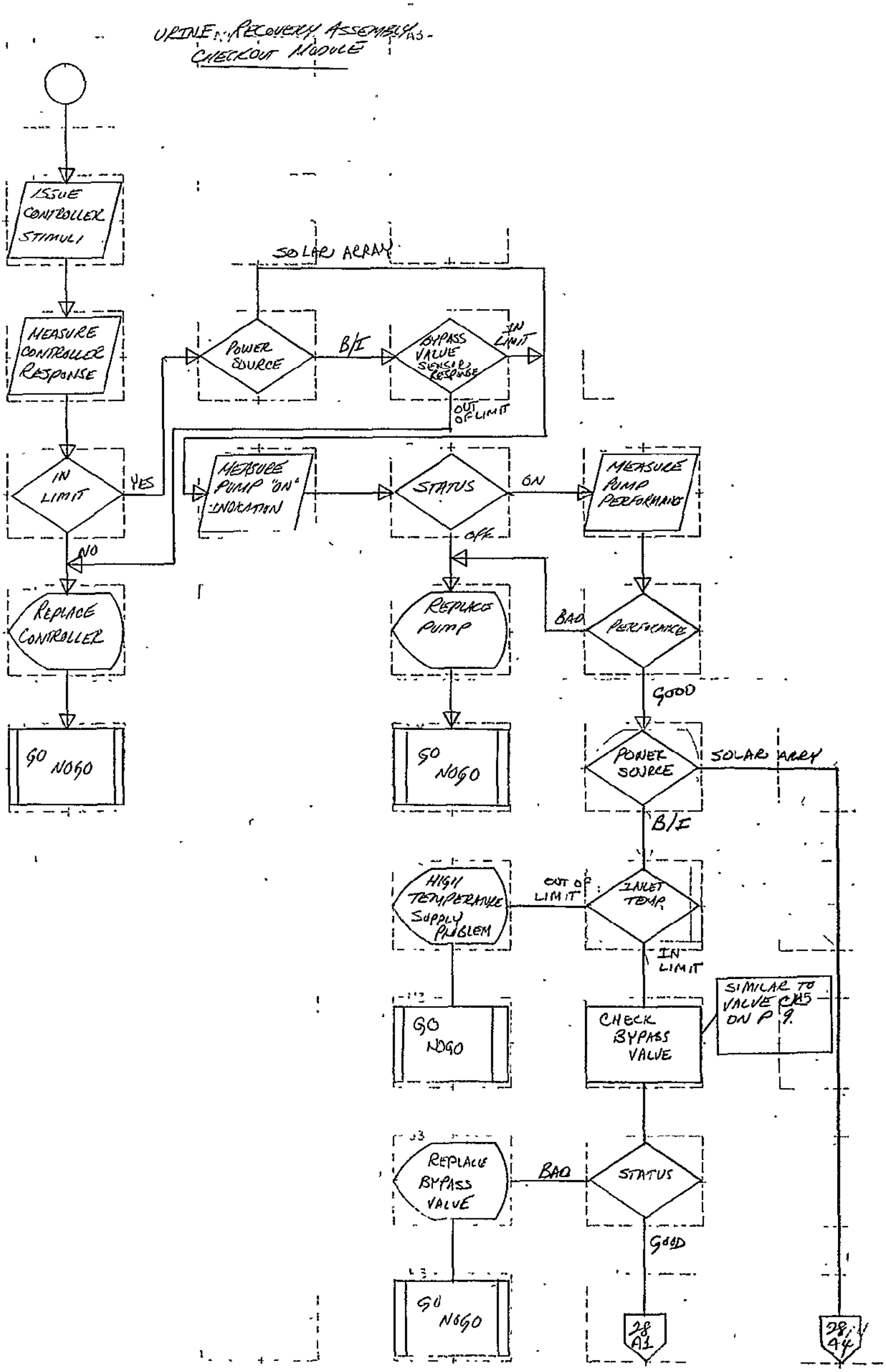


Figure 3-17. Urine Recovery Assembly Checkout Module (Sheet 4 of 6)

URINE RECOVERY ASSEMBLY
CHECKOUT MODULE

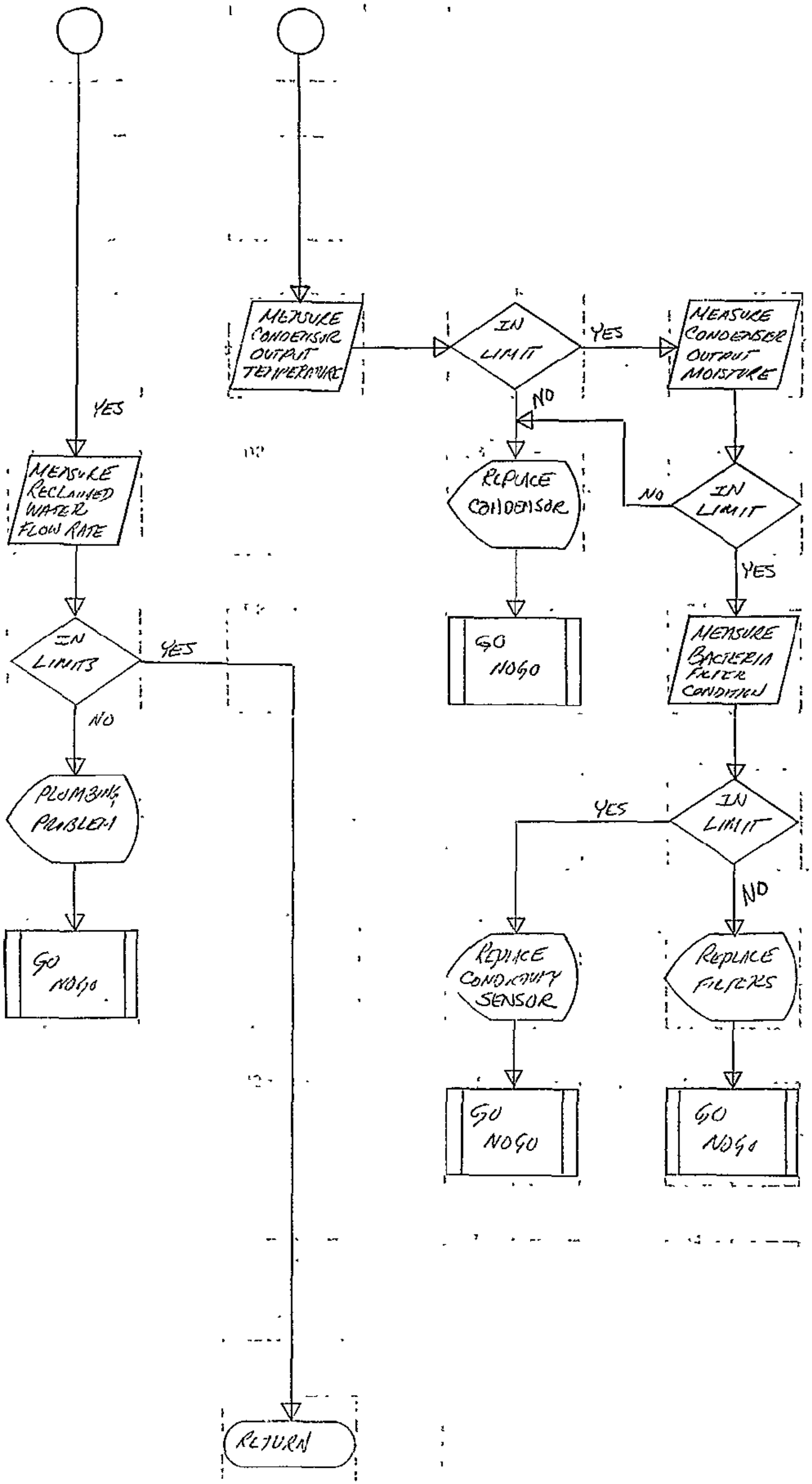


Figure 3-17. Urine Recovery Assembly Checkout Module (Sheet 5 of 6)

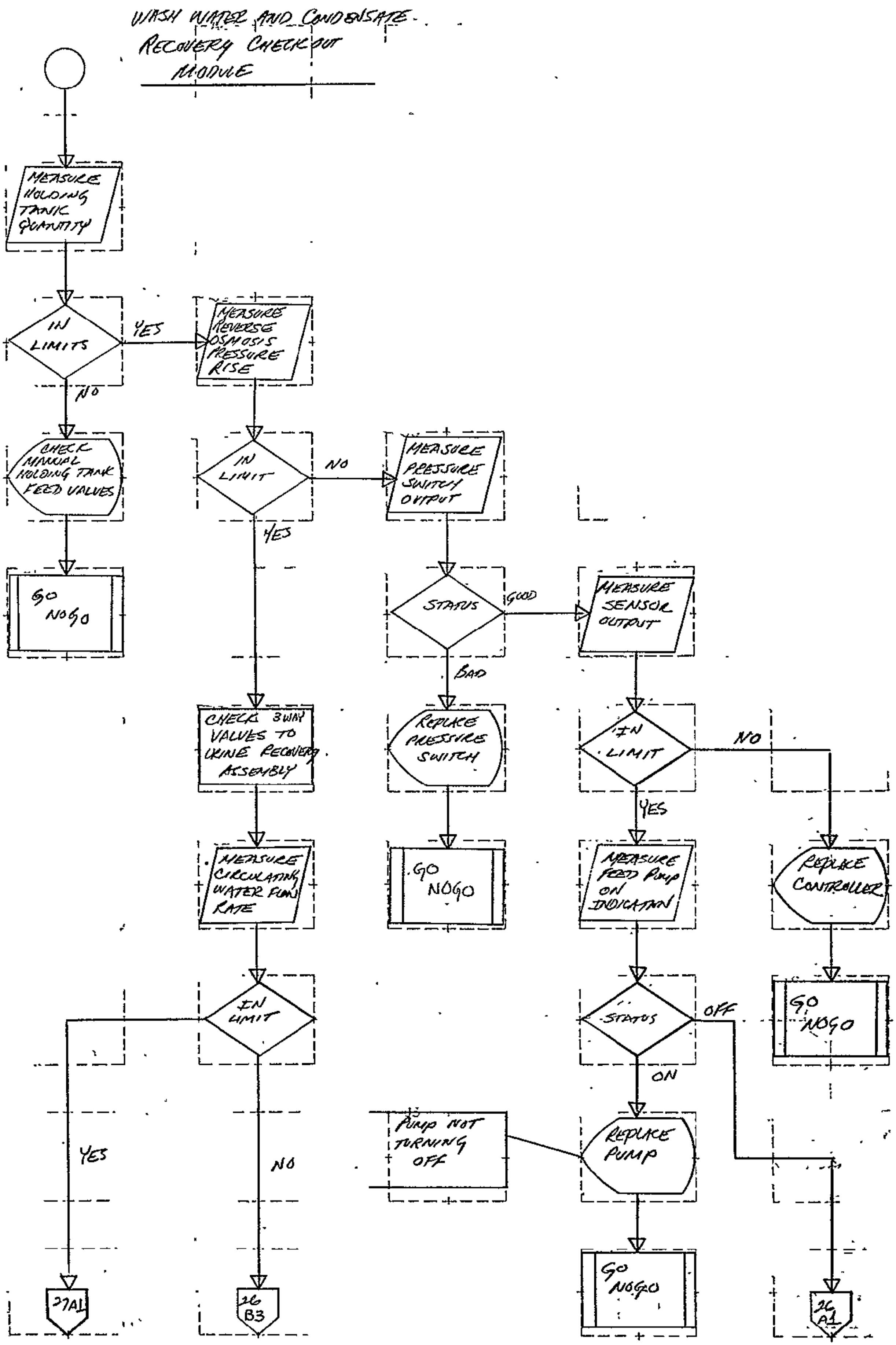


Figure 3-17. Urine Recovery Assembly Checkout Module (Sheet 6 of 6)

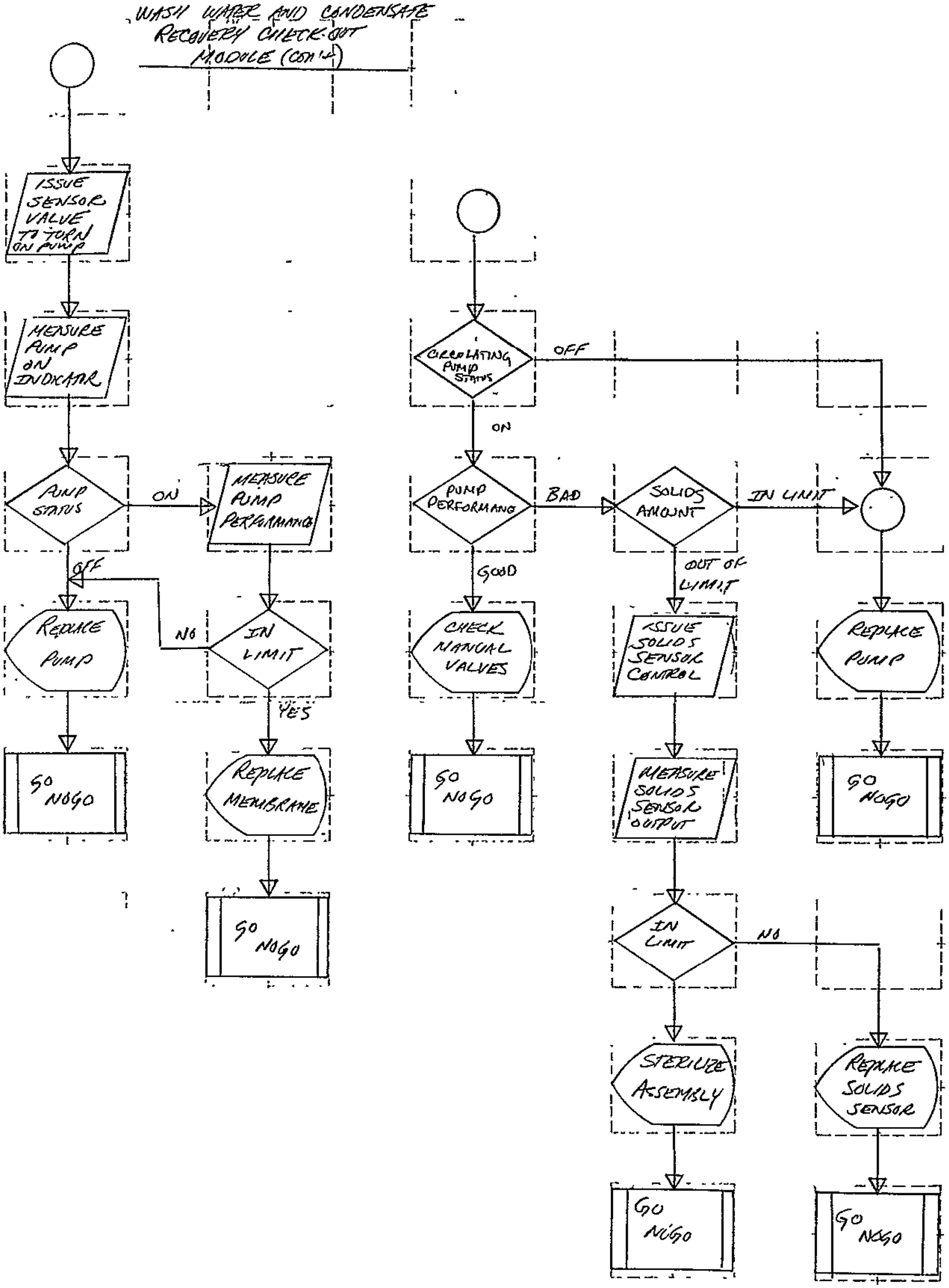


Figure 3-18. Wash Water and Condensate Recovery Checkout Module (Sheet 1 of 3)

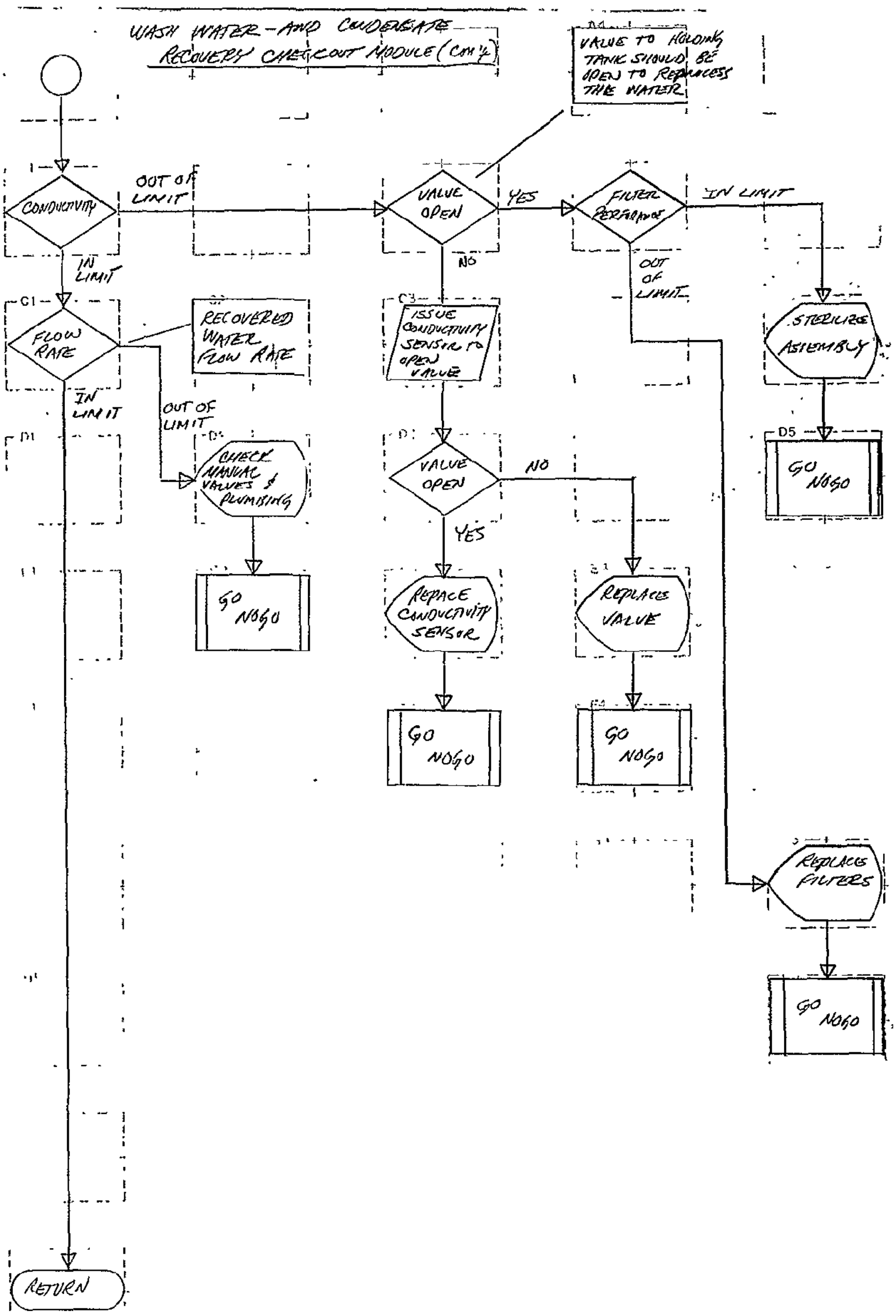


Figure 3-18. Wash Water and Condensate Recovery Checkout Module (Sheet 2 of 3)

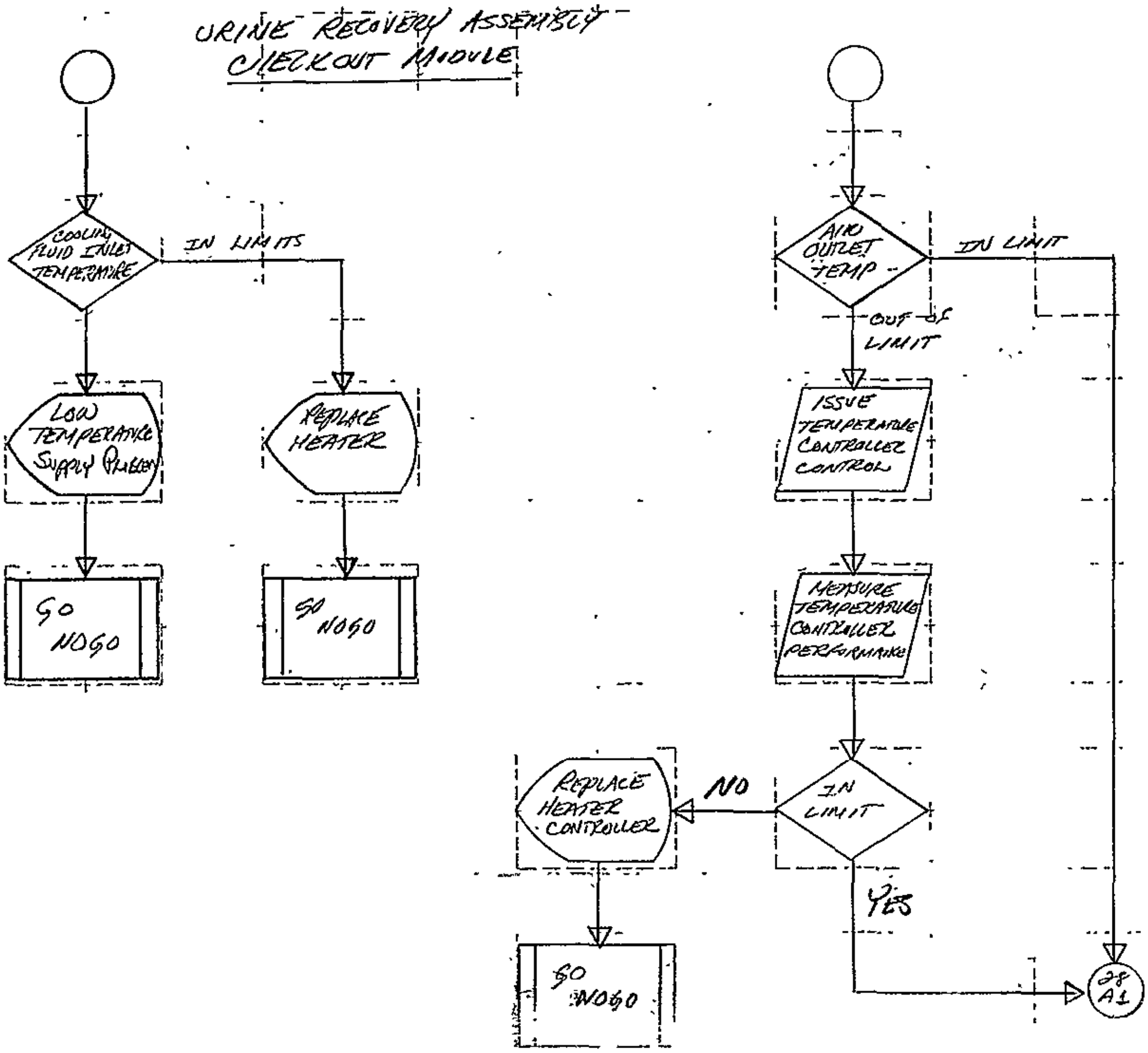


Figure 3-18. Wash Water and Condensate Recovery Checkout Module (Sheet 3 of 3)

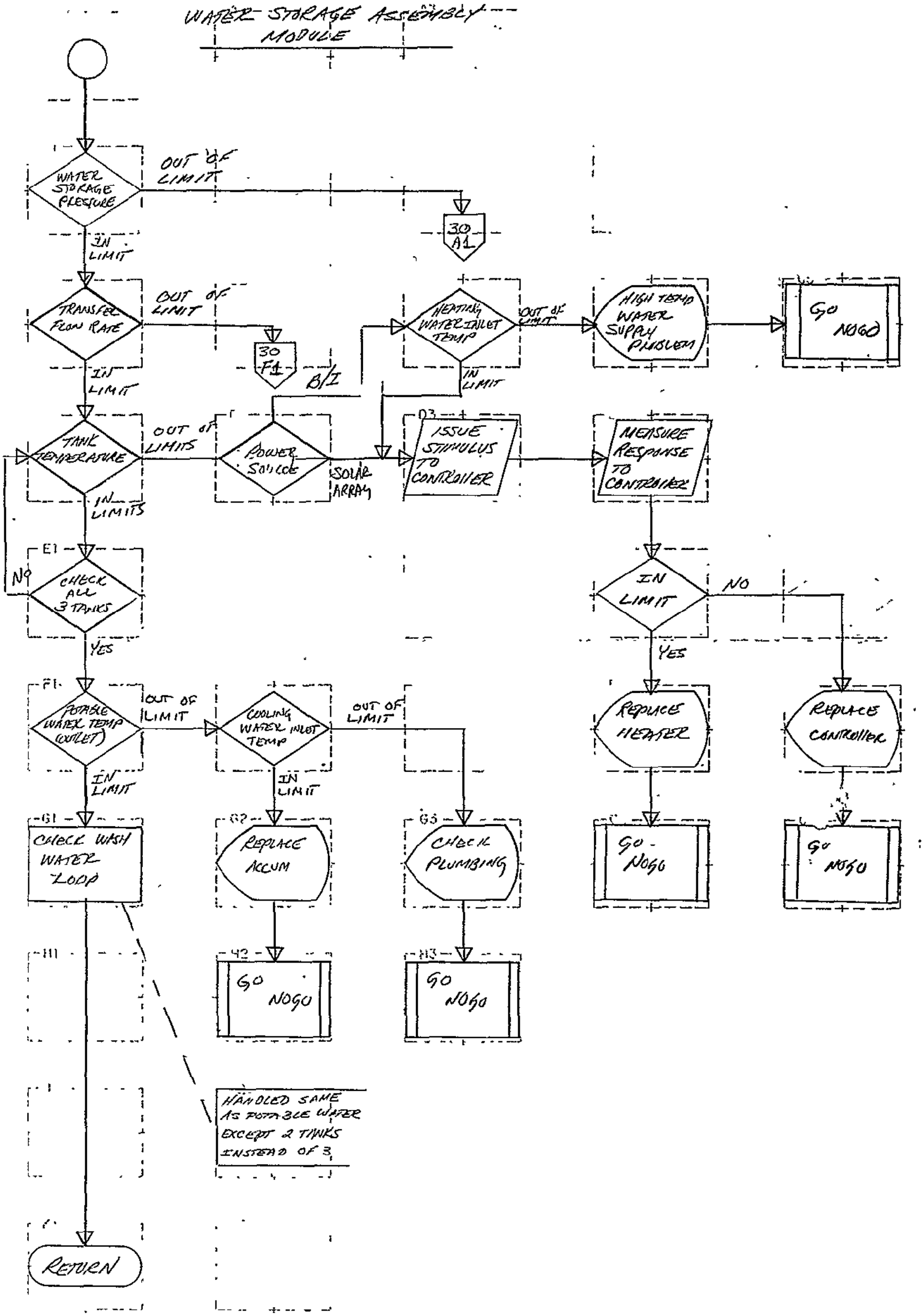


Figure 3-19. Water Storage Assembly Module (Sheet 1 of 2)

WATER STORAGE ASSEMBLY MODULE (Cont'd)

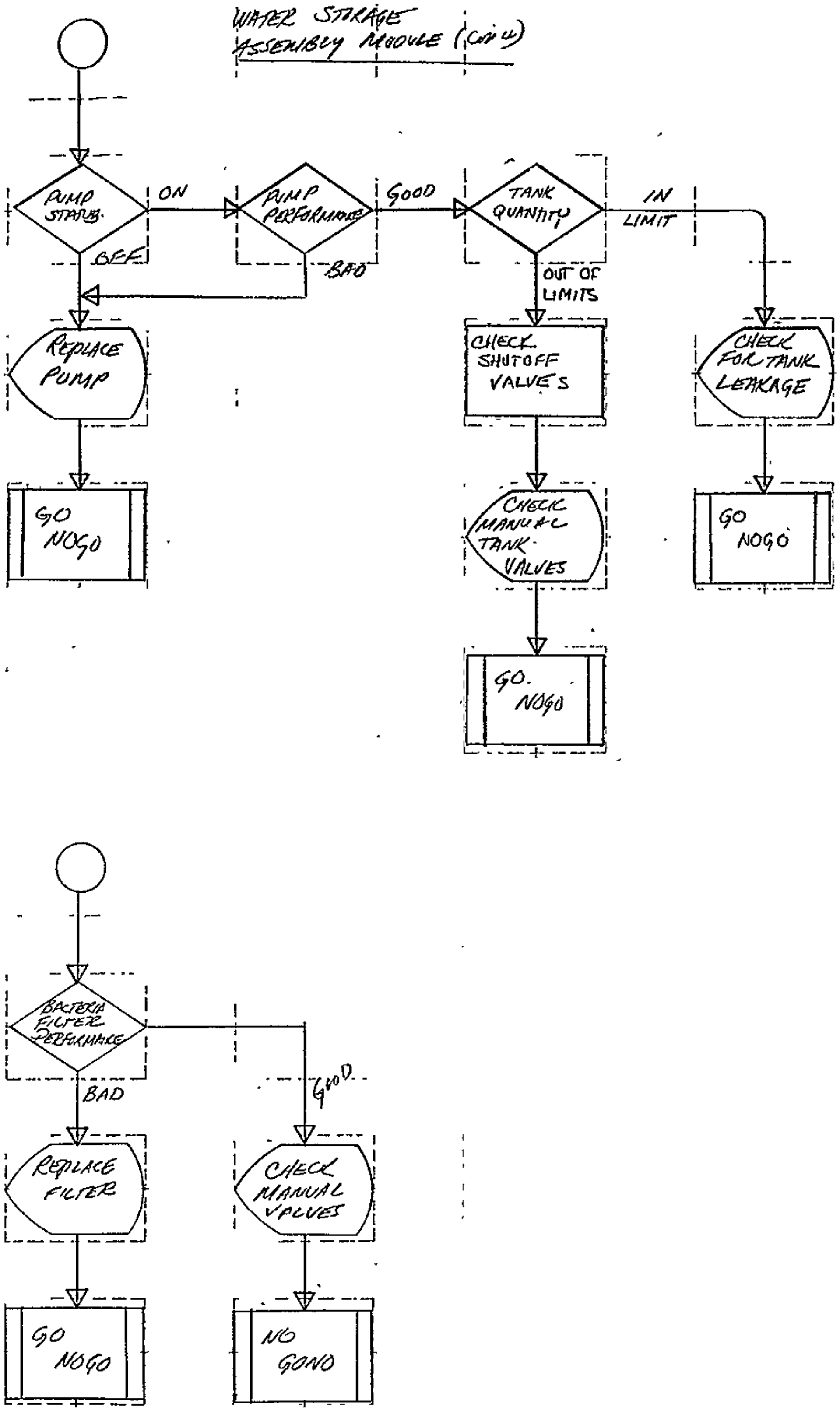


Figure 3-19. Water Storage Assembly Module (Sheet 2 of 2)

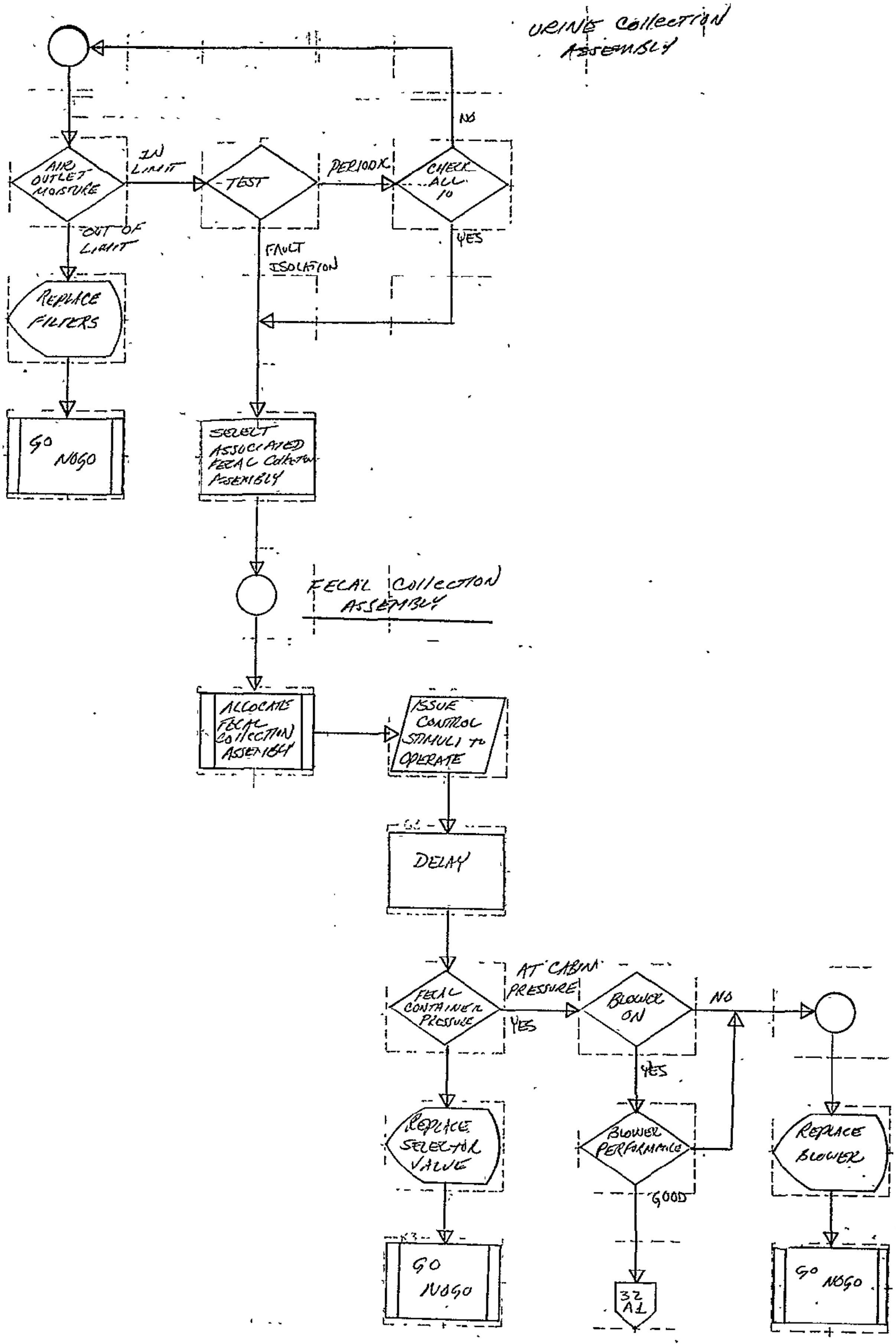


Figure 3-20. Urine Collection Assembly (Sheet 1 of 3)

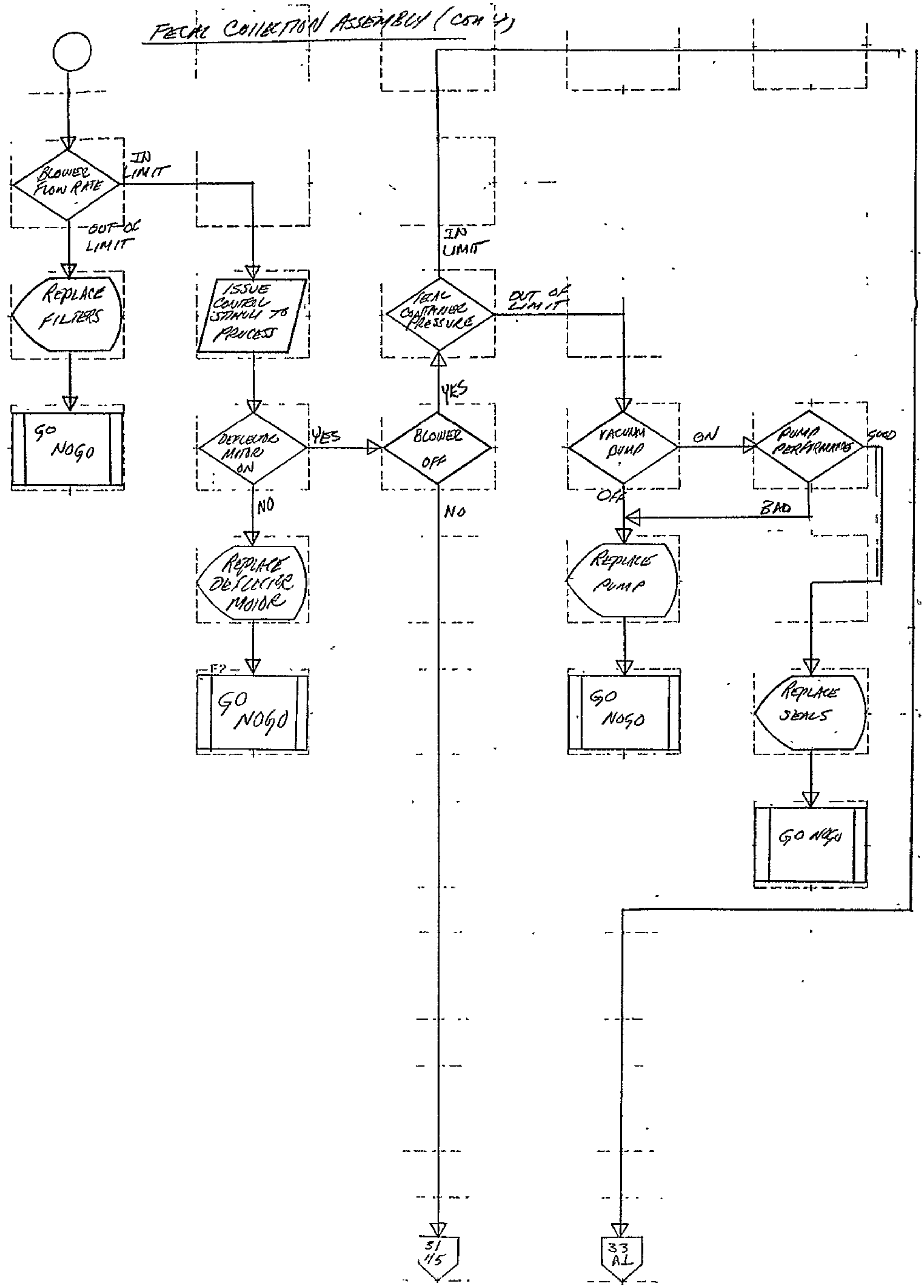


Figure 3-20. Fecal Collection Assembly (Sheet 2 of 3)

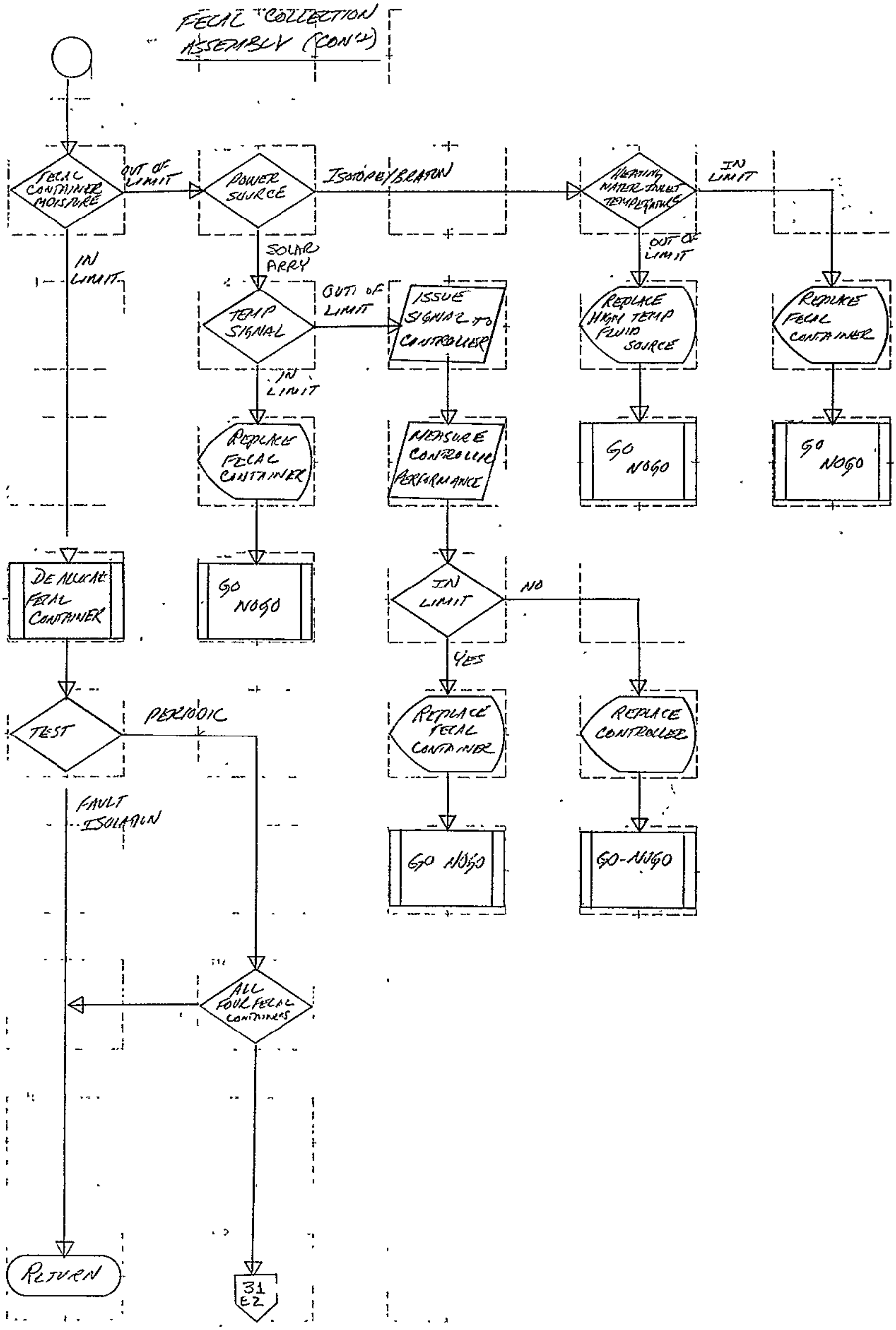


Figure 3-20. Fecal Collection Assembly (Sheet 3 of 3)

TEMPERATURE & HUMIDITY CONTROL

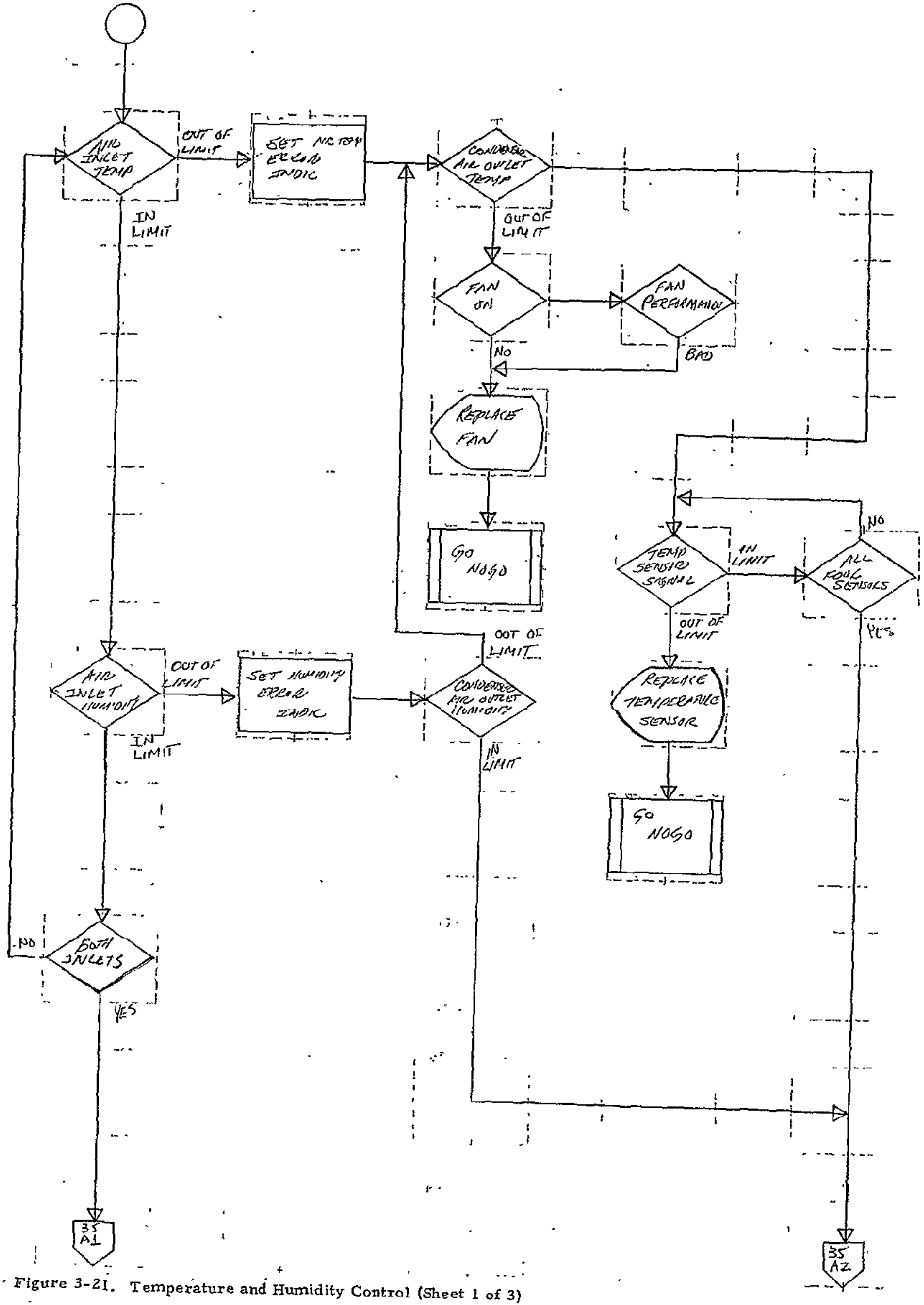


Figure 3-21. Temperature and Humidity Control (Sheet 1 of 3)

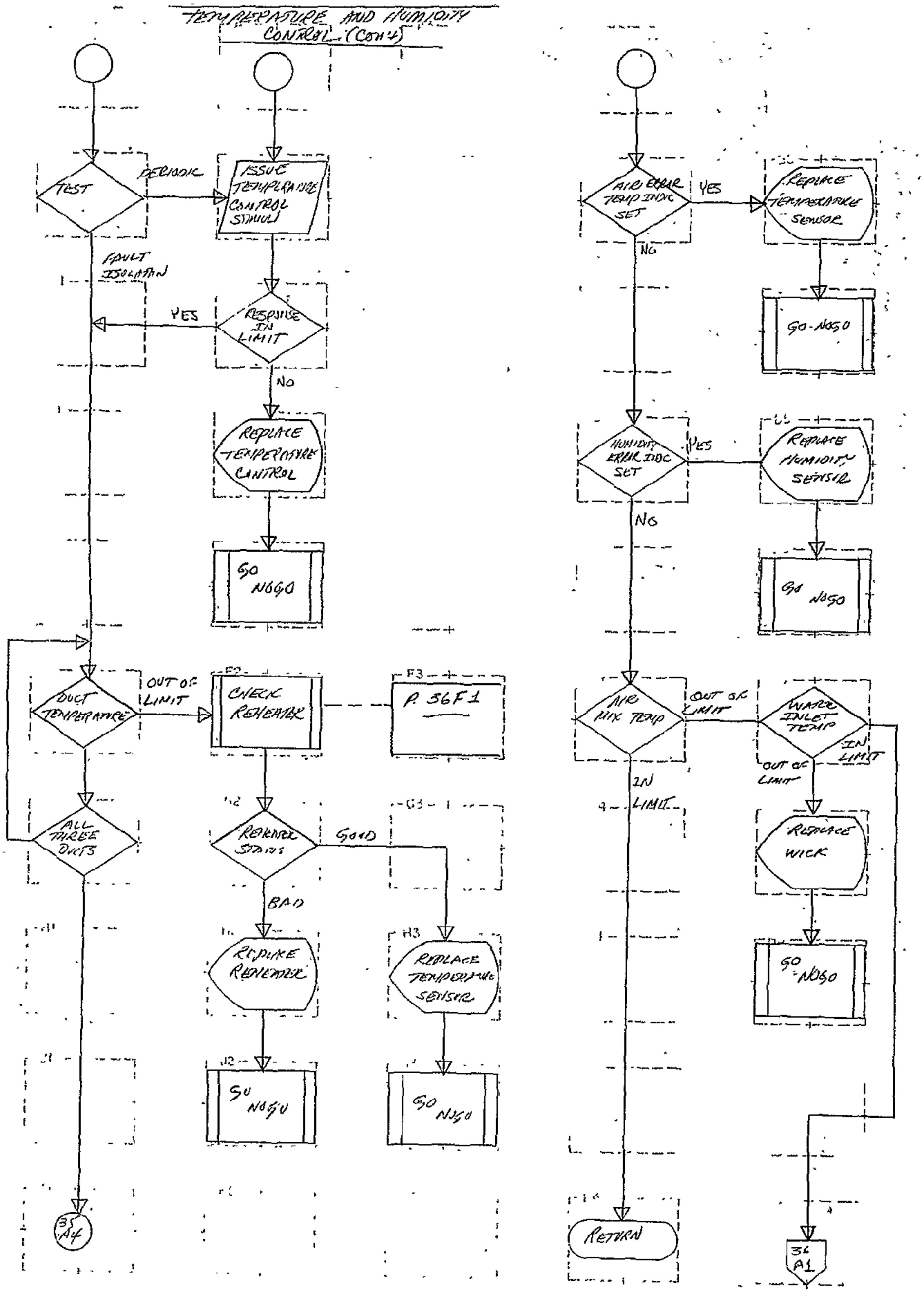
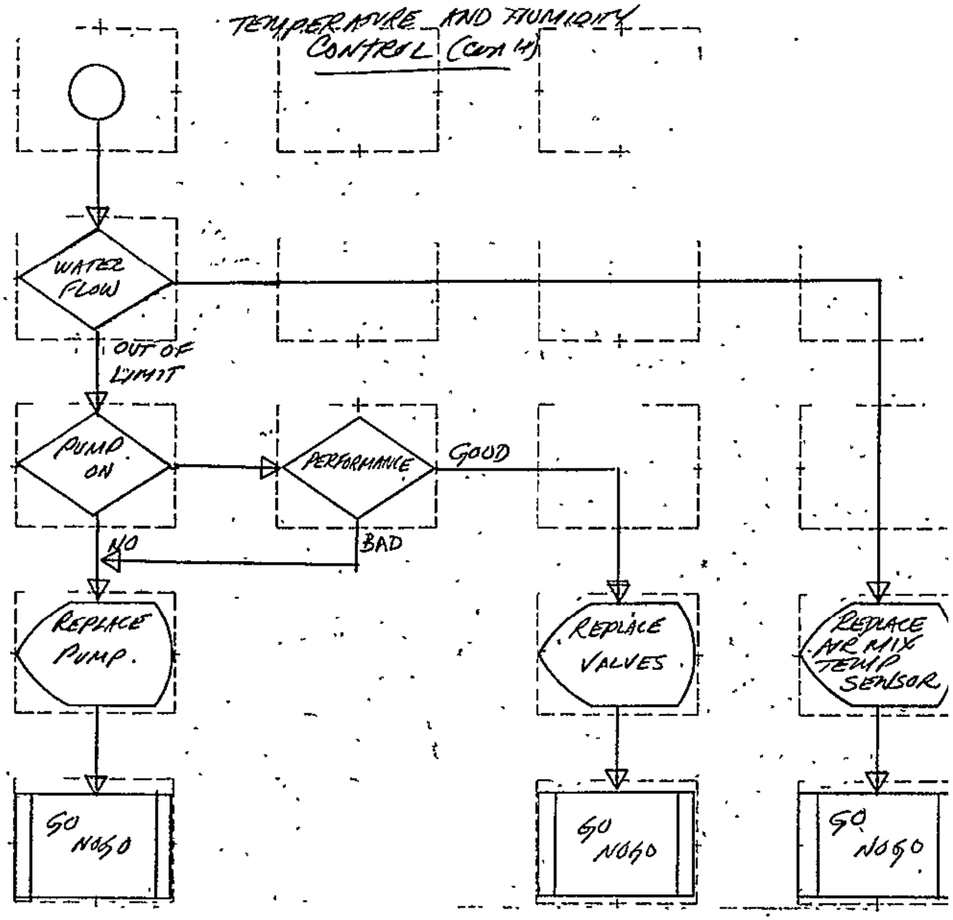


Figure 3-21. Temperature and Humidity Control (Sheet 2 of 3)

TEMPERATURE AND HUMIDITY CONTROL (Cont'd)



