Communications and Tracking Expert Systems Study

Preface

This research was conducted under the auspices of the Research Institute for Computing and Information Systems by T.F. Leibfried, Associate Professor of Computer Science, Terry Feagin, Professor of Computer Science, and David Overland, Research Associate, all at the University of Houston