CHECK REHEATERS

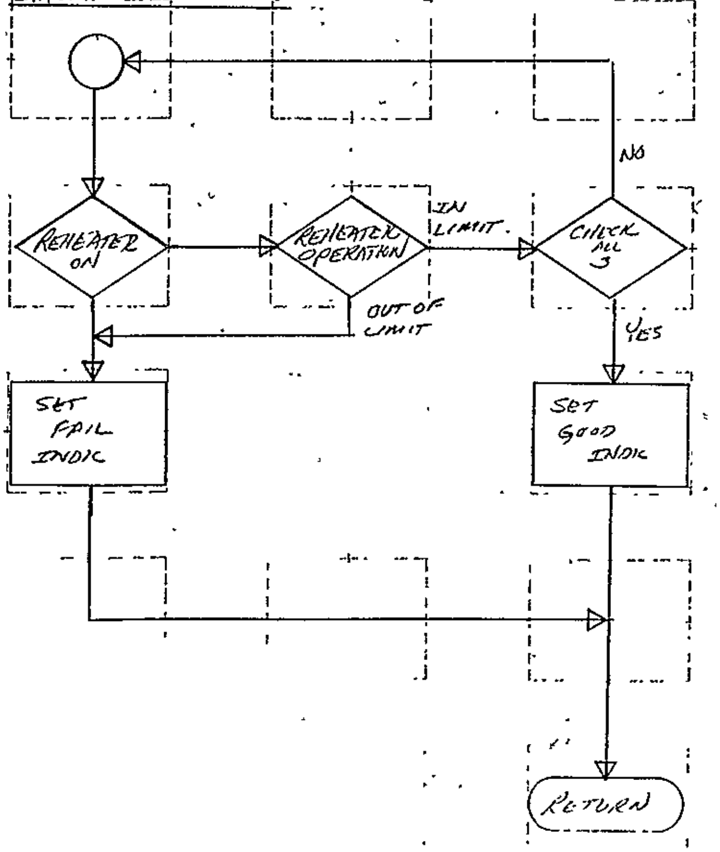


Figure 3-21. Temperature and Humidity Control (Sheet 3 of 3)

TRACE CONTAMINATION CONTROL

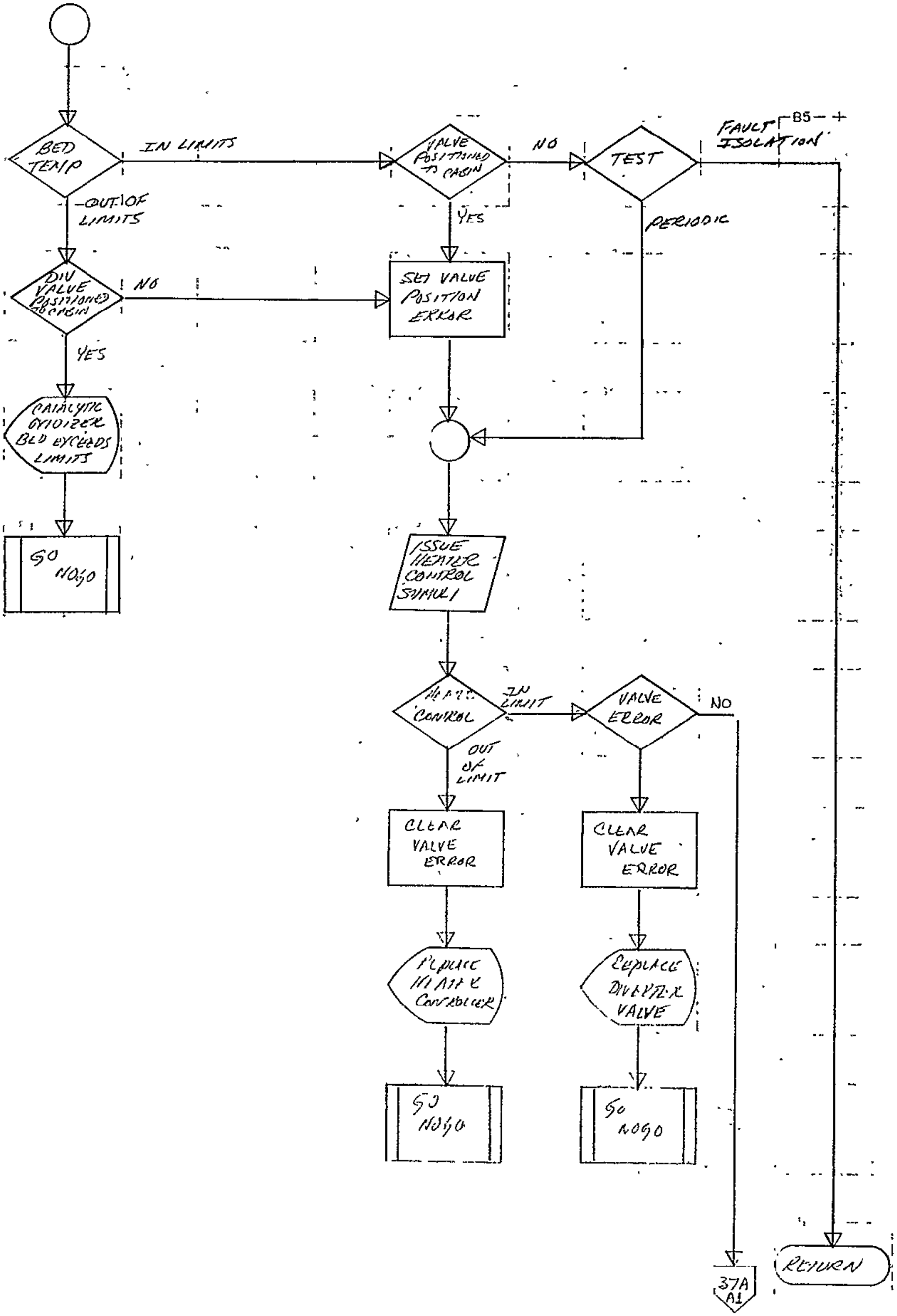


Figure 3-22. Trace Contaminant Control (Sheet 1 of 2)

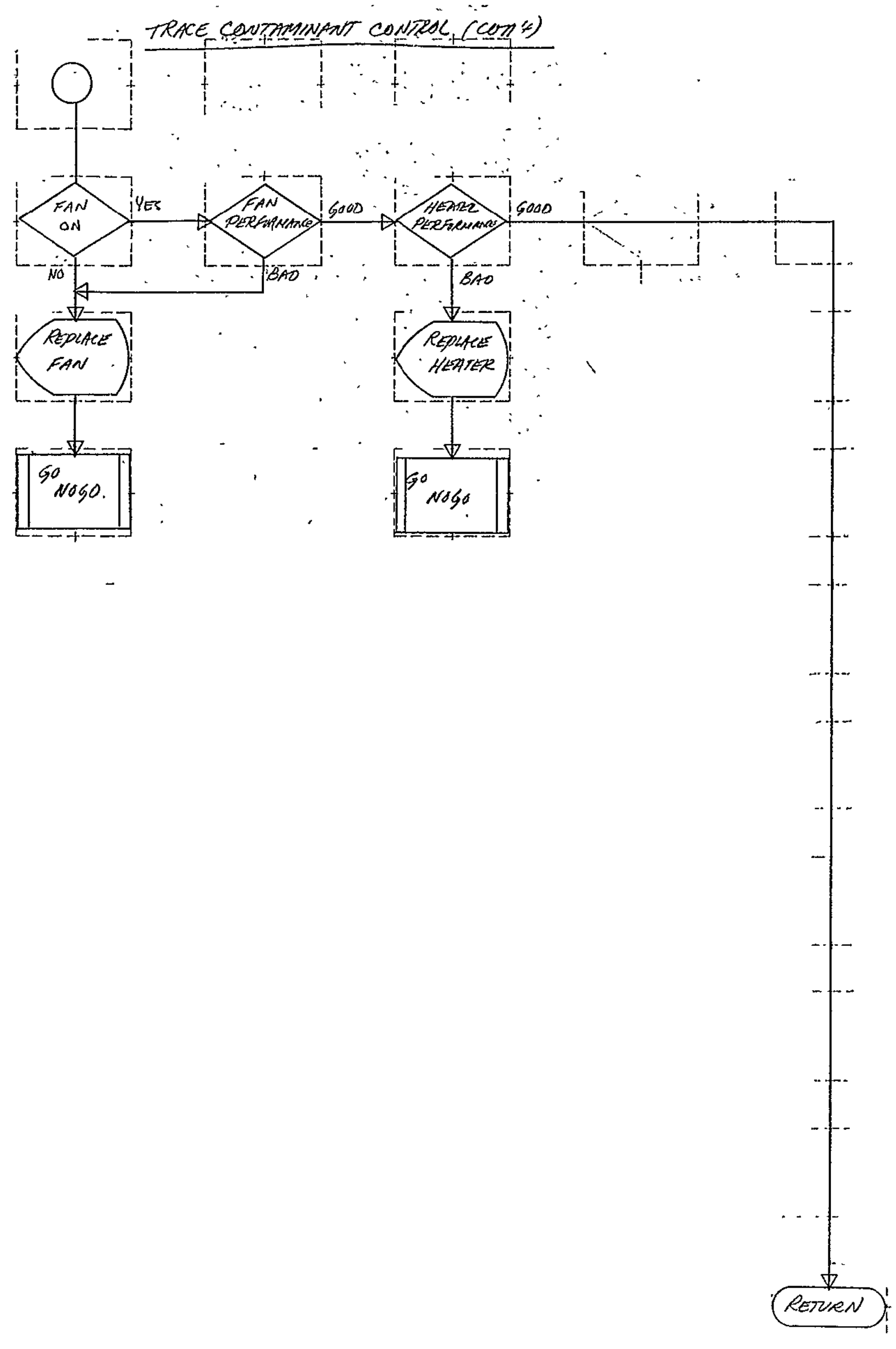


Figure 3-22. Trace Contaminant Control (Sheet 2 of 2)

CO2 CONTROL ASSEMBLY

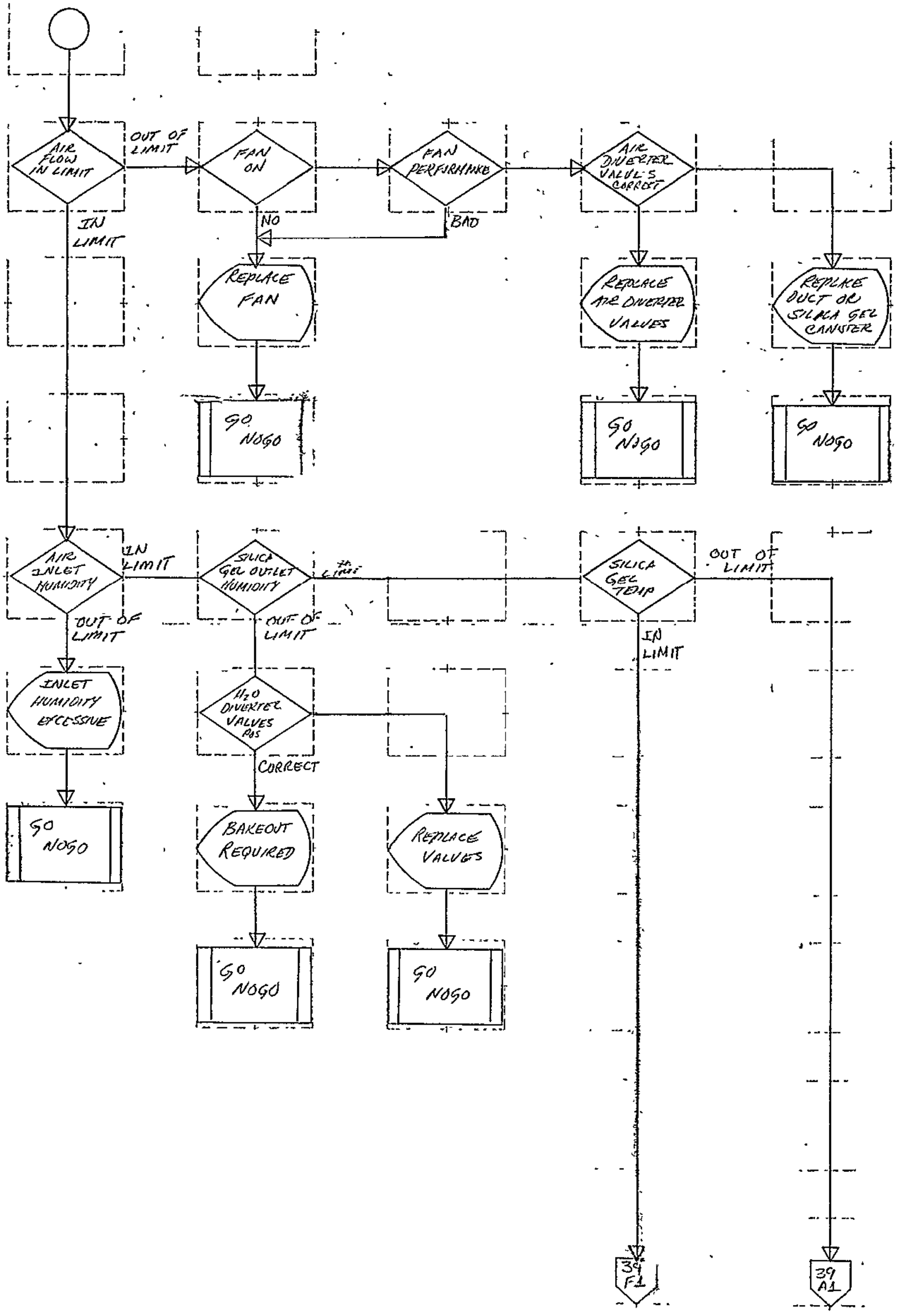


Figure 3-23. CO_2 Control Assembly (Sheet 1 of 7)

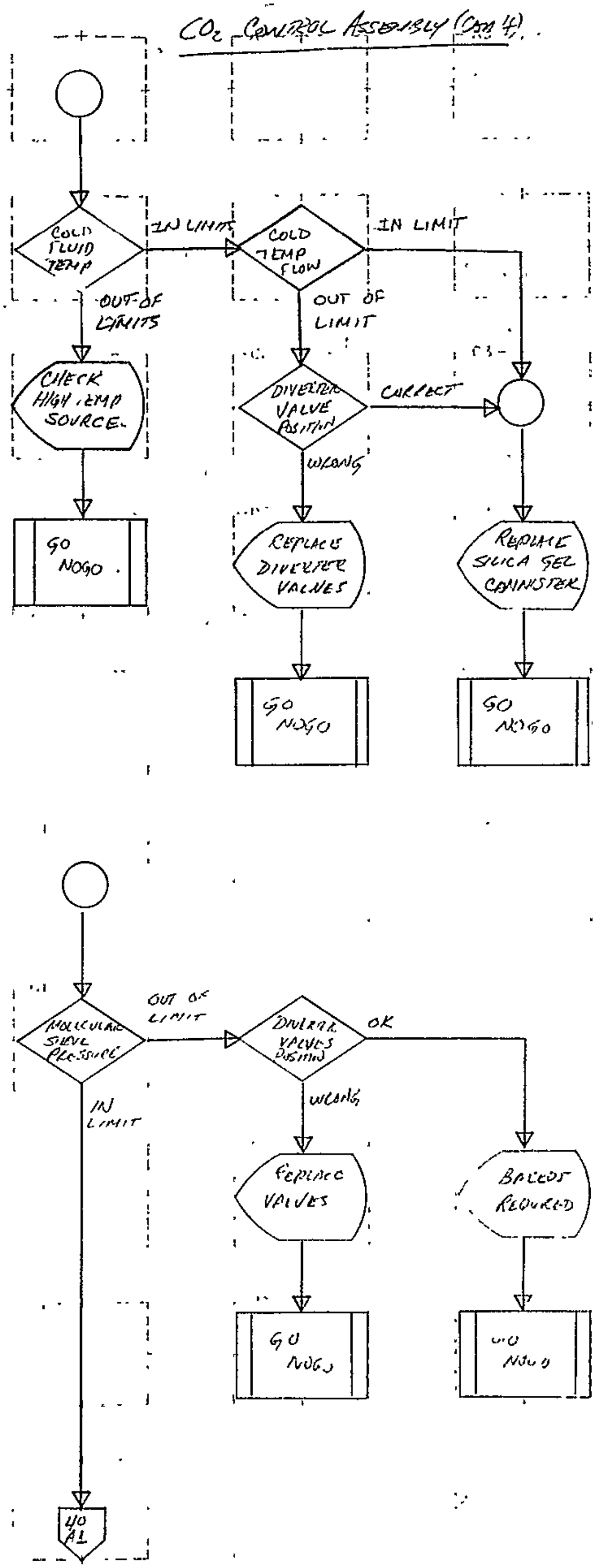


Figure 3-23. CO_2 Control Assembly (Sheet 2 of 7)

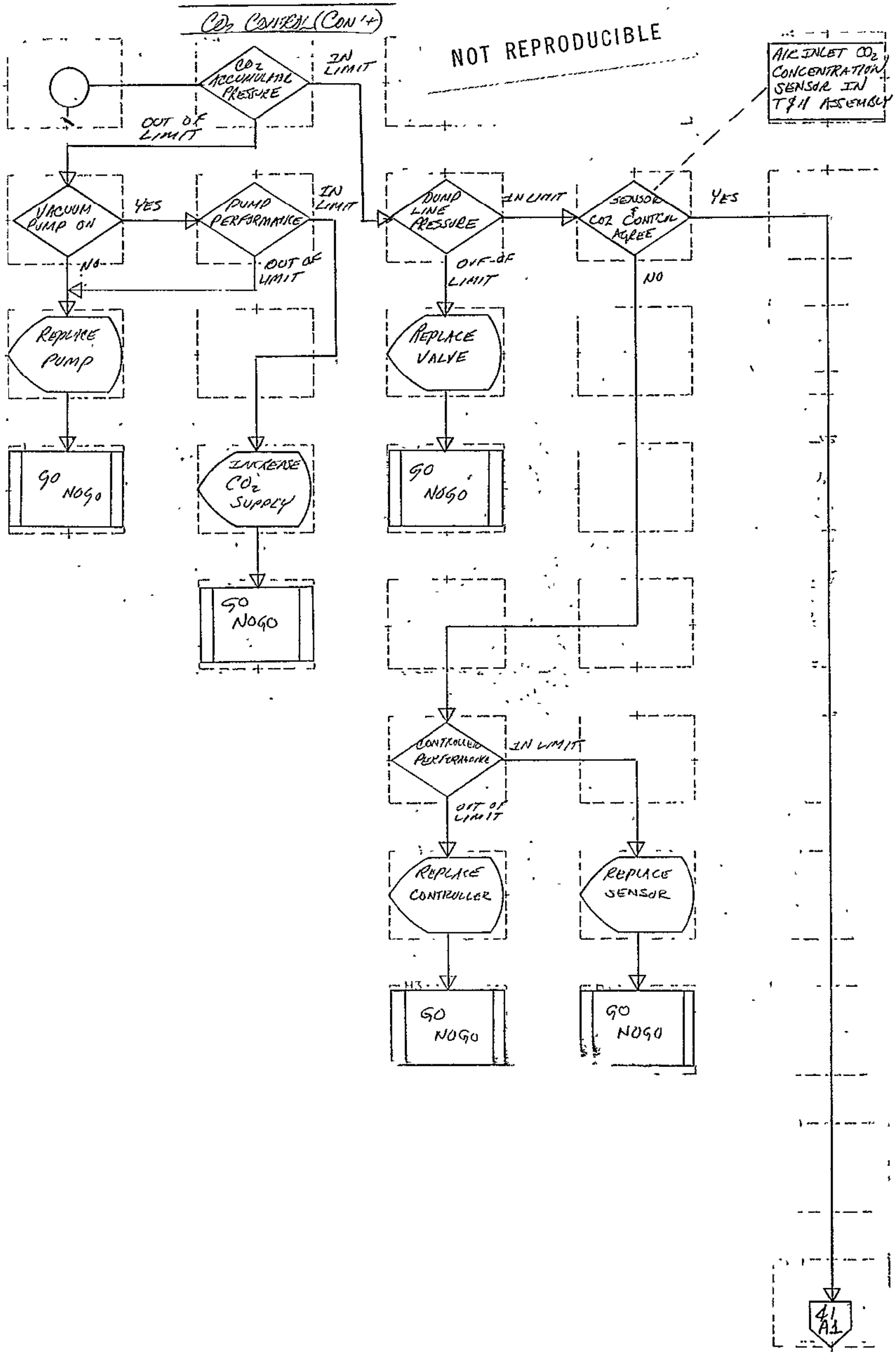


Figure 3-23. CO_2 Control Assembly (Sheet 3 of 7)

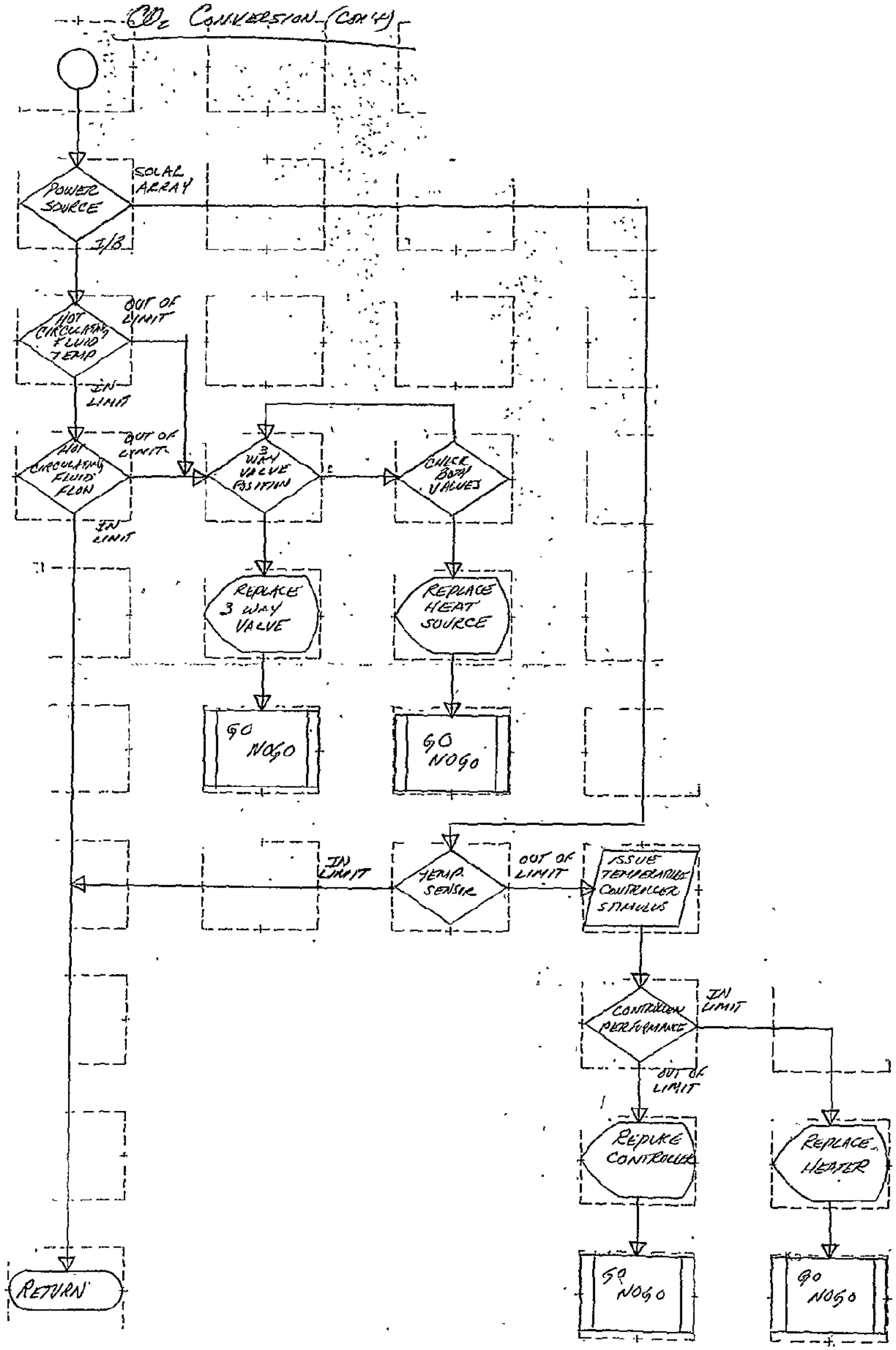


Figure 3-23. CO_2 Conversion Assembly (Sheet 4 of 7)

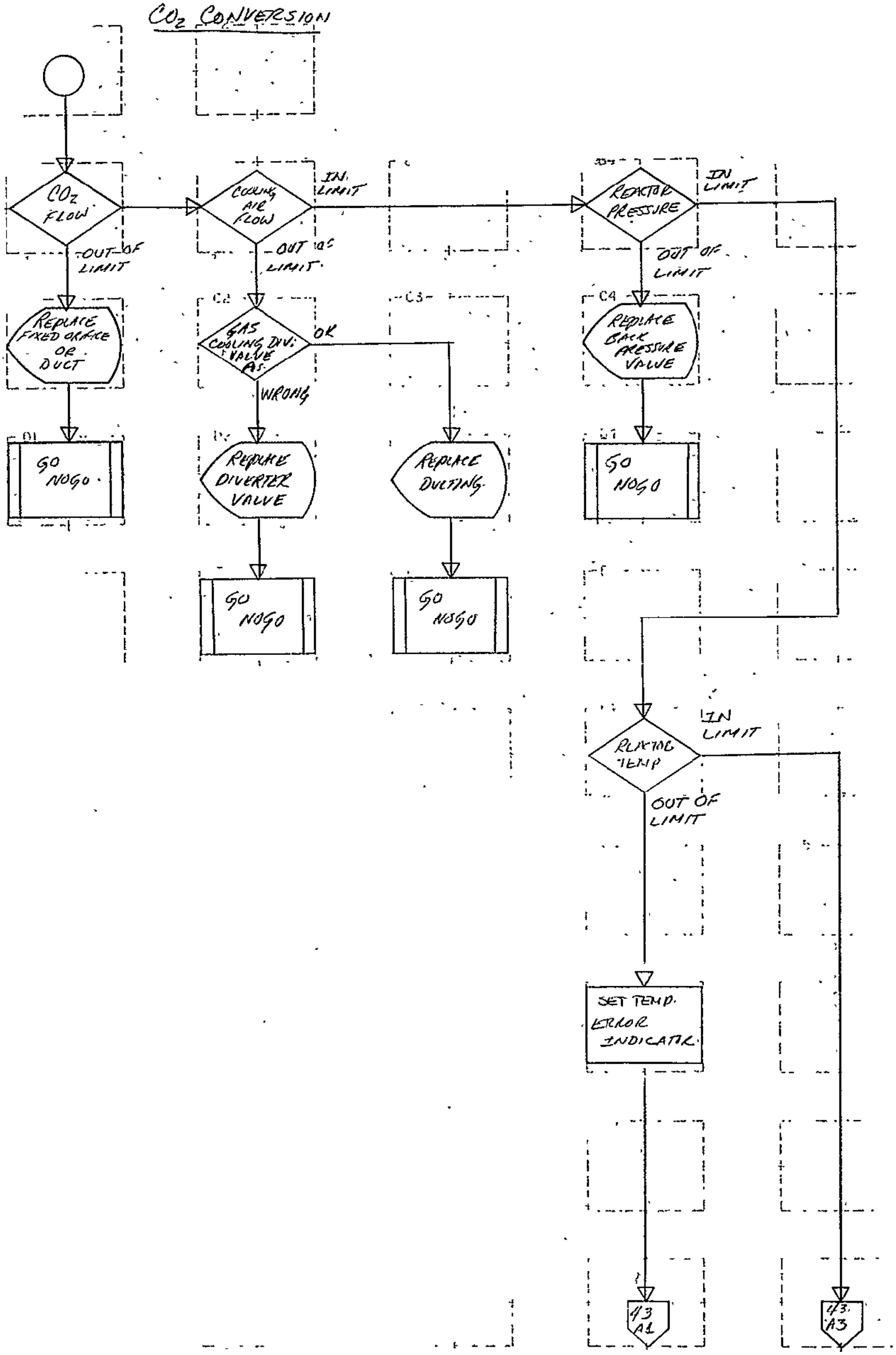


Figure 3-23. CO_2 Conversion (Sheet 5 of 7)

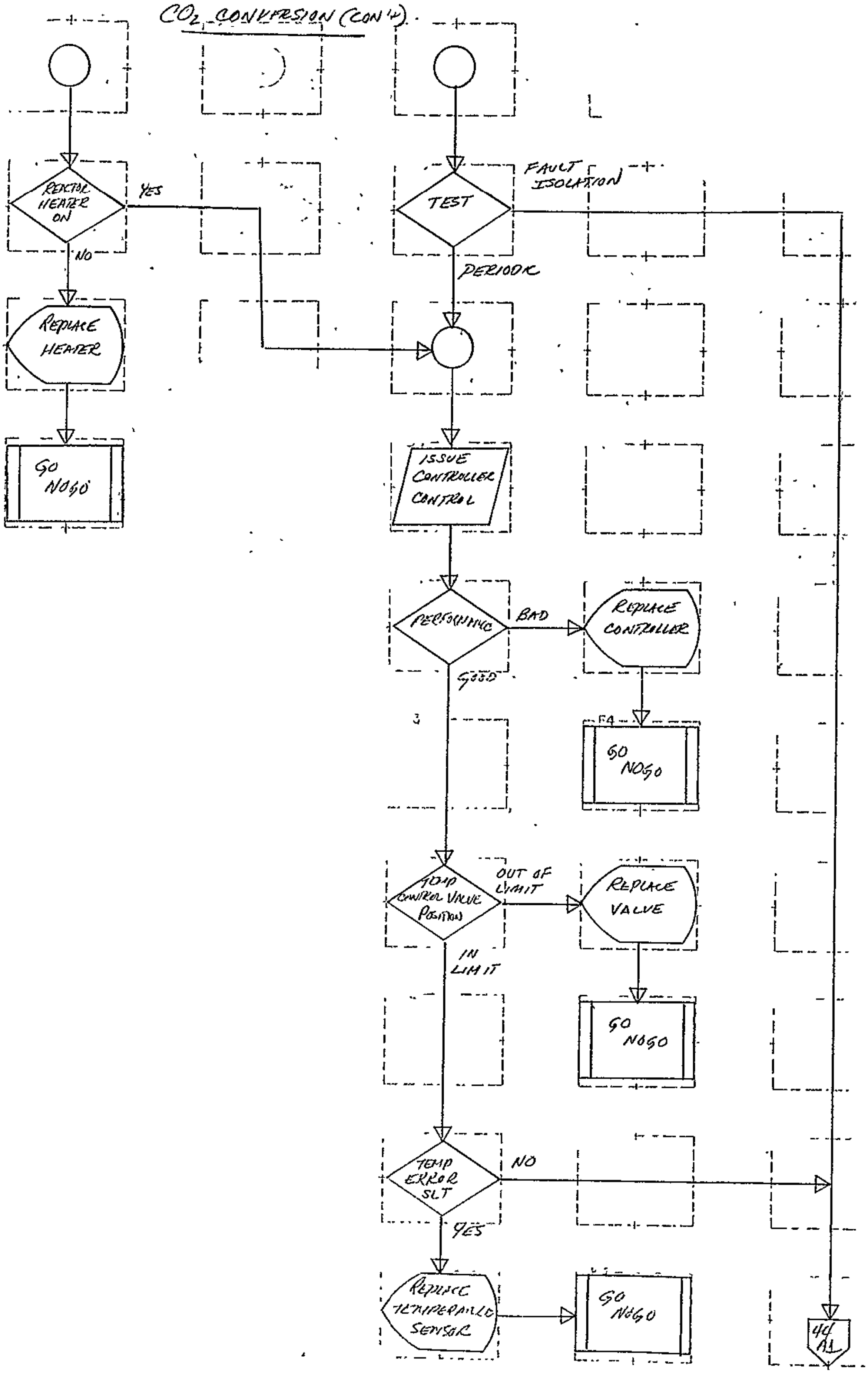


Figure 3-23. CO_2 Conversion (Sheet 6 of 7)

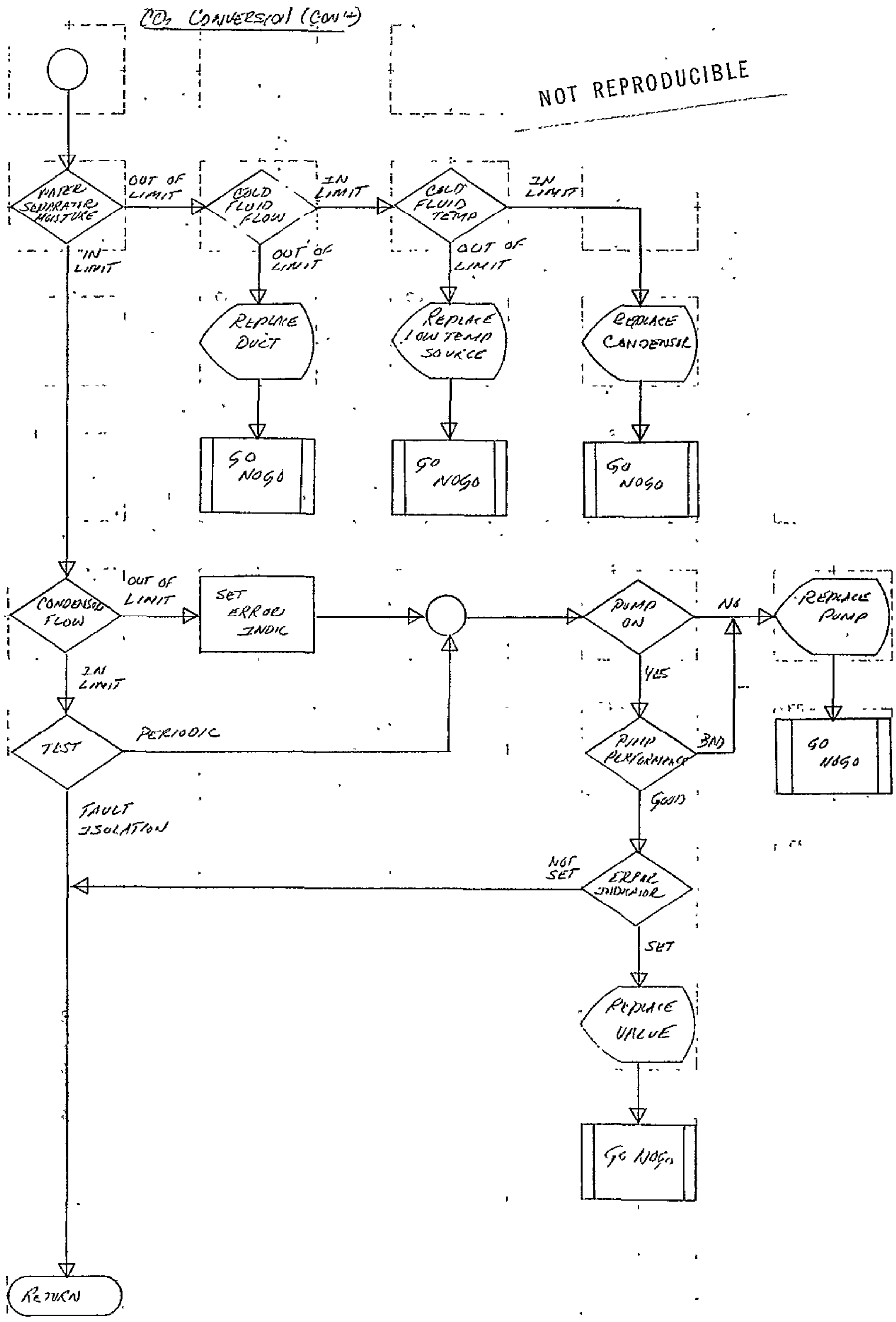


Figure 3-23. CO_2 Conversion (Sheet 7 of 7)

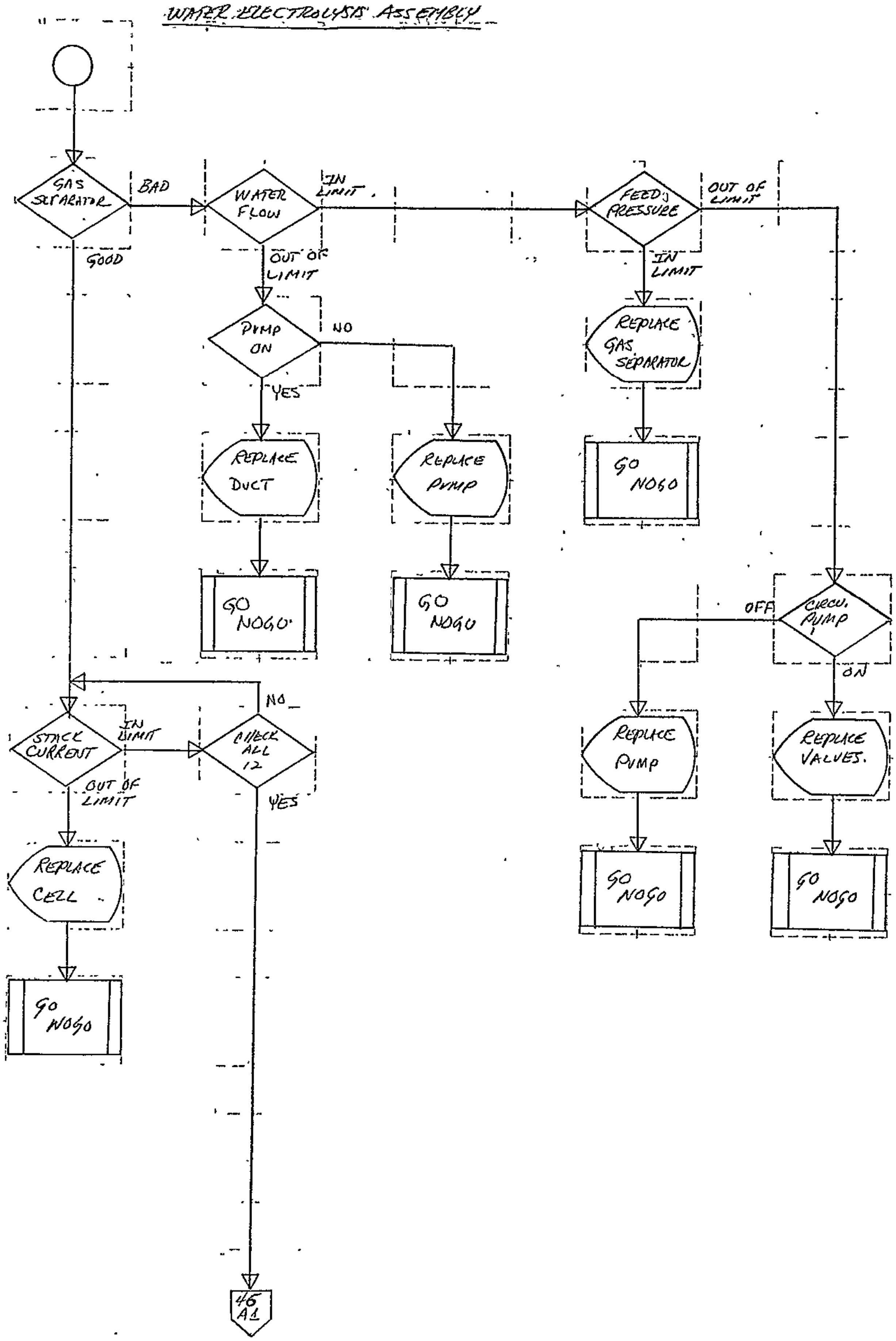


Figure 3-24. Water Electrolysis Assembly (Sheet 1 of 4)

WATER ELECTROLYSIS (CON'T)

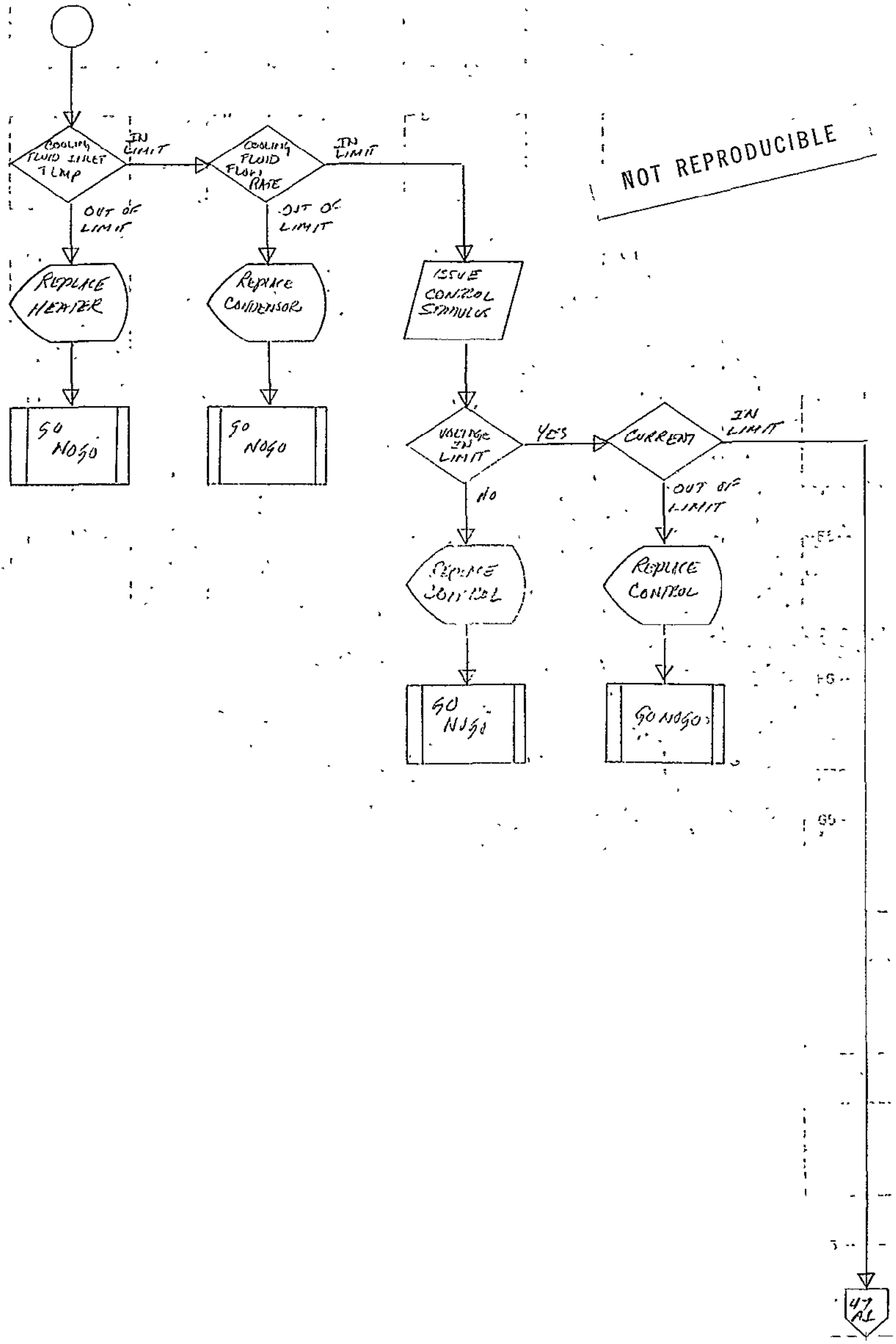


Figure 3-24. Water Electrolysis Assembly (Sheet 2 of 4)

WATER ELECTROLYSIS ASSEMBLY (CON'D)

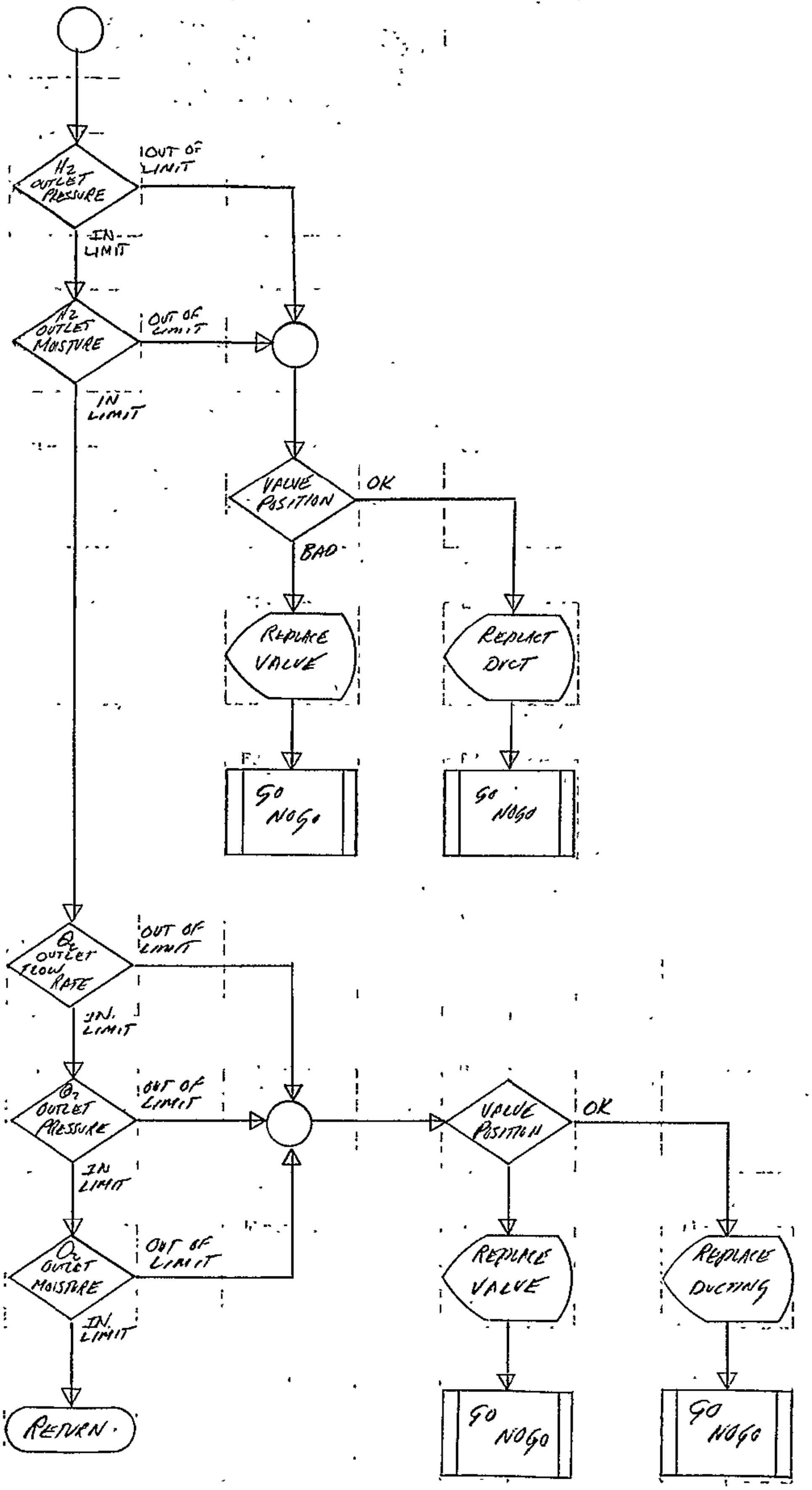


Figure 3-24. Water Electrolysis Assembly (Sheet 3 of 4)

VENTILATION ASSEMBLY

NOTE: THE DISPENSARY ATMOSPHERE PURIFICATION ASSEMBLY CONSISTS OF ONE FAN WHICH IS CHECKED IN THE SAME MANNER AS THE VENTILATION ASSEMBLY FANS

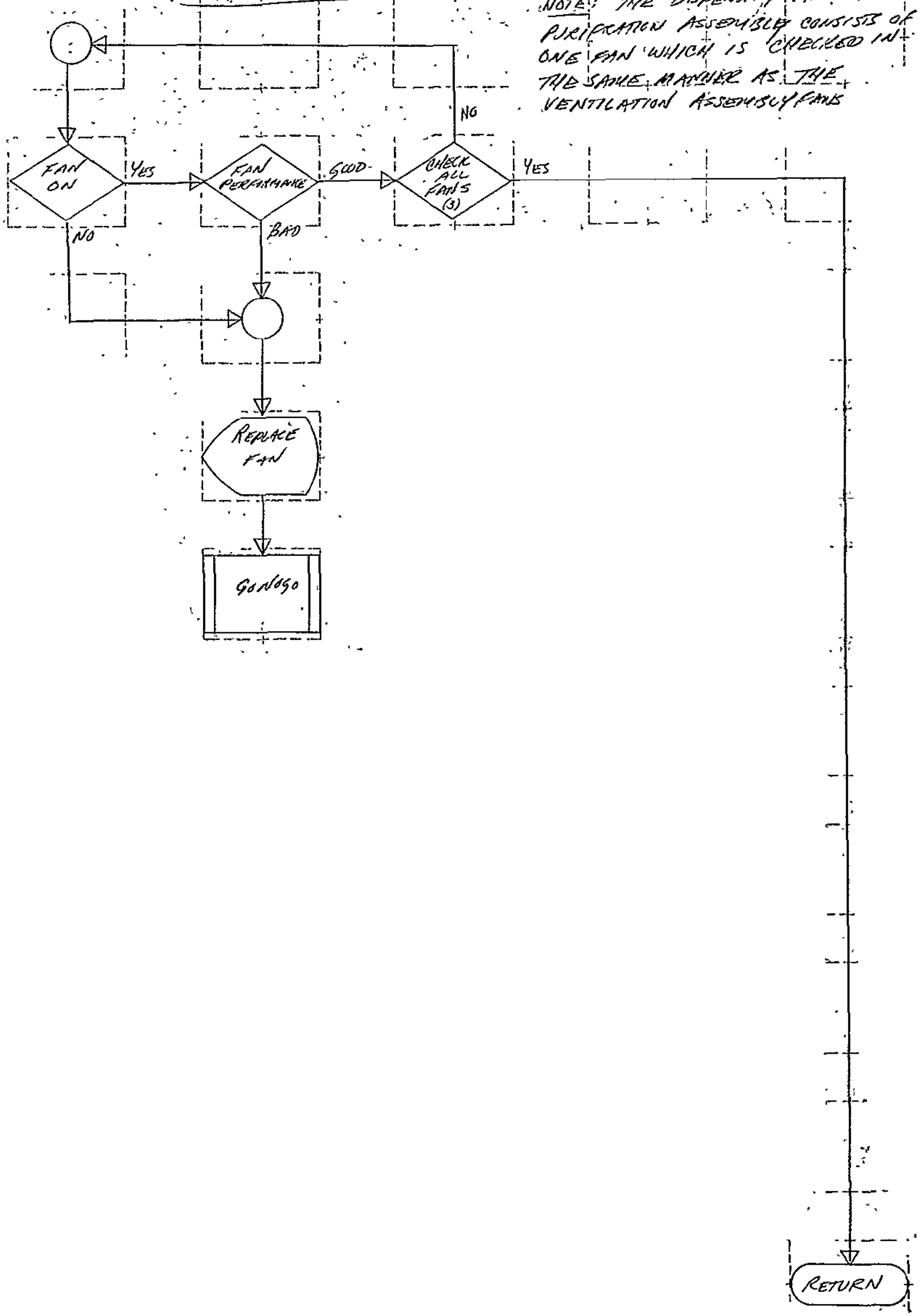


Figure 3-24. Water Electrolysis Assembly (Sheet 4 of 4)

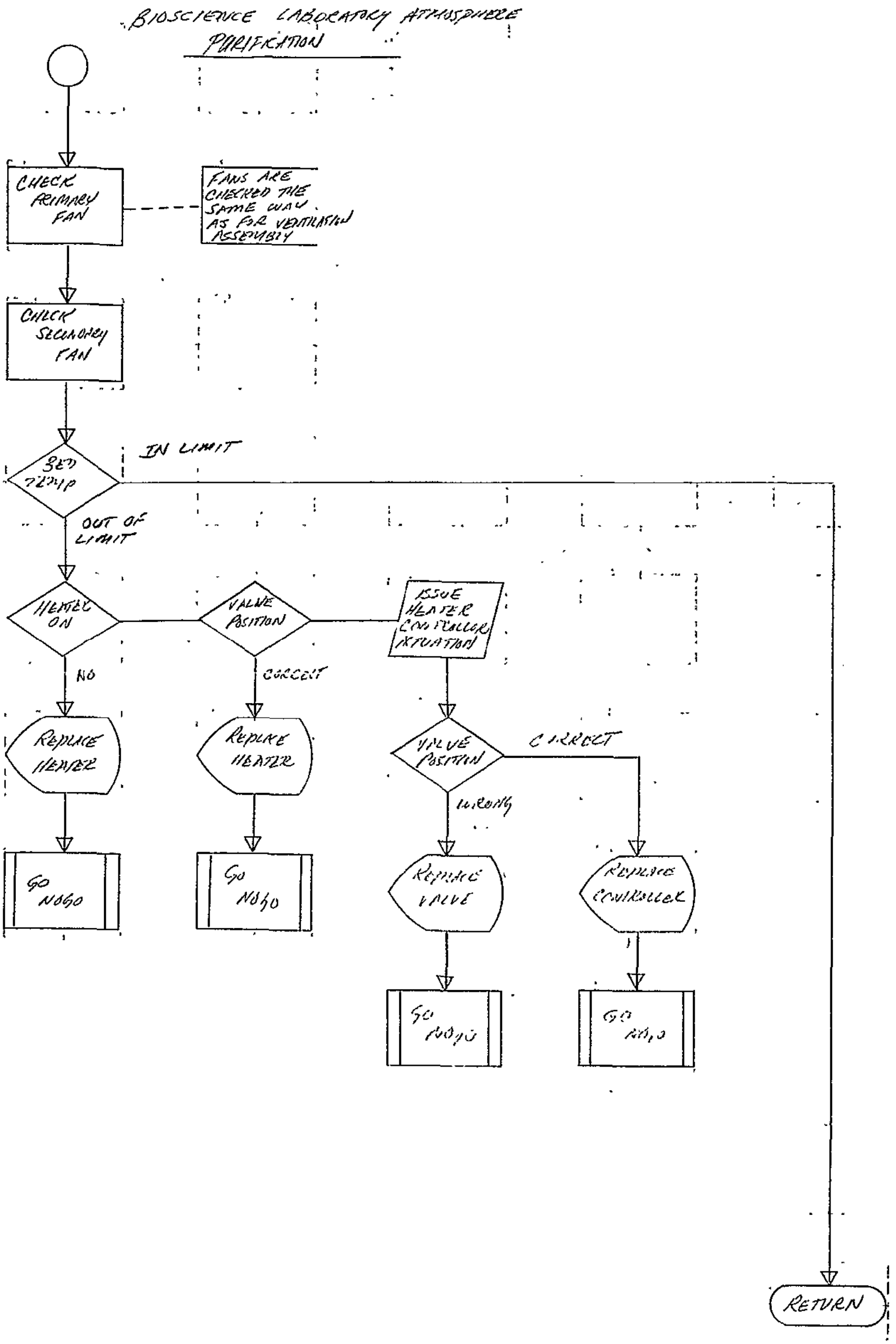


Figure 3-25. Bioscience Laboratory Atmosphere Purification

3.1.4 DATA BASE REQUIREMENTS

This program requires access to the data base for the following data:

- Association of symbolic test points with the actual device addresses
- Status and allocation of configurable components

3.1.5 HUMAN PERFORMANCE

The operator is required to communicate with the program to accomplish the desired function. Specifically, the operator must initiate the program using the system communication function. The program may be terminated prior to completion by using the TERM system communication function.

In addition, when errors are detected, the operator is provided with options to control program execution sequence. These options are referred to as GO-NOGO options and permit the operator to retest the LRU which failed, resume program execution, or to terminate program execution.

The operator must be capable of identifying the IVA and umbilical which are associated with the particular pressure suits.

3.2 INTERFACE REQUIREMENTS

This program must interface with the master executive, the OCS Executive, and the EC&LS Subsystem hardware. The EC&LS must also interface with the following subsystems.

- Power Subsystem
- Structure Subsystem
- Propulsion Subsystem
- Data Management Subsystem

3.2.1 INTERFACE DIAGRAMS

The interface between the EC&LS and other subsystems is depicted in Figure 3-26. Table 3-3 reflects the interface between the EC&LS Subsystem Checkout Program and the Executive Program.

Figures 3-27 through 3-32 depict the assembly interfaces within each assembly group.

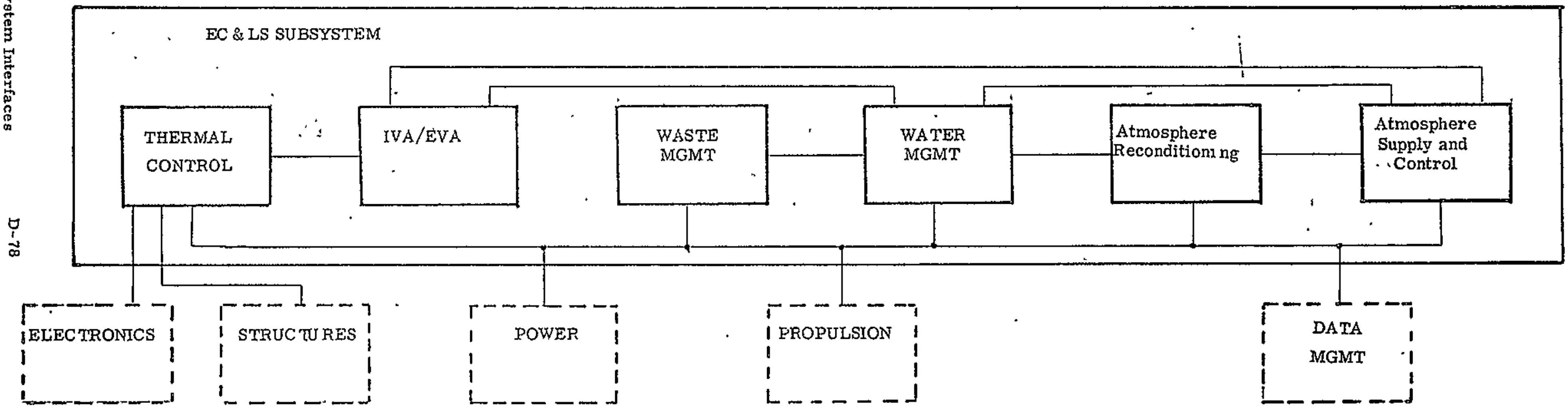


Figure 3-26. EC&LS Subsystem Interfaces

Table 3-3. EC&LS Checkout Program and Executive Program

		Task Specification	Processor Allocation	Task Termination	Memory Allocation	Peripheral Allocation	Data Base Allocation	Interrupt Monitoring	Output Processor	Input Processor	Special Processor	Data Retention	GO-NOGO Processor				
	Thermal Control								x		x		x				
	Atmosphere Supply								x		x		x				
	Atmosphere Recond.								x				x				
	Water Management								x				x				
	Waste Management								x		x		x				
	IVA/EVA								x	x	x		x				

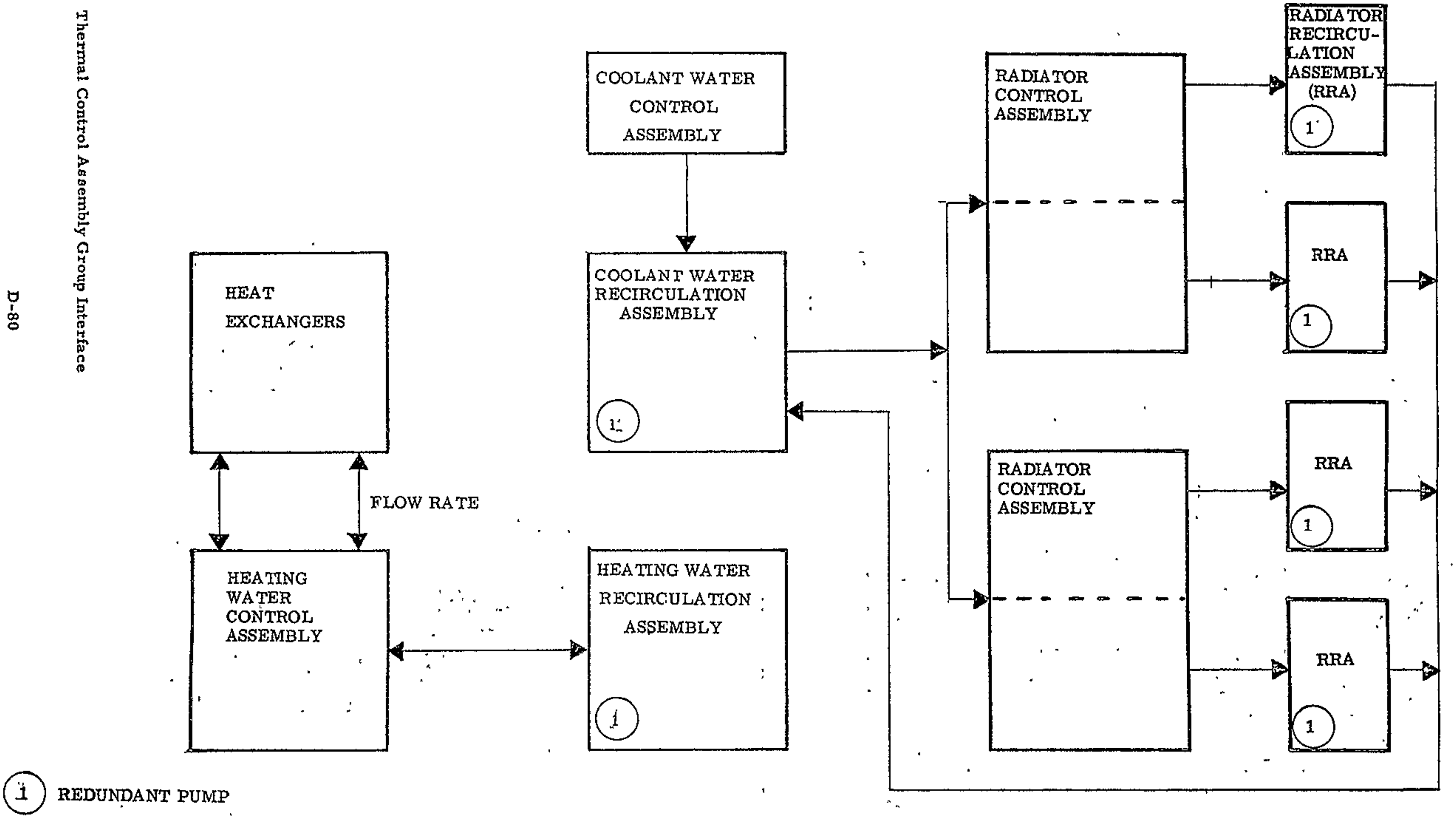


Figure 3-27. Thermal Control Assembly Group Interface

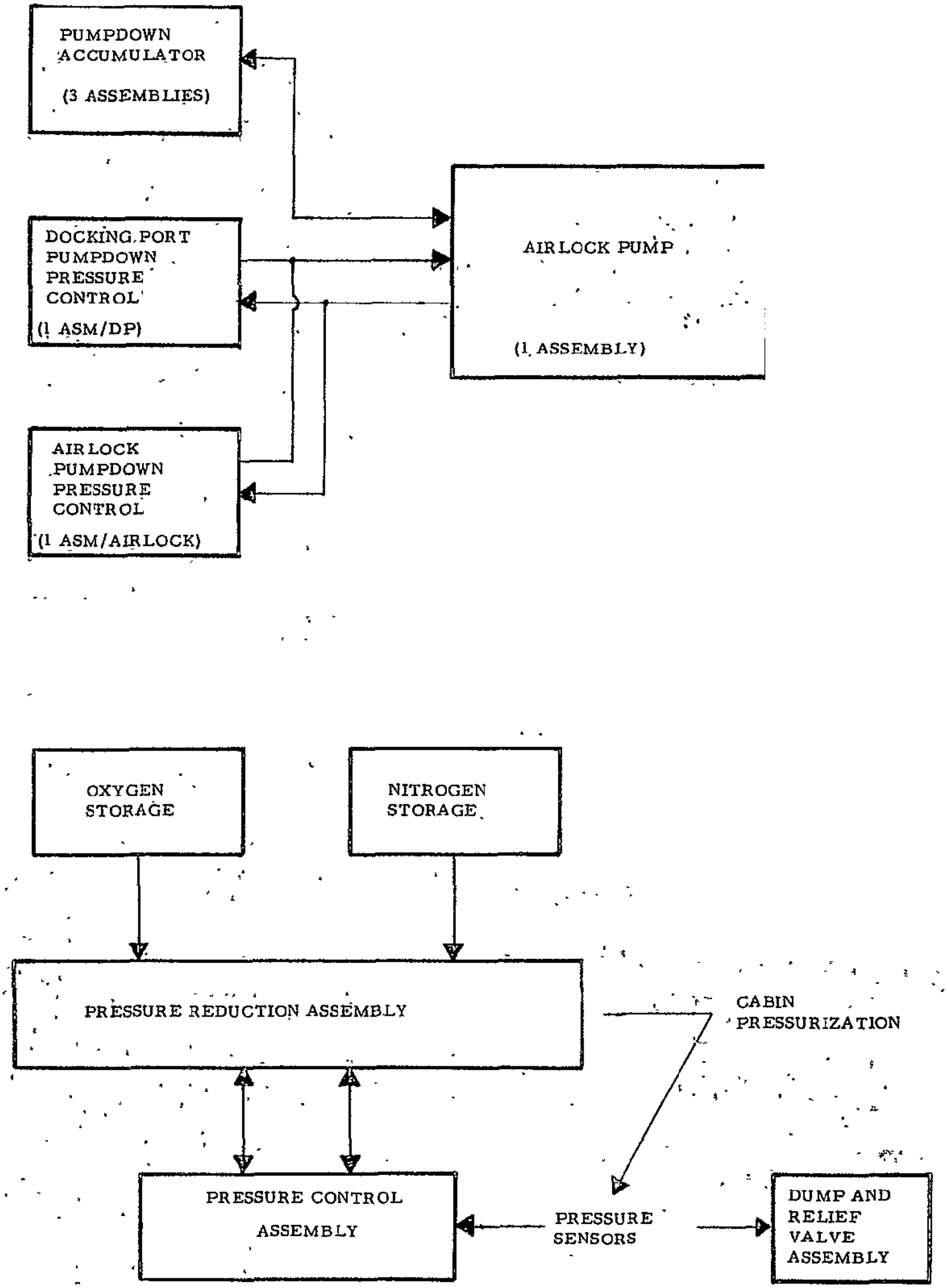


Figure 3-28. Atmosphere Supply and Control Assembly Group Interface

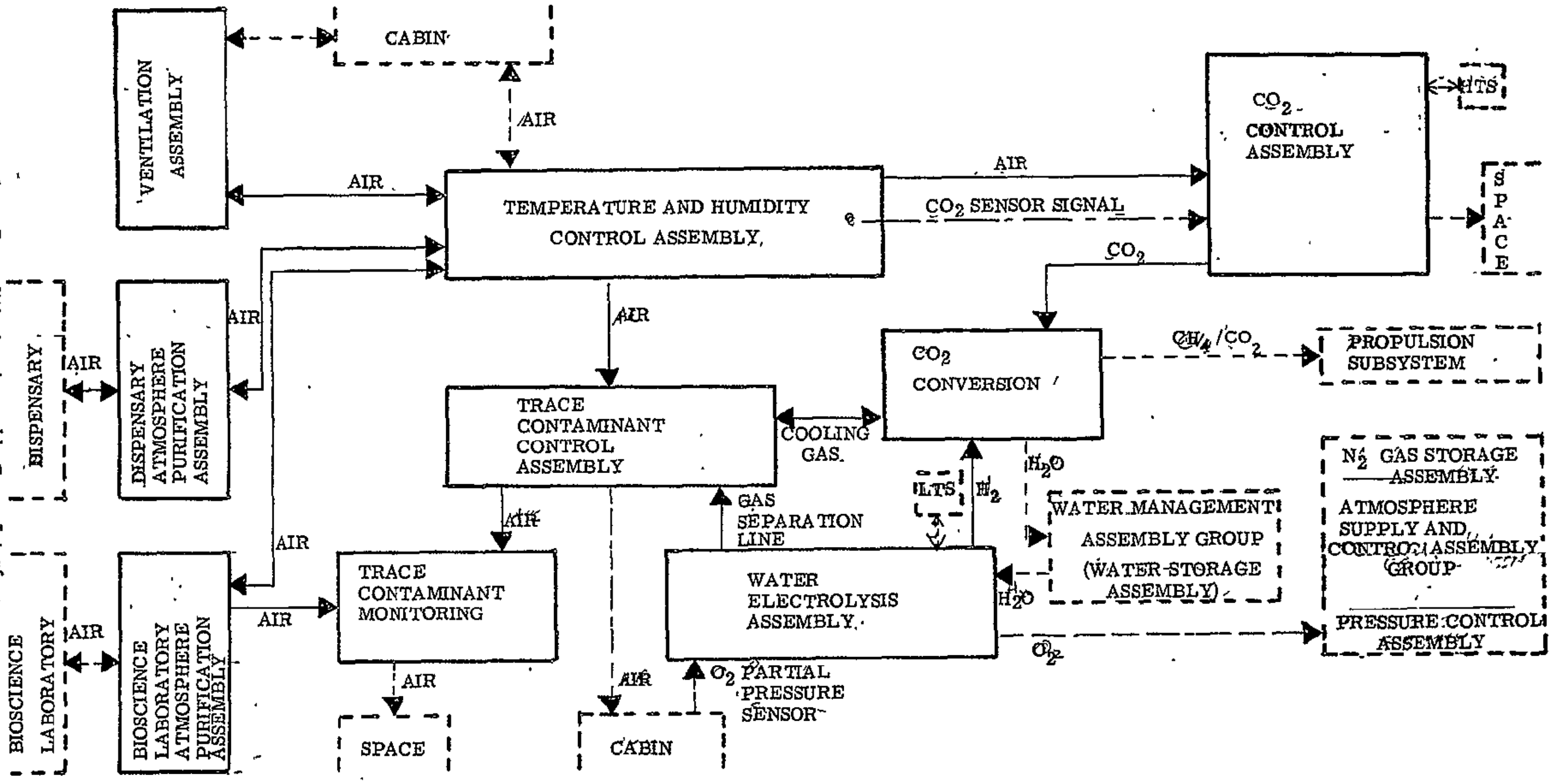


Figure 3-29. Atmosphere Reconditioning Assembly Group Interface

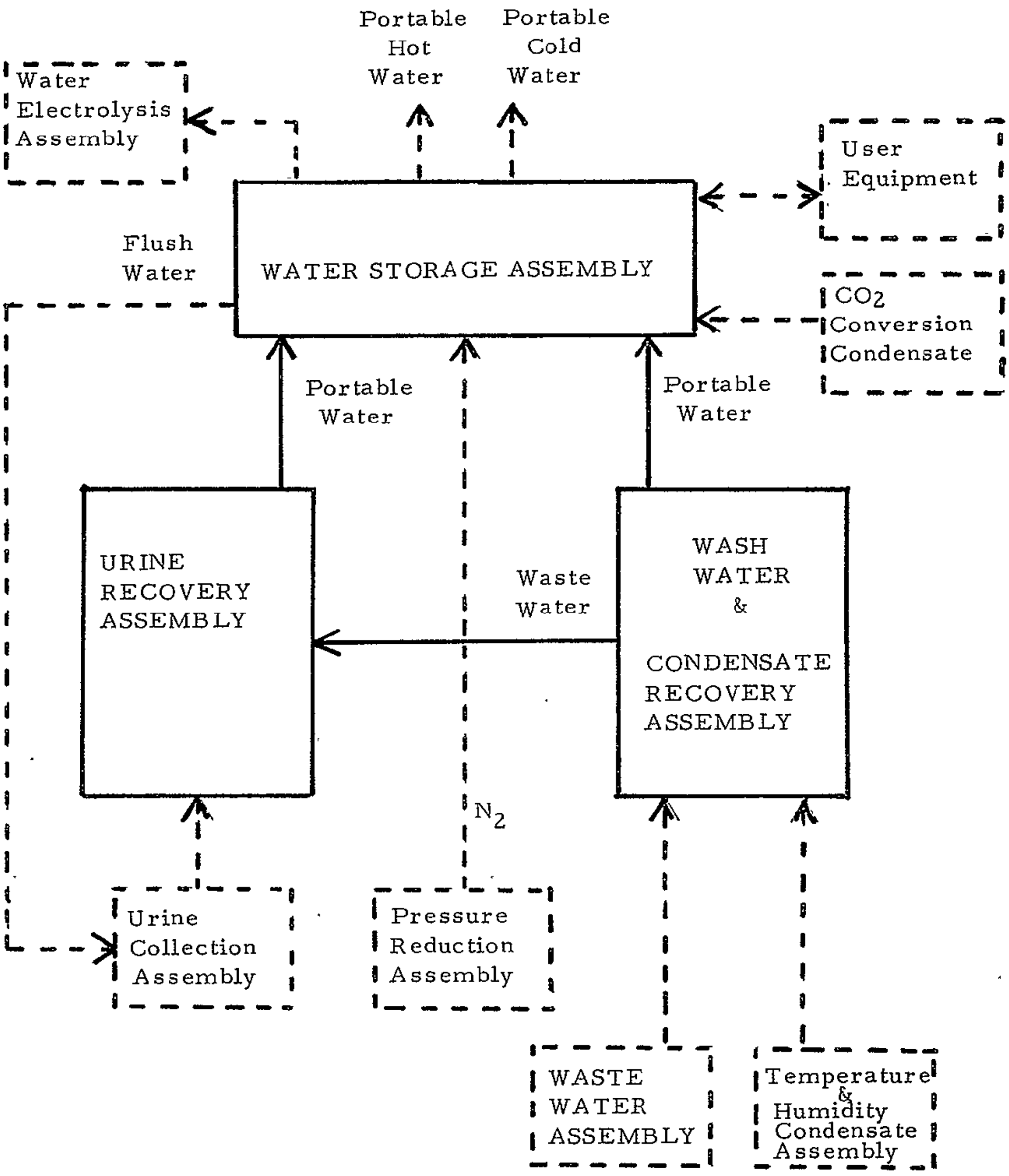


Figure 3-30. Water Management Assembly Group Interface Diagram

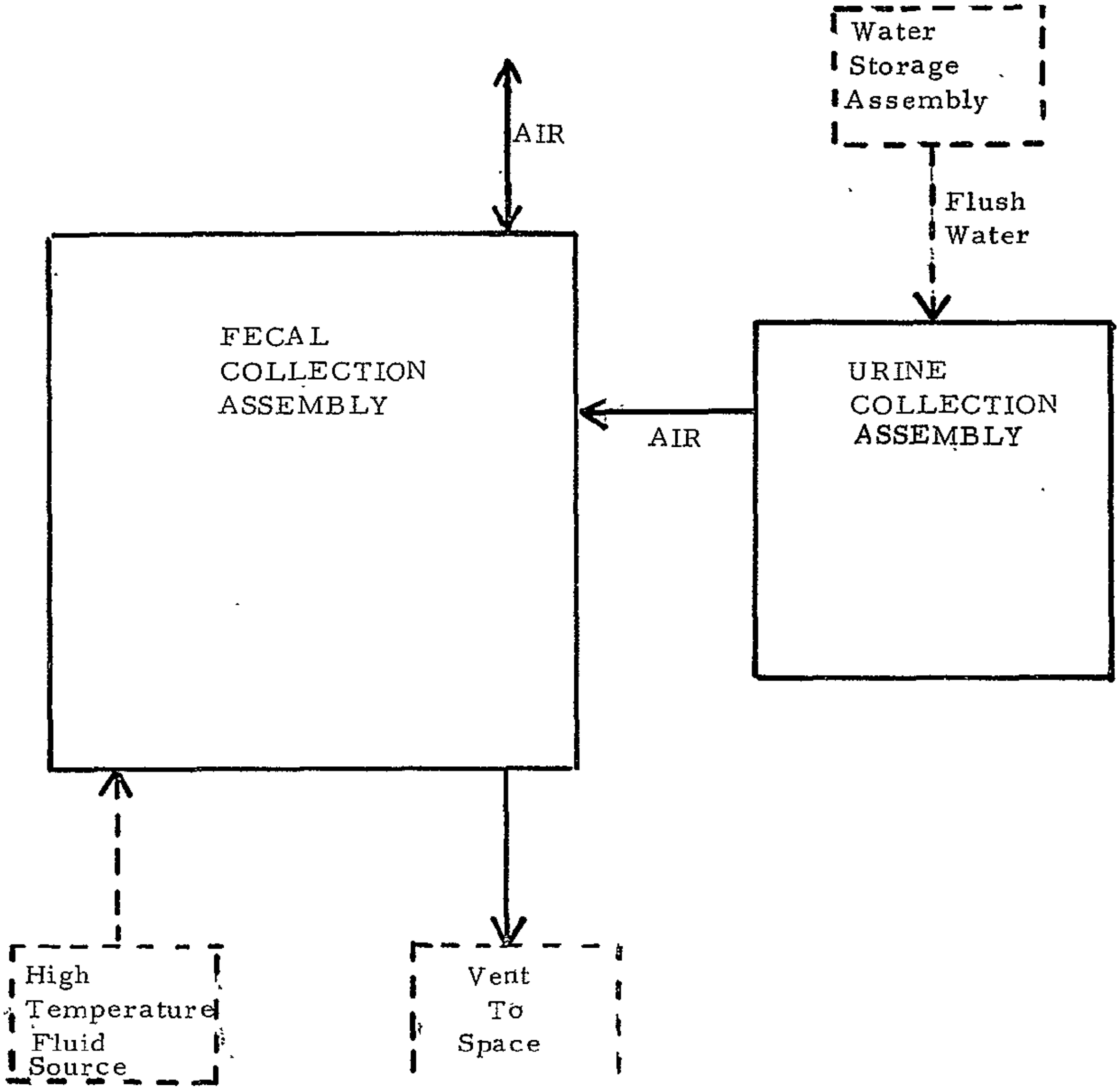


Figure 3-31. Waste Management Assembly Group Interface

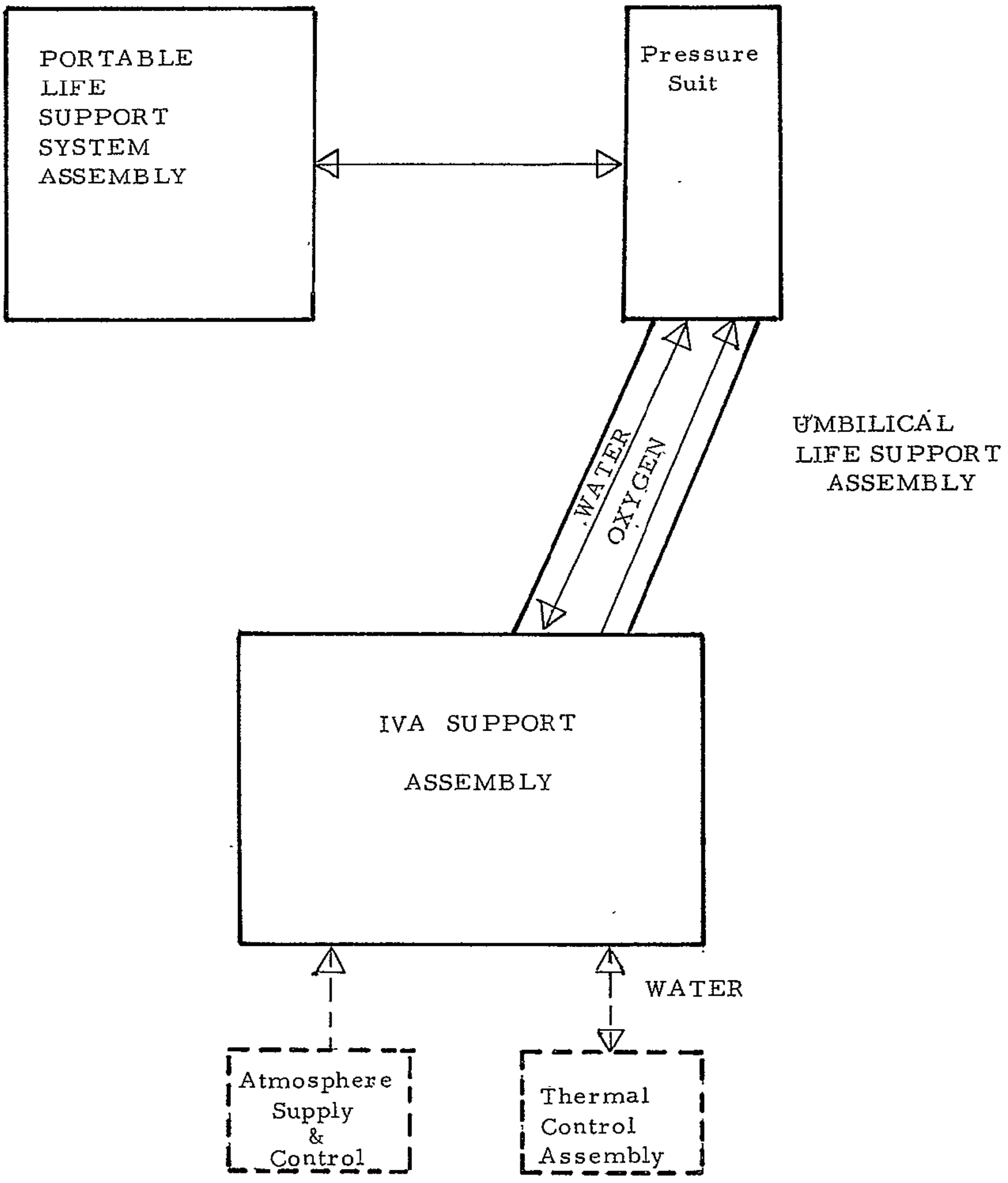


Figure 3-32. IVA/EVA Assembly Group Interfaces

3.2.2 DETAILED INTERFACE DEFINITION

This section describes the interfaces between the associated subsystems and program modules.

3.2.2.1 Subsystem Interfaces

Power Subsystem - The electrical power subsystem supplies power to all of the assemblies of the EC&LS subsystem requiring electrical power.

Structure Subsystem - The radiator tubes are integrated with frames of the meteroid shield. The EC&LS subsystem interfaces with the radiator at the inlet and outlet manifolds.

All EC&LS subsystem assemblies are mounted in the structure, and interconnecting plumbing and electrical wire harnesses are supported by the structure.

The EC&LS pumpdown system provides for pressurization and depressurization of the EVA airlock, the forward tunnel, and the equipment bay.

Propulsion Subsystem - The EC&LS subsystem supplies unreacted CO_2 from the CO_2 removal assembly to the propulsion subsystem. Excess water is also transferred to the propulsion system.

Data Management Subsystem - The data management and on-board checkout subsystems provide displays of the operational status of the EC&LS subsystem, provide spares and expendable inventory control for the EC&LS subsystem, and provide displays for fault isolation and repair.

3.2.2.2 Executive Program Interfaces

This section defines the interfaces which are required in the executive architecture to service this program.

3.2.2.2.1 Input Processor

Keyboard Inputs - This program requires the capability to access parameter information input by the operator from the keyboard to specify the selection of programmed options.

3.2.2.2.2 Output Processor

Measure Test Points - This program requires the capability to address specific test points through the Remote Data Acquisition Units (RDAU).

Issue Stimulus - This program requires the capability to issue stimuli to specified test points.

Display Control - This program requires the capability to present data at a display console to notify the operator of available options, or to present error messages.

3.2.2.2.3 Special Processor

Mode Control - This program requires the capability to have exclusive control over particular hardware components to accomplish required testing. This requires the capability to allocate a component to the program, and to permit the program to indicate a failure status on the device when an error is detected.

3.2.2.2.4 Miscellaneous

The IVA/EVA Assembly Group requires that the executive automatically detect when a pressure suit is attached to the system.

The Atmospheric Supply and Control Assembly Group requires access to the operational program which contains the algorithm used to determine the command issued to the cabin pressure control mechanism.

4.0 SUBSYSTEM SUMMARY

This section summarizes some of the information which has resulted from the analysis of the EC&LS subsystem.

4.1 LANGUAGE ANALYSIS

4.1.1 ELEMENT ANALYSIS

The analysis of this subsystem indicates the need to provide additional flexibility in acquiring data from test points. Consequently, the measure element has been modified to address multiple test points. This is accomplished by providing the capability to index through a table containing the test point addresses.

4.1.2 SIZING ESTIMATES

The sizing estimates developed for this subsystem are provided on a basis of element utilization in Table 4-1. Table 4-2 provides estimates on a basis of processor and memory utilization.

4.2 EXECUTIVE ANALYSIS

The EC&LS Subsystem requires the capability to address test points which are in a redundant capacity.

This subsystem requires direct interface (as opposed to data base interfaces) with the operational program modules.

Trend analysis requirements indicate the need for the executive to perform a data collection and program timing function to assure the timely accomplishment of trend analysis.

Table 4-1. Element Estimates Program Title: EC&LS Subsystem Checkout Program

ELEMENT NAME	ELEMENT UTILIZATION	PERIODIC TEST ELEMENT EXECUTION	PERIODIC TEST MEMORY REQUIREMENTS
ALLOCATE	160	232	122
CALL	454	133	157
DELAY	170	170	160
DISPLAY	334	36	133
GOTO	122	132	60
IF	483	845	277
MEASURE	374	597	146
READ	2	1	1
SEARCH	1	1	1
SOLVE	437	642	319
STATUS	48	216	42
STIMULI	97	308	60
TOTAL	2682	3313	1478

Table 4-2. EC&LS Subsystem

Periodic Test

Program text	15K	
Working buffer	1K	
Main Storage (words)		16K
Display	4.5K	
Program code	28.5K	
Auxiliary Storage (words)		33K
CPU time	0.02 minutes	
I/O time	4.58 minutes	
Time in Main Storage		4.60 minutes

Trend Analysis

Main Storage	150 words
Auxiliary Storage	1K words

	Per Execution	Per Day
CPU time	.00005	1.50
I/O time	.0026	78.13
Time in Machine (mins) (Minutes)	.00265	79.63

APPENDIX E

COMMUNICATIONS SUBSYSTEM CHECKOUT PROGRAM DESIGN

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

1.0 SCOPE

The purpose of this specification is to provide performance and design requirements for the Communications Subsystem checkout programs to be used for the Space Station during pre-flight testing and while in orbit. The detail to which these requirements are developed is a function of the engineering design details available in the Phase B Space Station studies.

The checkout functions identified are those for trend analysis, operation interface, fault isolation, periodic check, and calibration. Fault detection is feasible with human interface and the RDAU preprocessor; therefore, it is not identified as a checkout program function. However, management of the RDAU fault detection facilities is under control of the checkout executive program.

2.0 APPLICABLE DOCUMENTS

- "Communications", Space Station, MSFC-DRL-160, Line Item 13, Preliminary Systems Design Data, Volume I, Space Station Preliminary Design, Book 2, Contract NAS8-25140, July 1970.
- "Performance/Design and Product Configuration Requirements for Space Station Module, Space Station Test Articles, and Nose Cone System", Space Station, MSFC-DRL-160, Line Item 19, Preliminary Performance Specifications, Volume I, Contract NAS8-25140, July 1970.
- "Baseline Subsystem Descriptions", Interim Report RA-1, Contract NAS9-11189, August 1970.
- "Subsystem Checkout Strategy", Interim Report RA-5, Contract NAS9-11189, September 1970 .
- "Subsystem Test Definition and Measurement/Stimulus List", Interim Report RA-8, Contract NAS9-11189, November 1970.
- "Structure Subsystem Checkout Program", Interim Report OCS-2-2021, Contract NAS9-11189, November 1970.
- "GNC Subsystem Checkout Program", Interim Report OCS-2-2022, Contract NAS9-11189, November 1970.
- "EPS Subsystem Checkout Program", Interim Report OCS-2-2023, Contract NAS9-11189, December 1970.
- "Environmental Control and Life Support Subsystem Checkout Program", Interim Report OCS-2-2024, Contract NAS9-11189, December 1970.

3.0 REQUIREMENTS

In this section, the technical aspects of the Communication Subsystem checkout programs are described. Modularization concepts are utilized extensively in identifying the computer program components. Under the assumption that just as the Communications Subsystem itself is modularized, in the final implementation, the computer programs will be also. The functions, structures, processing, input, output, and data base requirements of the computer program components are discussed below.

3.1 PERFORMANCE

The Communications Subsystem requires trend analysis to assess assembly performance based on past history. Operation interface is used to enable and disable the RDAU fault detection capability, as well as to initiate certain forms of trend analysis. Fault isolation is required to identify the failed LRU or LRU group, and to assess assembly capability prior to use. Periodic check functions augment and cross-check the assessments made by fault isolation. Calibration verifies the operational characteristics of certain assemblies after LRU replacement.

3.1.1 SYSTEM REQUIREMENTS

Continuous monitoring is performed only for the non-critical measurements associated with RF power output levels and AGC output levels. These measurements provide the only means of automatic fault detection by the RDAU preprocessor technique. Fault detection relies extensively on human evaluation of operating results, in addition to RDAU limit checking.

Stimuli are required prior to making certain trend analysis measurements, and must be applied when the associated assembly is not providing an operational function.

Fault isolation is performed on a group of LRU's which are associated with the loss of a particular function.

After LRU replacement has been performed in the RF transmission system, which consists of the signal path from the power amplifier outputs through the antennas, the RF power level at the antennas is used for calibration purposes.

3.1.2 OPERATIONAL REQUIREMENTS

The communications checkout programs required for trend analysis, operation interface, fault isolation, periodic check, and calibration are discussed in the following sections.

3.1.2.1 Trend Analysis Function

The trend analysis checkout programs are used to periodically sample selected RF and AGC measurements for the purpose of detecting degraded performance prior to the time that the fault might otherwise be detected.

Input consists of single test point measurements and the intermediate results obtained after the last sample. In the case of certain AGC test points, output consists of stimuli to supply a reference for the measurement. Other output consists of a checkpoint just prior to termination, and parameters passed to other checkout functions such as fault isolation.

Information processing consists of averaging a series of measurements, comparing the most recent measurement to the average, and storing averages for later retrieval upon request and for use in the next cycle. A trend analysis function for RF power test points is initiated by the executive at a specified rate. For AGC test points, the trend analysis function is initiated by the shutdown program, since trending must be done during periods of inactivity because of the stimulus required. Executive checkpoint services are used to store results after each cycle. Restart services of the executive are employed to begin each successive cycle.

The trend analysis methods required for the Communications Subsystem are as follows:

1. Provide stimulus and make measurement, comparing with the average of 30 previous measurements. Retain, for retrieval on an as-required basis, 30 averages. The flow chart for this method is shown in Figure 3-6.

2. Compare measurement with the average of 48 previous measurements. Retain, for retrieval on an as-requested basis, the daily average for three months. A stimulus may or may not be required prior to the measurement. The flow chart for this method is shown in Figure 3-7.

In Table 3-1, the trend analysis requirements of the Communications Subsystem are summarized. Auxiliary storage requirements for data storage, assuming three checkpoints are available at any time, is approximately 5.7K words.

Table 3-1. Communications Trend Analysis

	Number of Applications	Trend Method	Stimulus Required	Measurement Rate	Averages and Measurements Retained Per Test Point
S-Band Video Receiver AGC	10	1	Yes	1/day	61
S-Band Data Receiver AGC	10	1	Yes	1/day	61
S-Band PM Receiver AGC	2	2	Yes	1/hour	139
Ku-Band FM Exciter RF Power	2	2	No	1/hour	139
S-Band FM Receiver AGC	2	2	Yes	1/hour	139
Ku-Band PA RF Power	5	2	No	1/hour	139
Ku-Band PM Exciter RF Power	5	2	No	1/hour	139
S-Band PM Transponder AGC	2	1	Yes	1/day	61
S-Band PM Transponder RF Power	2	2	No	1/hour	139
S-Band PA RF Power	2	2	No	1/hour	139
S-Band FM Exciter RF Power	2	2	No	1/hour	139
VHF Voice/Ranging RF Power	2	2	No	1/hour	139
VHF Voice/Ranging AGC	2	1	Yes	1/day	61
VHF Data Receiver AGC	2	1	Yes	1/day	61
VHF FM Transmitter RF Power	3	2	No	1/hour	139
VHF FM Receiver AGC	3	1	Yes	1/day	61
VHF Data Transmitter RF Power	2	1	Yes	1/day	61

3.1.2.2 Operation Interface Function

While fault detection is done by means other than software, there is a checkout program involvement in activating and deactivating the fault detection mechanism provided by the RDAU preprocessor. This is the operation interface function of communication checkout.

Input consists of the symbolic identity of the assembly being activated or deactivated, such as the Ku-band FM transmitter. Output consists of a memory update for the RDAUs which are connected to the status monitoring test points of the assembly.

Information processing consists of identifying the RDAUs involved, and either masking or unmasking the RDAU channels which measure the status test points. While the assembly is on standby, the channel is masked to prevent a limit check interrupt which might give a false error indication. After the assembly has been activated, the channel is unmasked to allow limit checking to proceed. Figure 3-8 and Figure 3-9 show typical flow charts for "start-up" and "shut-down", respectively.

For assemblies requiring procedures beyond simply applying or disconnecting electrical power, special modules are utilized. For example, these modules supply logic necessary to complete the warmup cycle of a power amplifier or to assure signal acquisition. Other assemblies require an AGC stimulus in order that the trend measurement can be made. Since this stimulus must be made while the assembly is not in normal operational use, the shutdown routine will invoke the trend analysis program, provided sufficient time has elapsed since the last measurement.

3.1.2.3 Fault Isolation Function

Upon detection of a fault by the crew, ground, or RDAU's, the fault isolation functions are invoked to identify the failed assembly or LRU. Although automatic isolation to the failed LRU is attempted, in cases such as the high gain antenna group, it is only feasible to isolate to a larger assembly because of the instrumentation problems involved. In some cases, additional test points would increase the system insertion losses; therefore increasing the power required to compensate for these losses, compared with that required for operational purposes.

No unusual input or output requirements for fault isolation beyond those required for GNC, Structures, EP and EC&LS Subsystems have been identified.

Information processing for each fault isolation function follows the same form identified in the referenced Space Station OCS Interim Reports. In most cases, communications fault isolation requires exclusive use of the LRU's associated with the function which is suspected of containing a fault.

The fault isolation modules for the Communications Subsystem relate to functions such as video reception, voice transmission, etc. These routines are then applied to the hardware assembly groups of S-band low gain, Ku-band low gain, high gain, and VHF low gain shown in Figure 3-1. Further breakdown of each group, detailing LRU interfaces is shown in Figure 3-2 through Figure 3-5. In each block of the LRU interface diagrams, the LRU quantity is indicated in parenthesis.

The LRU's associated with the VHF low gain group are shown in Figure 3-2. The fault isolation functions are identified below, with the LRU coverage for each function indicated in parentheses:

- VHF RF Transmission (VHF low-gain antennas, VHF diplexers, VHF multiplexer/power divider/switches)
- VHF Voice (VHF voice XMTR/RCVR's, signal modems)
- VHF PM Data (VHF PM Data XMTR/RCVR's)
- VHF Voice/Ranging (VHF FM voice/ranging XMTR/RCVR's, voice transceiver modems, ranging modems)

The LRU's associated with the Ku-band low-gain group are shown in Figure 3. The fault isolation functions are identified below, with the LRU coverage for each function indicated in parentheses:

- Ku-Band RF Transmission (Ku-band low-gain antennas, Ku-Band preamp/mixer/diplexer/switches, S-band multiplexers and circulators)
- S-Band Data (S-band data receivers)
- S-Band Video (S-band video receivers, video reciver modems)

The LRU's associated with the S-band low-gain group are shown in Figure 4. The fault isolation functions are identified below, with the LRU coverage for each function indicated in parentheses:

- S-Band RF Transmission (S-band low-gain antennas, S-band triplexers/switches)
- S-Band Transponder (S-band PM Transponders, transponder modems)
- S-Band FM Transmitter (S-band power amplifiers, S-band FM exciters, transmitter modems)

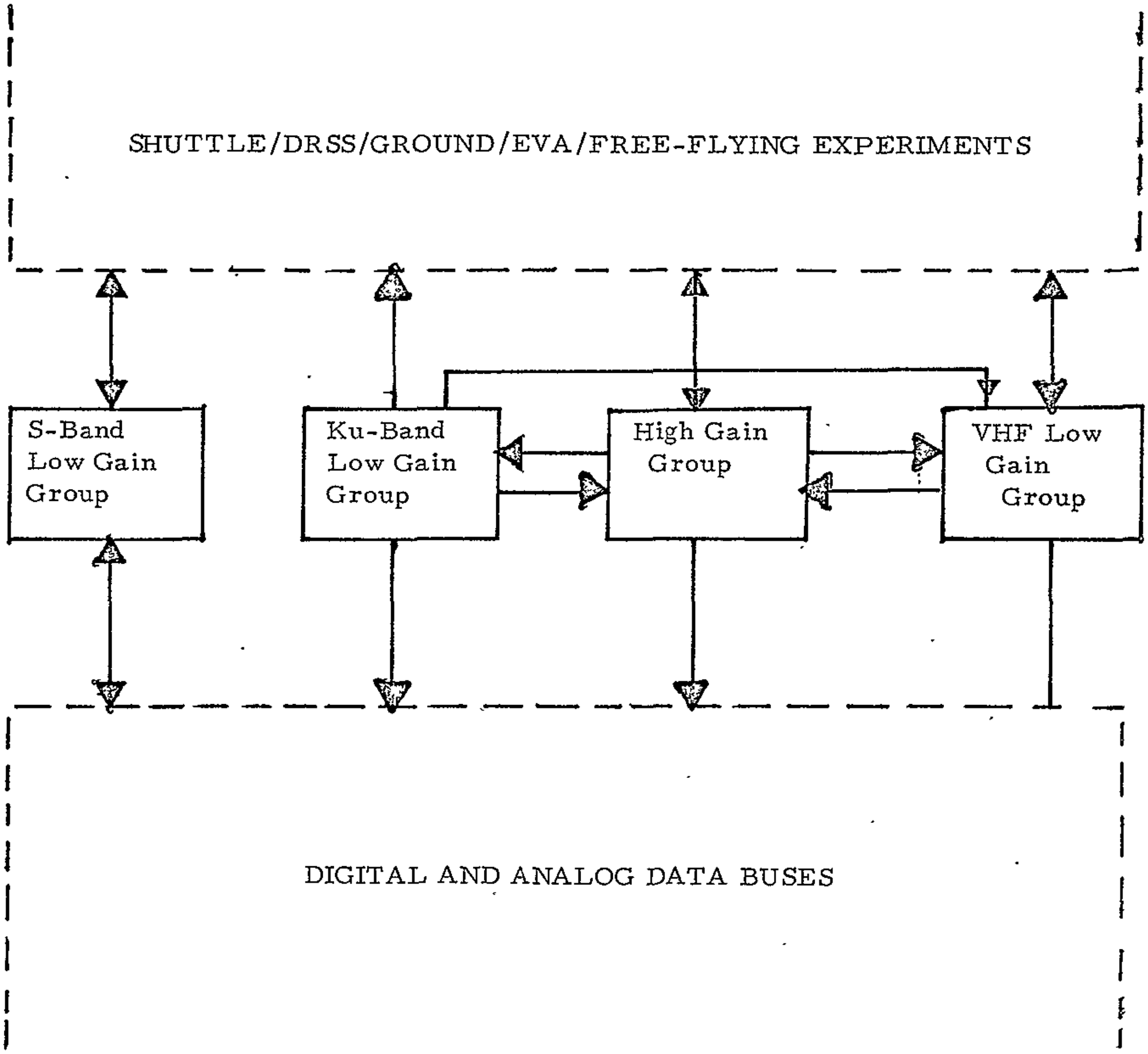


Figure 3-1. Communications Assembly Group Interfaces

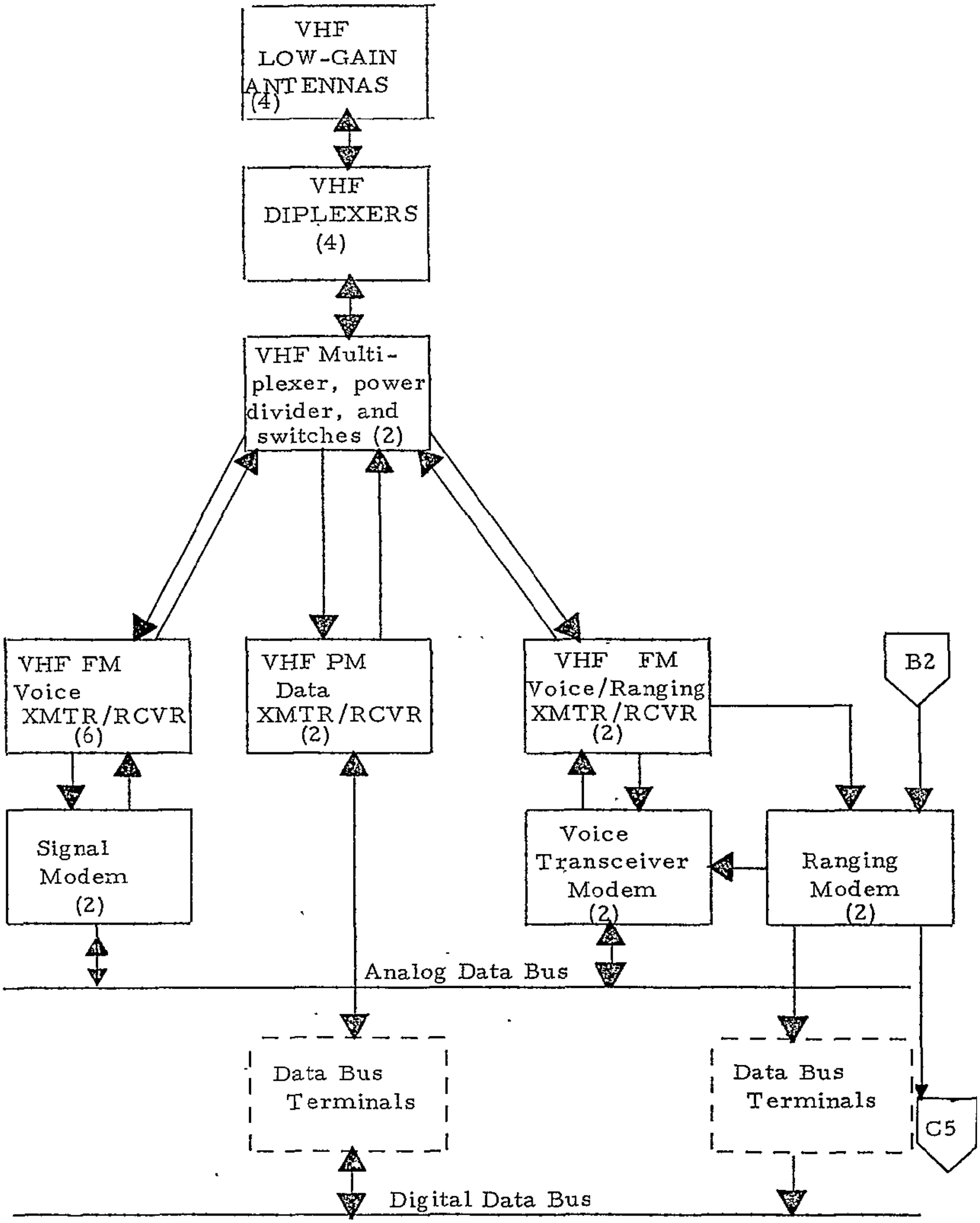


Figure 3-2. LRU Interface Diagram VHF Low-Gain Group

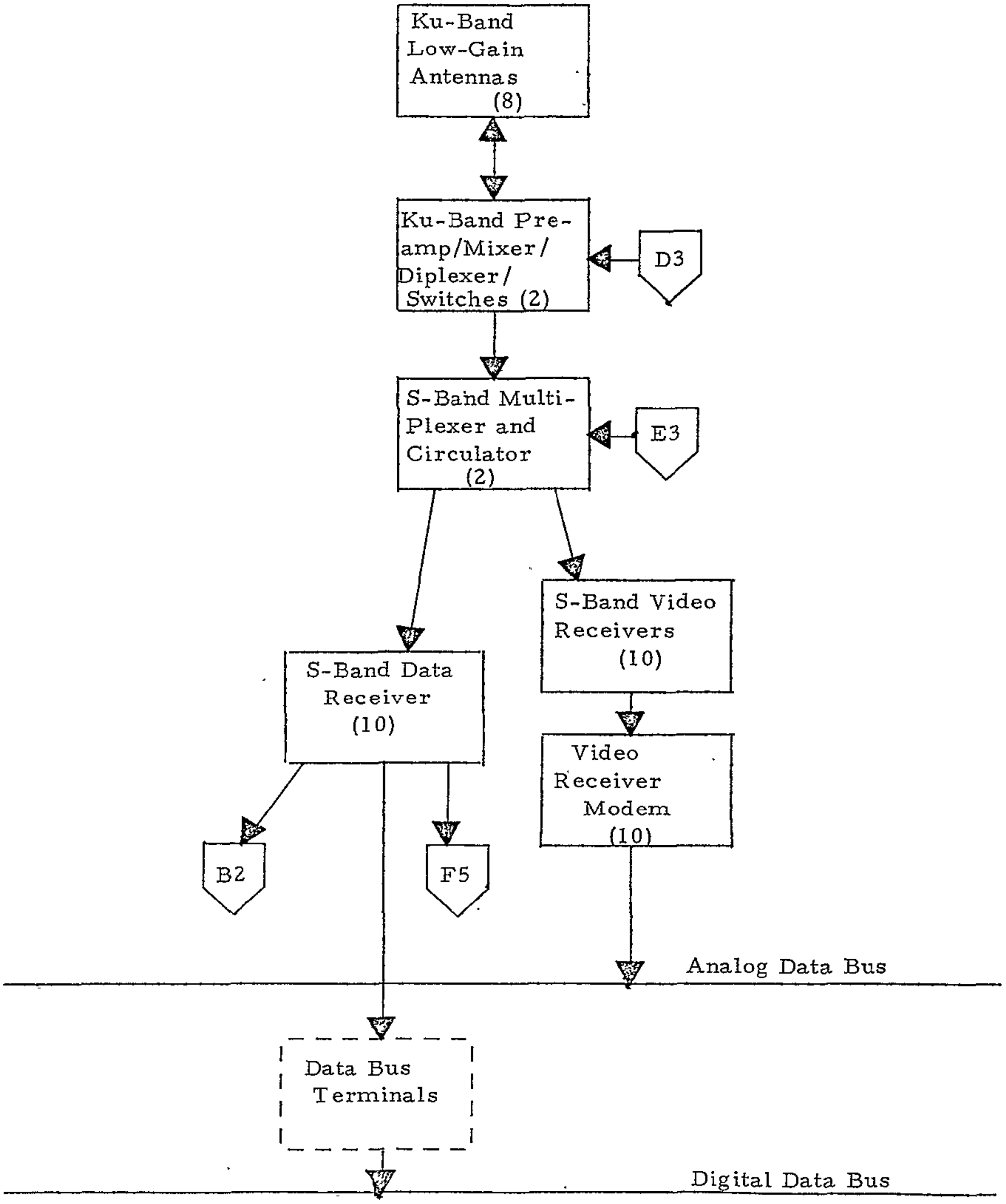


Figure 3-3. LRU Interface Diagram Ku-Band Low-Gain Group

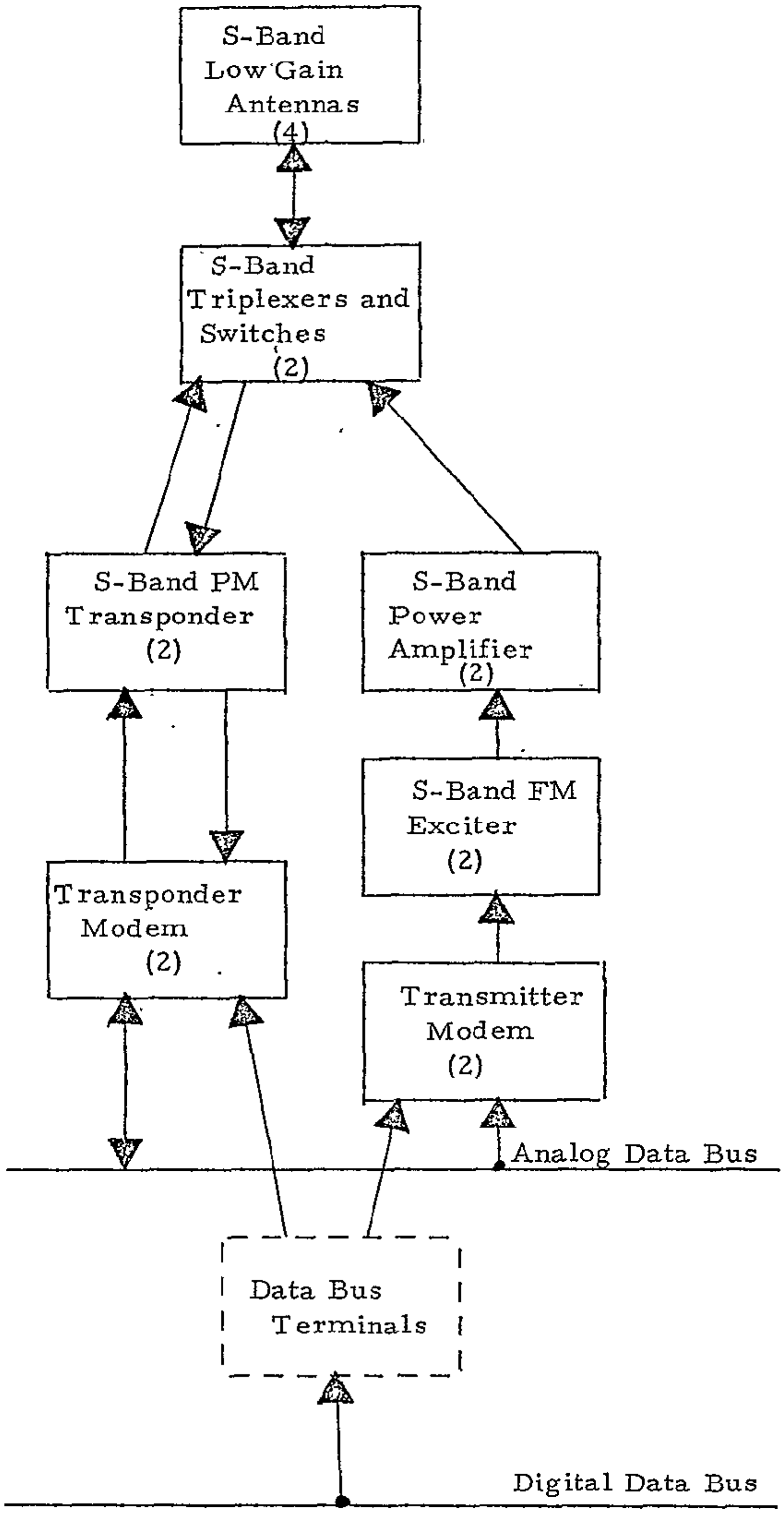


Figure 3-4. LRU Interface Diagram S-Band Low-Gain Group

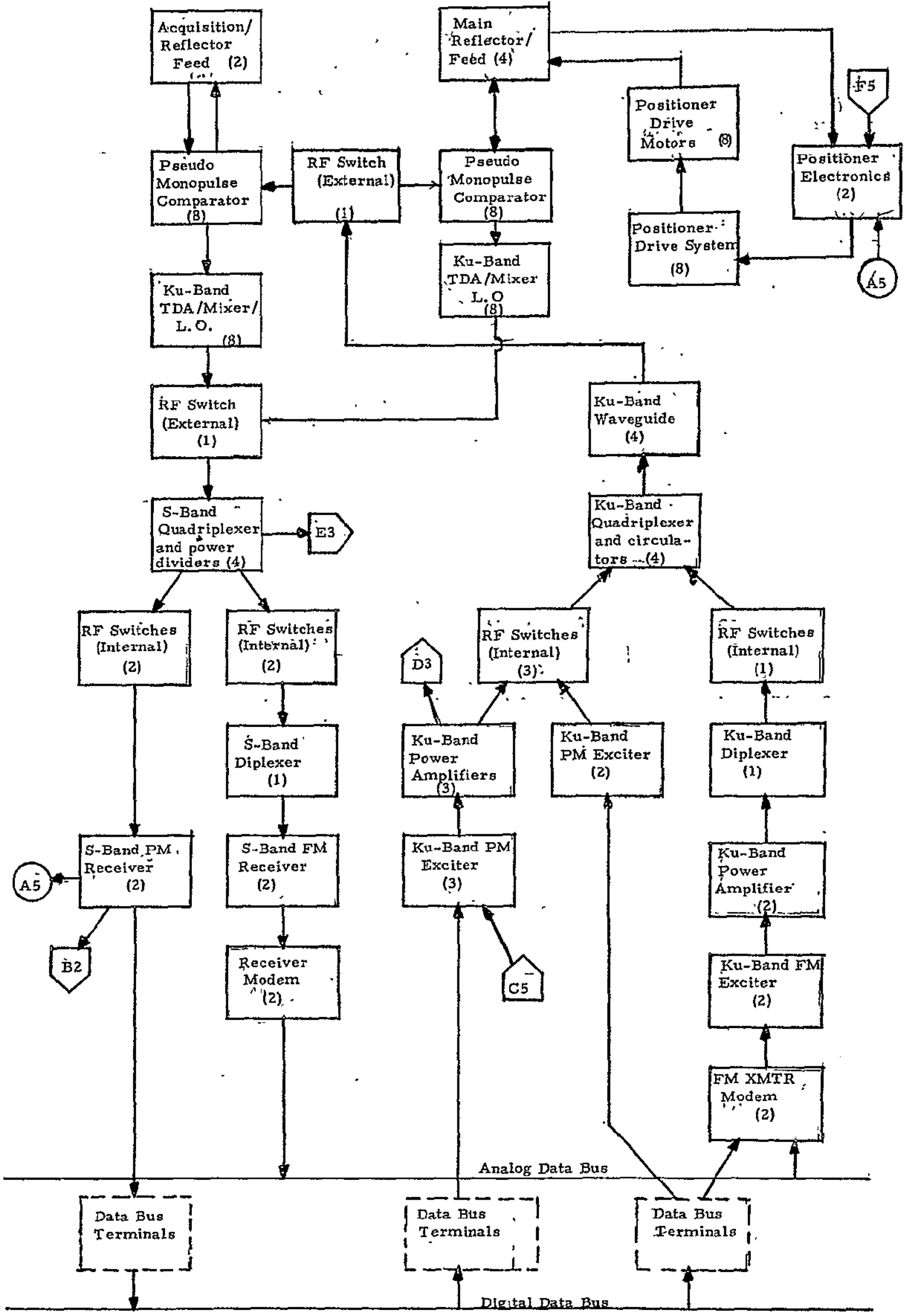


Figure 3-5. LRU Interface Diagram High-Gain Group

The LRU's associated with the high-gain group are shown in Figure 3-5. The fault isolation functions are identified below, with the LRU coverage for each function indicated in parentheses:

- Reception (main reflector/feeds, acquisition reflector/feeds, pseudo monopulse comparators, external RF switches, Ku-band TDA/mixers/L. O. 's, S-band quadriplexers and power dividers, internal RF switches)
- Transmission (main reflector/feeds, acquisition,reflector/feeds, pseudo monopulse comparators, external RF switch, Ku-band waveguide, Ku-band quadriplexers/circulators, internal RF switches)
- S-Band PM Receiver (S-band PM receivers)
- S-Band FM Receiver (S-band diplexer, S-band FM receivers, receiver modems)
- Ku-Band High Power (Ku-band power amplifiers, Ku-band PM exciters)
- Ku-Band Low Power (Ku-band PM exciters)
- FM Transmitter (Ku-band diplexer, Ku-band power amplifiers, Ku-band FM exciters, FM XMTR modems)

3.1.2.4 Periodic Check Function

The periodic check functions identified in this section augment the periodic tests which can be performed by executing the fault isolation function for an assembly group.

Input consists of test point measurements, as well as configuration, status, and limit information from the data base. Output consists of exceptional conditions displayed to crew or ground, and data base updates where exclusive use of the assembly being checked is required.

Information processing consists of obtaining the limit values from the data base, ascertaining that the test point is within these limits, and that the RDAU memory contains the proper values.

3.1.2.5 Calibration Function

The calibration functions required for the Communications Subsystem measure insertion loss in the RF transmission system, and compare with values which have been established in pre-flight and operational evaluations. The RF transmission system is defined as those assemblies which form the signal paths between power amplifiers and exciters, and the antennas of the Space Stations.

Input consists of test point measurements associated with RF power and the Calibration Table. Output consists of messages to the crew.

Information processing consists of using RF power output at the power amplifier or exciter side of the RF transmission system, and RF power level at the antenna to compute insertion loss in decibels, the most convenient form for human interface. Logarithmic calculations required for this function will be performed by a library routine written in machine language, and invoked by high-level language programs, using the CALL language element.

3.1.3 FLOWCHARTS

This section contains flowcharts of typical Communications Subsystem Checkout Programs.

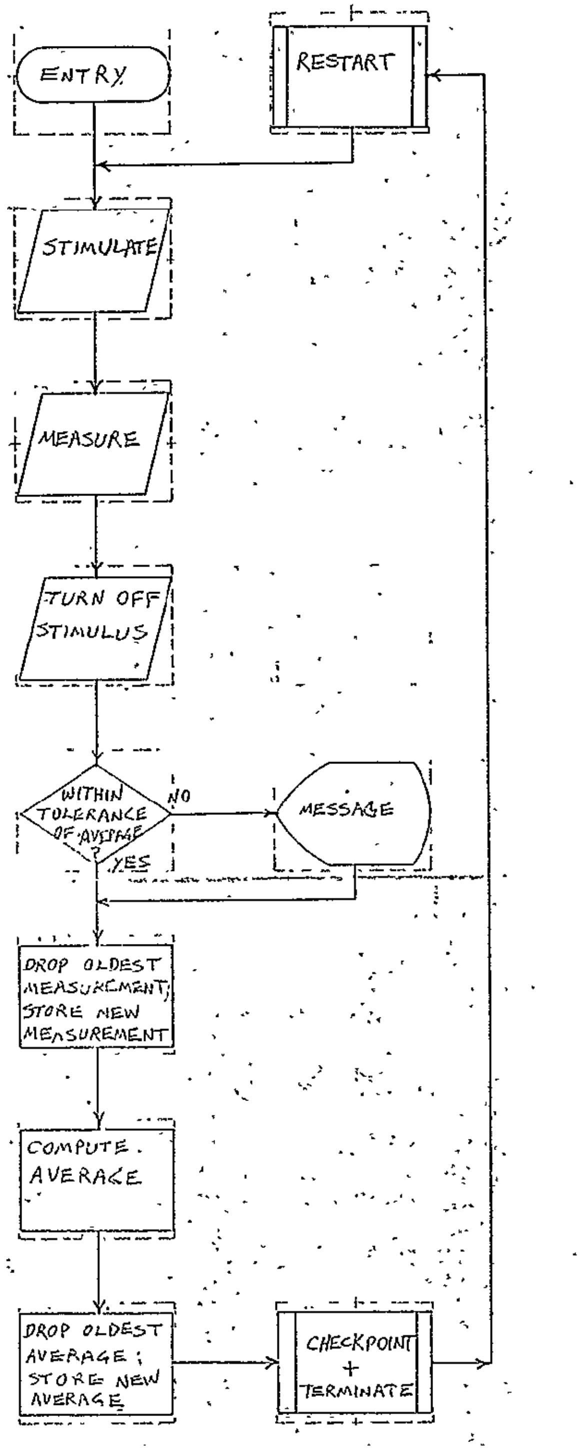


Figure 3-6. Method 1 - Trend Analysis

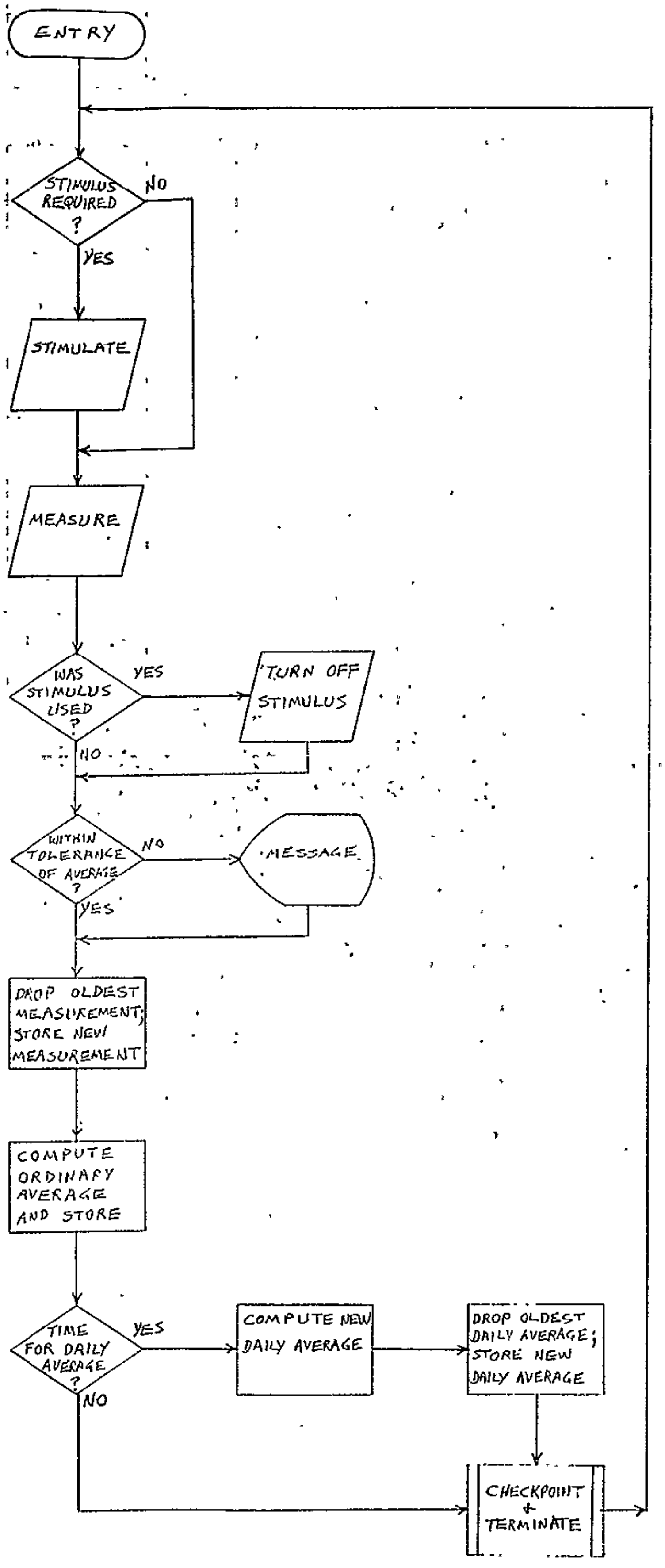


Figure 3-7. Method 2 - Trend Analysis

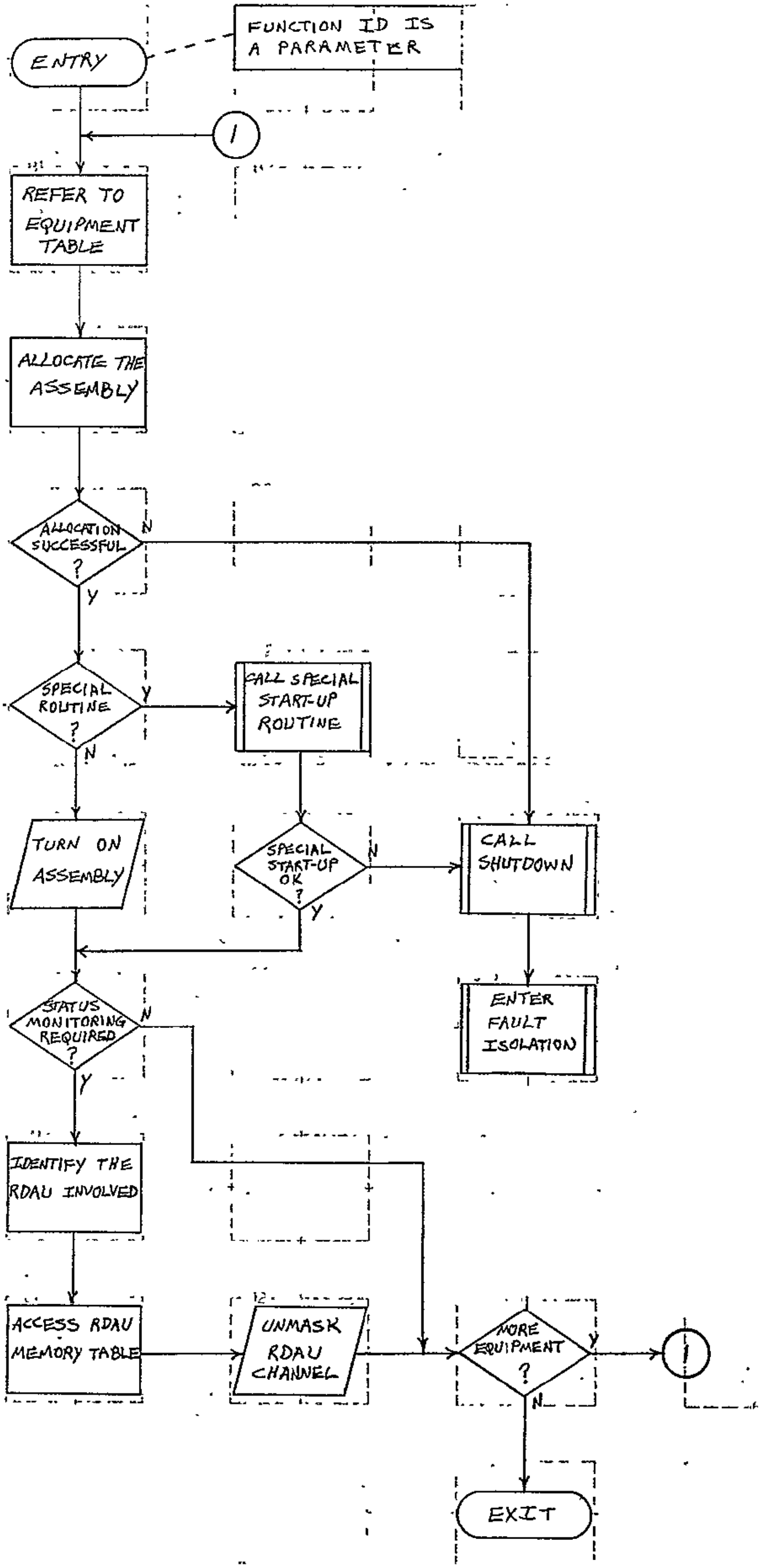


Figure 3-8. Start-Up Routine

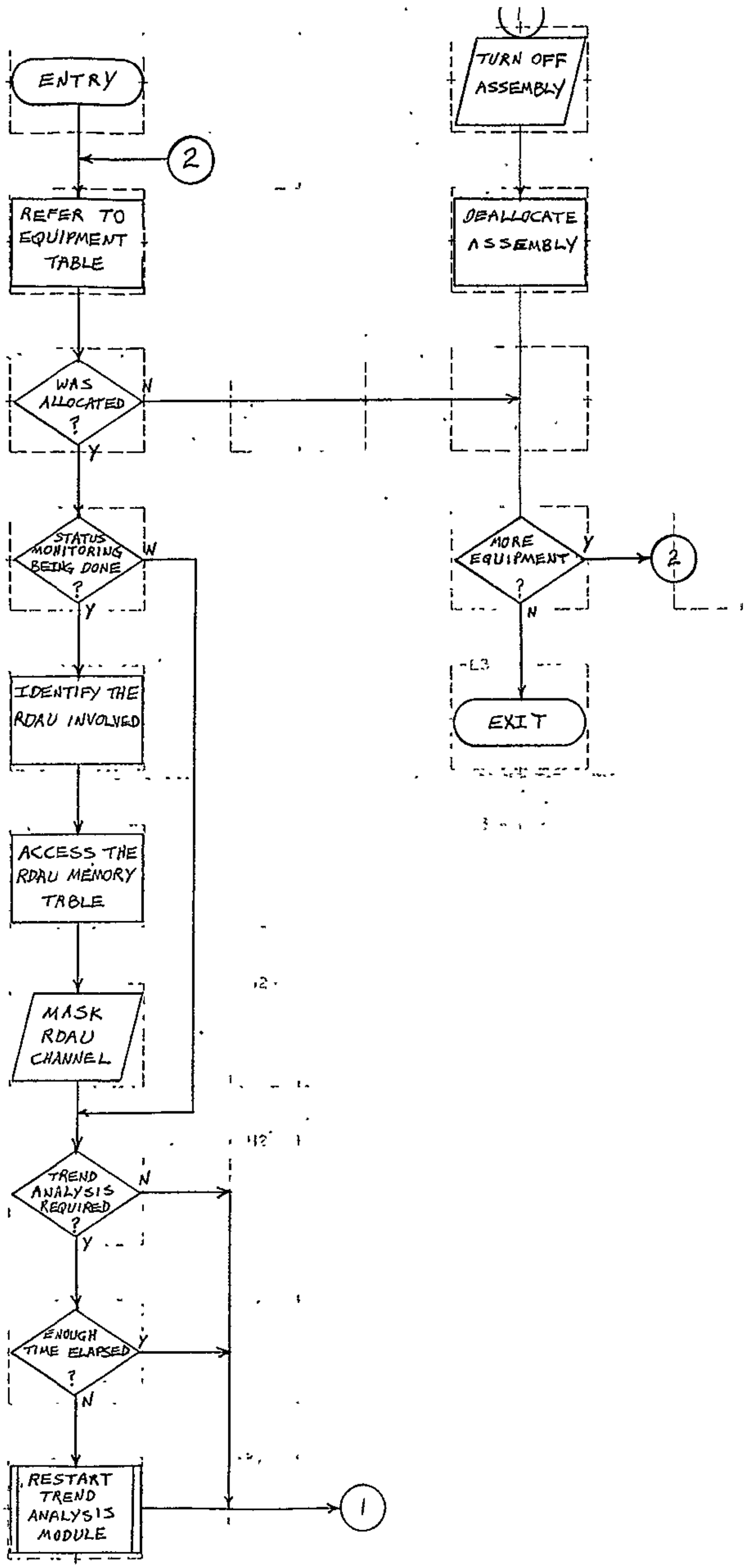


Figure 3-9. Shut-Down Routine

3.1.4 DATA BASE REQUIREMENTS

As in other subsystems, the configuration portion of the data base plays an important role for checkout programs. The dynamic nature of the subsystem configuration must be accurately reflected in the data base at all times so that both operational and checkout programs will address the proper assemblies. Since fault isolation is performed upon the loss of a particular function, it is essential that the LRU's which have been switched in to perform that function are made known to the checkout program.

By designing programs to refer to a set of symbolic test points and treating the set as a configuration which changes as the hardware changes, a higher level of program design can be achieved which will avoid such problems as the following:

- Having to develop a checkout program for every possible hardware configuration.
- Changing programs because of hardware reconfiguration.
- Assuring that the impact of design changes can be completely assessed.
- Having operational programs attempt to use LRU's which have already been fault isolated.

A data base requirement, in addition to those of other subsystems, is the use of an Equipment Table by the operation interface modules to identify any RDAU memory changes which are necessary in order to mask or unmask RDAU channels when certain assemblies are powered off or on; and to identify the need for trend analysis measurements prior to deactivating an assembly.

A Calibration Table stores insertion loss characteristics which are determined in preflight analysis and compared with actual measurements after LRU replacement in the RF transmission system.

3.1.5 HUMAN PERFORMANCE

The Communications Subsystem relies more extensively on human performance for the detection of faults than most other Space Station subsystems. When voice or video quality deteriorates such that fault isolation is desired, the identity of the assembly associated with the problem must be supplied by the crew member at the time fault isolation is invoked.

Insertion losses calculated by the calibration functions are expressed in terms of decibels to allow the acceptance levels to be expressed in a form compatible with engineering practice.

3.2 INTERFACE REQUIREMENTS

In this section, program to program, program to data base, and program to DMS I/O, assembly interfaces are indicated.

The checkout program to executive interfaces required for the Communication Subsystem are as follows:

- Data base access is gained through the services of the executive in order to control data base changes and to assure the ability to reconfigure memory without impacting program performance.
- Reference to all I/O devices, including those associated with test points, is done through executive services in order to keep clerical details in the programming language to a minimum.
- Programs refer symbolically to test points and rely on the executive to translate the symbols into actual addresses at execution time by using the configuration information maintained in the data base. This assures use of the latest hardware configuration without program modification.

3.2.1 INTERFACE DIAGRAM

The relationships among the Communications Subsystem checkout functions are shown in Figure 3-10.

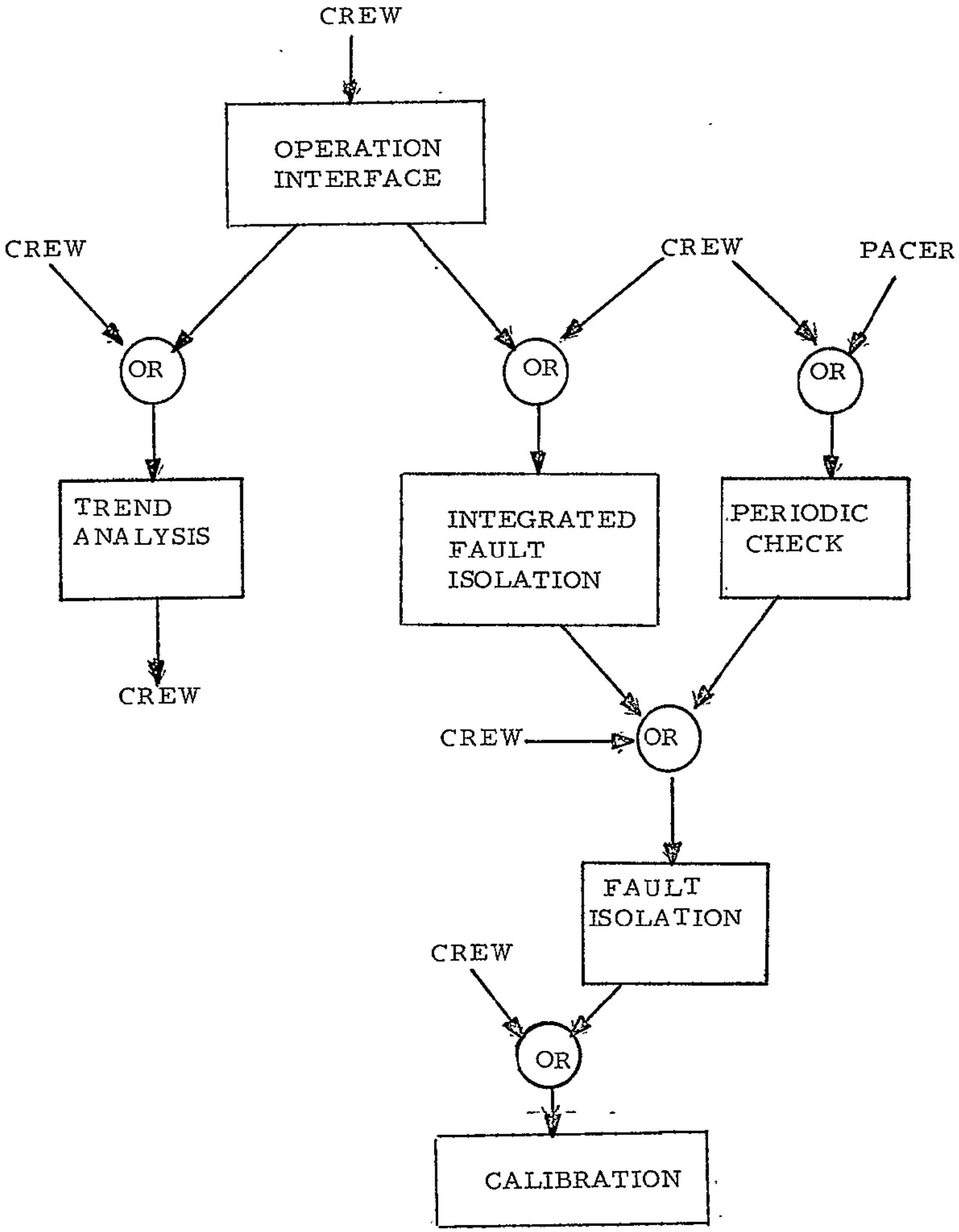


Figure 3-10. Communication Checkout Interface

3.2.2 DETAILED INTERFACE DEFINITION

Utilization of the data base and basic DMS I/O assemblies is indicated for each Communications Subsystem checkout function in Figure 3-11 through Figure 3-15. Use of the Configuration Table is made throughout; however, it is explicitly shown only when a particular function updates it.

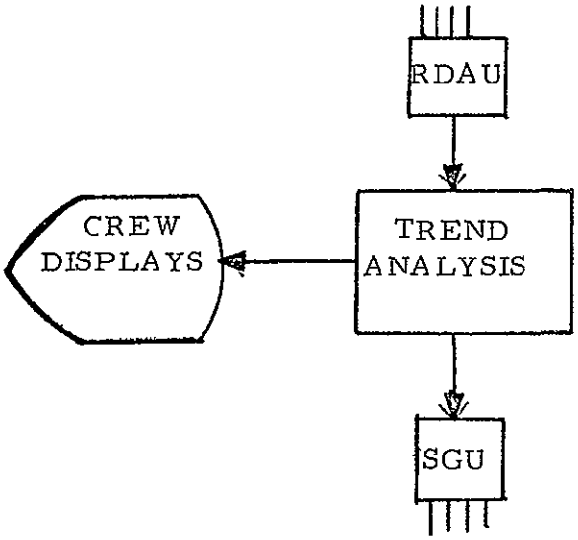


Figure 3-11. Trend Analysis Function Interface Diagram

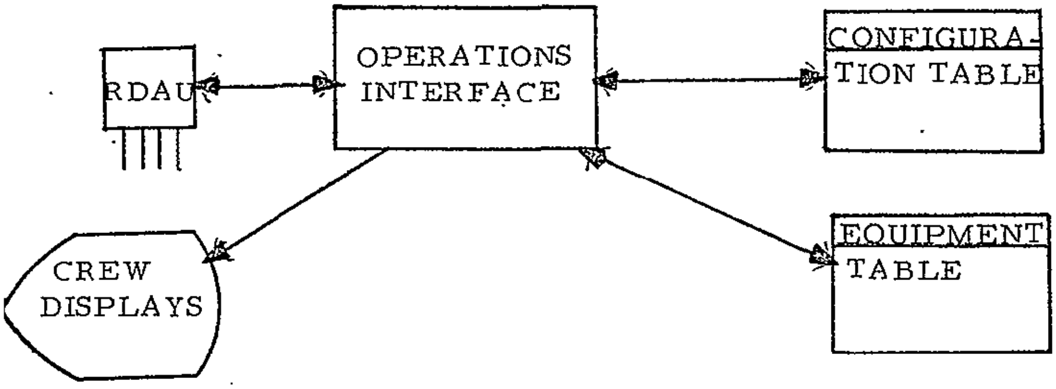


Figure 3-12. Operations Interface Function Interface Diagram

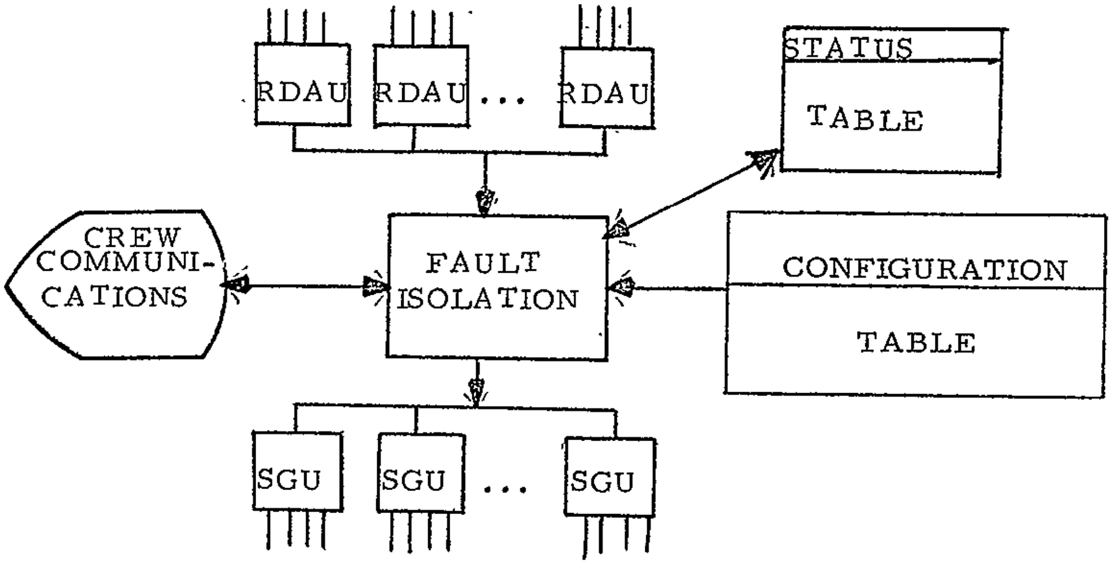


Figure 3-13. Fault Isolation Function Interface Diagram

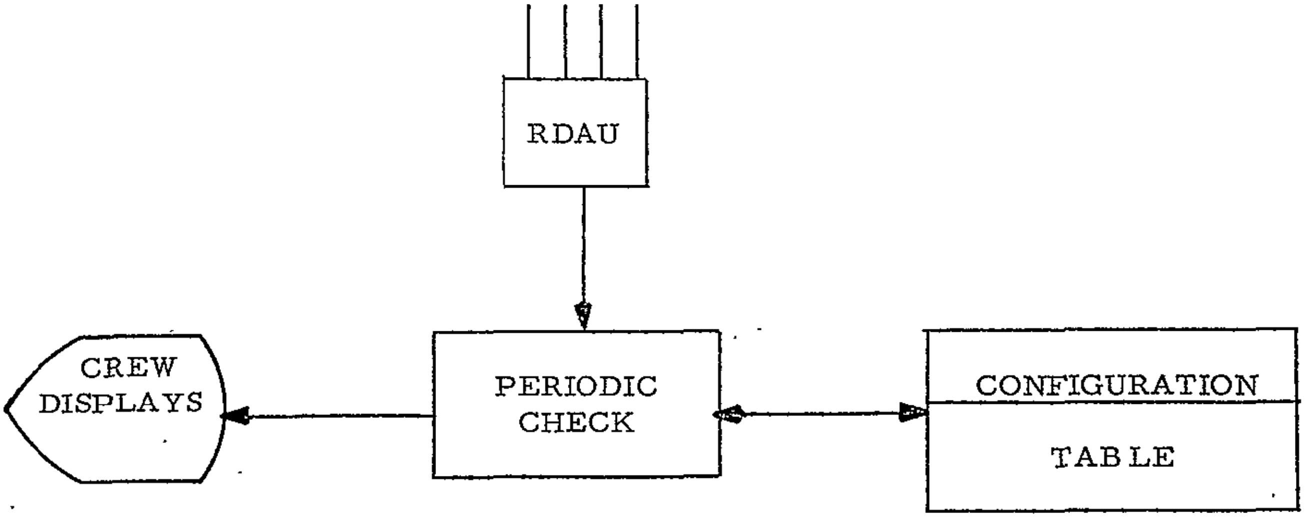


Figure 3-14. Periodic Check Function Interface Diagram

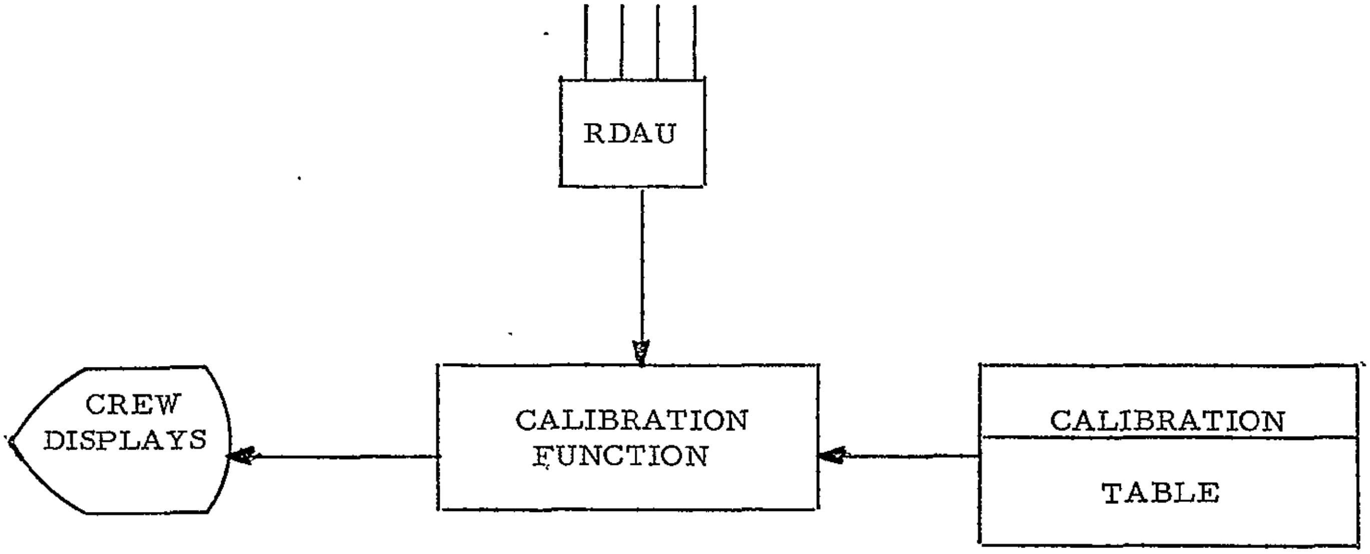


Figure 3-15. Calibration Function Interface Diagram

4.0 SUBSYSTEM SUMMARY

The checkout requirements study of the Communications Subsystem results in requirements regarding language, executive, and the Data Management Subsystem which are discussed in the following paragraphs.

4.1 LANGUAGE ANALYSIS

The language requirements of the Communications Subsystem are within the set of requirements indicated for the other subsystems, with the exception of the need for computing logarithms to the base 10. This requirement arises from the use of decibels by communications engineers to express certain measurement comparisons. While this function could be implemented as a language element, its frequency of use would appear to warrant only a CALL to a library subroutine.

Because many assemblies such as antennas and power amplifiers have multiple functions, the importance of maintaining an accurate, real-time expression of the configuration in the data base is underscored.

The ability to refer to symbolic test points and to use a single program with one of many sets of test points will reduce from approximately 220 to 80 the number of Communication System checkout program modules. In Table 4-1, the trend analysis sizing estimates are indicated, based on the assumption that these modules would be implemented in machine language. In Table 4-2 through Table 4-7, a typical module breakdown for the remaining functions is indicated, together with estimates of the high-level language elements which are required. The column entitled "Number of Applications" indicates the number of different uses for a single program module, provided the program is written to refer to symbolic, rather than actual test points, and provided the symbolic test points are expressed as groups in the configuration data base.

4.2 EXECUTIVE ANALYSIS

No additional executive services beyond those identified for other subsystems have been identified. Previously identified services which are particularly useful for the Communications Subsystem are:

- Data base maintenance interface and access control
- Pacer services
- Symbolic test point translation
- Program parameter assembly prior to initiation
- Utility services for RDAU memory management
- I/O services

- Rate table management utilities
- Allocation of DMS resources to programs

4.3 PROGRAM SIZING

In order to evaluate the effect of the Communications Subsystem checkout programs on the Data Management Subsystem, individual program modules are identified in Tables 4-1 through 4-6, and listed with certain characteristics which must be considered together in an overall analysis. For trend analysis modules which are implemented in machine language, the performance parameters are listed in terms of average processor time, I/O time, and storage words in Table 4-1. The characteristics of the high level language modules are tabulated according to the frequency with which the program executes the language elements. After estimating the DMS requirements of the individual language elements, the element utilization data in Tables 4-2 through 4-4 were reduced to DMS performance parameters as indicated in Tables 4-7 through 4-12.

Table 4-1. Trend Analysis Program Sizing

TREND PROGRAM	PROCESSOR TIME	I/O TIME	MAIN STORAGE	TIME IN MAIN STORAGE	AUXILIARY STORAGE
METHOD 1	800±300 usec	216±24 usec	120±36 words	1016±300 usec	120±30 words
METHOD 2	900±400 usec	144±100 usec	150±50 words	1200±300 usec	220±60 words

Table 4-2. Operation Interface Language Usage Estimates Start-Up Modules

			LANGUAGE ELEMENTS													
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	S-Band Video Rcvr	10	1	1	1	2	2	2		1	1	1	1	2	1		
	S-Band Data Rcvr	10	1	1	1	2	2	2		1	1	1	1	2	1		
	S-Band PM Rcvr	2	1	1	1	2	2	2		1	1	1	1	2	1		
	Ku-Band FM Xmtr	2	4	3	2	4	2	4	1	2	2	2	1	4	1		
	Ku-Band PM Xmtr	3	4	3	2	4	2	4	1	2	2	2	1	4	1		
	Ku-Band PM Exciter	2	1		1	1	1	1		1	1	1	1	1	1		
	S-Band Transponder	2	2	1	1	2	2	2		2		2	1	2	1		
	S-Band FM Xmtr	2	2	2	2	2	2	4	1	2	2	2	1	2	1		
	VHF Voice/Rng. T/R	2	4	3	2	4	2	4	1	2	3	2	1	4	1		
	VHF Data T/R	2	4	3	2	4	2	4	1	2	2	2	1	4	1		
	VHF FM T/R	3	4	3	2	4	2	4	1	2	2	2	1	4	1		

Table 4-3. Operation Interface Language Usage Estimates Shut-Down Modules

			LANGUAGE ELEMENTS													
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	S-Band Video Rcvr	10	1	1	1	3	2	2		1	2	1	1	3	1		
	S-Band Data Rcvr	10	1	1	1	3	2	2		1	2	1	1	3	1		
	S-Band PM Rcvr	2	1	1	1	3	2	2		1	2	1	1	3	1		
	S-Band FM Rcvr	2	1	1	1	3	2	2		1	2	1	1	3	1		
	S-Band PM Trnspndr.	2	1	1	1	3	3	2		2	2	2	1	3	1		
	VHF Voice/Ranging	2	1	1	1	3	2	2		1	2	1	1	3	1		
	VHF Data T/R	2	2	2	2	4	3	4		2	3	2	1	4	1		
	VHF FM Rcvr	3	1	1	1	3	2	2		1	2	1	1	3	1		

Table 4-4. Fault Isolation Language Usage Estimates
Low Gain Group

			LANGUAGE ELEMENTS													
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	VHF RF Transmission	2	21	2		15	4	3		1	6		1	15	1		
	VHF Voice	2	41	29	8	45	30	8	4	1	4	3	1	45	1		
	VHF PM Data	2	3	3	2	4	2	4	1	1	1		1	4	1		
	VHF Voice/Ranging	2	27	25	6	25	4	6	3	1	8	4	1	25	1		
	Ku-Band RF Trans.	2	17	10	12	15	8	10		4	8		1	15	1		
	S-Band Data	10	3	3	2	2	2	2	1	1			1	2	1		
	S-Band Video	10	6	6	3	3	3	3	2	2	4	2	1	3	1		
	S-Band RF Trans.	2	5	1	4	4	3	2		1	2		1	4	1		
	S-Band Transponder	2	13	10	4	8	4	4	2	2	4	4	1	8	1		
	S-Band FM Trnsmtr.	2	12	10	6	10	5	4	3	3	3	2	1	10		1	

Table 4-4. Fault Isolation Language Usage Estimates (Continued)
High Gain Group

PROGRAM MODULES		LANGUAGE ELEMENTS													
	Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
Reception	1	76	76	30	30	20	12		12	20	12	1	30	1		
Transmission	1	61	40	20	30	30	10		10	20	10	1	30	1		
S-Band PM Receiver	2	3	3	2	2	1	2	1	1			1	2	1		
S-Band FM Receiver	2	9	9	4	12	6	8	2	2	4	6	1	12	1		
Ku-Band High Power	3	6	5	4	4	2	4	2	2	2		1	4	1		
Ku-Band Low Power	2	3	2	2	2	1	2	1	1			1	2	1		
FM Transmitter	2	13	11	4	10	8	6	3	3	6	2	1	10	1		

Table 4-5. Communications Periodic Test Language Usage Estimates

			LANGUAGE ELEMENTS													
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	Primary Power	1	77			78	1	2				77	1	78	1		
	Switch Position	1	84			2	1	2				84	1	2	1		
	Mode	1	42			2	1	2				42	1	2	1		
	Receiver AGC Output	1	31			2	1	2				31	1	2	1		
	Receiver Mod. Output	1	31			2	1	2				31	1	2	1		
	Modem Mod. Output	1	28			2	1	2				28	1	2	1		
	XCTR/XMTR RF Pwr	1	16			2	1	2				16	1	2	1		
	XCTR/XMTR Mod.	1	19			2	1	2				19	1	2	1		
	PA RF Power	1	7			2	1	2				7	1	2	1		
	Preamp/Mixer/Lo O/F	1	6			2	1	2				6	1	2	1		
	HGA Feed Power	1	8			2	1	2				8	1	2	1		
	HGA Position	1	8			2	1	2				8	1	2	1		
	HGA Feed VSWR	1	88			2	1	2				8	1	2	1		
	LGA Ele. RF power	1	8			2	1	2				8	1	2	1		
	LGA Element VSWR	1	8			2	1	2				8	1	2	1		
	Ranging Moded Data	1	2			4	3	2				2	1	4	1		

Table 4-6. Calibration Language Usage Estimates

		▨	LANGUAGE ELEMENTS														
		Number of Applications	MEASURE	STIMULATE	ALLOCATE	IF	SOLVE	DISPLAY	DELAY	WRITE	CALL	READ	BEGIN	GOTO	END		
PROGRAM MODULES	High Gain System	4	2			2	2	2					1	2	1		
	Ku-Band Low Gain	8	2			2	2	2					1	2	1		
	S-Band Low Gain	4	2			2	2	2					1	2	1		
	VHF Low Gain	4	2			2	2	2					1	2	1		

The column headings in Tables 4-7 through 4-12 have the following meanings:

PROGRAM MODULE:	The name of the program.
RATE:	Frequency with which this program module will be executed for each application.
APPLICATIONS:	Number of uses for the module. As an example, if four identical assemblies required a periodic test once per hour, the number of applications would be four and the rate 1/H. Viewed from the multiprocessor, the traffic rate would be 4/H.
PROCESSOR TIME:	(Milliseconds). The time required for actual instruction execution; the average time that a processor will be busy because of the program. Note that this time is exclusive of I/O time, and that the work load for any module is shared among the processors without prejudice.
I/O TIME:	(Milliseconds). The time in which execution of the program in main storage must be suspended while a data acquisition path program is performed by a data bus controller or channel.
MAIN STORAGE:	(Words). The amount of shared main storage required by the program text (instructions), plus that required for working storage (data and constants).
TIME IN MAIN STORAGE:	The average time during which the program must occupy shared main storage. In some cases, this is more than the sum of processor and I/O times because of the need to delay briefly while the subsystem responds to stimuli.
AUXILIARY STORAGE:	(Words). The amount of auxiliary storage required for program text (instructions), tables, constants intermediate results, results which must be retained for later use, etc.

Where encircled numbers appear in Tables 4-7 through 4-12, they have the following meanings:

1. Rate of execution is determined by the failure rate of the LRU's addressed by the program.
2. Time in main storage is the sum of processor and I/O times.
3. Auxiliary storage requirement is estimated to be the same as the main storage requirement.
4. Rate depends on duty cycle of the associated assembly.
5. Time in main storage determined by the warm-up time or time required for the associated assembly to reach steady-state.

Table 4-7. Operation Interface (Start-Up) Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
S-Band Video Receiver	④	10	4.92	914	161	⑤	③
S-Band Data Receiver	④	10	4.92	914	161	⑤	③
S-Band PM Receiver	④	2	4.92	914	161	⑤	③
Ku-Band FM XMTR	④	2	11.0	2130	312	⑤	③
Ku-Band PM XMTR	④	3	11.0	2130	312	⑤	③
Ku-Band PM Exciter	④	2	3.57	812	104	⑤	③
S-Band Transponder	④	2	6.25	1220	169	⑤	③
S-Band FM XMTR	④	2	8.72	1830	270	⑤	③
VHF Voice/Ranging T/R	④	2	11.4	2340	325	⑤	③
VHF Data T/R	④	2	11.0	2130	312	⑤	③
VHF FM T/R	④	3	11.0	2130	312	⑤	③

Table 4-8 . Operation Interface (Shut-Down) Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
S-Band Video Receiver	④	10	5.55	1120	186	⑤	③
S-Band Data Receiver	④	10	5.55	1120	186	⑤	③
S-Band PM Receiver	④	2	5.55	1120	186	⑤	③
S-Band FM Receiver	④	2	5.55	1120	186	⑤	③
S-Band PM Transponder	④	2	6.97	1520	209	⑤	③
VHF V/R Receiver	④	2	5.55	1120	186	⑤	③
VHF Data T/R	④	2	9.79	2030	315	⑤	③
VHF FM Receiver	④	3	5.55	1120	186	⑤	③

Table 4-9 . Fault Isolation (Low Gain) Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
VHF RF Transmission	①	2	22.4	3760	580	②	③
VHF Voice	①	2	72.4	9550	1632	②	③
VHF PM Data	①	2	8.37	1220	273	②	③
VHF Voice/Ranging	①	2	49.4	8530	987	②	③
Ku-Band RF Trans.	①	2	36.8	6400	950	②	③
S-Band Data	①	10	7.09	1020	172	②	③
S-Band Video	①	10	15.3	3150	335	②	③
S-Band RF Trans.	①	2	9.72	1630	250	②	③
S-Band Transponder	①	2	25.3	4770	517	②	③
S-Band FM Transmitter	①	2	25.8	4470	541	②	③

Table 4-10. Fault Isolation (High Gain) Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Reception	①	1	146	27400	2322	②	③
Transmission	①	1	110	20400	2072	②	③
S-Band PM Receiver	①	2	6.79	1020	161	②	③
S-Band FM Receiver	①	2	25.7	4672	692	②	③
Ku-Band High Power	①	3	13.6	2340	335	②	③
Ku-Band Low Power	①	2	6.19	915	159	②	③
FM Transmitter	①	2	28.4	5080	676	②	③

Table 4-11. Periodic Tests Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
Primary Power	1/Mo	①	66.2	7820	1599	②	③
Switch Position	1/W	①	96.1	25600	1201	②	③
Mode	1/W	①	48.7	127800	655	②	③
Receiver AGC Output	1/W	①	36.3	9440	512	②	③
Receiver Modulation O/P	1/W	①	36.3	9440	512	②	③
MODEM Modulation O/P	1/W	①	32.9	8520	473	②	③
XCTR/XMTR RF Power	1/W	①	19.4	4870	317	②	③
XCTR/XMTR Modulation	1/W	①	22.8	5780	356	②	③
PA RF Power	1/W	①	9.23	2130	200	②	③
Preamp/Mixer Lo Output	1/W	①	8.10	1830	187	②	③
HGA Feed Power	1/W	①	10.4	2440	213	②	③
HGA Position	1/W	①	10.4	2440	213	②	③
HGA Feed VSWR	1/W	①	10.4	2440	213	②	③
LGA Element RF Power	1/W	①	10.4	2440	213	②	③
LGA Element VSWR	1/W	①	10.4	2440	213	②	③
Ranging MODEM Data	1/W	①	4.67	610	179	②	③

Table 4-12. Calibration Program Sizing

Program Module	Rate	Applications	Processor Time (Millisec.)	I/O Time (Millisec.)	Main Storage Words	Time in Main Storage	Auxiliary Storage
High Gain System	①	4	2.85	204	135	②	③
Ku-Band Low Gain	①	8	2.85	204	135	②	③
S-Band Low Gain	①	4	2.85	204	135	②	③
VHF Low Gain	①	4	2.85	204	135	②	③

APPENDIX F

PROPULSION SUBSYSTEM CHECKOUT PROGRAM DESIGN

TABLE OF CONTENTS

Table of Contents (Cont)

LIST OF ILLUSTRATIONS

LIST OF TABLES

1.0 SCOPE

The propulsion subsystem consists of high thrust and low thrust propellant systems. Both systems interface with the GN&C subsystem through the Data Management subsystem for operational control. The low thrust system also interfaces with the EC&LS subsystem for gases and water which are used as propellants.

The fault detection function required for the propulsion subsystem is accomplished by tables containing the parameters which must be monitored to assure subsystem performance. These tables are transferred to the Remote Data Acquisition Unit (RDAU) via the executive program. Exception monitoring is then accomplished. Figure 1-1 provides a graphic description of this function. Table 1-1 has been provided to indicate the extent of the overall fault detection requirements.

The program described by this document is required for periodic checkout and fault isolation.

Initiation of the periodic checkout function is accomplished as the result of a keyboard entry by a crew member. It is anticipated that periodic checkout will be accomplished both daily and on a tri-monthly basis with somewhat different requirements.

The fault isolation function utilizes the same software modules as the periodic checkout; however, analysis of the detected error by the sequence logic module permits selection of the appropriate module to begin the required fault isolation. If the error is not detected in the selected area, the program module provides this information and recommends that the periodic test be executed.

Subsystem Calibration is performed in conjunction with the periodic test. Trend analysis is executed on a basis of varying requirements by the executive. Tables 1-2 and 1-3 have been included to provide insight to the requirements in this area.

2.0 APPLICABLE DOCUMENTS

- Space Station MSFC-DRL-160 Line Item 8, Volume V Book Mechanical, McDonnell-Douglas Astronautics Co.
- Baseline Subsystem Descriptions, Interim Report RA-1, August 1970, Contract NAS9-11189, IBM.
- Line Replaceable Unit Definition, Interim Report RA-6, September 1970, Contract NAS9-11189, IBM.

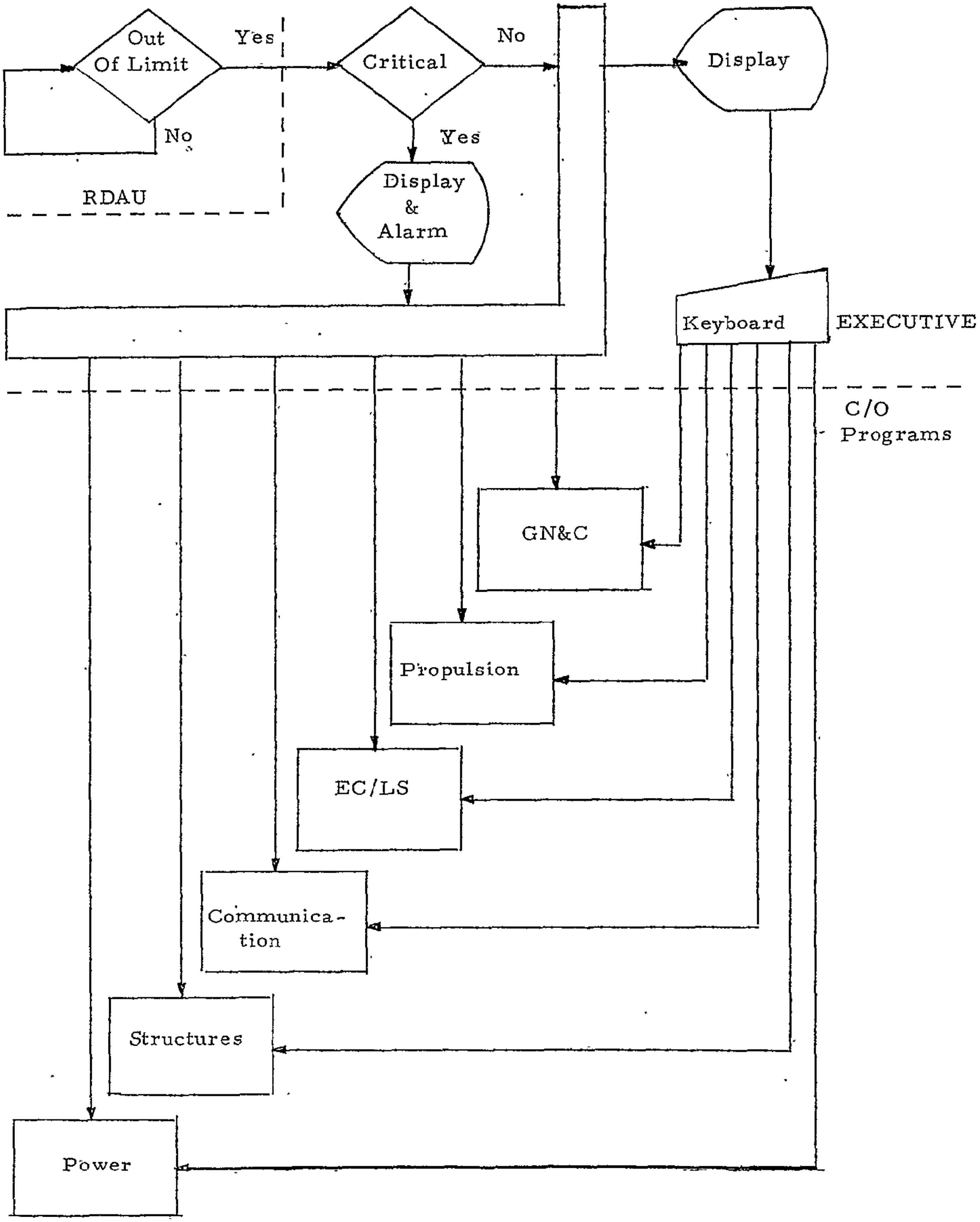


Figure 1-1. Fault Detection Logic

Table 1-1. Propulsion Subsystem Fault Detection Summary

ASSEMBLY / SAMPLE RATE	1/Sec
Low Thrust System	
Collection/Storage Assembly	20
Water Supplement Assembly	4
Flow Control Assembly	12
Thrustor Assembly	104
Power Distribution & Control	64
High Thrust System	
Pressure Storage Assembly	8
High Pressure Manifold	2
Pressure Control Assembly	2
Low Pressure Manifold	2
Propellant Storage Assembly	28
Propellant Manifold	3
Thrustor Module	32
Purge Assembly	6
Resupply Assembly *	2
High Pressure Assembly *	8
Low Pressure Assembly *	22
Misc Temperatures	100
Total Per Second	419
Total Per Minute	25, 140
Total Per Hour	1, 508, 400
Total Per Day	36, 201, 600

* Only during Resupply Operation

Table 1-2. Propulsion Subsystem Trend Analysis Summary

SAMPLE FREQUENCY ASSEMBLY	1/Day
Low Thrust System	
Collection Storage Assembly	16
High Thrust System	
Pressure Storage Assembly	8
High Pressure Manifold	2
Low Pressure Manifold	2
Propellant Storage Assembly	42
Propellant Manifold	3
Thrustor Modules	56
Purge Assembly	4
Total Per Day	133

Table 1-3. Propulsion Subsystem Calibration Summary

ASSEMBLY / CALIBRATION FREQUENCY	1/3 Mon	1/6 Mon
Low Thrust System		
Collection Storage Assembly		24
Water Supplement Assembly		8
Flow Control Assembly		20
Thrustor Assembly		6
High Thrust System		
Pressure Storage Assembly	8	
High Pressure Manifold	2	
Pressure Control Assembly	4	
Low Pressure Manifold	2	
Propellant Storage Assembly	42	
Propellant Manifold	3	
Thrustor Modules	92	
Purge Assembly	4	
Resupply Assembly	2	
Misc Temperatures		100
TOTAL	159	158

- Subsystem Test Descriptions and Measurement Stimulus List, Interim Report RA-8, November 1970, Contract NAS9-11189, IBM.

- Failure Mode and Effects Analysis, Interim Report RA-4, September 1970, Contract NAS9-11189, IBM.

- Subsystem Checkout Strategy, Interim Report RA-5, September 1970, Contract NAS9-11189, IBM.

- Space Station MSFC-DRL-160 Line Item 19, Volume I, Performance/Design and Product Configuration Requirements, July 1970, Contract NAS8-25140, McDonnell-Douglas Astronautics Co.

3.0 REQUIREMENTS

This program meets the periodic testing and fault isolation requirements for the Propulsion Subsystem.

3.1 PERFORMANCE

The Propulsion Subsystem consists of two independent subsystems for propulsion. This division between the high and low thrust system was used to provide definition of functional areas for the program.

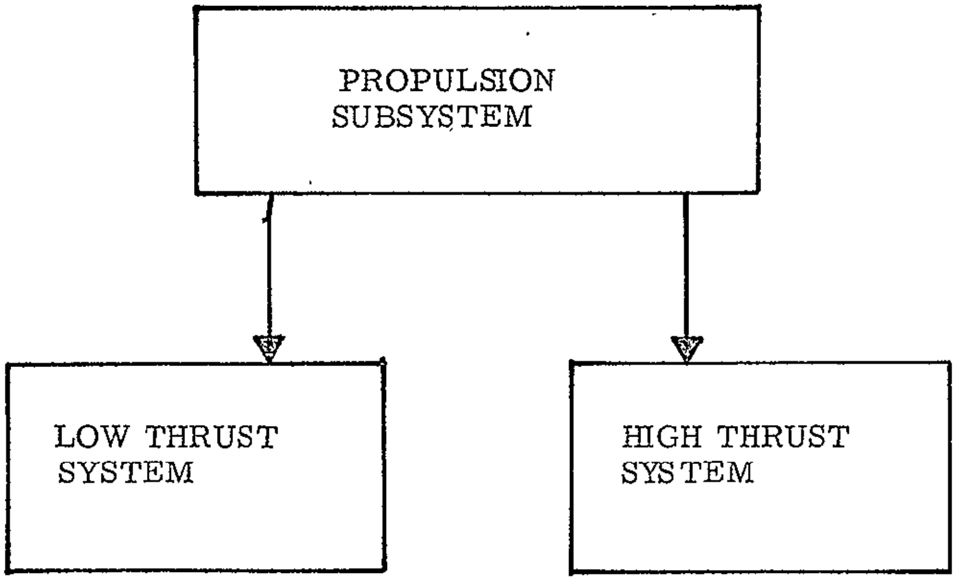
Figure 3-1 provides a functional breakdown of this subsystem and indicates the associated assemblies.

3.1.1 SYSTEM REQUIREMENTS

This section describes system constraints and requirements which have influenced design.

3.1.1.1 Subsystem Definition

This program specification is based upon the subsystem definition which is available as a result of this study contract. Some test points in this subsystem are defined at the assembly level; and consequently, every failure which is detected cannot currently be identified with an LRU. Also, the correlation between the assembly test points identified in the "Subsystem Test Descriptions and Measurement Stimulus List" (RA-8), and the LRU's identified in the "Line Replaceable Units Definition", is not always apparent.



- Collection & Storage Assembly
- Water Supplement Assembly
- Flow Control Assembly
- Thrustor Assembly
- Power Dist. & Control Assembly

- High Pressure Assembly
- Low Pressure Assembly
- Resupply Assembly
- High Pressure Manifold
- Purge Supply System
- Propellant Manifold
- Thrustor Modules
- Propellant Storage Assembly
- Low Pressure Manifold
- Pressure Control Assembly
- Pressure Storage Assembly

Figure 3-1. Propulsion Subsystem Block Diagram

3.1.1.2 Collection Storage Assembly

The algorithm required to compute gas level (mass) in the storage bottles, based upon the temperature and pressure, has not been defined.

3.1.1.3 Trend Analysis and Calibration Constants

The algorithms required for Trend Analysis and the calculation of calibration constants have not been defined, and could significantly impact the sizing estimates, which are a least squares technique.

3.1.1.4 Miscellaneous Temperature

The placement of the 100 miscellaneous temperature sensors, which are defined for the subsystem, has been assumed.

3.1.1.5 Fault Detection

The operational program is responsible for maintaining the proper test points in the RDAU memory. This selection is dependent upon whether the storage assemblies are being resupplied, or the subsystem is in a "ready to fire" status.

3.1.2 OPERATIONAL REQUIREMENTS

This program specification defines specific operational requirements for automated checkout of the Space Station Propulsion Subsystem. The sequence of testing attempts to examine the least dependent functional groups first.

3.1.2.1 Sequence Logic Module

This software module is used to select the appropriate sequence of program modules to be executed in the event an error is detected in this subsystem. It also provides the sequencing required for both the daily and tri-monthly periodic tests.

3.1.2.1.1 Source and Type of Inputs

This module provides the entry points for the periodic and fault isolation testing. In the event that a fault is detected, the failing test point will also be identified to this module.

3.1.2.1.2 Destination and Types of Outputs

The only outputs from this module are the displays associated with the progress of the testing.

3.1.2.1.3 Information Processing

This module determines whether fault isolation or periodic testing is to be accomplished. In the event that fault isolation is required, the detected error is isolated to an assembly. If the program is unable to isolate an error in the selected and associated assemblies, a message is presented to the operator recommending execution of the periodic test.

When an error is detected in the Collection/Storage Assembly, it is examined; and if it was the bottle pressure or relief valve, the isolation valve is closed prior to execution of the Collection/Storage Assembly checkout module. If the detected error was the upstream pump flow, the CO_2/CH_4 flow from the EC&LS subsystem is verified prior to execution of the Collection/Storage Assembly checkout module.

If this checkout module fails to isolate an error and the problem was detected in the propellant control valves, the checkout module for the Power Distribution and Control Assembly is executed.

When an error is detected in the Water Supplement Assembly, a check is accomplished to determine if a bottle pressure problem exists. If so, the Collection/Storage Checkout module is executed prior to the Water Supplement Assembly Checkout module to assure the current CO_2 pressure level.

The detection of an error in the Flow Control Assembly associated with the regulator requires identification of the CO_2/CH_4 or H_2O line. In this instance, the Collection/Storage Assembly checkout module, or Water Supplement Assembly checkout module, is executed to insure an adequate supply of propellant. The Flow Control Assembly checkout module is then executed to isolate the problem.

The occurrence of a module manifold pressure problem in the Thrustor Assembly results in the execution of the Flow Control Assembly checkout module prior to executing the Thrustor Assembly checkout module. This assures the supply of propellants to the Thrustor Assembly. If the problem cannot be isolated, and the detected error was associated with the thrustor control valve, a final check is accomplished by executing the Power Distribution and Control Checkout module.

The occurrence of an error in the Power Distribution and Control Assembly results in execution of the checkout module for this assembly.

Errors detected in the majority of the high pressure system assemblies result in execution of the associated checkout module. In the instance of failure of the hig pressure manifold, propellant manifold, and the propellant storage assembly, the program must determine if the system is in the resupply or "ready to fire" configuration in order to select the proper sequence of module execution.

The sequence of module execution for the periodic testing is shown on sheets 4 and 5 of Figure 3-2.

3.1.2.2 Collection/Storage Assembly Checkout Module

This assembly takes the bio-waste gases (CO_2 and CH_4) from the EC&LS Subsystem and compression pumps them to the storage supply or to the flow control assembly. The gases may be stored separately or mix values can be used to combine them.

3.1.2.2.1 Source and Type of Input

The inputs associated with this module are the test points on the assembly.

3.1.2.2.2 Destination and Type of Outputs

The outputs from this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.2.3 Information Processing

The program module which assesses the status of this assembly meets the requirements for fault isolation, and both daily and tri-monthly periodic testing. This module assumes that the supply of CO_2 and CH_4 gases from the EC&LS subsystem has been verified prior to execution.

The general program flow checks the bottle, isolation valves, and propellant control valves. The fault isolation module tests only the lines (CO_2 and CH_4) in which an error was detected; but the periodic test checks all loops in both assemblies. The last components examined are the mix values. The periodic tests include all fault isolation sequences and additional tests in the area of valve control, trend analysis, and calibration.

The daily periodic test computes the level of gas in both storage bottles based upon temperature and pressure data. This information is then transferred to the data base for operational purposes.

This routine also uses the average of upstream and downstream pump flow rates for comparison with the average of the readings from the previous ten days. If the delta between these averages exceeds a predefined limit, the operator is notified.

The tri-monthly periodic check exercises both the propellant control and isolation valves. These valves are only exercised in the fault isolation and daily periodic test when a positional error is detected.

The tri-monthly logic computes calibration constants for the storage bottles, high and low pressure manifold temperature, and high and low manifold pressure. This data is also used to accomplish pump leak checks.

3.1.2.3 Power Distribution and Control Checkout Module

This assembly uses signals received from the EC&LS and GN&C subsystems to activate the thrustors through valve control.

3.1.2.3.1 Source and Types of Inputs

The inputs associated with this module are the test points which are part of the assembly, and valve position measurements in the thrustor and Collection/ Storage Assembly.

3.1.2.3.2 Destination and Type of Outputs

The outputs from this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.3.3 Information Processing

This program module meets both the fault isolation and tri-monthly periodic testing requirements for this assembly. This module assumes that both the thrustor valves and propellant flow control valves have been checked out. This requires that the collection/storage and water supplement assemblies be checked out prior to execution of this module.

The program checks the voltage and current supply to the thrustor assembly for the operation of the heaters. These values are then used to calculate a heater resistance value. If this value is not in limit, the power subsystem requires checking out. The program then checks the propellant flow control valves and the thrustor control valves.

The essential difference between fault isolation and the periodic test is that all values are exercised during periodic testing; while the fault isolation test is limited to a value where a failure is detected.

3.1.2.4 Flow Control Assembly Checkout Module

This assembly regulates and selects propellant flow. The signals which control both the regulator isolation valves and the crossfeed valves are received from the power distribution and control assembly.

3.1.2.4.1 Source and Types of Inputs

The inputs associated with this module are the test points which are located on the flow control assembly.

3.1.2.4.2 Destination and Type of Outputs

The outputs from this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.4.3 Information Processing

This program module meets both the fault isolation and periodic testing requirements for this assembly. The program sequentially checks the regulator isolation valves, the crossfeed valves, and the regulators. The significant difference between the periodic and the fault isolation test is that the periodic test examines all the items, while the fault isolation test examines only those elements directly associated with the detected error. In addition, tri-monthly periodic test exercises the valve control mechanisms. These mechanisms are not exercised by the fault isolation or daily periodic test, unless an error is detected.

Calibration constants are calculated relative to the crossfeed pressures and the regulators as a function of the tri-monthly periodic test.

3.1.2.5 Water Supplement Assembly Checkout Module

This assembly stores and provides the water for supplemental impulse. The water in this tank is pressurized by the use of CO_2 from the Collection/Storage Assembly.

3.1.2.5.1 Source and Type of Inputs

The inputs associated with this module are the test points associated with the water supplement assembly.

3.1.2.5.2 Destination and Type of Outputs

The outputs from this module are the normal operational messages indicating out-of-tolerance conditions, and the progress of testing.

3.1.2.5.3 Information Processing

This program module meets both the fault isolation and periodic testing requirements for this assembly.

The pressure inlet and outlet valves, tank isolation valve, and the flow control valve are checked for proper position. If the tri-monthly periodic test is executed, or an error is detected, the valve control mechanism is exercised. The module then proceeds to sequentially examine the bottle temperature, vaporizor temperature, and bottle pressure. The final step performed by this module is the check of the fill/drain valve.

The program module then computes calibration constants relative to the storage bottle and H_2O vaporizor as a function of the tri-monthly periodic test.

3.1.2.6 Low Thrust Thrustor Assembly Module

The thrustor assembly module consists of eight thrustor modules, each containing four thrustors. The thrustor operation is initiated by commands from the power control assembly which simultaneously opens thrustor valves, sets power level, and turns on the heater elements.

3.1.2.6.1 Source and Types of Inputs

The inputs associated with this module are the test points associated with the thrustor assembly.

3.1.2.6.2 Destination and Types of Outputs

The outputs from this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.6.3 Information Processing

This program module meets both the fault isolation and periodic testing requirements for this assembly.

The program sequentially examines the module isolation valves, the thrustor valves, and heater. The tri-monthly periodic test exercises the valve control mechanisms. These mechanisms are not exercised by the fault isolation or daily periodic test unless an error is detected.

The tri-monthly periodic check also computes constants related to the module manifold pressure.

3.1.2.7 High Pressure Assembly Module

This module is contained in the cargo module and is connected to the Space Station through interface lines in the resupply assembly. This assembly provides pressure to the pressure storage assembly.

3.1.2.7.1 Source and Types of Inputs

The inputs associated with this module are the test points which are part of the assembly.

3.1.2.7.2 Destination and Types of Outputs

The outputs from this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.7.3 Information Processing

This program module meets both the fault isolation and periodic testing requirements for this assembly. It is anticipated that periodic checkout of this assembly will be accomplished only during the period when the cargo module is docked to the space station.

This program module assures the correct valve position prior to closing the isolation valve related to the sphere in which an error has been detected. If either pressure or temperature are out of limits, the module identifies the failure as being associated with the measurement transducer, or the relief valves. In the event of a pressure build up in this assembly, which is not isolated through the transducer, it is assumed that the cause is due to a stoppage in the pressurant flow. During a resupply operation, the pressure limits are adjusted by the operational program as the pressurant transfers from the high pressure assembly to the pressure storage assembly. Consequently, a stoppage in the line would result in the pressure failing to reduce, and the limit dropping below the present pressure. When the periodic test is conducted, both spheres within each line are checked out. In addition, all valves are exercised to assure proper operation.

3.1.2.8 Low Pressure Assembly Module

This program module is used to check out the low pressure assembly. This assembly is used to resupply propellant to the propellant storage assembly.

3.1.2.8.1 Source and Types of Inputs

The inputs associated with this module are the test points which are part of the assembly.

3.1.2.8.2 Destination and Types of Ouputs

The outputs of this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.8.3 Information Processing

The checkout technique for both periodic testing and fault isolation of the low pressure assembly is identical to that which is required for the high pressure assembly; the only difference being the number of storage tanks and test points included with the assembly.

3.1.2.9 Resupply Assembly Module

This program module provides checkout of the resupply assembly. This assembly interfaces between the high pressure assembly and the pressure storage assembly, and the low pressure assembly and the propellant storage assembly.

3.1.2.9.1 Source and Types of Inputs

The inputs associated with this module are the test points which are part of the assembly.

3.1.2.9.2 Destination and Types of Outputs

The outputs of this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.9.3 Information Processing

This program module meets both the fault isolation and periodic testing requirements for this assembly. The module initially examines valve positions and if correct, the isolation valves for this assembly are closed. The pressure is then examined to isolate the problem to a stoppage of flow or a leakage condition. If pressure is within limits, the isolation valves are repositioned to their initial status.

In addition to those functions accomplished during fault isolation, the periodic test executed on a tri-monthly basis computes calibration constants for the pressure transducers in this assembly, and exercises the valves. This is not done during fault isolation, unless an error in the valve position is detected.

3.1.2.10 Pressure Storage Assembly

This assembly is used to store the pressurant which is used to feed the propellant into the thrustors.

Four storage spheres are included in this assembly. They are reloaded through the resupply assembly, or supply pressure to the high pressure manifold, depending on valve positions. The fault isolation and periodic testing which is required for this module is identical to that which is provided for the high pressure assembly. The only variations are the number of spheres and valves involved.

3.1.2.11 High Pressure Manifold Module

This assembly provides an interface for the high pressure lines between the pressure storage, purge supply, pressure control, and resupply assemblies.

3.1.2.11.1 Source and Types of Inputs

The inputs with this module are the test points which are associated with the assembly.

3.1.2.11.2 Destination and Types of Outputs

The outputs of this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.11.3 Information Processing

This program module meets both the fault isolation and periodic testing requirements for this assembly.

The basic program flow checks the propellant storage isolation valves. If the pressure measurement is not within tolerance, the program attempts to isolate this to the transducer, leaky valves, or a cause of increased pressure. The module then checks the isolation valve to the pressure control assembly to assure proper operation.

The program will check only the line in which the pressure failure was detected during fault isolation. The periodic test checks both lines and accomplishes trend analysis on the pressure measurement. The tri-monthly periodic test also computes calibration constants for the pressure measurement.

When this module is used for the purpose of fault isolation, the assembly which provides the upstream pressure will be examined initially. During a resupply operation, this would be the resupply assembly.

During firing, or while in ready to fire state, the pressure storage assembly would be checked out.

3.1.2.12 Pressure Control Assembly Module

This assembly regulates and controls the flow of high pressure gas to propellant storage assemblies.

3.1.2.12.1 Source and Types of Inputs

The inputs associated with this module are the test points which are associated with the assembly.

3.1.2.12.2 Destination and Types of Outputs

The outputs from this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.12.3 Information Processing

This program module meets both the fault isolation and periodic testing requirements for this assembly.

The general flow of this module involves checkout of the regulator, isolation valves, and the failure indicators available for both the high pressure and low pressure switches. The program checks the high pressure loop by isolating it and applying a test pressure from the purge supply assembly. While this is being accomplished, the downstream pressure is monitored to assure that the pressure switch is activated properly, and that downstream pressure is maintained at the proper level. The checkout of the low pressure loop is accomplished by venting the low pressure manifold creating a gradual pressure reduction. The downstream pressure is monitored to assure that the low pressure switch actuates the low pressure isolation valve at the proper pressure. The fault isolation routine will not examine the valve operation, nor will it test the loop regulator or pressure switches unless an error is indicated in this area. The tri-monthly periodic test does exercise the valves and will check both the high and low pressure loops. In addition, the pressure switches will be recalibrated on a tri-monthly basis.

3.1.2.13 Low Pressure Manifold Assembly Module

This program module was used to accomplish checkout of the low pressure manifold assembly. This assembly provides the interface between the pressure control assembly and the propellant storage assemblies.

3.1.2.13.1 Source and Types of Inputs

The inputs associated with this module are the test points which are part of the assembly.

3.1.2.13.2 Destination and Types of Outputs

The outputs of this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.13.3 Information Processing

This program module meets both the fault isolation and periodic testing requirements for this assembly. This program initially examines the valve positions and pressures within the manifolds. The daily periodic test performs trend analysis on the pressure transducer. The tri-monthly periodic test exercises the valves and computes calibration constants for the pressure measurement. The fault isolation test will not exercise the valves unless an error is detected within the valves.

3.1.2.14 Propellant Manifold Assembly Module

This program module accomplishes checkout of the propellant manifold assembly. During a resupply operation, this assembly provides interface to the propellant storage assembly. During the normal operational sequence, this assembly provides the interface between the propellant storage assembly and the thrustor modules.

3.1.2.14.1 Source and Types of Inputs

The inputs associated with this module are the test points which are part of the assembly.

3.1.2.14.2 Destination and Types of Outputs

The outputs of this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.14.3 Information Processing

This program module meets both the fault isolation and periodic testing requirements for this assembly. The program initially checks valve positions, If an error is detected during fault isolation, or if the tri-monthly periodic test is being executed, the valve's operation is examined. Subsequent to this, pressures are tolerance checked, and attempts made to isolate the problem if out-of-limit conditions are detected. Trend analysis on the pressure measurements is accomplished as a function of the daily periodic checkout. The tri-monthly periodic check also involves the computation of calibration constants for the pressure measurements.

3.1.2.15 Propellant Storage Assembly Module

This program module provides the checkout required to assure proper operation of the propellant storage assembly. This assembly provides propellant to the thrustor module by interfacing with the propellant manifold assembly. Pressurant, provided by interface with the low pressure manifold assembly, forces the propellant into the thrustor.

3.1.2.15.1 Source and Types of Inputs

The inputs associated with this module are the test points which are part of the assembly.

3.1.2.15.2 Destination and Types of Outputs

The outputs of this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.15.3 Information Processing

This program module meets both the fault isolation and periodic testing requirements for the assembly. The program initially accomplishes the valve check to make sure the valves are positioned properly for the current operational configuration. Any discrepancies during the fault isolation will result in checkout of valve operation. The tri-monthly periodic test always checks the valve for operation regardless of error status. The program then assures that the fuel level is consistent with the trend data. If not, replacement of the quantity gauging assembly is recommended. The program then examines the status of the relief valve and checks the pressure within the storage tank. The daily periodic test accomplishes trend analysis for the pressure, temperature, and quantity gauging LRU's. The tri-monthly test exercises the valves and provides calibration constants for the pressure and temperature transducers, and the quantity gauging assemblies.

3.2.1.16 High Thrust Thrustor Module

This assembly consists of 36 thrustors divided into forward and aft sections, each containing two modules. The aft modules contain ten thrustors each the forward modules contain eight thrustors each. Isolation valves are incorporated in this assembly to provide the capability to isolate the modules and thrustors.

3.1.2.16.1 Source and Types of Inputs

The inputs associated with this module are the test points which are part of the assembly.

3.1.2.16.2 Destination and Types of Outputs

The outputs of this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.16.3 Information Processing

This program module meets both the fault isolation and periodic testing requirements for this assembly. This module sequentially checks the section (fore, aft), and module isolation valves. The module power and pressure are then examined. A pressure check within each group of thrustors is then accomplished. The operation of the thrustor control valves is then examined. This is accomplished by issuing a stimulus and measuring the chamber pressure response. Daily periodic testing also involves trend analysis which must be accomplished by line pressure, chamber pressure, and the chamber temperature. The tri-monthly periodic test additionally computes calibration constants related to line pressure, chamber pressure and temperature, and verifies valve operation.

3.1.2.17 Purge Supply Assembly

This assembly is used to provide a pressurant supply and control purging of the pressure manifolds and storage tanks.

3.1.2.17.1 Source and Types of Inputs

The inputs associated with this module are the test points associated with this assembly.

3.1.2.17.2 Destination and Types of Outputs

The outputs associated with this module are the normal operational messages indicating out-of-tolerance situations, and the progress of testing.

3.1.2.17.3 Information Processing

This module meets both the fault isolation and periodic testing requirements for this assembly.

The program sequentially examines the isolation valves and the relief valves. If a relief valve is found open, the problem is isolated to an increase of pressure, or failure of the pressure transducer or relief valve. The low pressure and high pressure lines are than examined. Trend analysis is accomplished by the daily periodic test and calibration constants computed by the tri-monthly periodic test.

3.1.2.18 Termination Routine

This module is included to assure the orderly return of control to the executive at the completion of execution.

3.1.2.18.1 Source and Types of Inputs

The inputs to this module indicate a successful or unsuccessful completion of program execution. In the event termination is required due to operator intervention, or a detected failure, additional inputs as to the status of the associated components are required.

3.1.2.18.2 Destination and Type of Outputs

None.

3.1.2.18.3 Information Processing

This module must accomplish the functions required to assure the subsystem is properly configured before control is returned to the executive.

3.1.3 PROGRAM FLOWCHARTS

Program flowcharts are provided in this section.

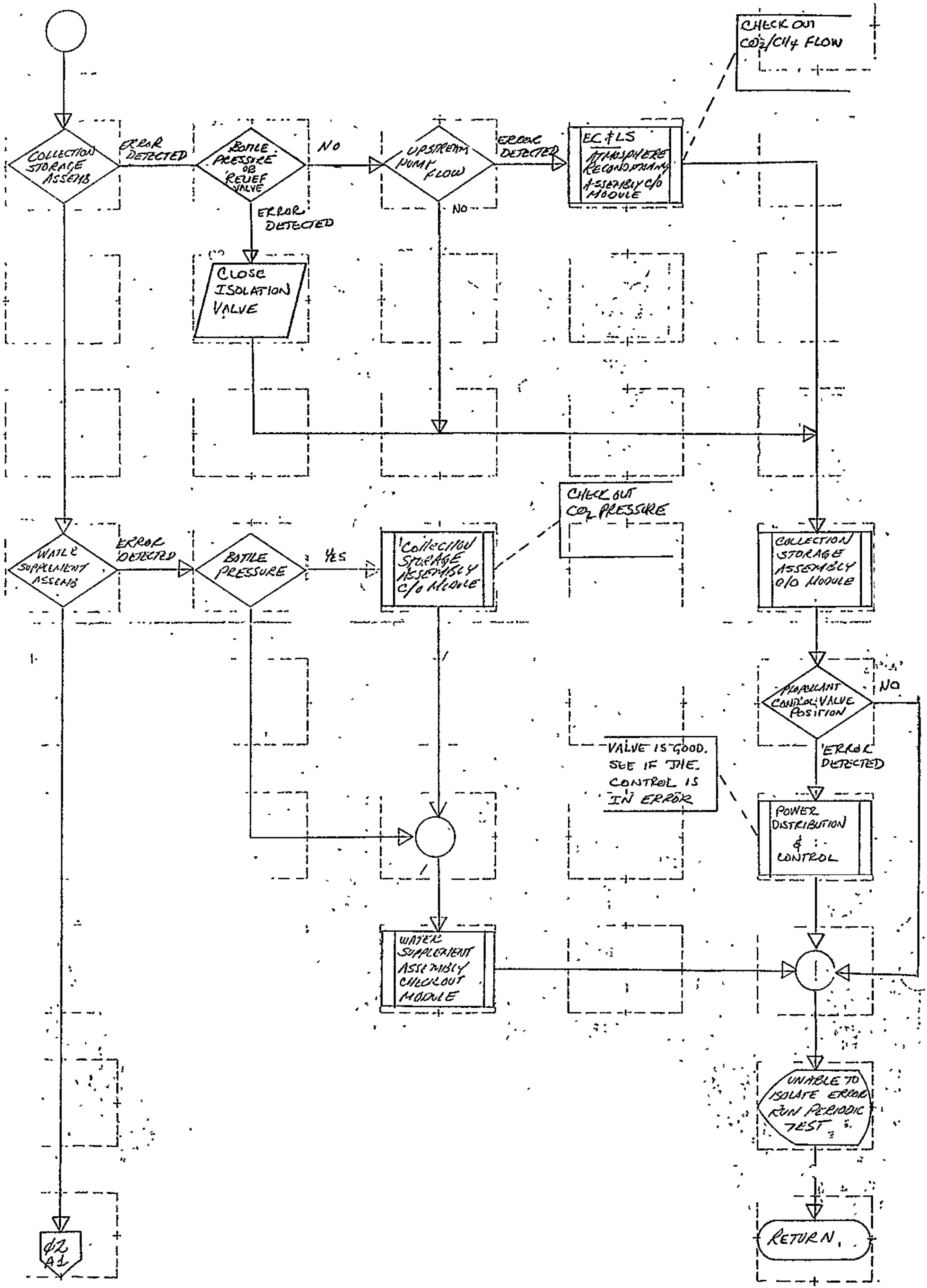


Figure 3-2. Sequence Logic Module (Sheet 1 of 5)

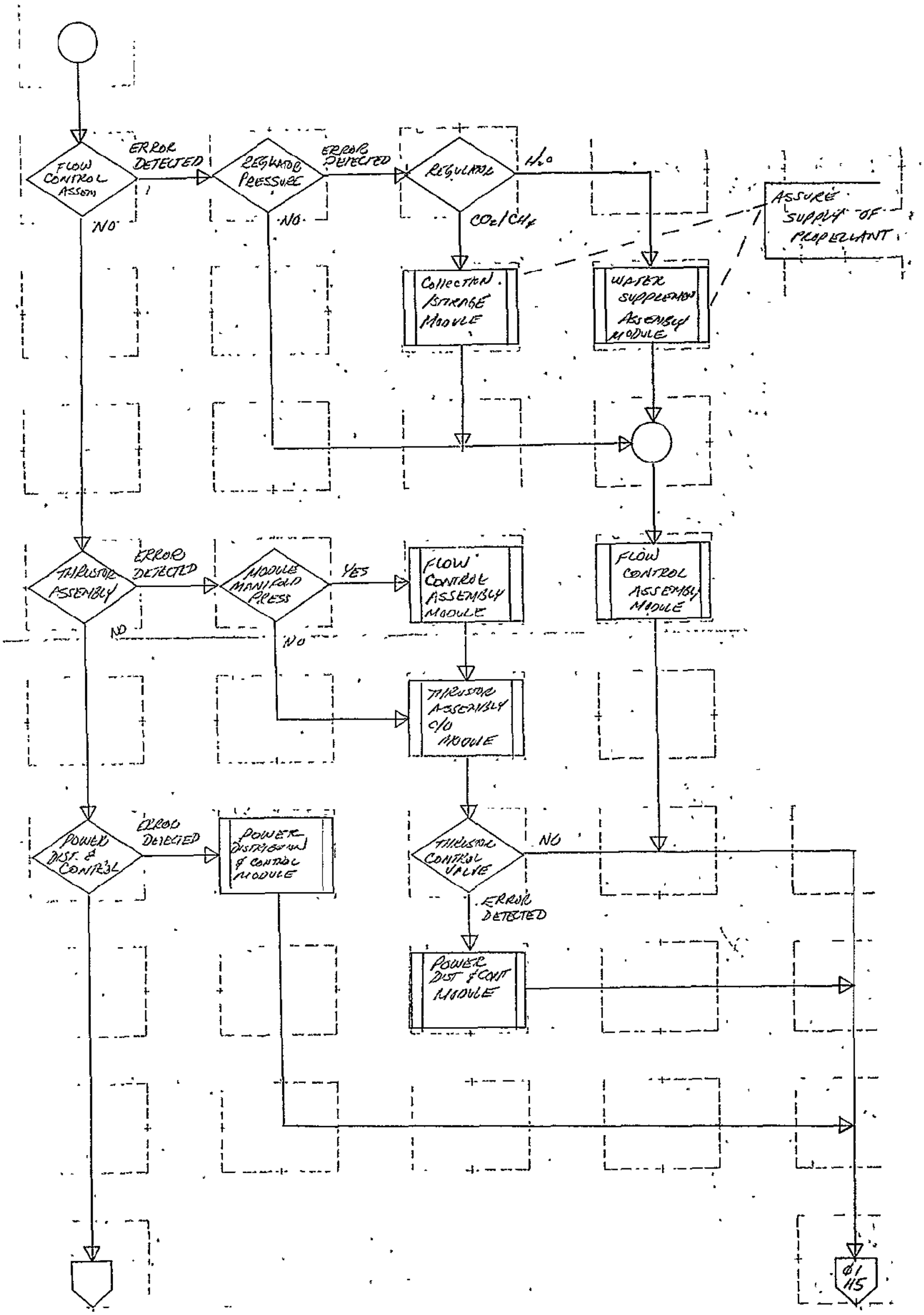


Figure 3-2. Sequence Logic Module (Sheet 2 of 5)

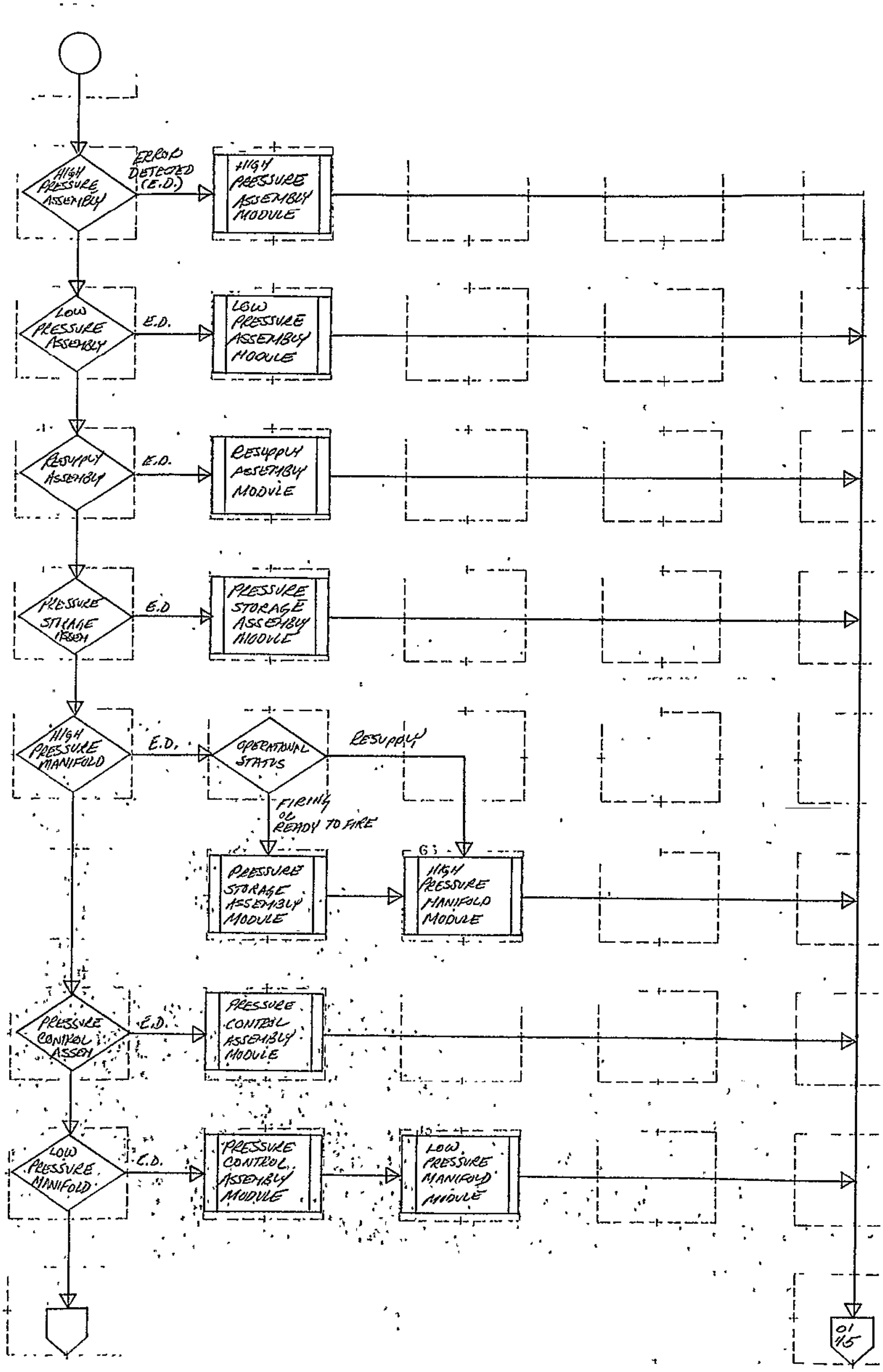


Figure 3-2. Sequence Logic Module (Sheet 3 of 5)

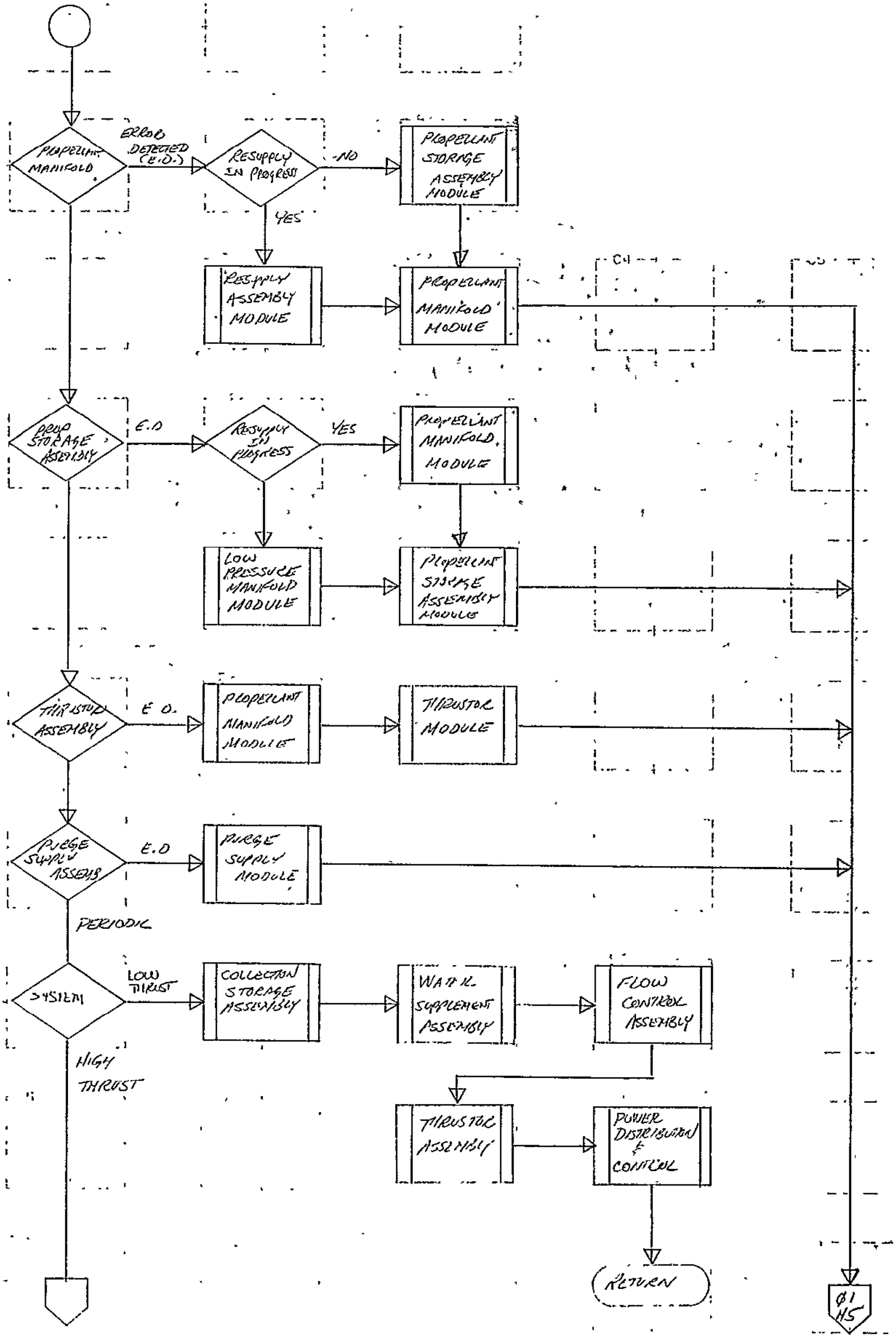


Figure 3-2. Sequence Logic Module (Sheet 4 of 5)

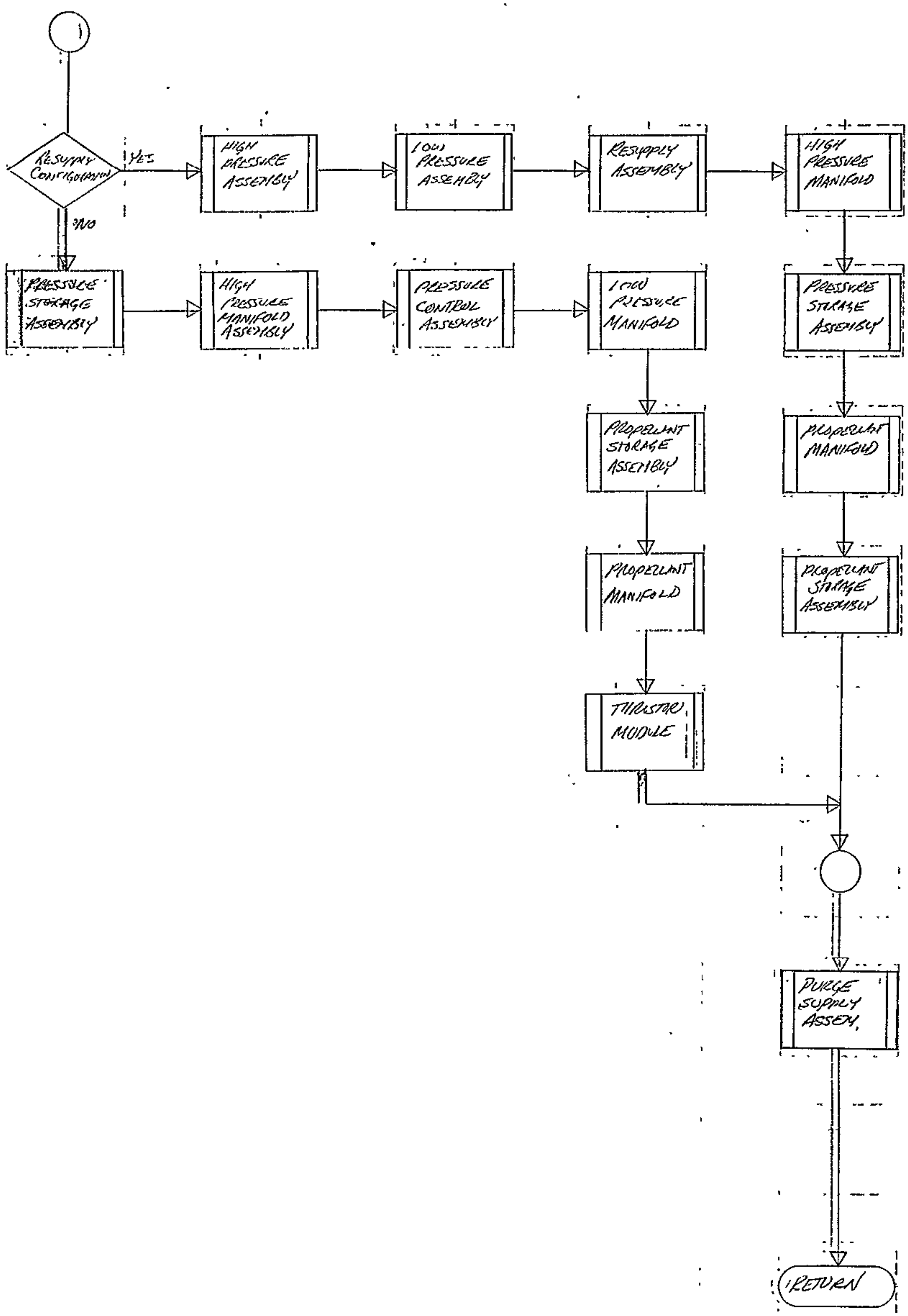


Figure 3-2. Sequence Logic Module (Sheet 5 of 5)

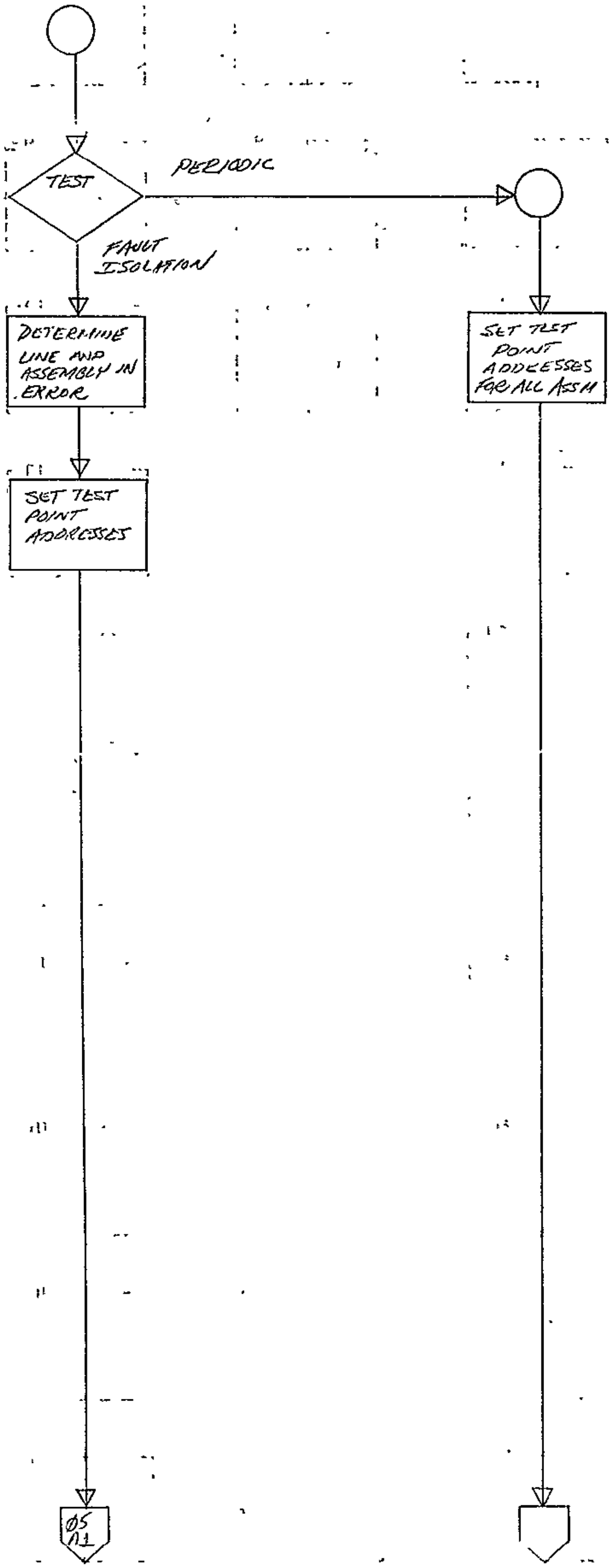


Figure 3-3. Collection/Storage Assembly Checkout Module (Sheet 1 of 5)

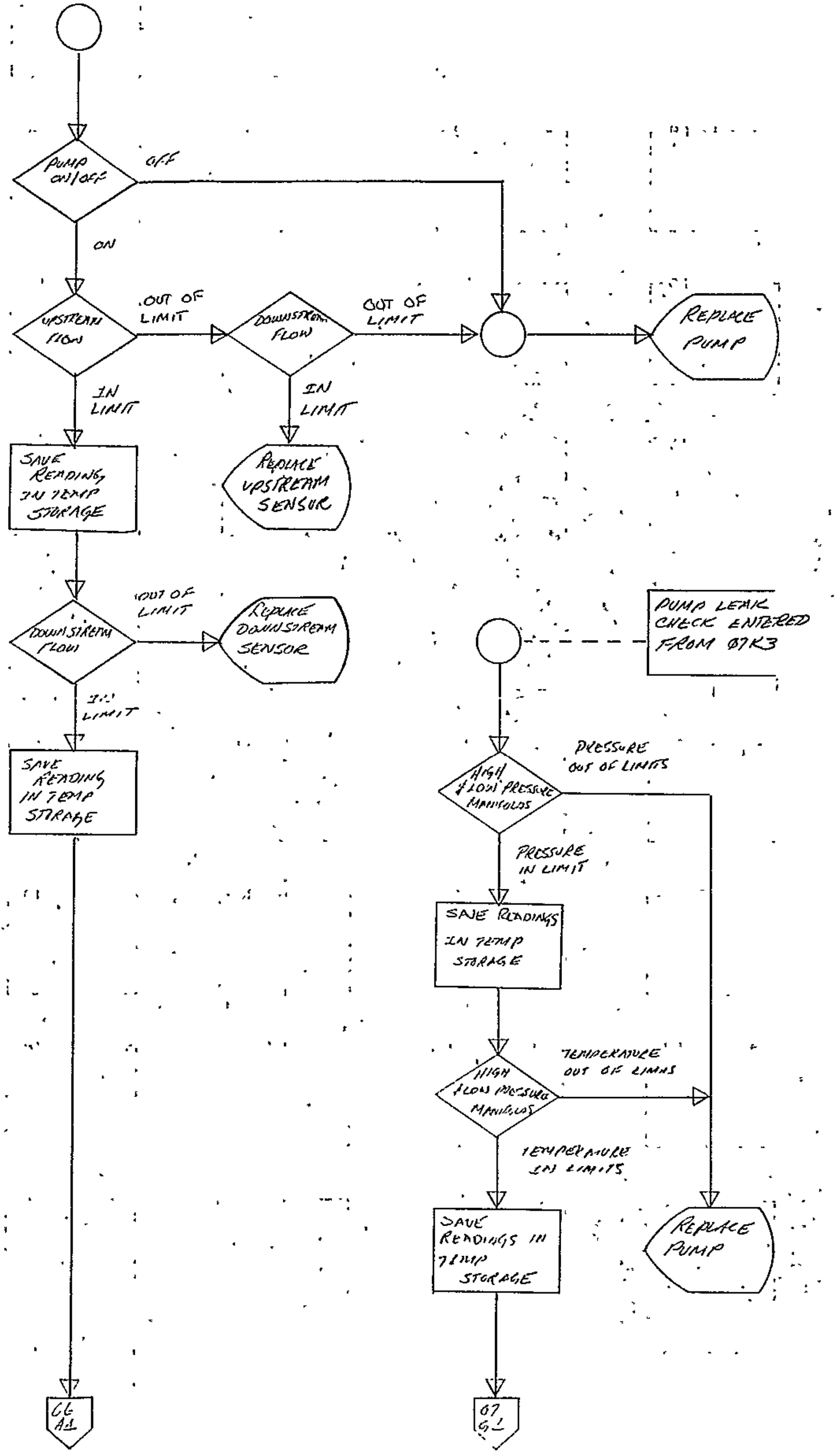


Figure 3-3. Collection/Storage Assembly Checkout Module
(Sheet 2 of 5)

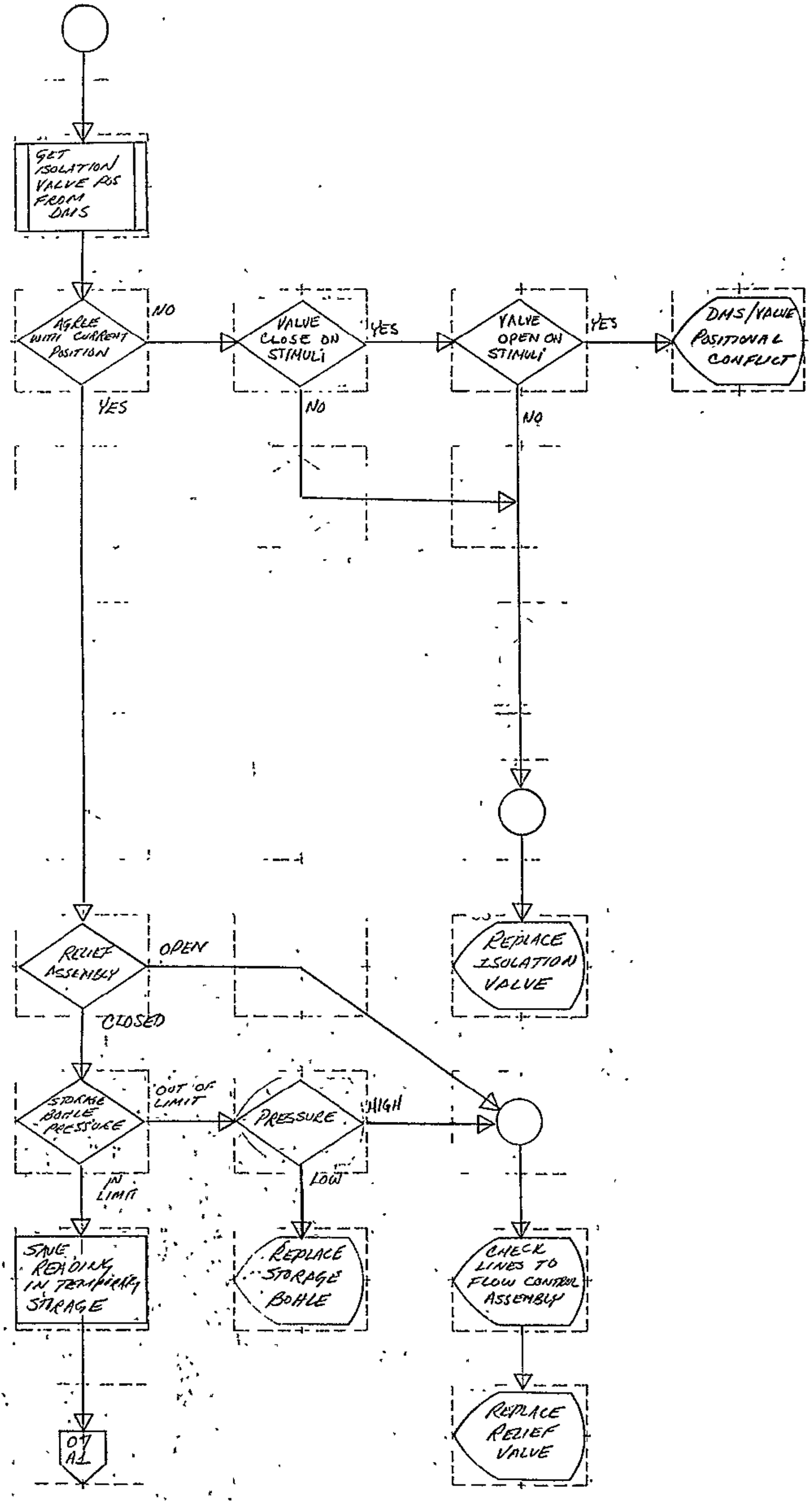


Figure 3-3. Collection/Storage Checkout Module
(Sheet 3 of 5)

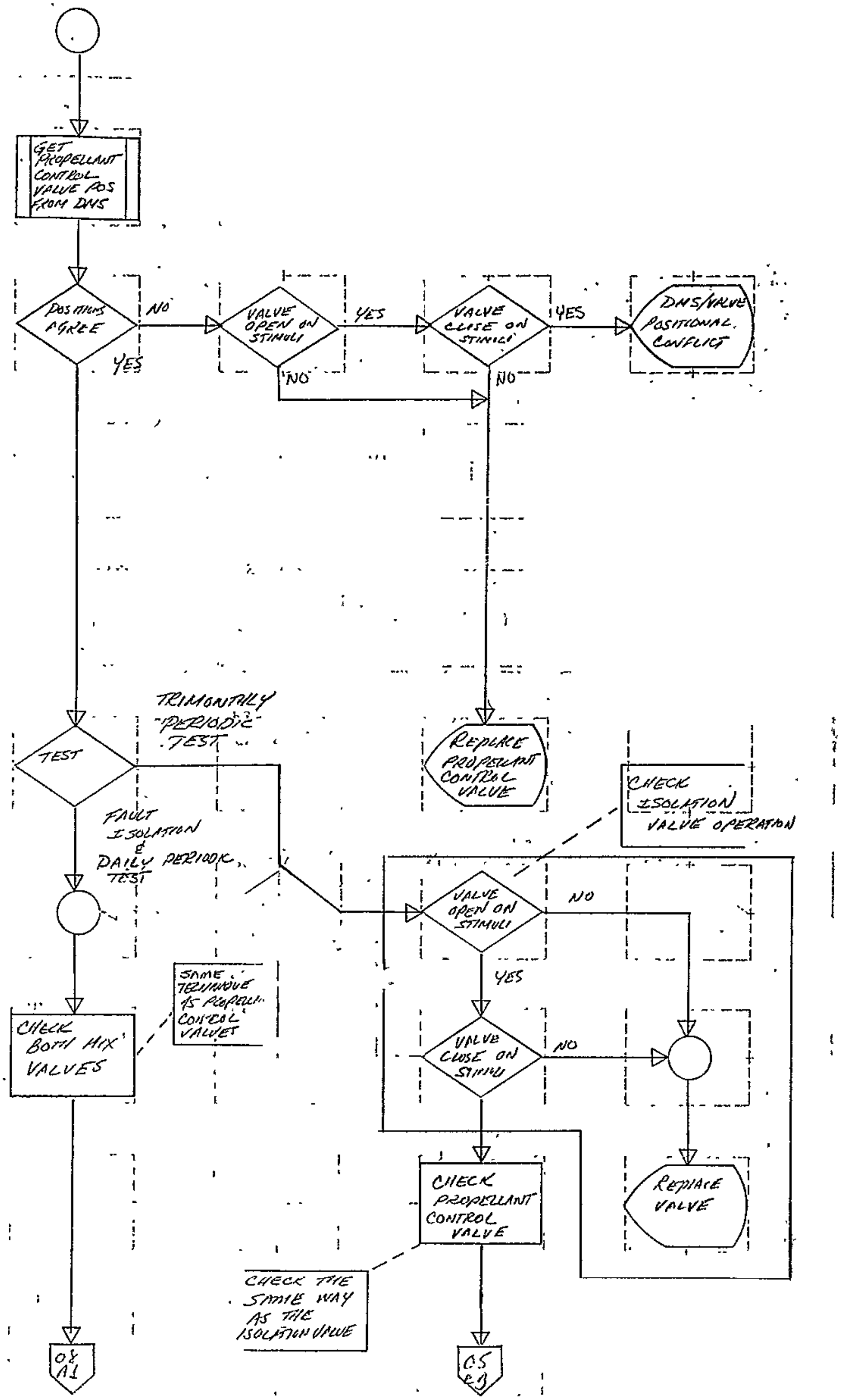


Figure 3-3. Collection/Storage Checkout Module
(Sheet 4 of 5)

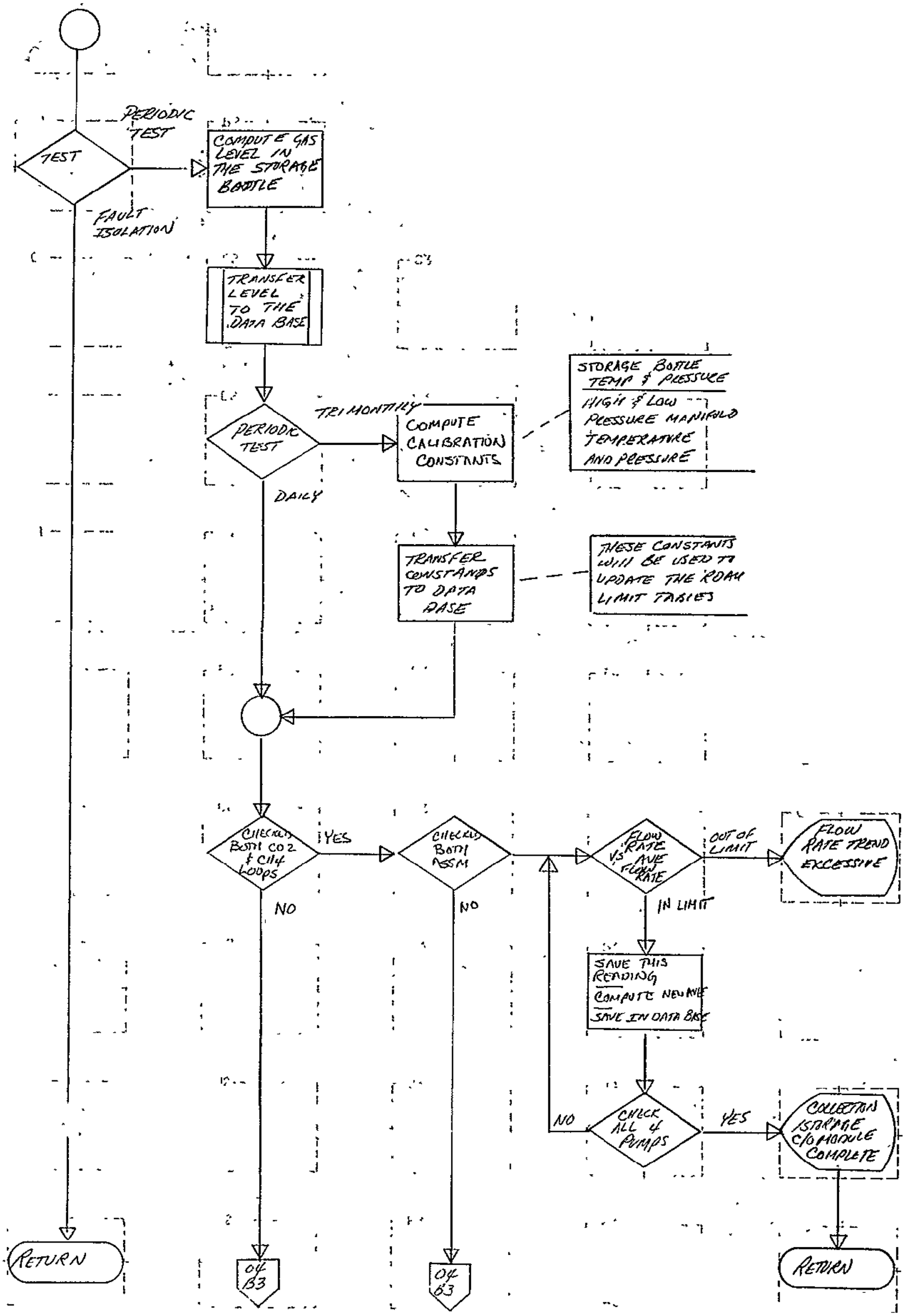


Figure 3-3. Collection/Storage Checkout Module
(Sheet 5 of 5)

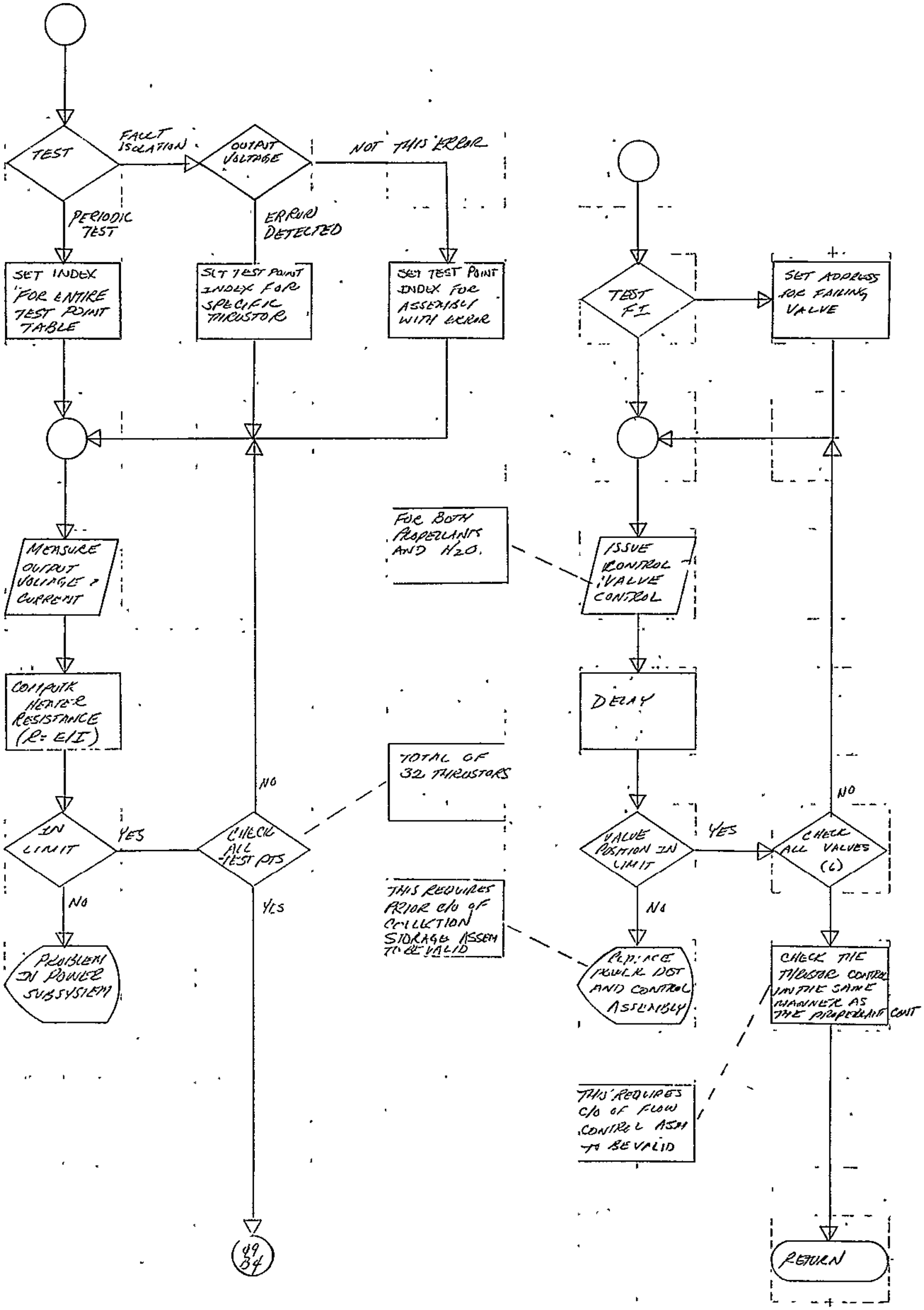


Figure 3-4. Power Distribution and Control Checkout Module

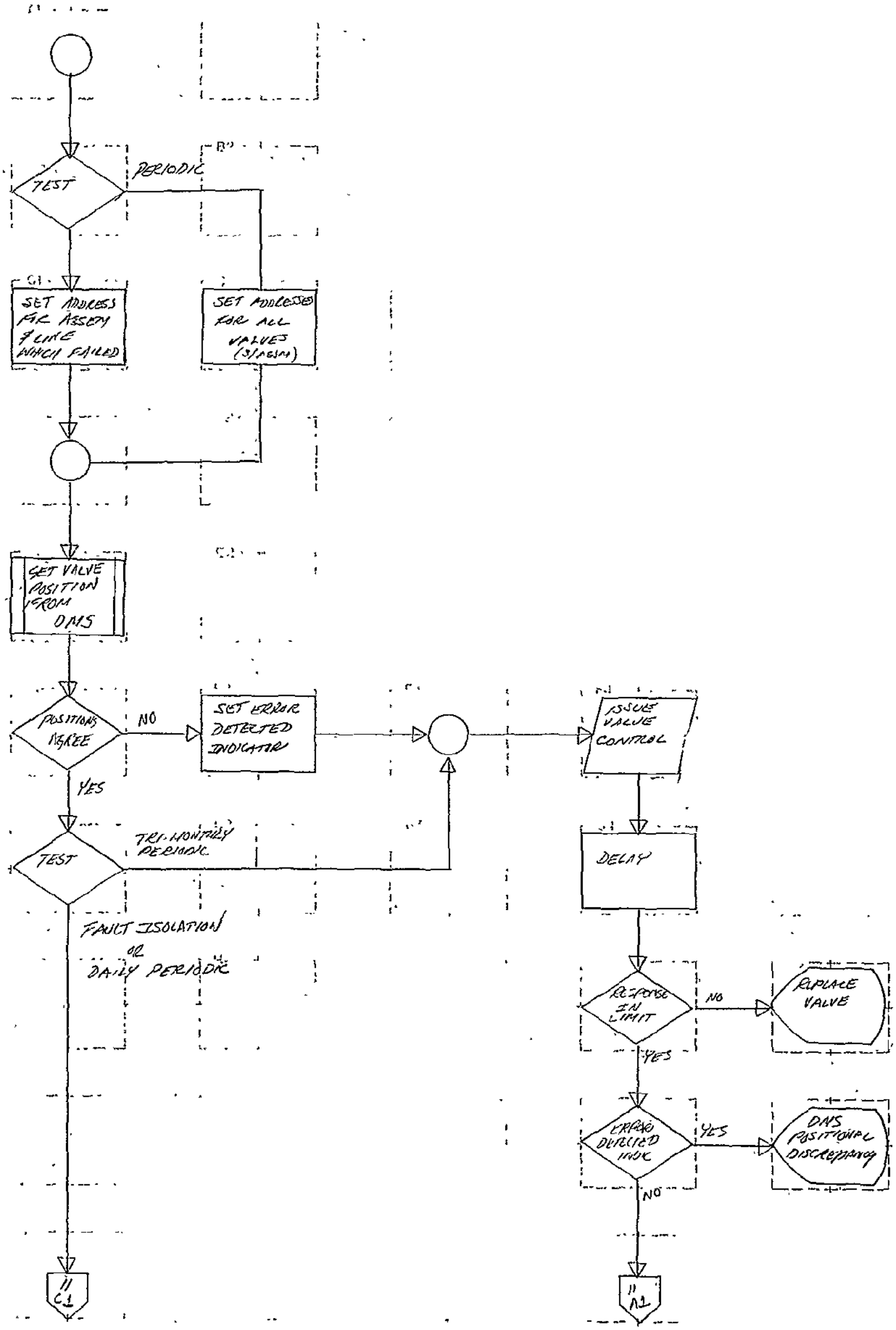


Figure 3-5. Flow Control Assembly (Sheet 1 of 2)

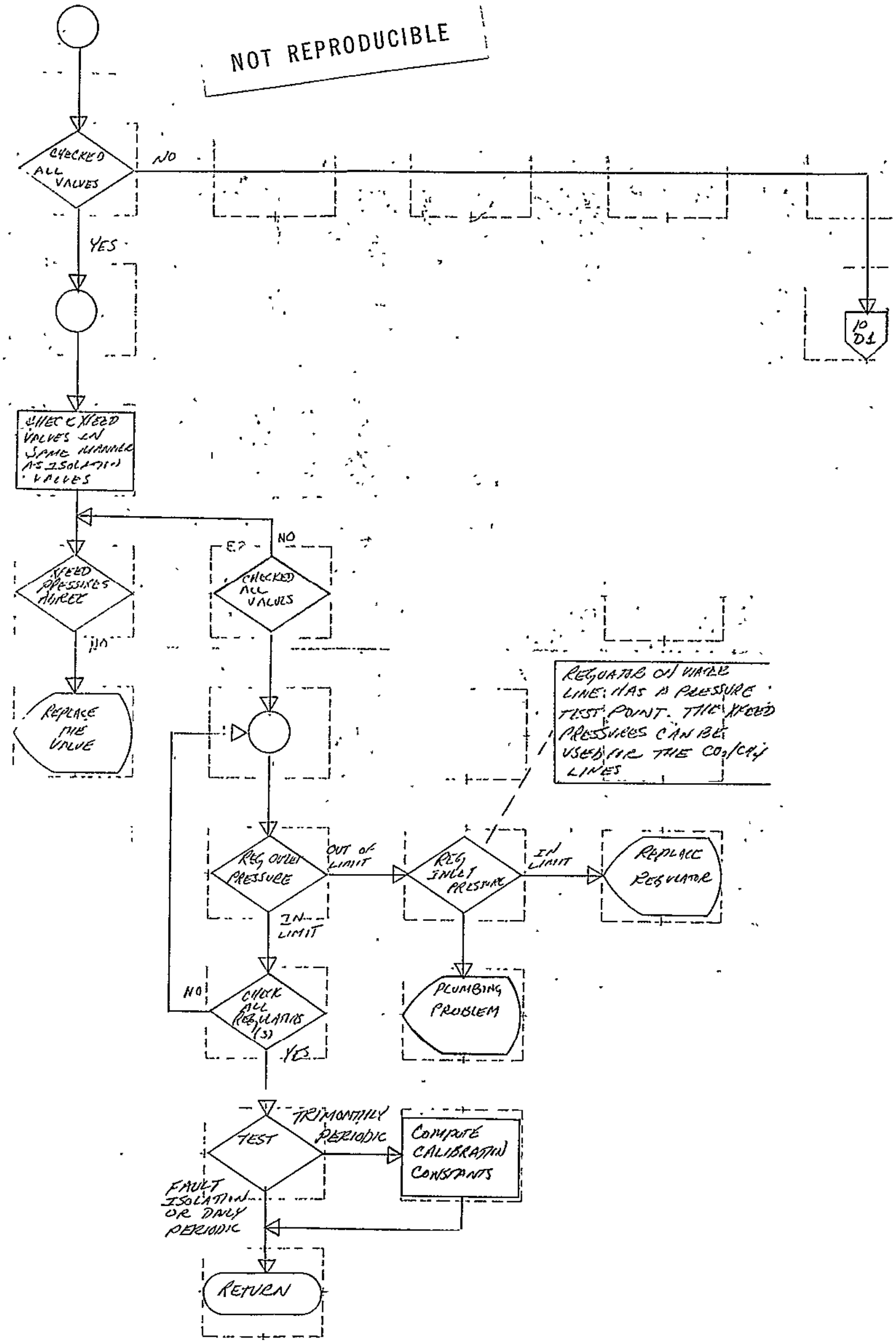


Figure 3-5. Flow Control Assembly (Sheet 2

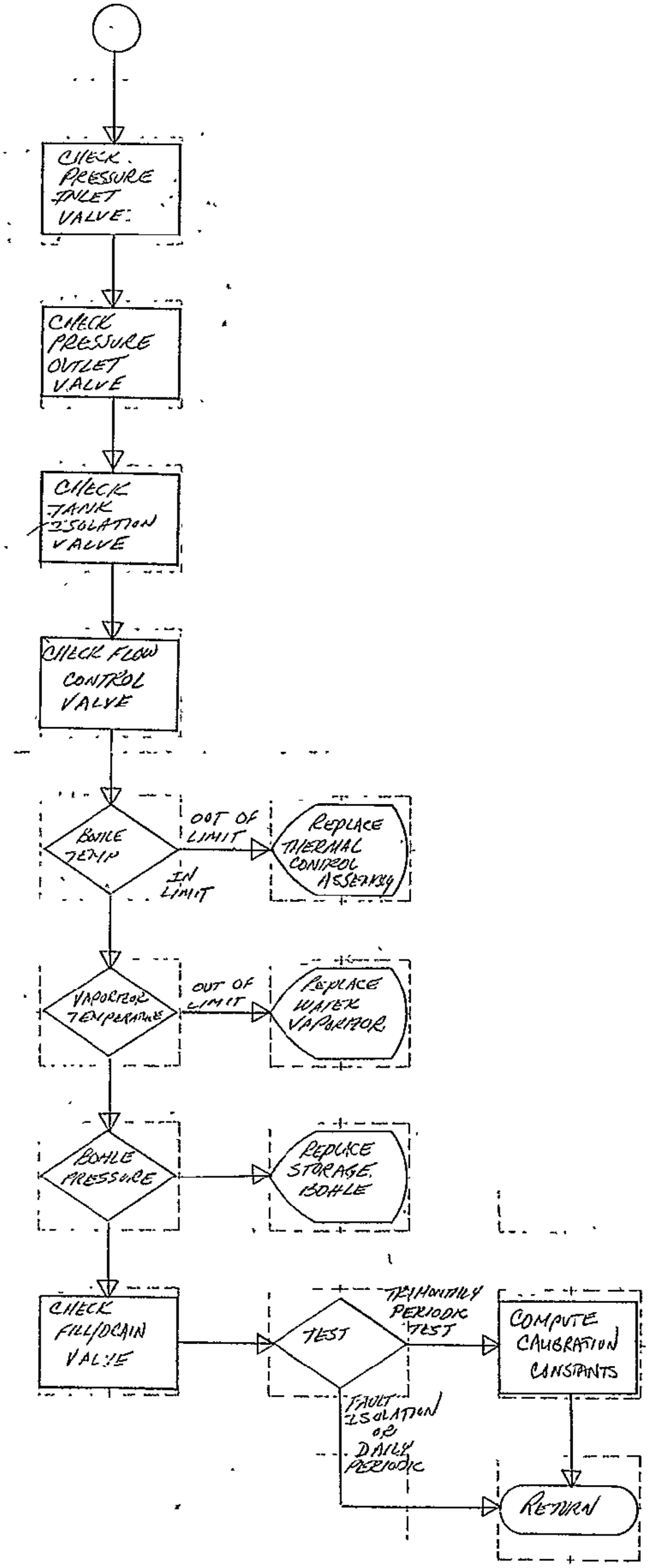


Figure 3-6. Water Supplement Assembly

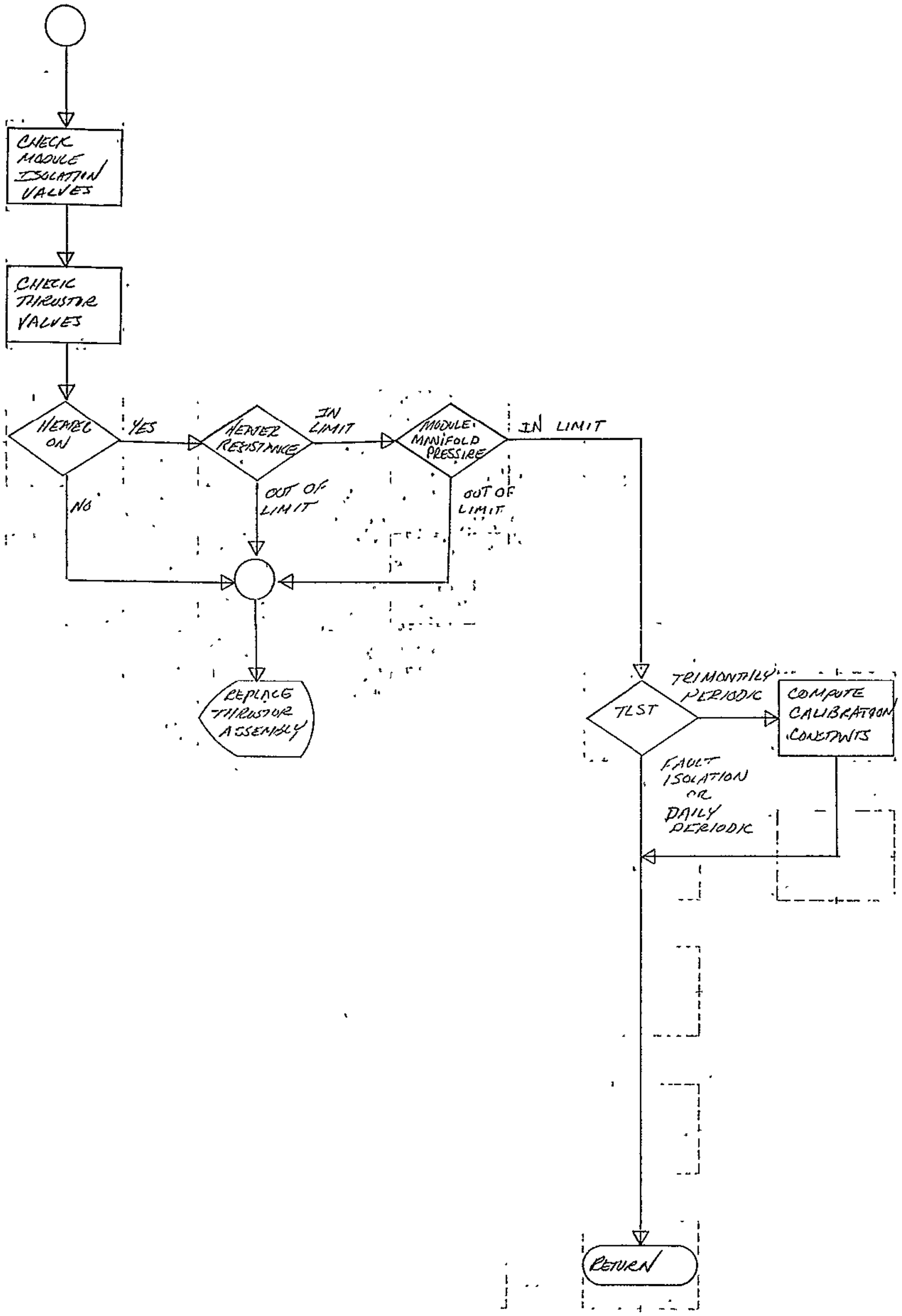


Figure 3-7. Thrustor Assembly Module

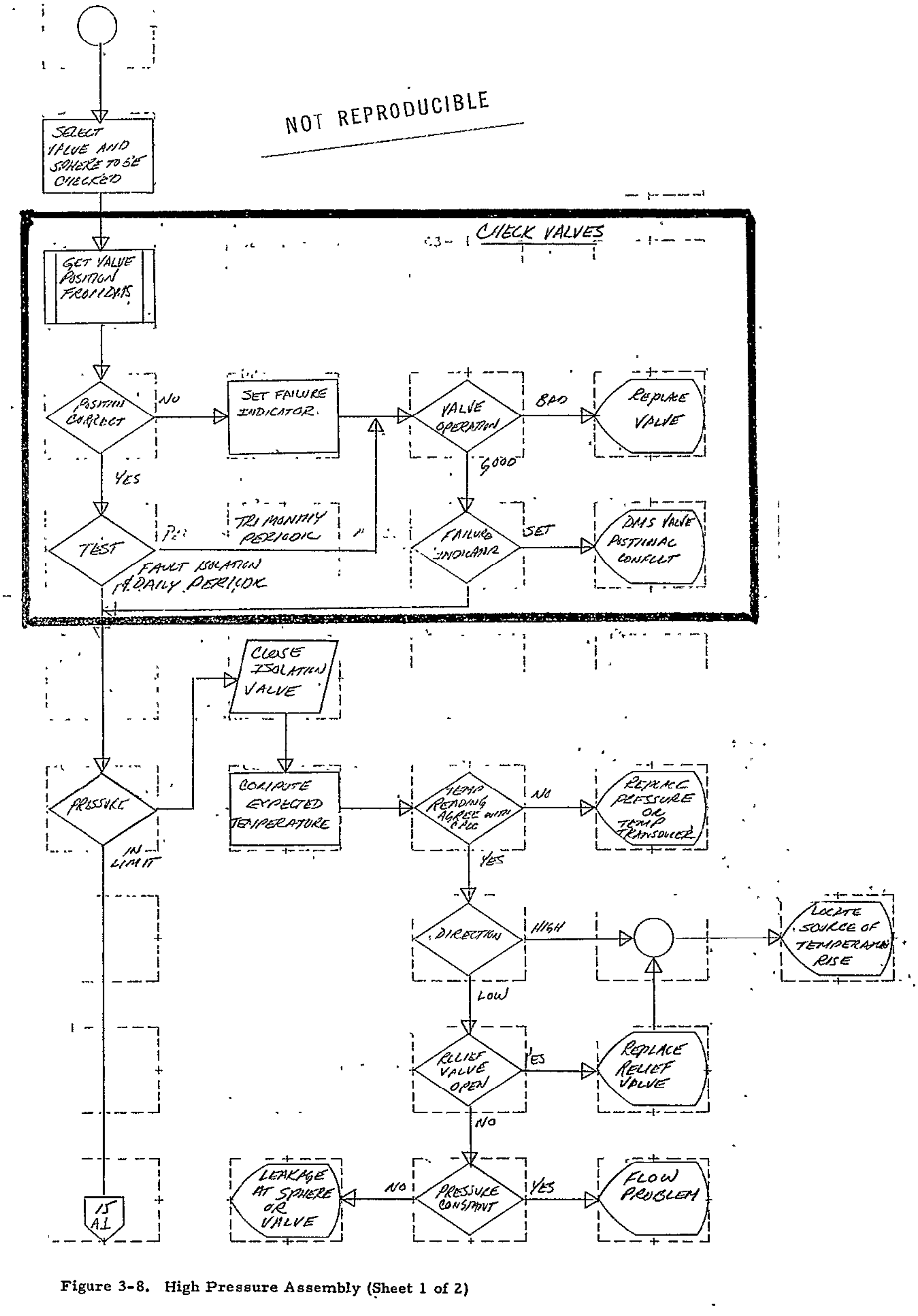


Figure 3-8. High Pressure Assembly (Sheet 1 of 2)

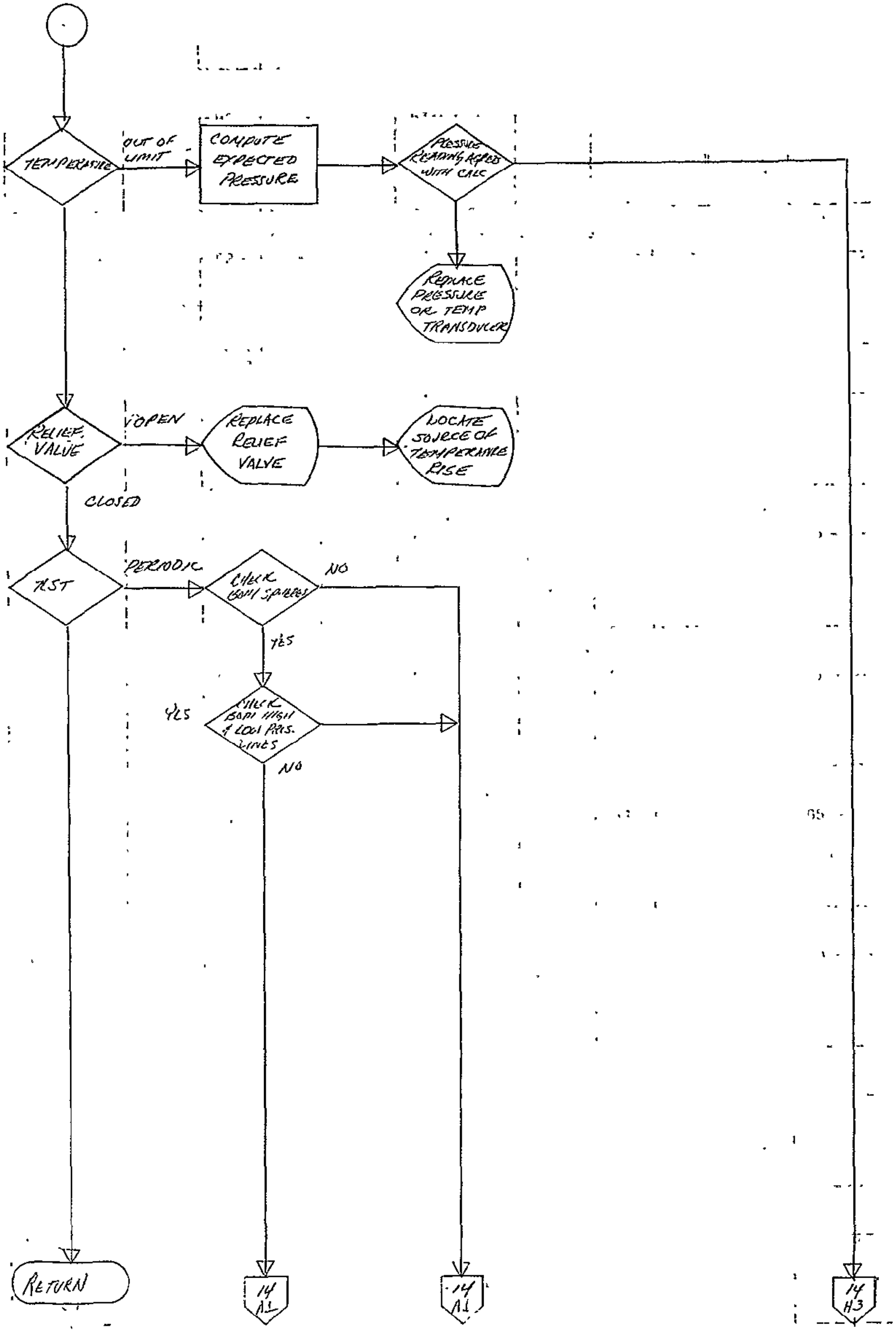


Figure 3-8. High Pressure Assembly (Sheet 2 of 2)

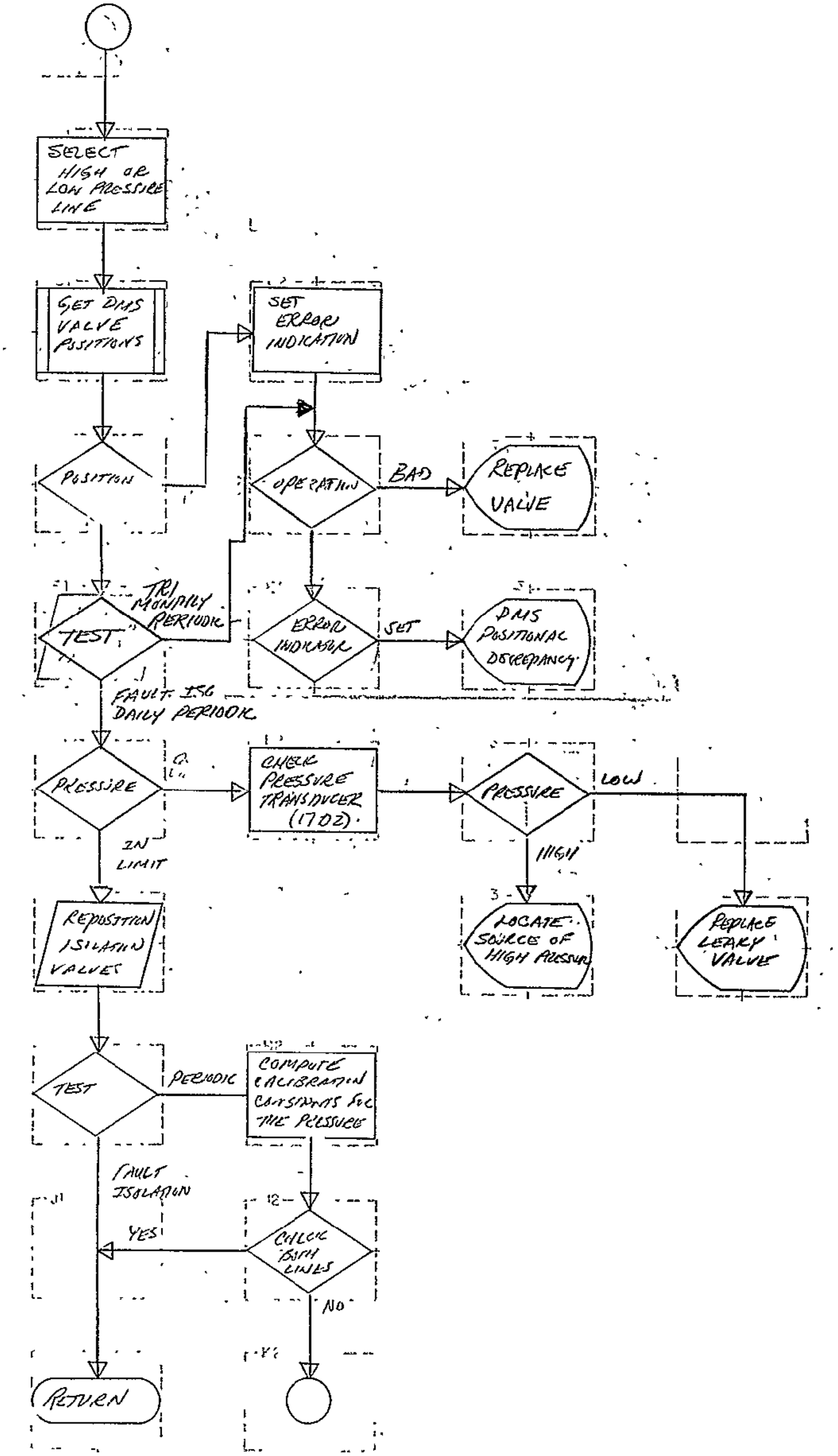


Figure 3-9. Resupply Assembly Module

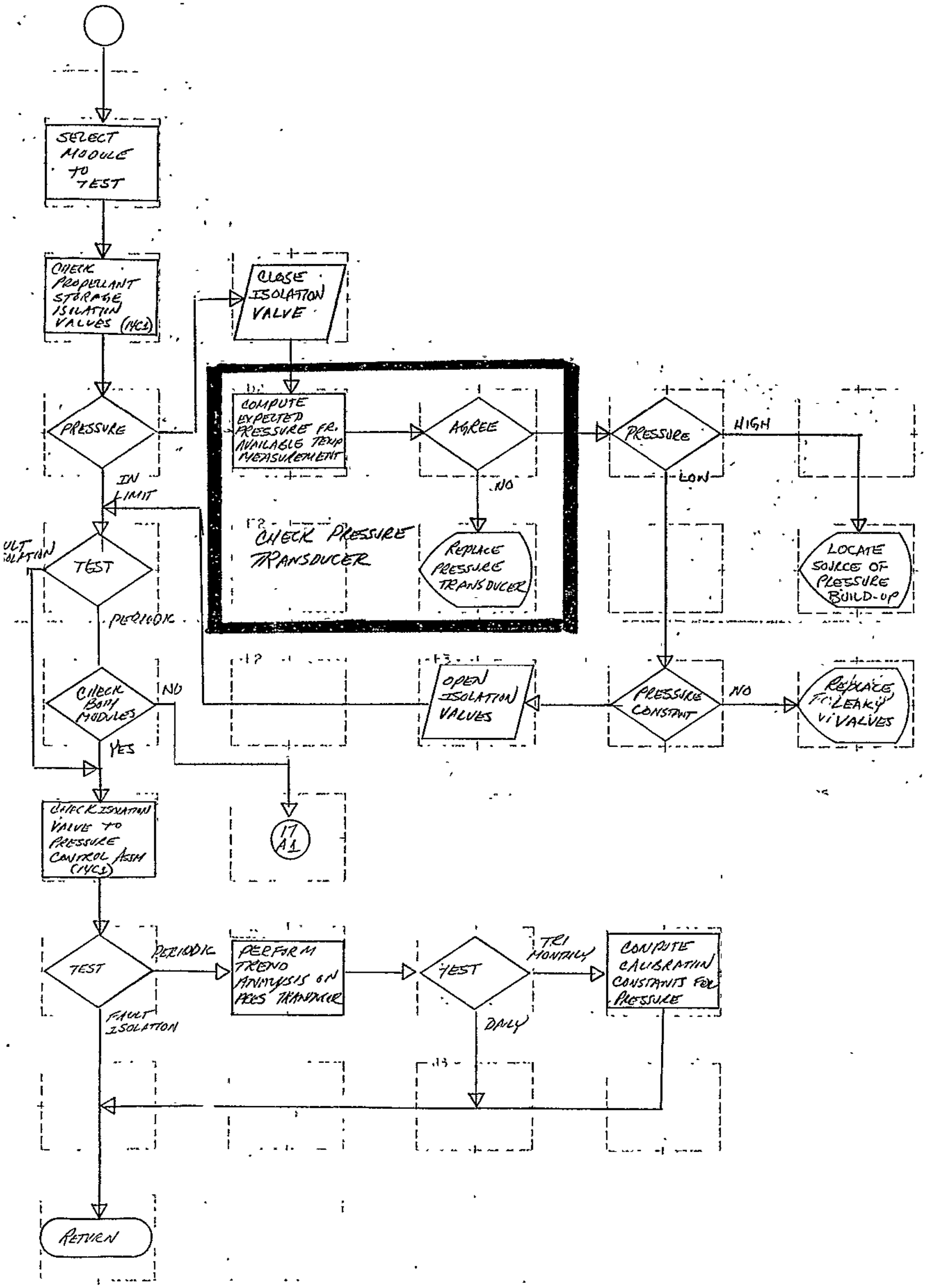
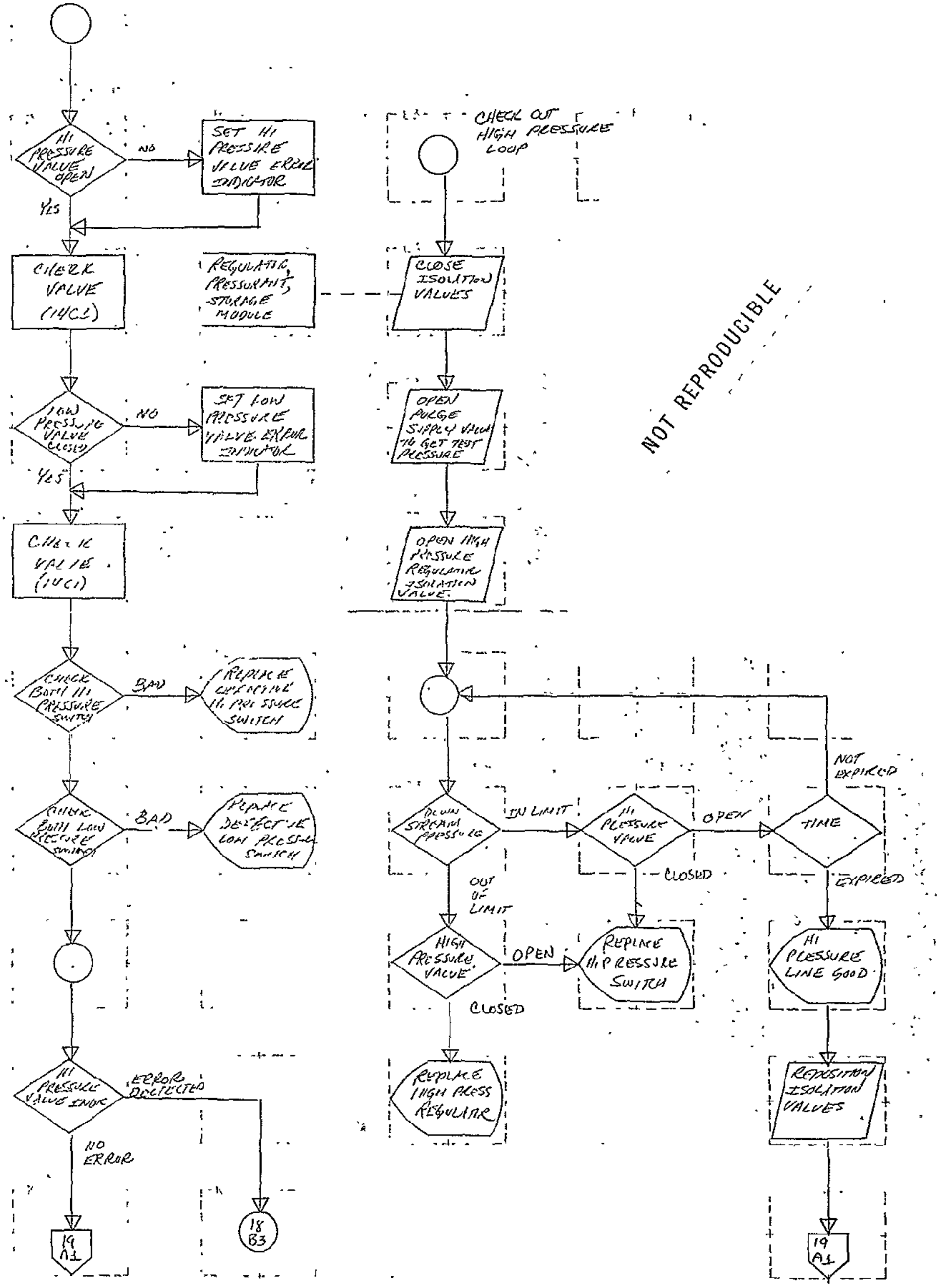


Figure 3-10. High Pressure Manifold Module



NOT REPRODUCIBLE

Figure 3-11. Pressure Control Assembly Module (Sheet 1 of 2)

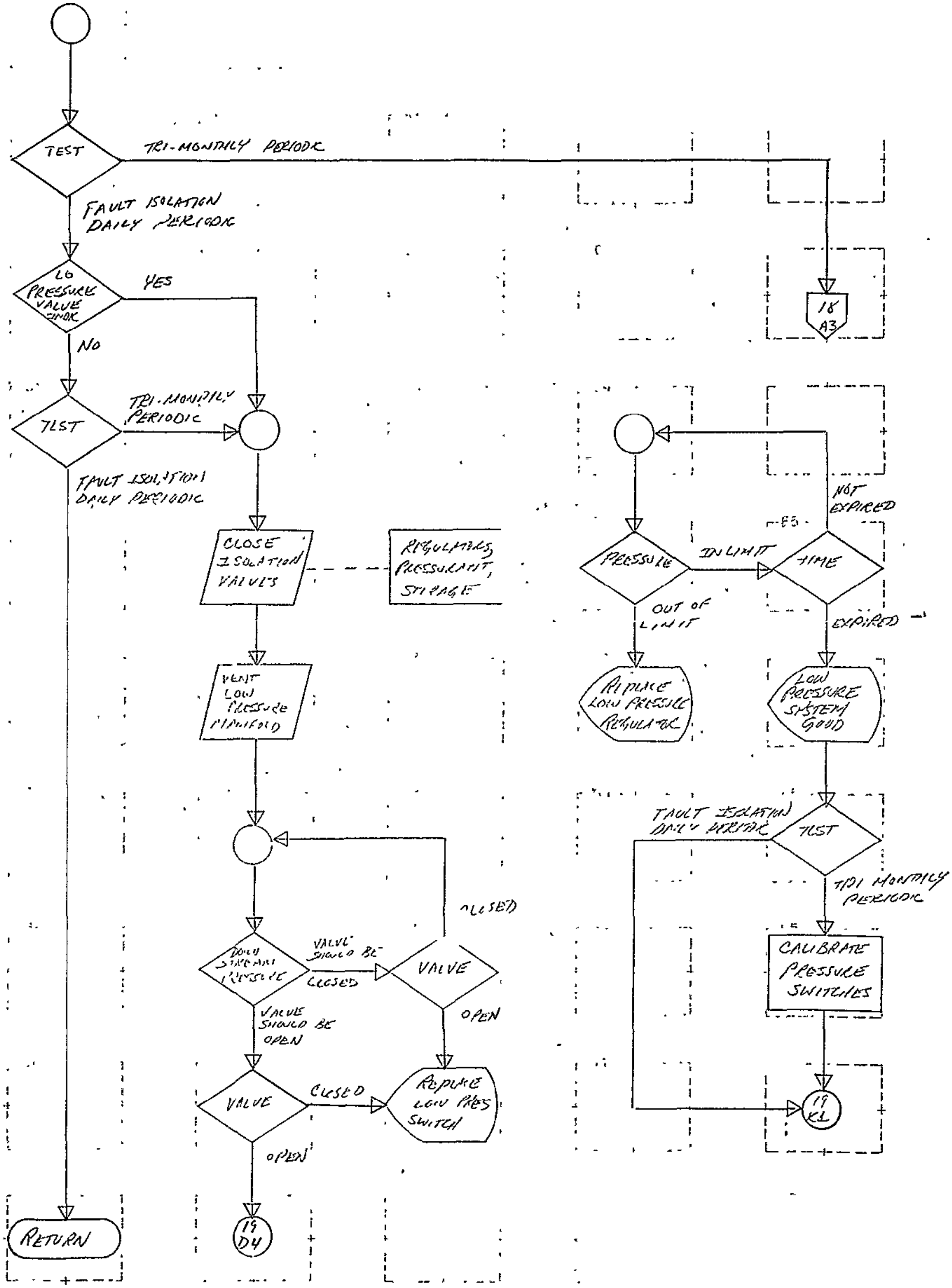


Figure 3-11. Pressure Control Assembly Module (Sheet 2 of 2)

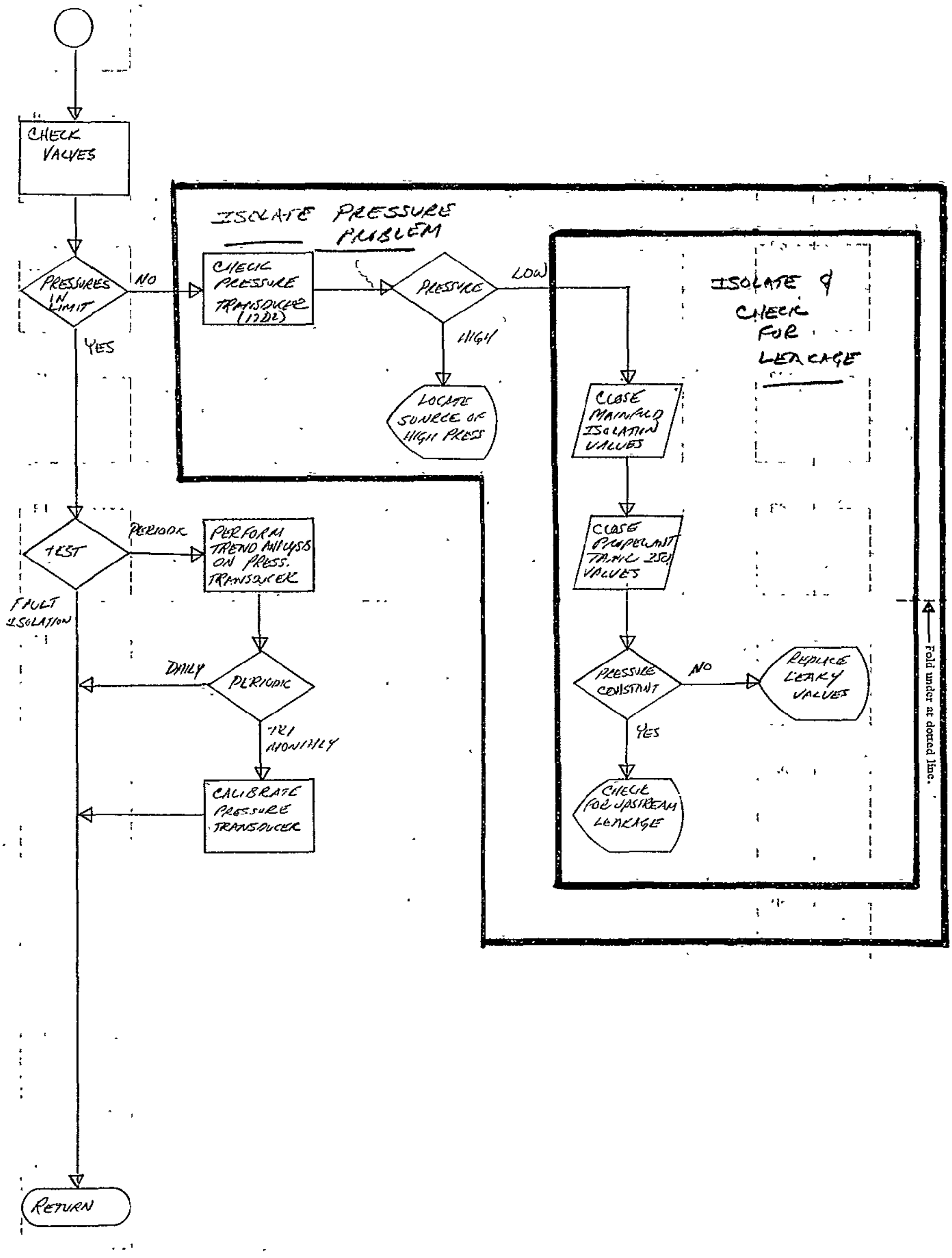


Figure 3-12. Low Pressure Manifold Assembly

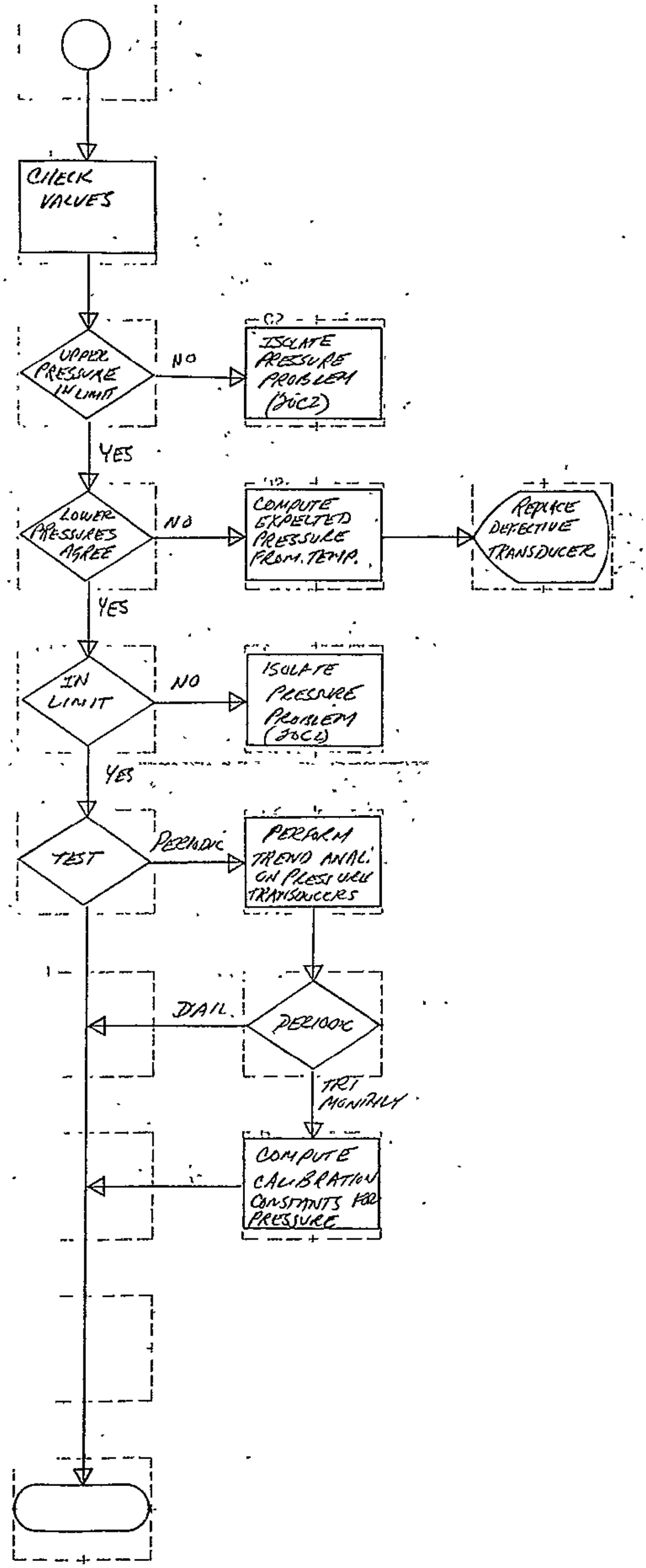


Figure 3-13. Propellant Manifold Assembly Module

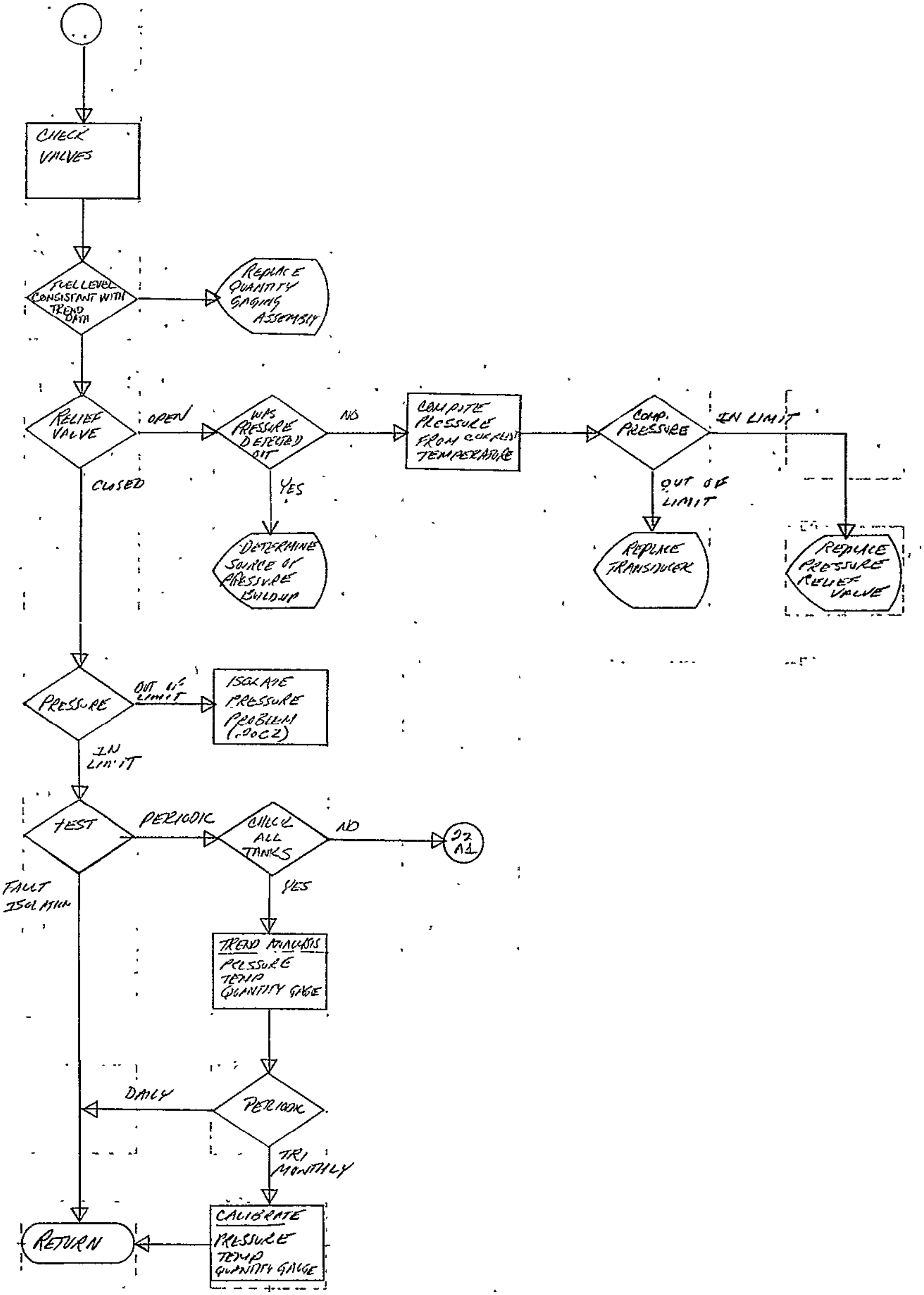


Figure 3-14. Propellant Storage Assembly

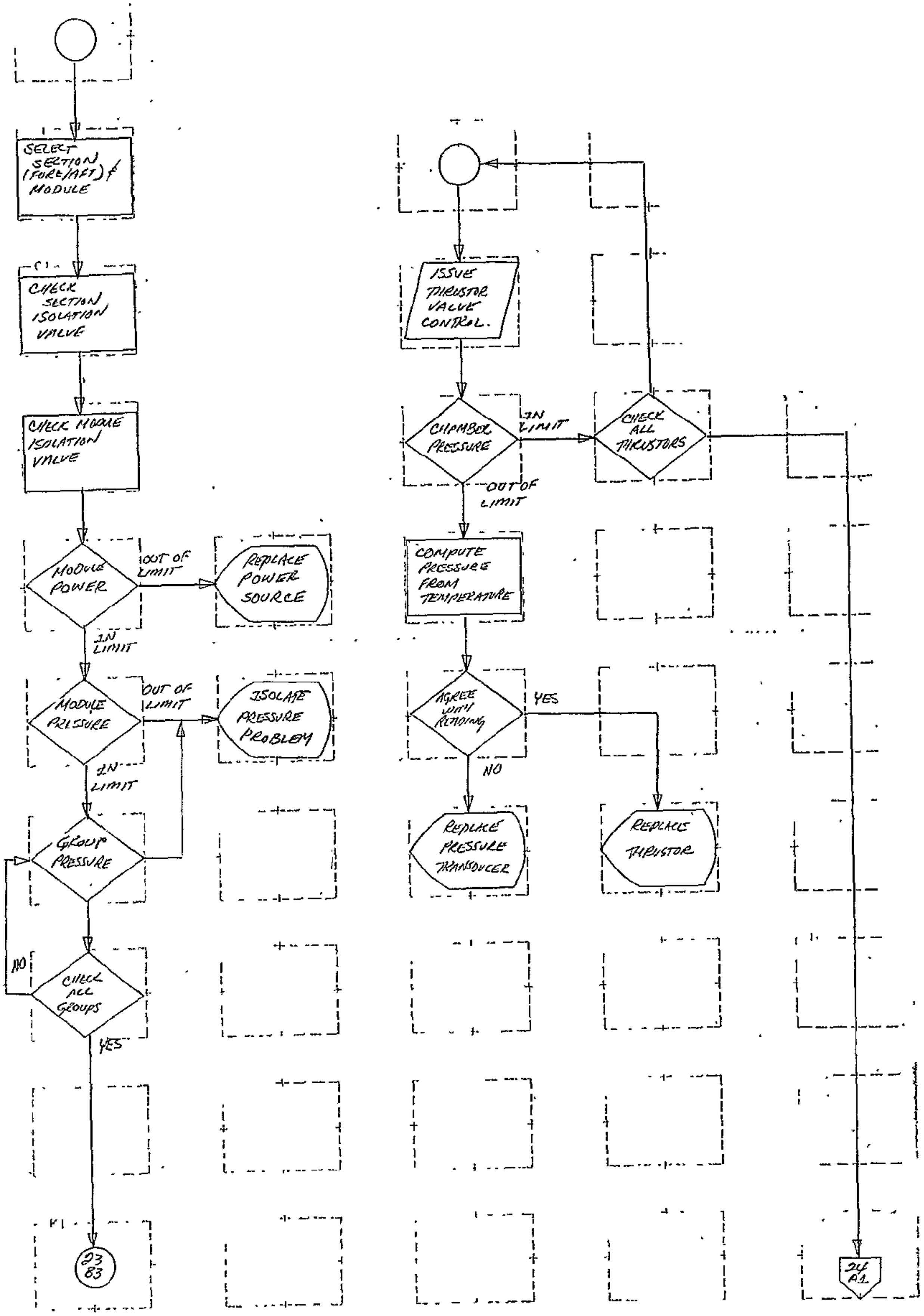


Figure 3-15. Thrustor Assembly (Sheet 1 of 2)

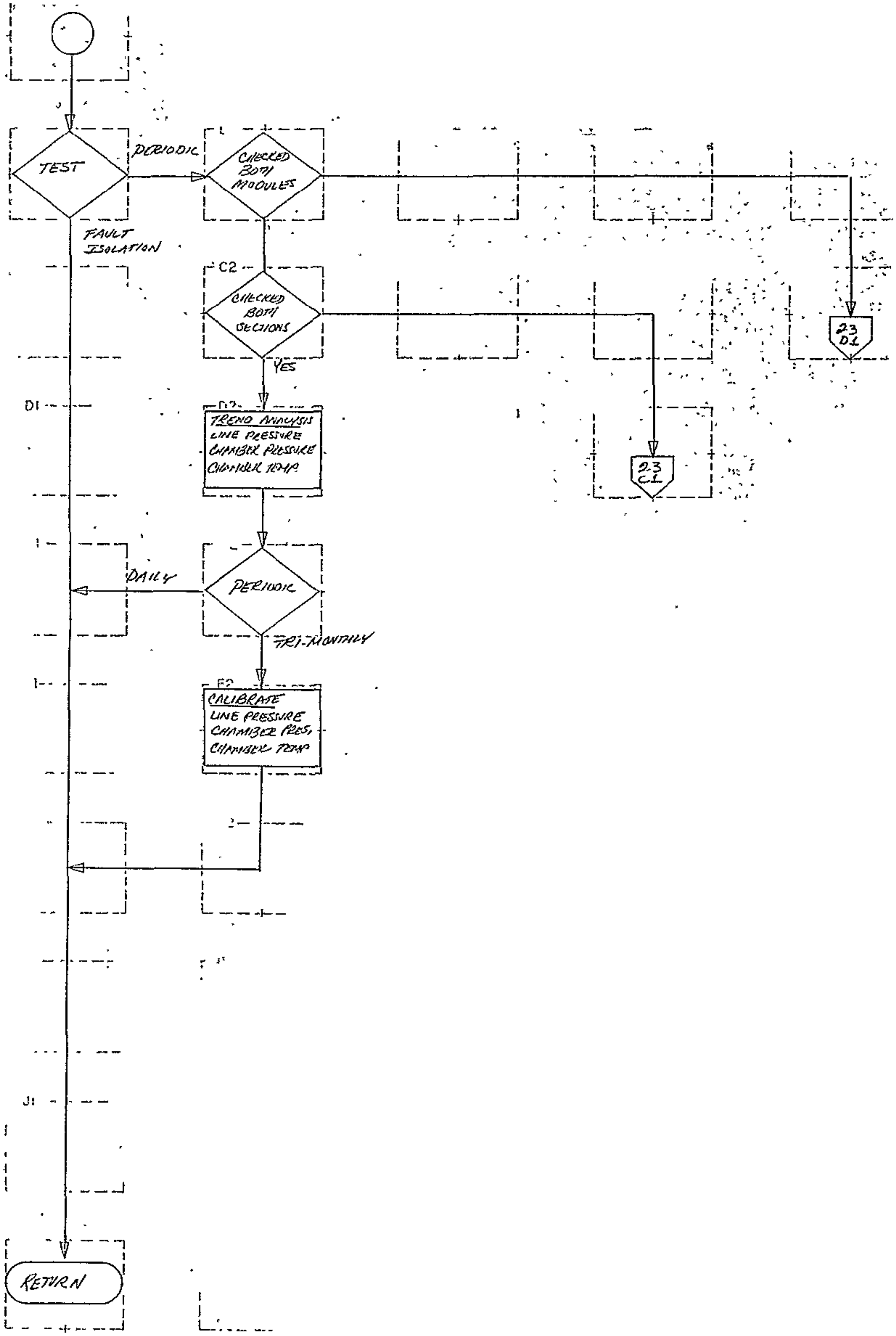


Figure 3-15. Thrustor Assembly (Sheet 2 of 2)

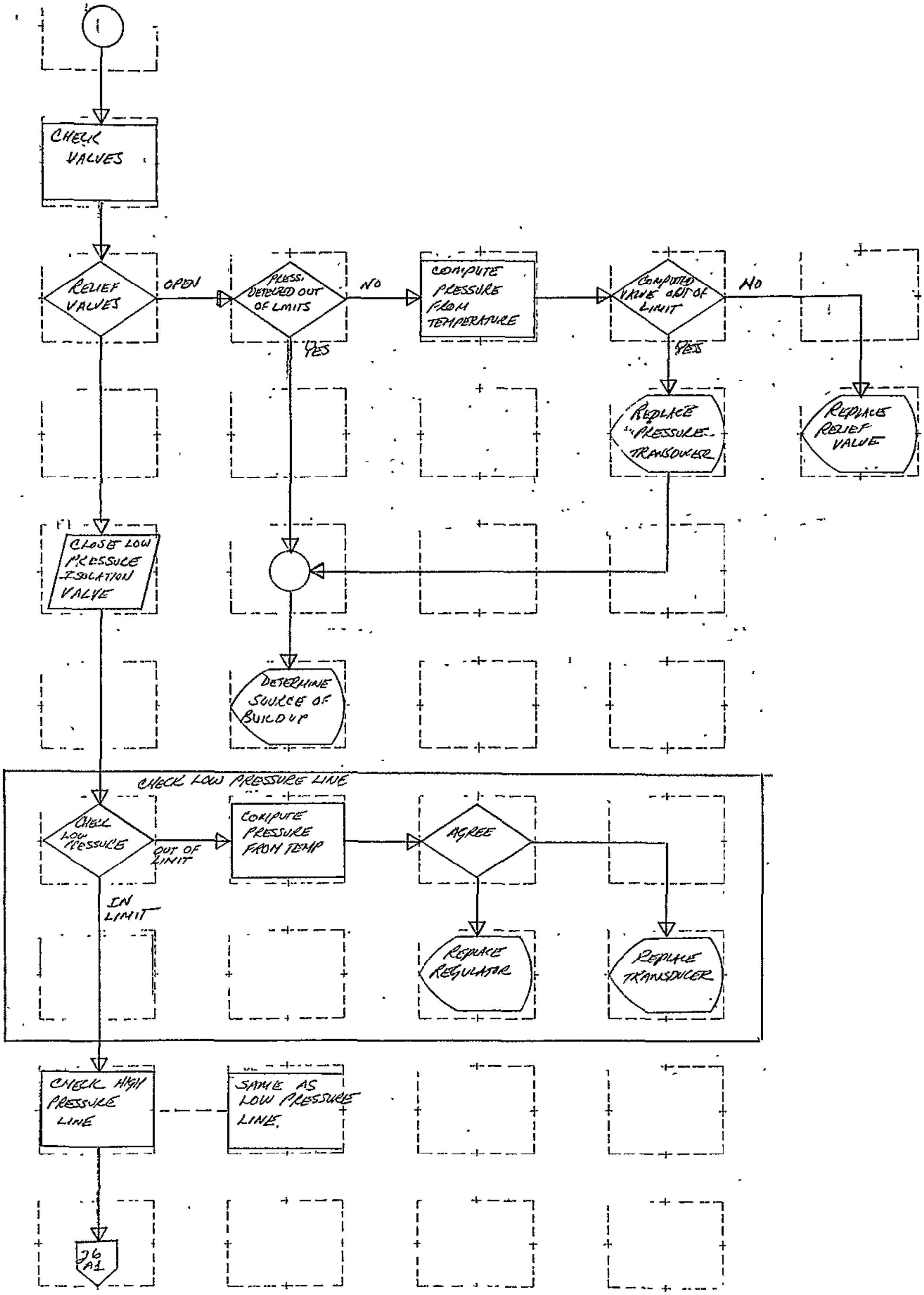


Figure 3-16. Purge Supply Assembly (Sheet 1 of 2)

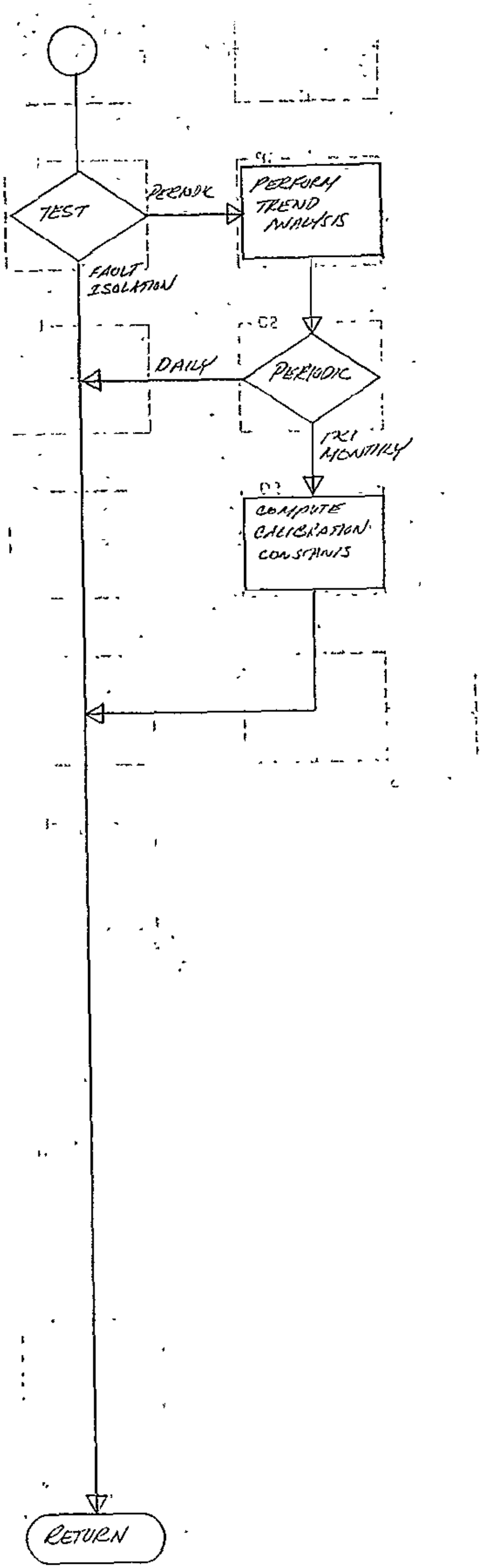


Figure 3-16. Purge Supply Assembly (Sheet 2 of 2)

3.1.4 DATA BASE REQUIREMENTS

This program requires access to the data base for the following data:

- Association of symbolic test points with the actual device addresses
- Status and allocation of configurable components

3.1.5 HUMAN PERFORMANCE

The operator is required to communicate with the program to accomplish the desired function. Specifically, the operator must initiate the program using the EXECUTE system communication element. The program may be terminated prior to completion by using the system communication function.

In addition, when errors are detected, the operator is provided with options to control program execution sequence. These options are referred to as GO-NOGO options and permit the operator to restart the LRU which failed, resume the program execution, or to terminate program execution.

3.2 INTERFACE REQUIREMENTS

This program must interface with the Master Executive, the OCS executive, and the propulsion subsystem hardware. The propulsion subsystem must also interface with the following subsystems:

- Environment Control and Life Support
- Power
- Guidance Navigation and Control
- Data Management

3.2.1 INTERFACE DIAGRAMS

The interface between the propulsion and other subsystems is depicted in Figure 3-17. Figure 3-18 diagrams the assembly interfaces in the Low Thrust Subsystem. Figures 3-19 through 3-23 provide detailed information regarding the Low Thrust Assemblies.

Figure 3-24 represents the interface between the assemblies in the High Thrust propulsion system. Figures 3-25 and 3-35 provide detailed information regarding the high thrust assemblies.

SUBSYSTEM INTERFACE DIAGRAM

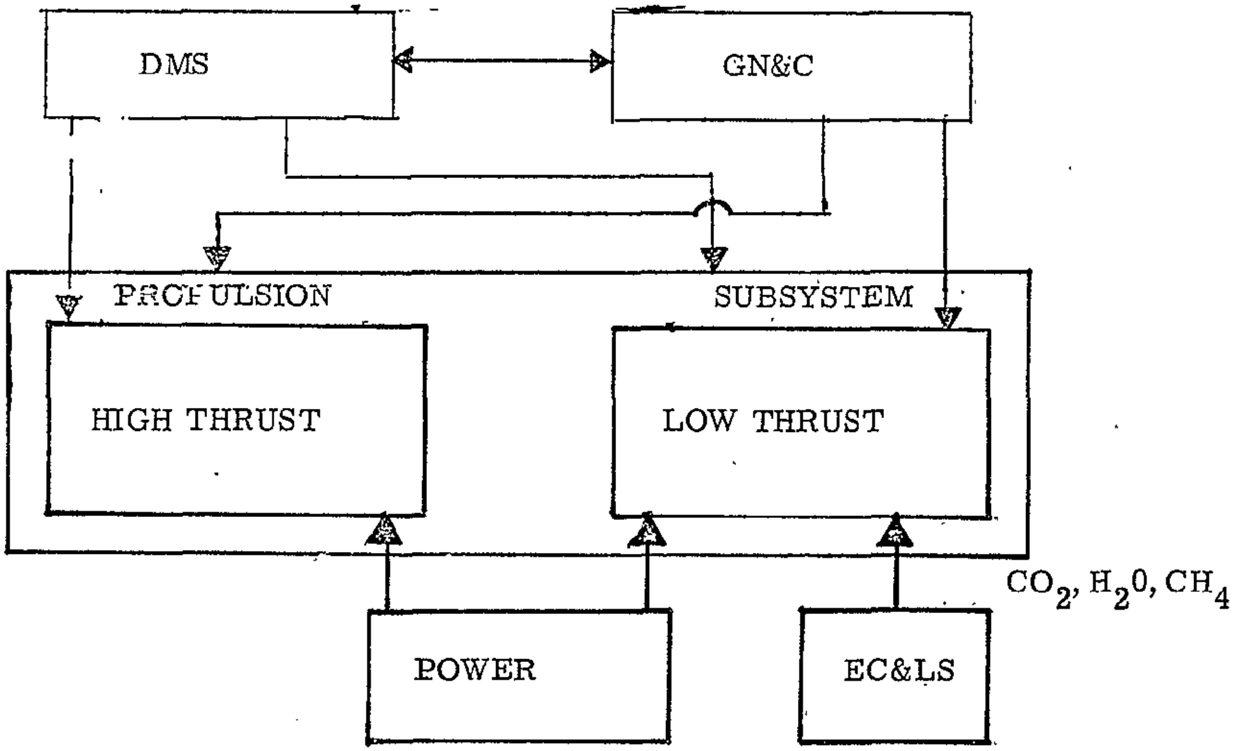


Figure 3-17. Subsystem Interface

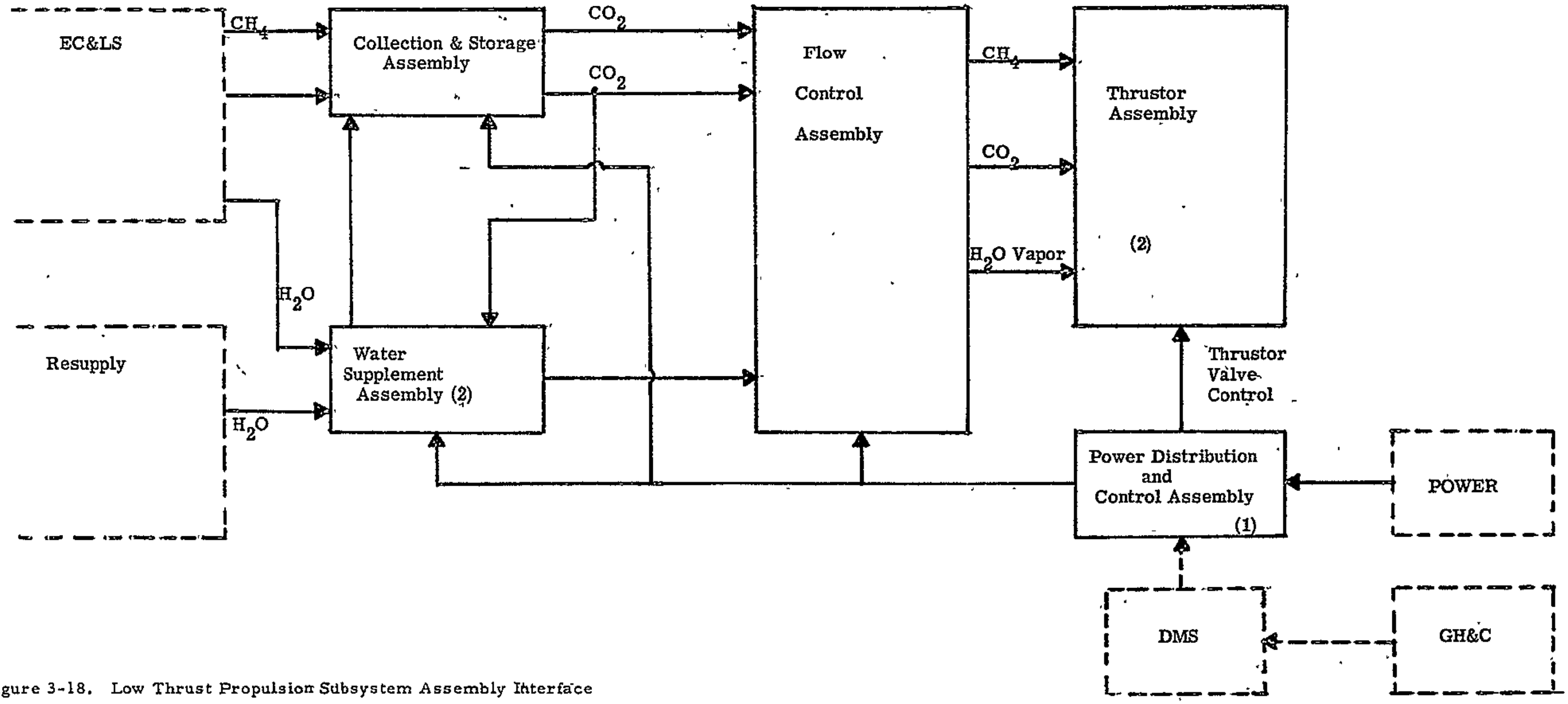
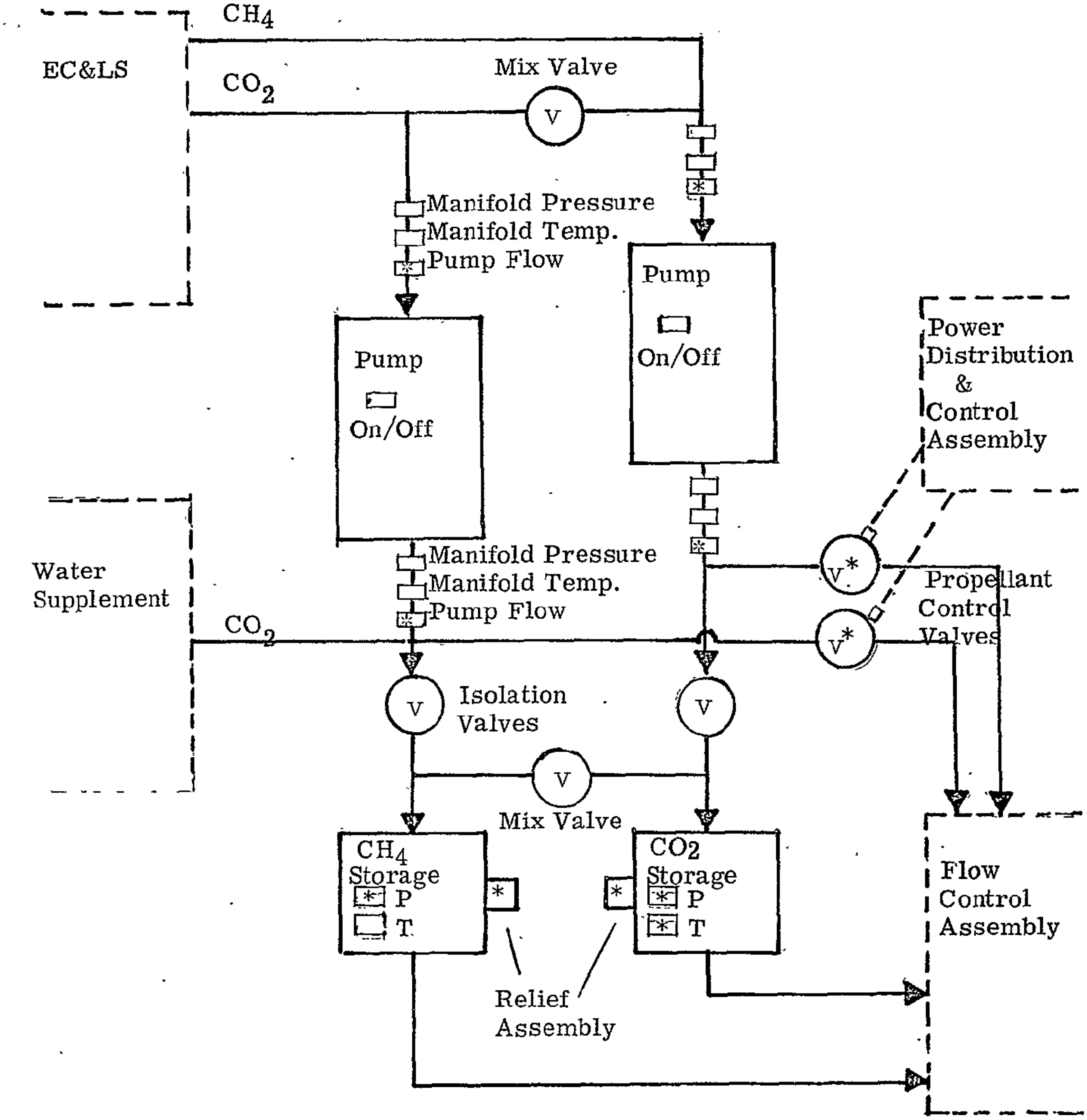


Figure 3-18. Low Thrust Propulsion Subsystem Assembly Interface



* Monitored for Fault Detection

Figure 3-19. Collection Storage Interface Assembly

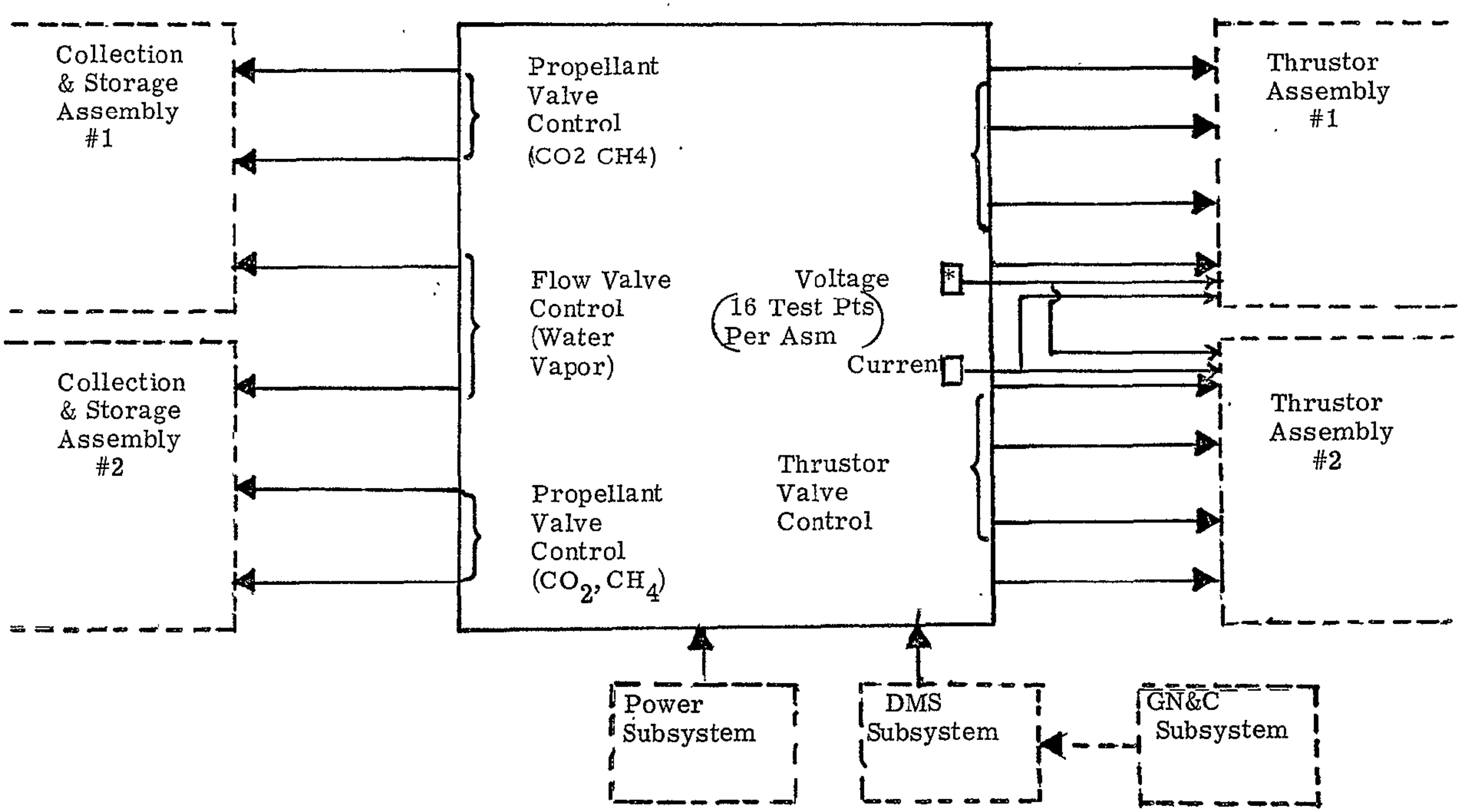


Figure 3-20. Power Distribution & Control Assembly Interfaces

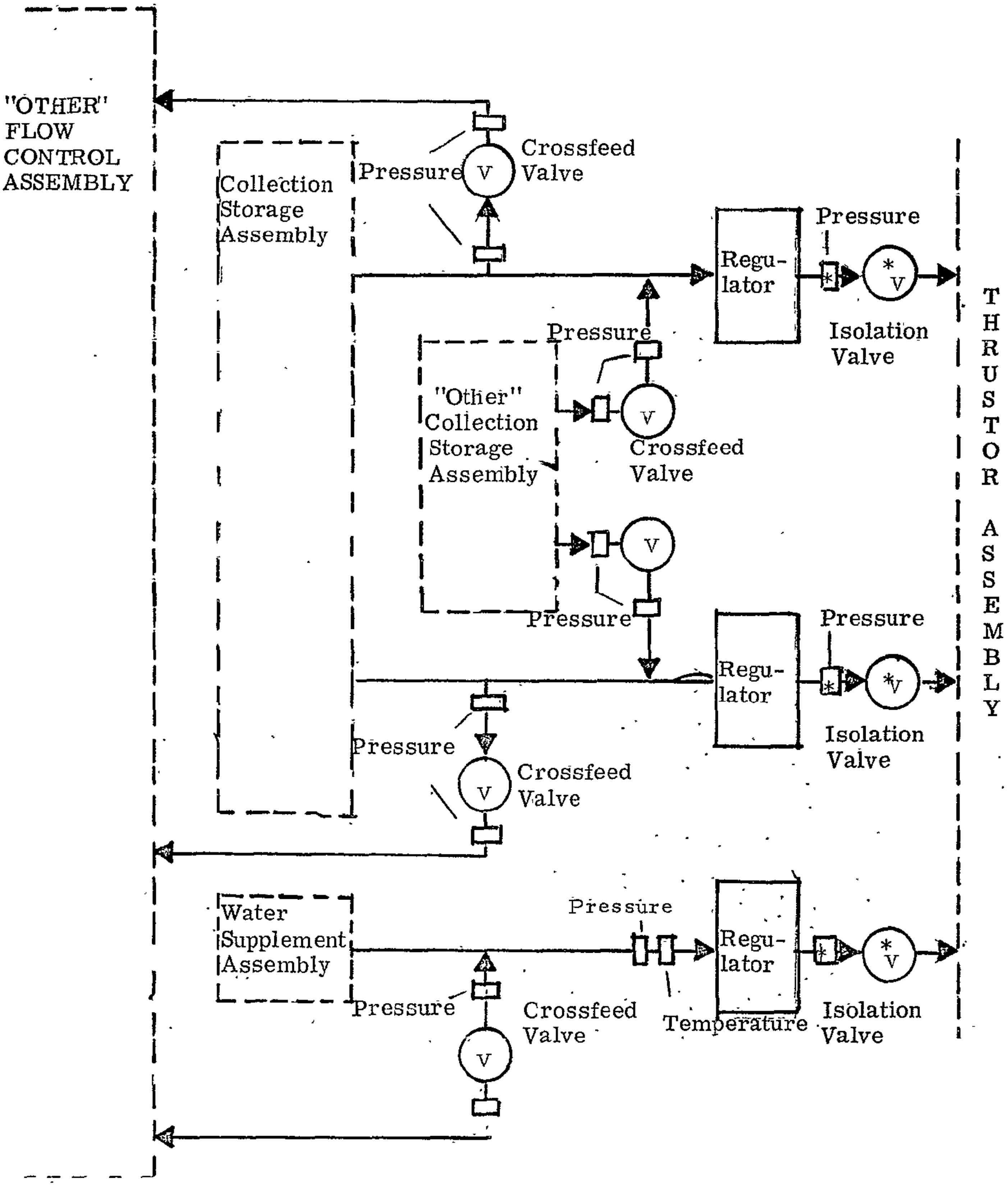


Figure 3-21. Flow Control Assembly Interfaces

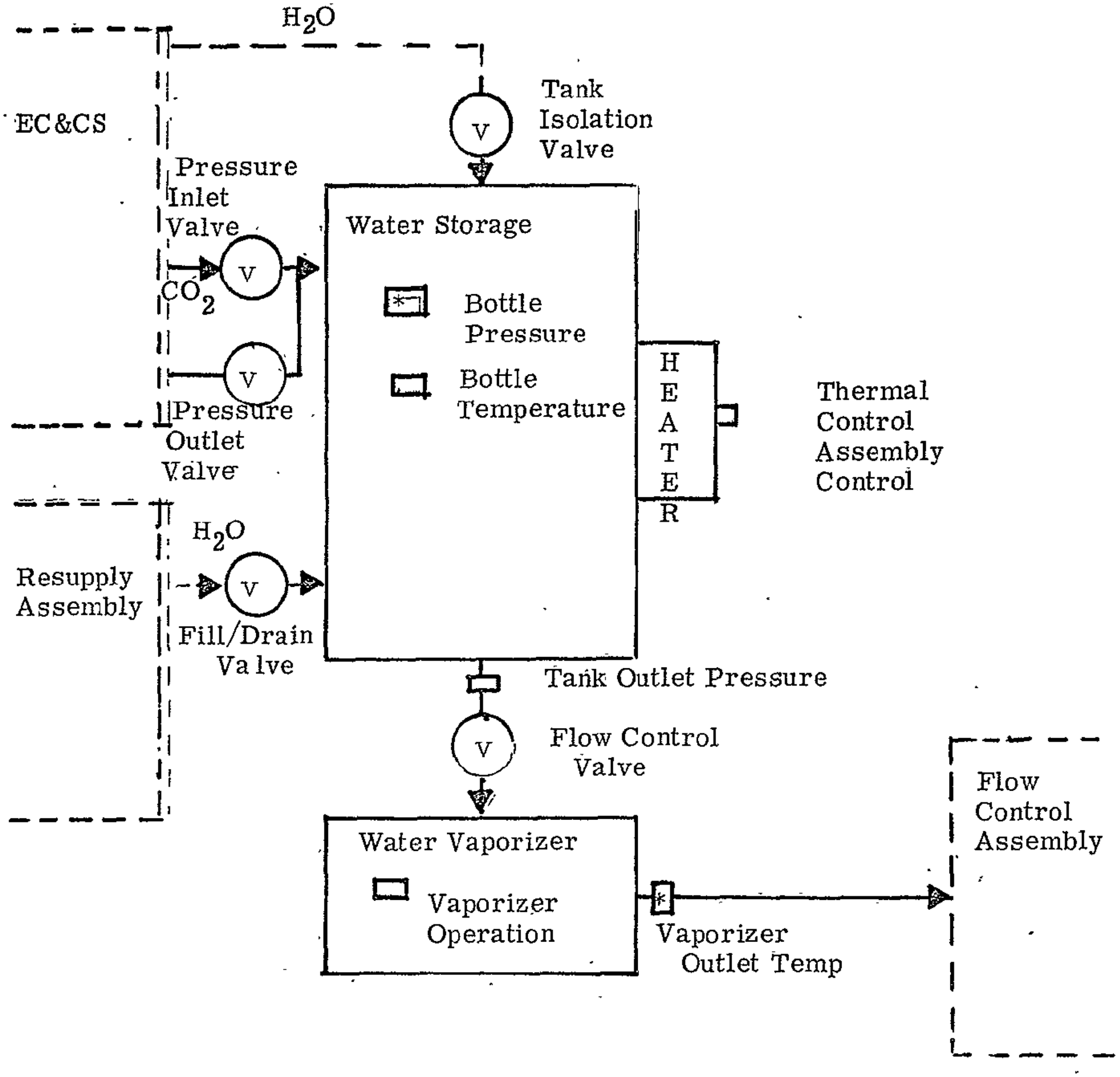


Figure 3-22. Water Supplement Assembly Interfaces

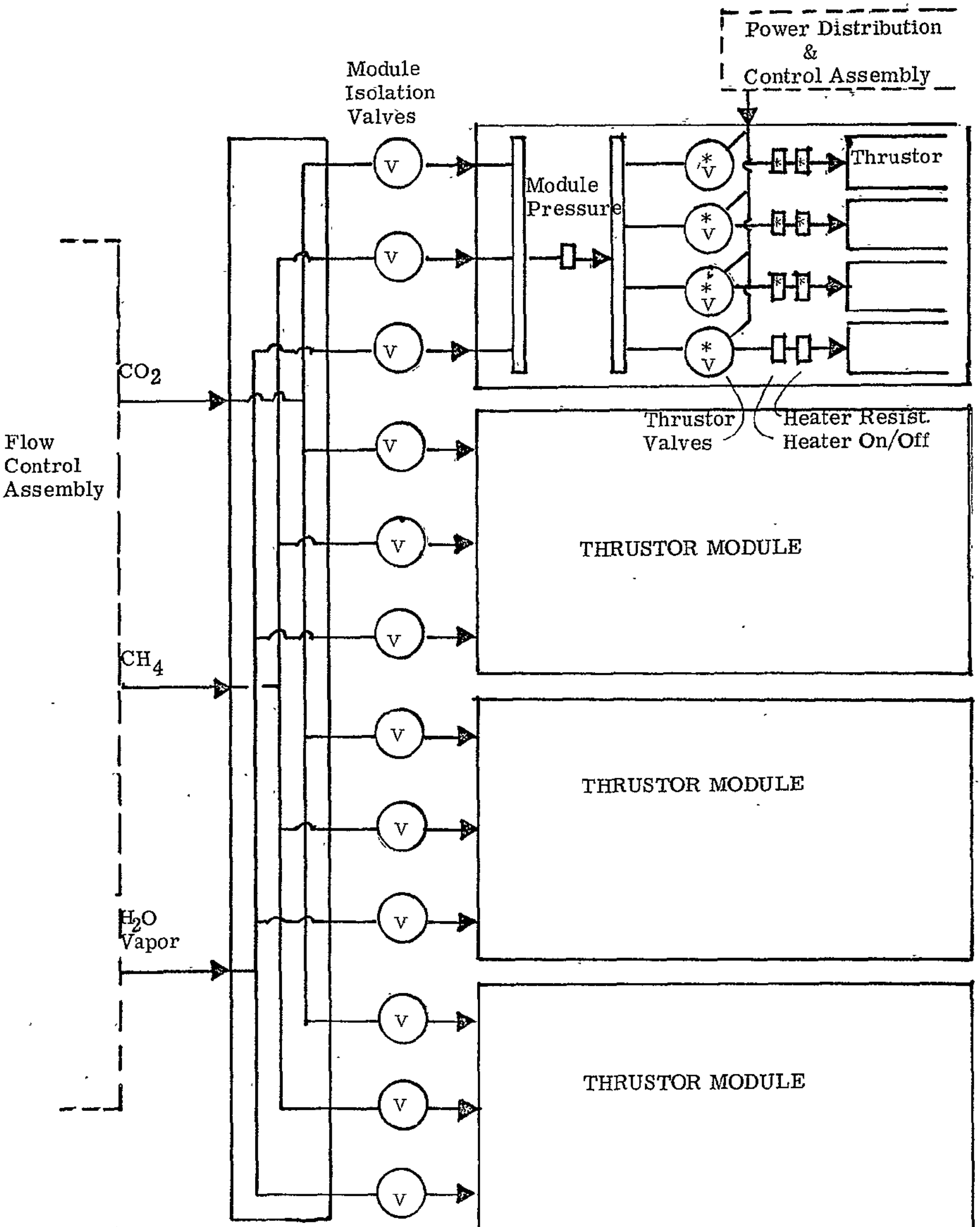


Figure 3-23. Thrustor Assembly Interfaces

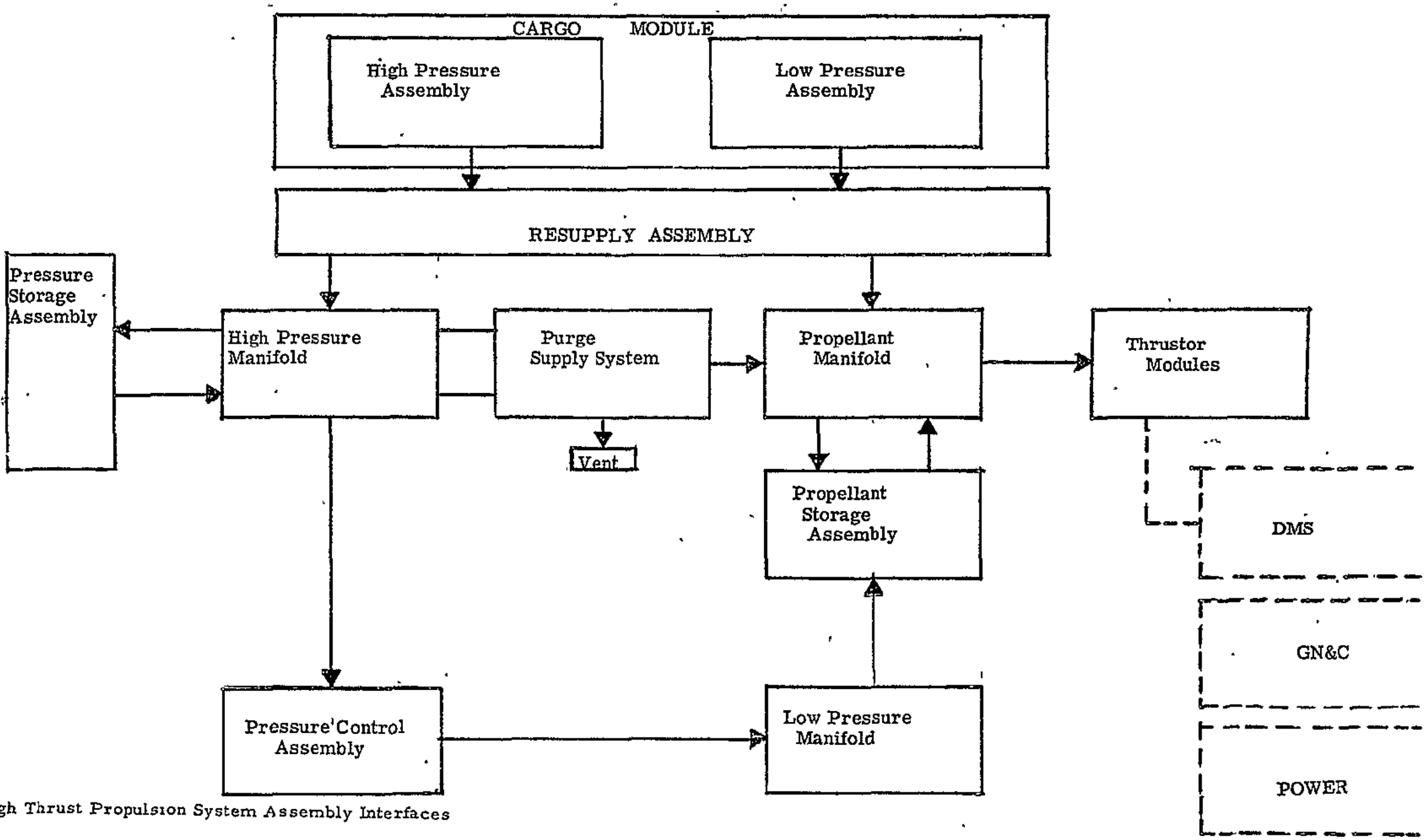


Figure 3-24. High Thrust Propulsion System Assembly Interfaces

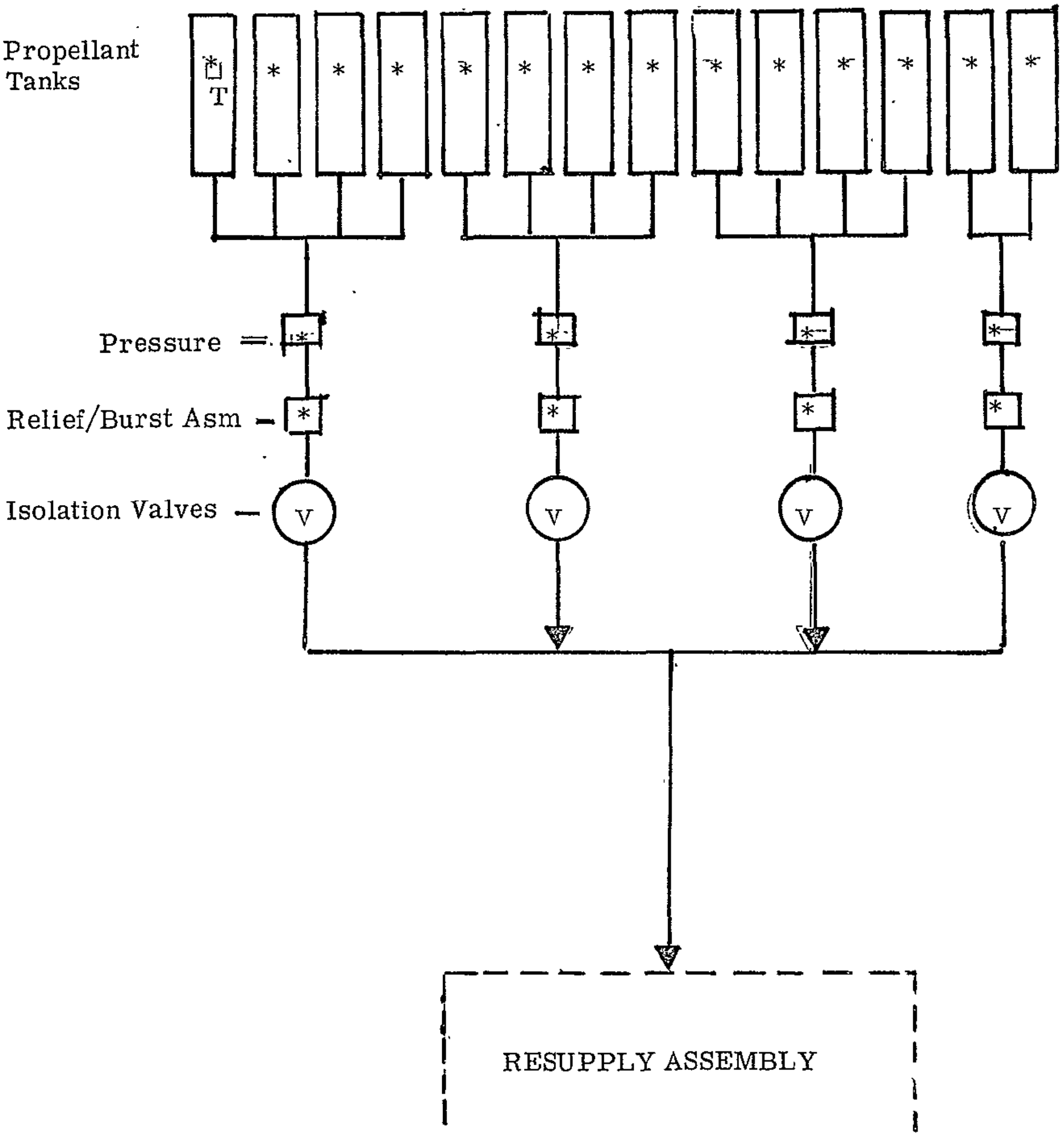


Figure 3-26. Low Pressure Assembly

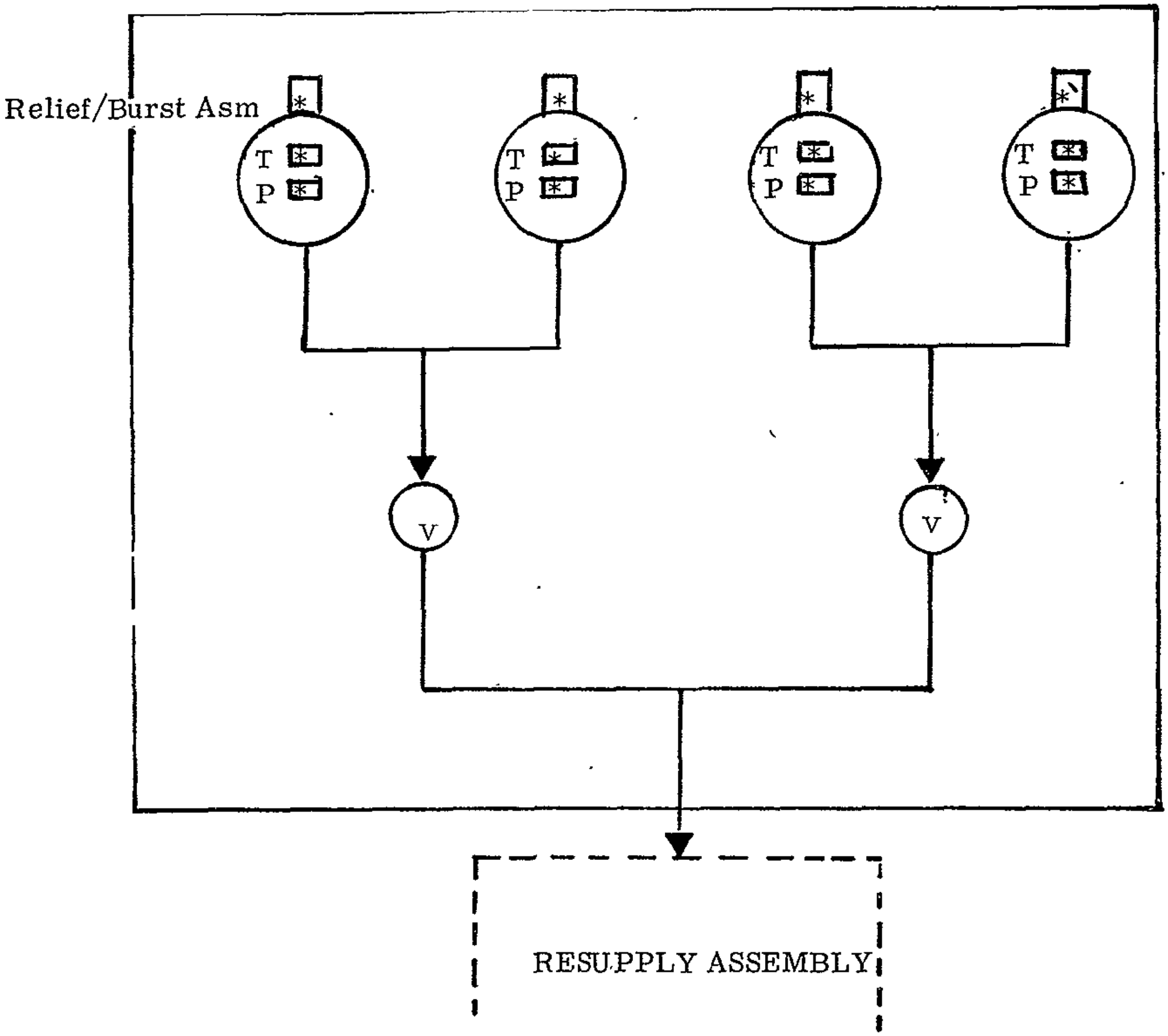


Figure 3-25. High Pressure Assembly

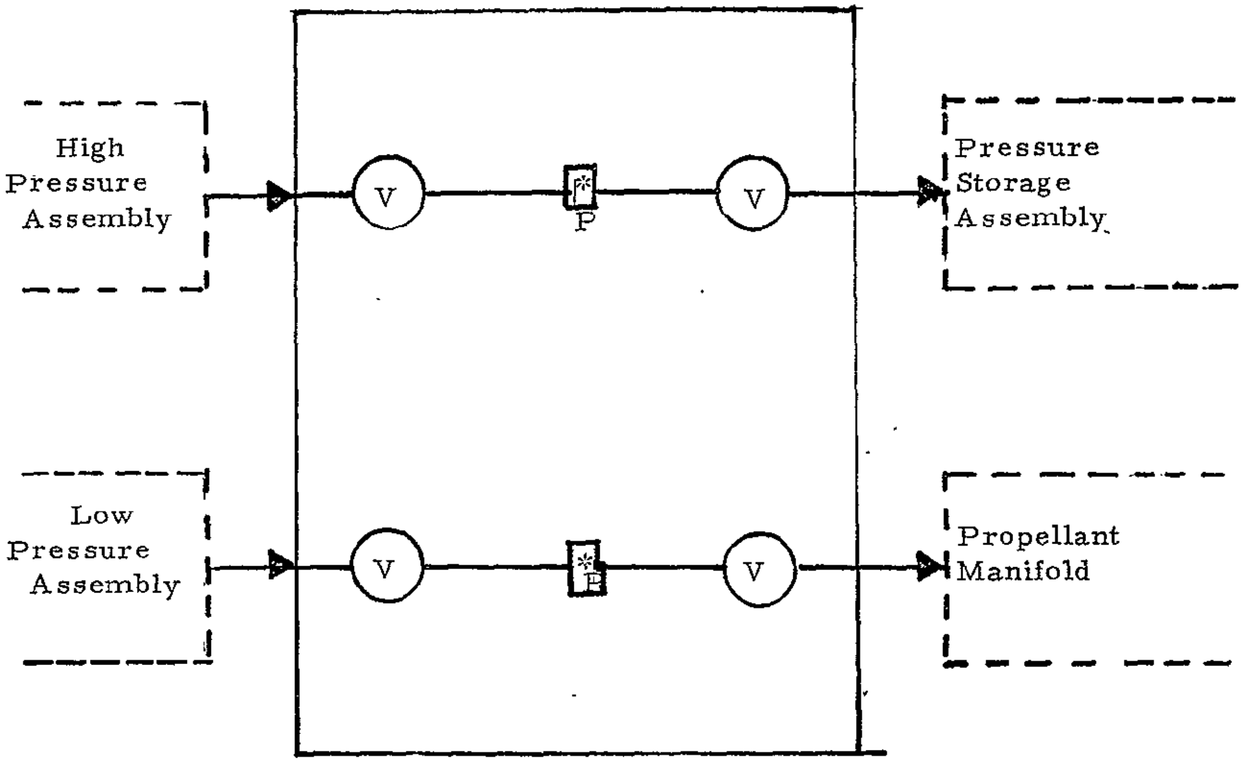


Figure 3-27. Resupply Assembly

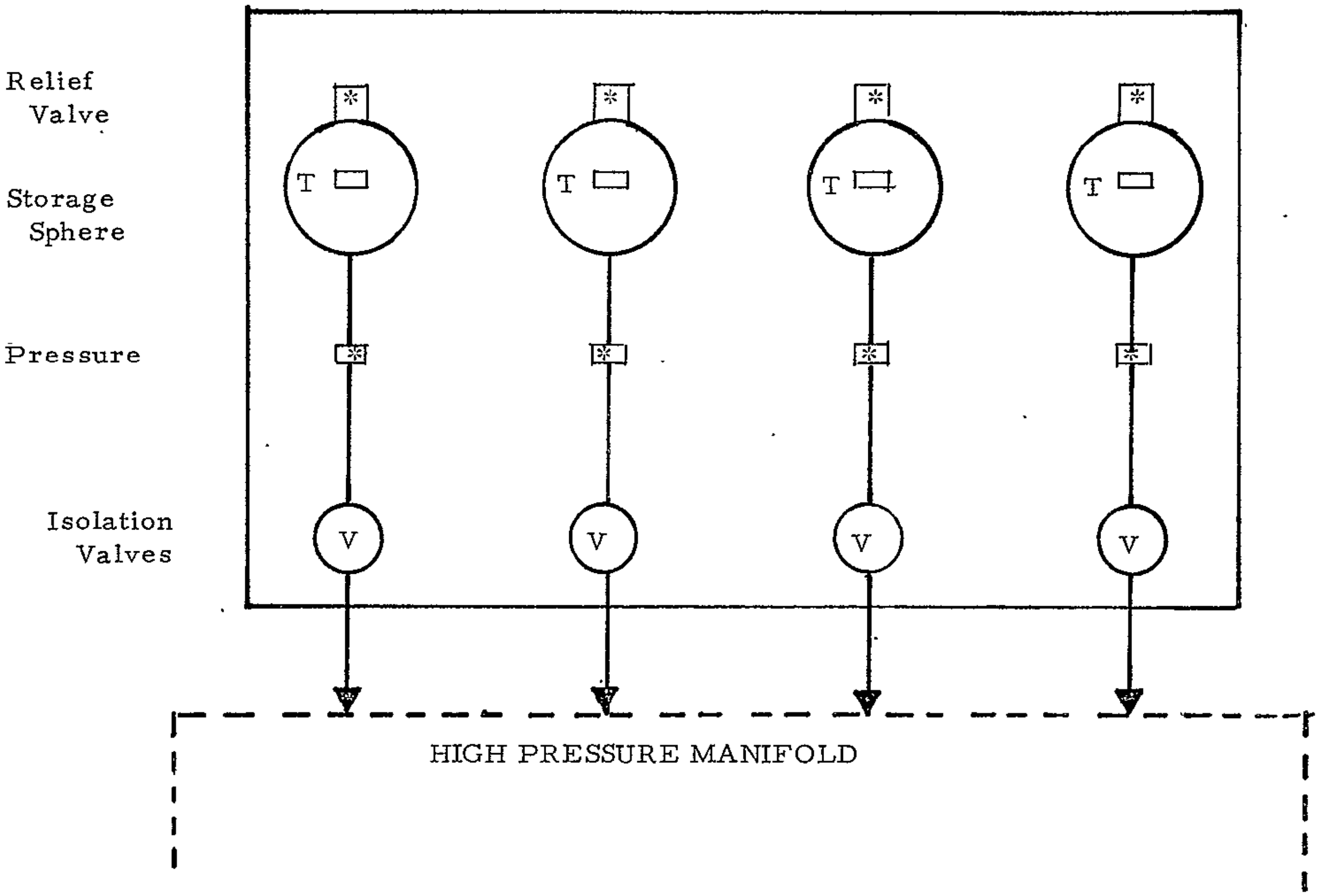


Figure 3-28. Pressure Storage Assembly

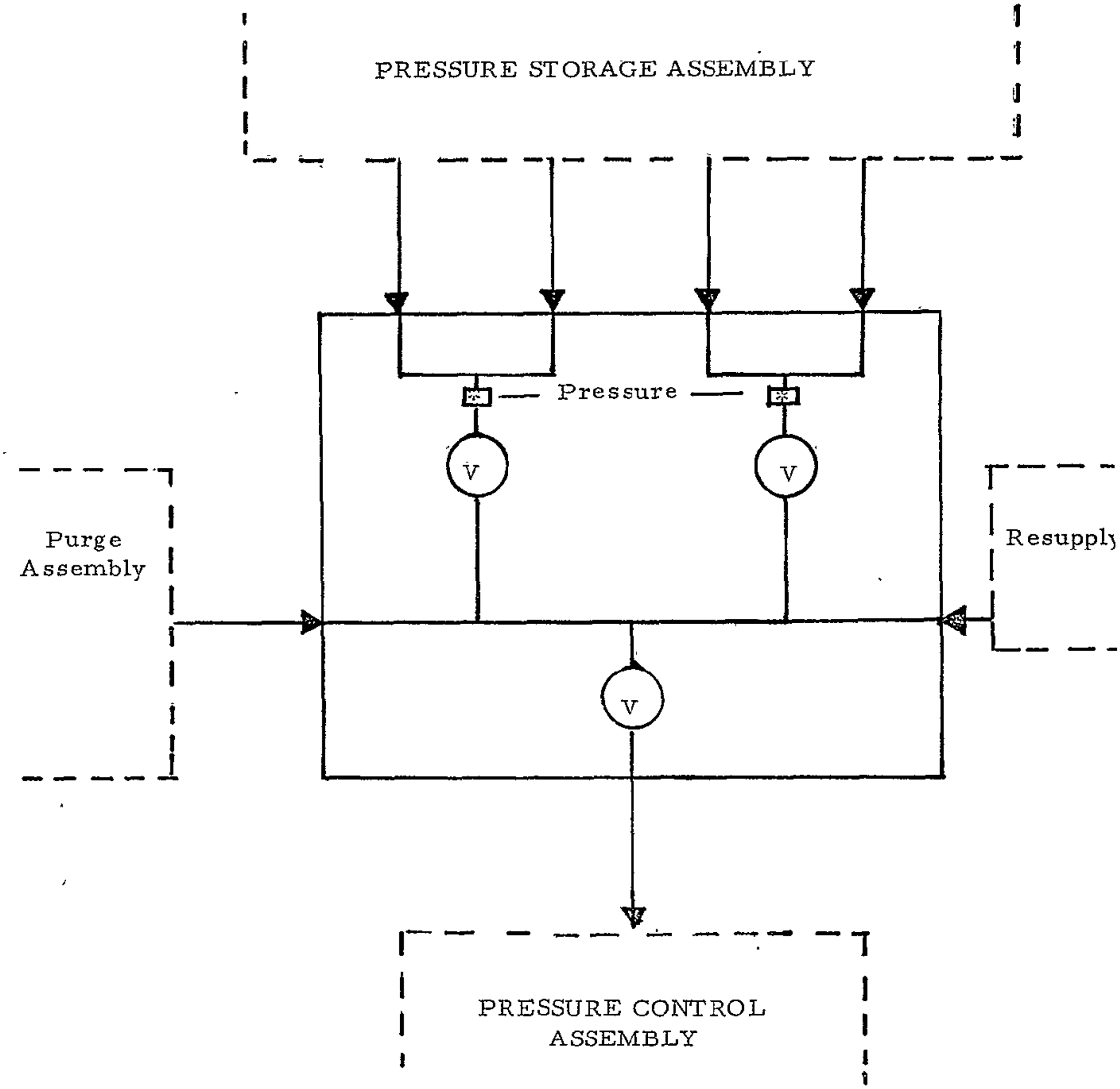


Figure 3-29. High Pressure Manifold

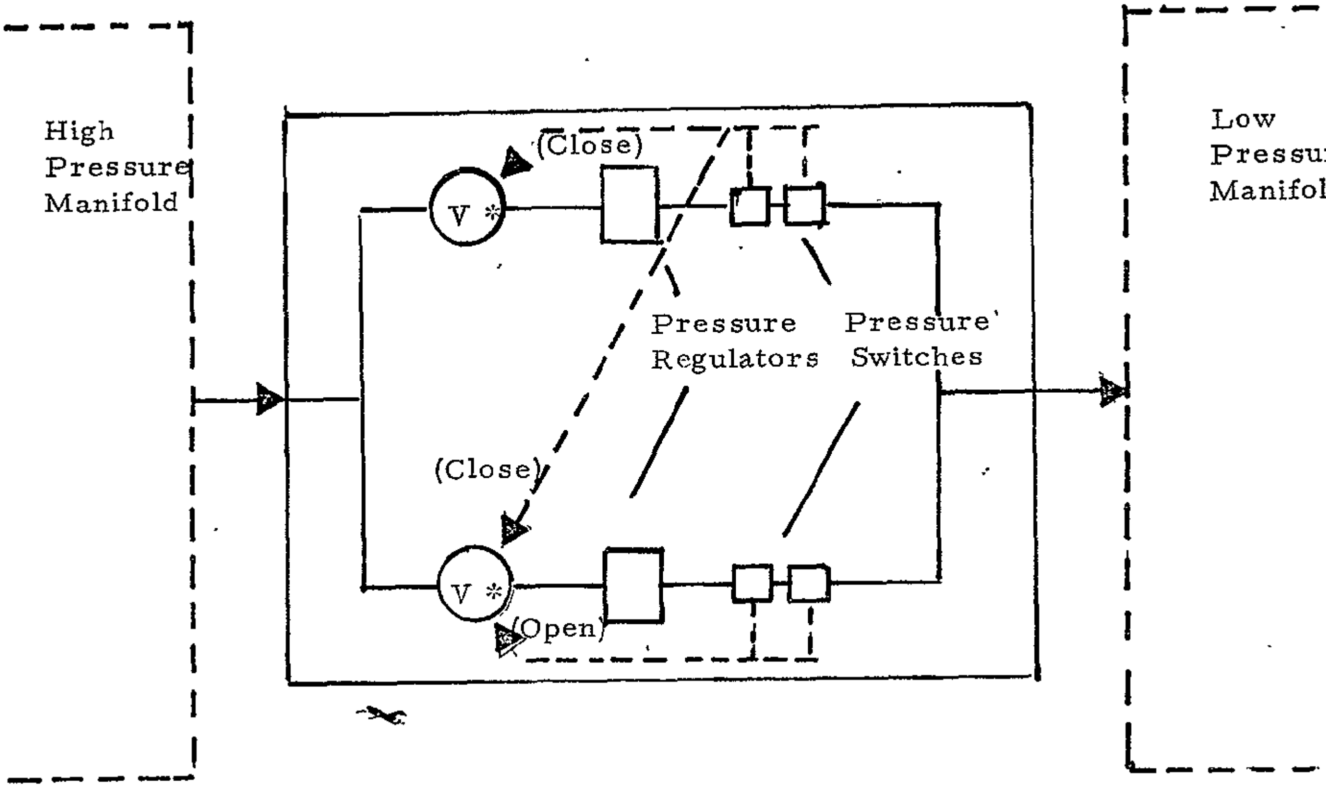


Figure 3-30. Pressure Control Assembly

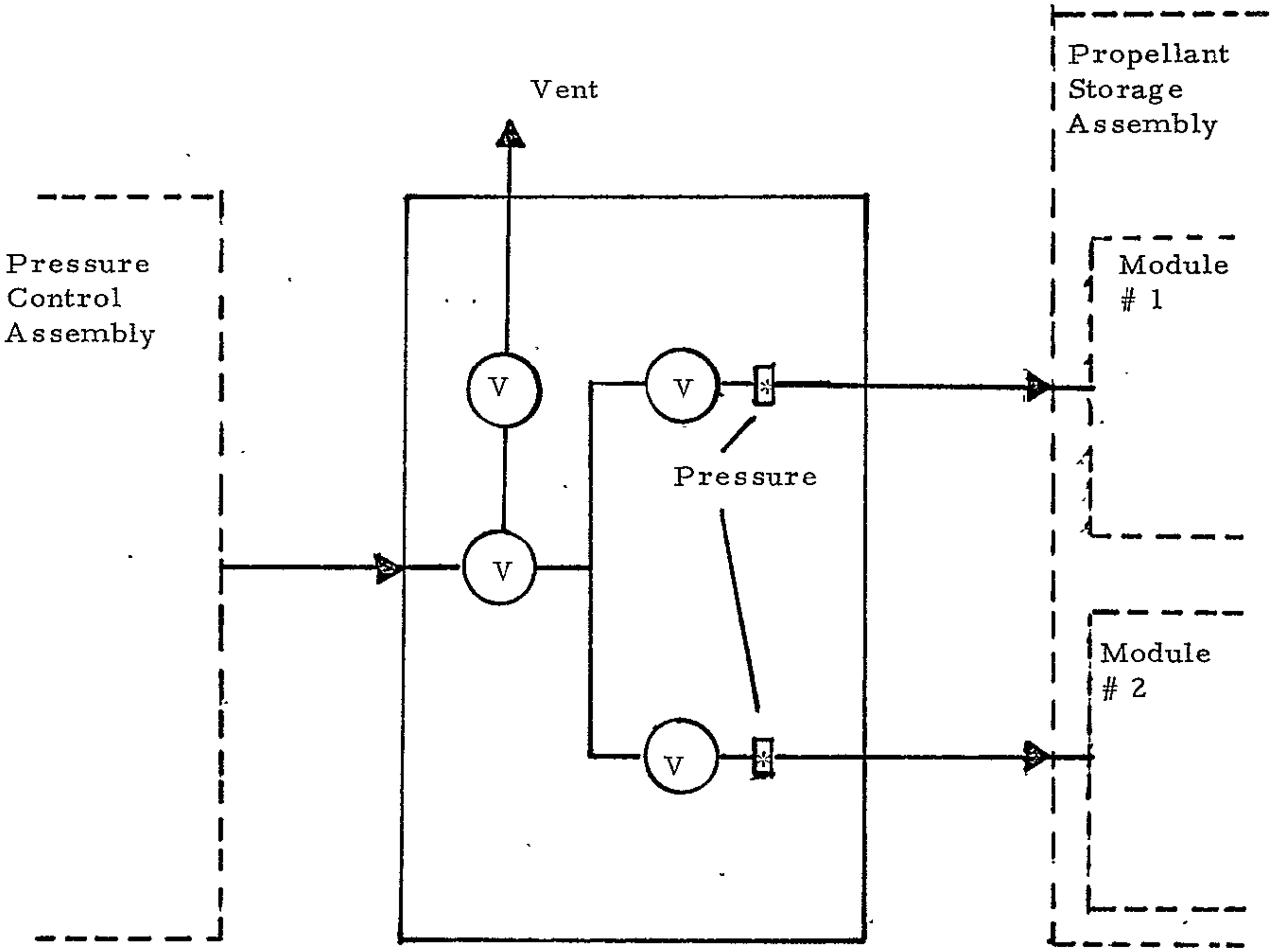


Figure 3-31. Low Pressure Manifold

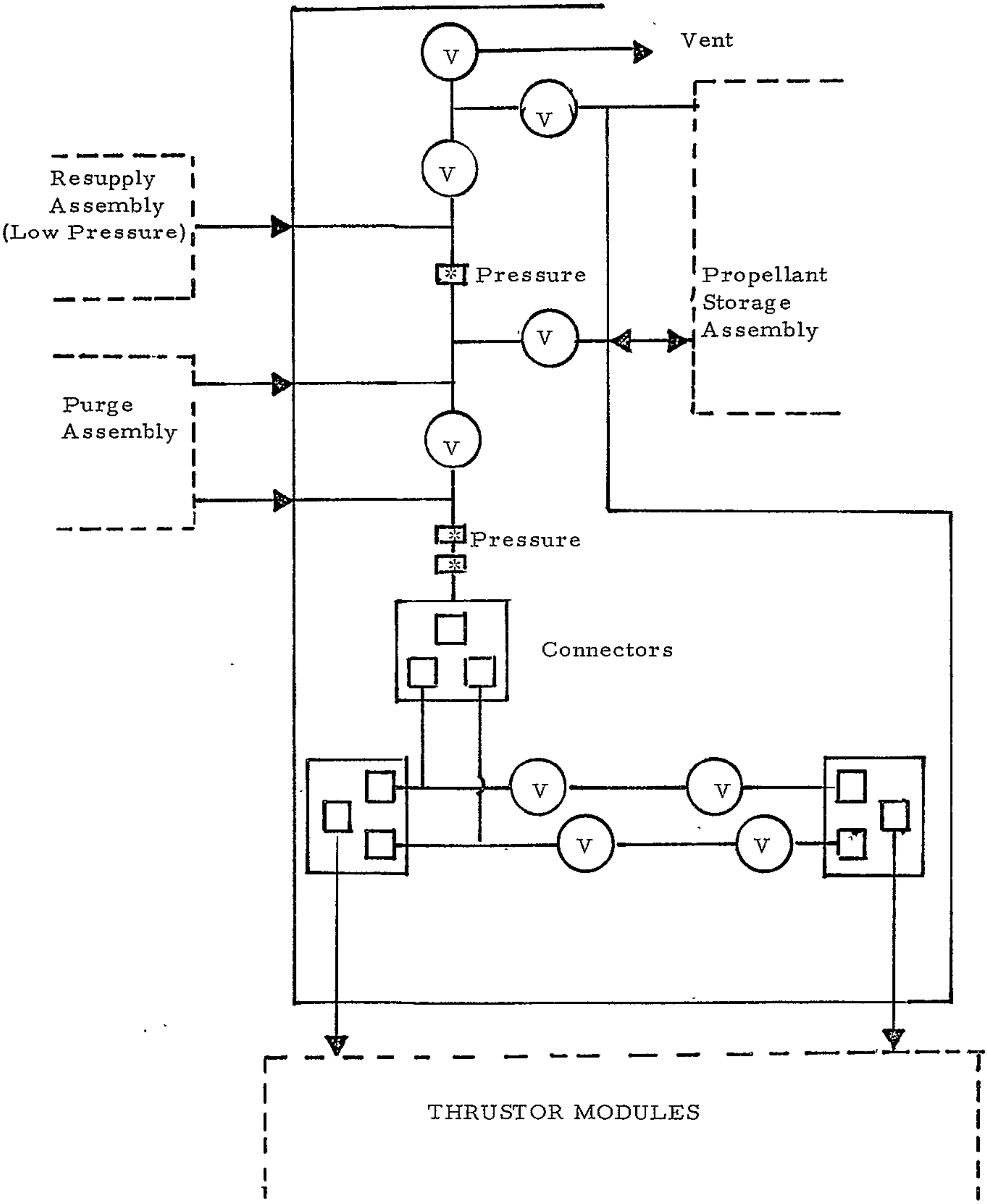


Figure 3-32. Propellant Manifold

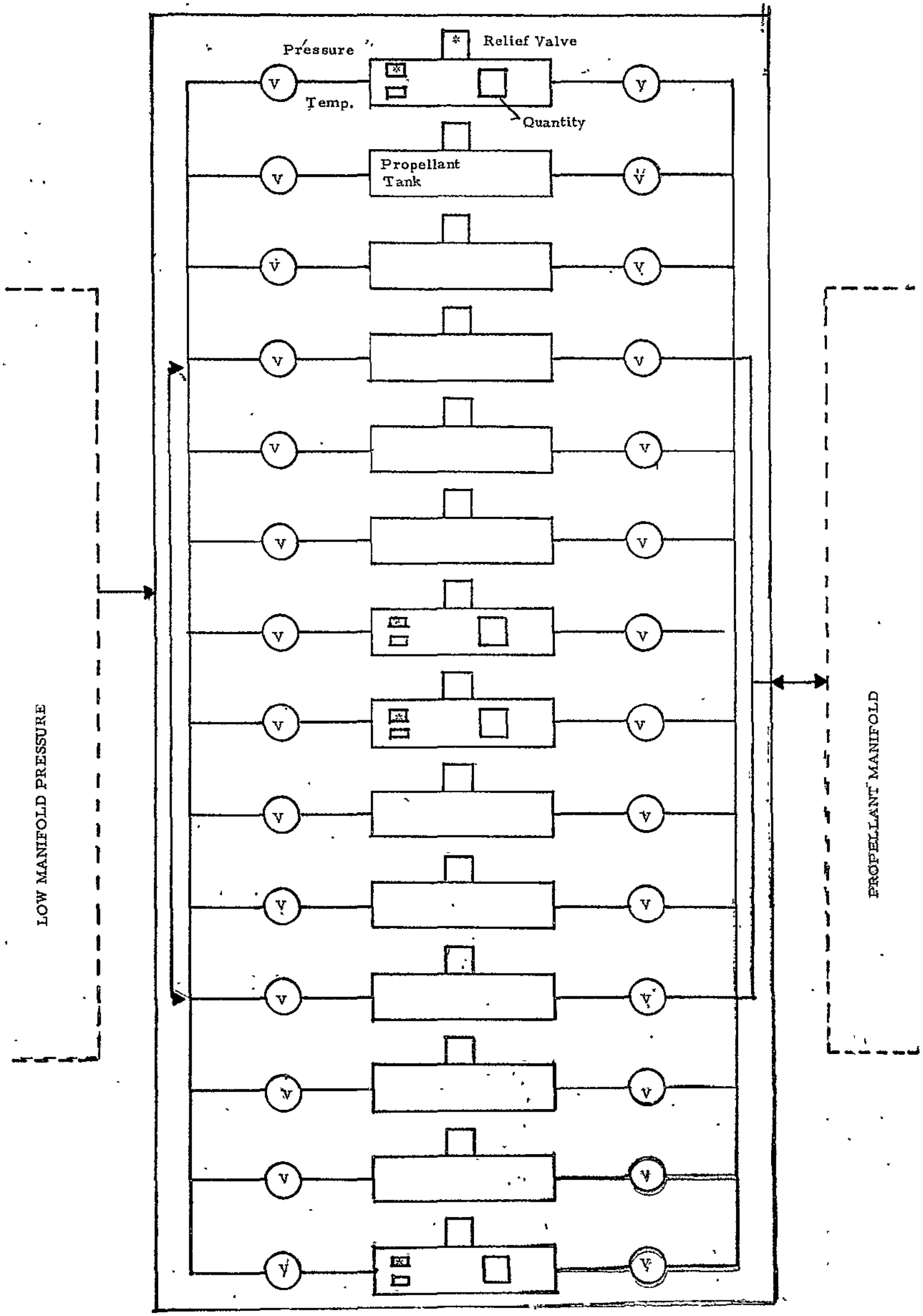


Figure 3-33. Propellant Storage Assembly

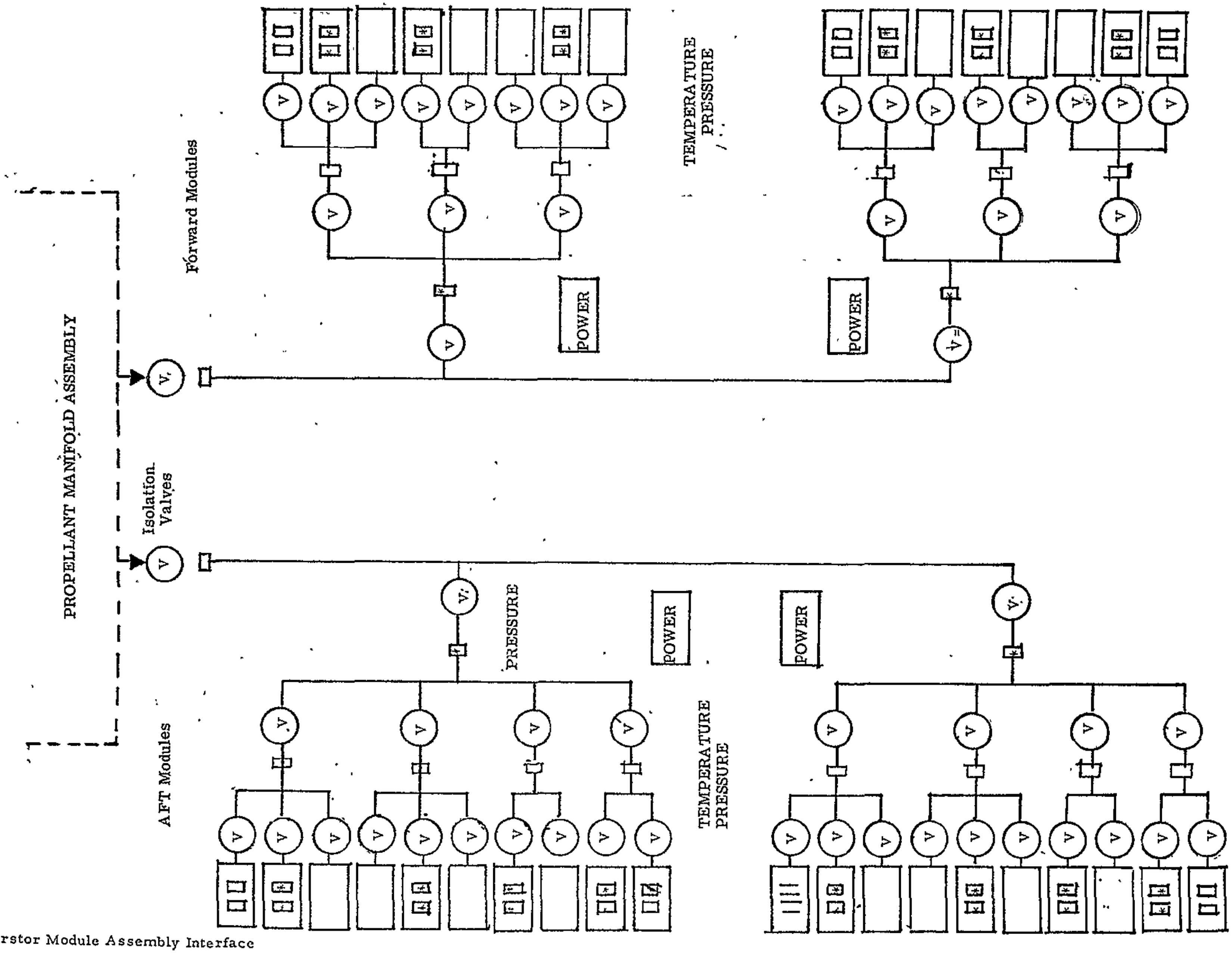


Figure 3-34. Thurstor Module Assembly Interface

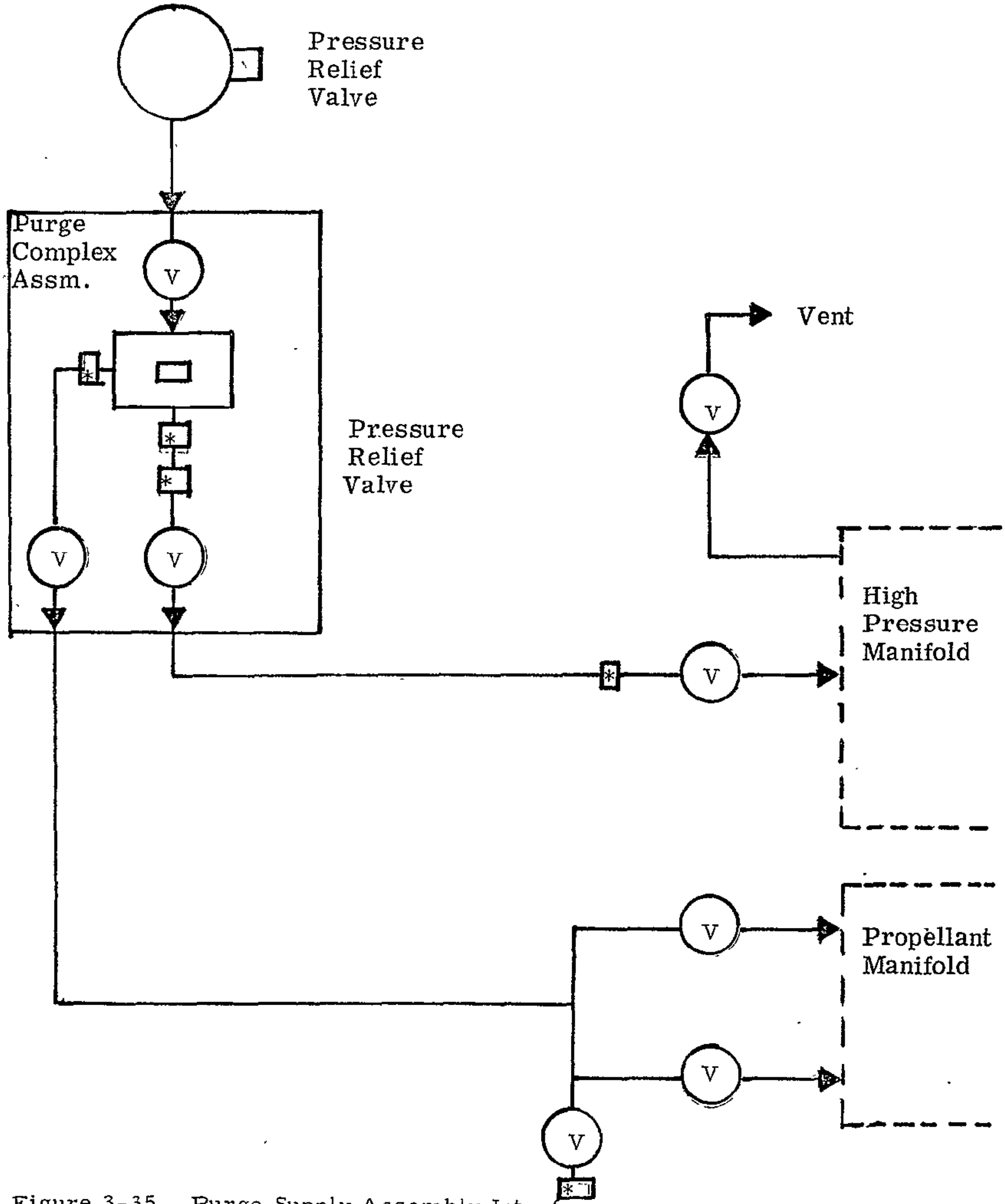


Figure 3-35. Purge Supply Assembly Interface

3.2.2 DETAILED INTERFACE DEFINITION

This section describes the interfaces between the associated subsystems and program modules.

3.2.2.1 Subsystem Interfaces

- Power Subsystem - The electrical power subsystem supplies all of the assemblies of the propulsion subsystem requiring electrical power.
- GN&C Subsystems - Commands to control the guidance of the space station are transferred from the GN&C subsystem to the Propulsion subsystem through the Data Management subsystem.
- EC&LS Subsystem - The EC&LS subsystem provides CO_2, H_2O, and CH_4 gases for use as a propellant for the low thrust propulsion system.
- Data Management Subsystem - The Data Management subsystem provides displays of the operational status of the propulsion subsystem, accomplishes fault detection, provides spares and inventory control, and provides displays associated with periodic testing, fault isolation, and subsystem maintenance and repair.

3.2.2.2 Executive Program Interfaces

The interfaces which are required in the executive architecture to service this program are depicted in Tables 3-1 and 3-2.

3.2.2.2.1 Input Processor - Keyboard Inputs

This program requires the ability to access parameter information input by the operator from the keyboard to specify the selection of programmed options.

3.2.2.2.2 Output Processor

- Measure Test Points - This program requires the capability to address specific test points through the Remote Data Acquisition Units (RDAU).
- Issue Stimulus - This program requires the capability to issue stimuli to specified test points.
- Display Control - This program requires the capability to present data at a display console to notify the operator of available options, or to present error messages.

Table 3.1 Propulsion Checkout Program

And

Executive Program Interfaces
(High Thrust System)

	Task Specification	Processor Allocation	Task Termination	Memory Allocation	Peripheral Allocation	Data Base Allocation	Interrupt Monitoring	Output Processor	Input Processor	Special Processor	Data Retention	GO-NOGO Processor				
High Pressure Asm								x		x		x				
Low Pressure Asm								x		x		x				
Resupply Asm								x		x		x				
Pressure Storage Asm								x		x		x				
High Pres. Manifold								x		x		x				
Purge Supply Asm								x		x		x				
Propellant Manifold								x		x		x				
Thrustor Module								x		x		x				
Pressure Control Asm								x				x				
Low Pressure Manifold								x				x				
Propellant Storage Asm								x		x		x				

Table 3.2 Propulsion Checkout Program

And

Executive Program Interfaces
(Low Thrust System)

		Task Specification	Processor Allocation	Task Termination	Memory Allocation	Peripheral Allocation	Data Base Allocation	Interrupt Monitoring	Output Processor	Input Processor	Special Processor	Data Retention	GO-NOGO Processor				
	Collection/Storage								x		x		x				
	Water Supplement								x				x				
	Flow Control								x		x		x				
	Thrustor Assembly								x		x		x				
	Power Dist. & Cont.								x		x		x				

3.2.2.2.3 Special Processor - Mode Control

This program requires the capability to have exclusive control over particular hardware components to accomplish required testing. This requires the capability to allocate a component to the program and to permit the program to indicate a failure status on the device when an error is detected.

4.0 SUBSYSTEM SUMMARY

This section summarizes some of the information which has resulted from the analysis of the propulsion subsystem.

4.1 LANGUAGE ANALYSIS

4.1.1 ELEMENT ANALYSIS

The element analysis which resulted from the study of this subsystem confirmed our previous results.

4.1.2 SIZING ESTIMATES

Tables 4-1, 4-2, and 4-3 provide the sizing estimates which have been developed for this subsystem.

4.2 EXECUTIVE ANALYSIS

The Propulsion Subsystem requires the capability to address test points which are in a redundant capacity.

Table 4.1 Element Utilization

	ELEMENT														
	ALLOCATE	BEGIN	CALL	DELAY	DISPLAY	GOTO	IF	MEASURE	READ	SEARCH	SOLVE	STATUS	STIMULATE	TIME	WRITE
Seq Logic Module		1	50		15	42	39			4	8		1		1
Collection/Storage	1	1		12	17	11	25	23			24	2	12		10
Power Dist & Cont	1	1		18	3	5	109	69			140		36		
Flow Control Asm.	1	1		2	7	6	12	9	1		12	1	2		1
Water Supp Asm	1	1		4	11	5	21	11	8		24	4	4		8
Thrustor Asm	1	1		2	5	4	11	7	6		14	2	2		6
Low Pressure Asm	1	1		3	8	7	15	12			14	1	3		
Resupply Asm	1	1		5	6	3	12	3	2		10	1	3		2
Pressure Storage	1	1		6	28	18	22	36			16	1	6		
High Pressure Man.	1	1		2	8	5	19	5	2		12	2	4		2
Pressure Control	1	1		12	17	12	28	25	4		26	4	14	2	4
Low Pressure Man.	1	1		9	16	2	12	13	2		7	1	8		2
Propellant Manifold	1	1		9	17	1	13	13	3		10	1	8		3
Propellant Storage	1	1		9	20	2	15	14			11	1	1		
Thrustor Assembly	1	1		3	8	7	20	11	3		20	2	3		3
Purge Supply	1	1		5	7	2	11	6	4		11	1	2		4
High Pressure Asm	1	1		2	8	8	15	10	4		13	1	2		
TOTAL	6	16	50	103	201	138	399	267	267	4	372	25	11	2	46

Table 4.2. Periodic Test Element Execution

	ALLOCATE	BEGIN	CALL	DELAY	DISPLAY	GOTO	IF	MEASURE	READ	SEARCH	SOLVE	STATUS	TIME	WRITE		
Seq. Logic Module			13		13	20	22			4	8					
Collection Storage	1	1		32	1	20	72	68			80	8	32	40		
Power Dist & Cont.	1	1		18		2	103	68			134		36			
Flow Control Asm.	1	1		11		16	44	29	20		59	6	11	20		
Water Supp. Asm.	1	1		9		10	46	21	8		44	9	9	8		
Thrustor Asm.	1	1		56		58	29	115			230	56	56	6		
Low Pressure Asm.	1	1		8		8	44	32			64	4	12			
Resupply Asm.	1	1		8	16	10	34	18	4		44	8	12	4		
Pressure Storage	1	1		8	8	8	44	24			44	4	8			
High Pressure Man.	1	1		9		13	62	20	2		38	9	9	2		
Pressure Control	1	1		13	2	10	33	23	4		30	5	15	4		
Low Pressure Man.	1	1		3		4	17	8	2		15	3	3	2		
Propellant Manifold	1	1		9		9	47	19	3		40	9	9	2		
Propellant Storage	1	1		36	57	33	165	70	14		184	28	37	3		
Thrustor Assembly	1	1		142		25	73	324	92		140	2	160	14		
Purge Supply	1			9		8	40	20	4		36	8	9	92		
High Pressure Asy.	1			16		20	100	44			84	16	16	4		

Table 4.3. Propulsion Subsystem Sizing Estimates

Low Thrust System

Program text	5K	
Working buffer	1K	
Main Storage (words)		6K
Program code	8K	
Displays	.5K	
Auxiliary Storage (words)		8.5K
CPU Time	.010	
I/O Time	2.649	
Time in Main Storage		2.659 minutes

High Thrust System

Program text	4K	
Working buffer	1K	
Main Storage (words)		5K
Program code	8.5K	
Displays	1.5K	
Auxiliary Storage (words)		10K
CPU time	.015	
I/O time	4.745	
Time in Main Storage		4.760 minutes

Trend Analysis

Main Storage	150 words
Auxiliary Storage	1K words

	Per Execution	Per Day
CPU time	.00005	.00665
I/O time	.0026	.3458
Time in Main Storage (Mins)	.00265	.35245

International Business Machines Corporation
Federal Systems Division
150 Sparkman Drive
Huntsville, Alabama 35805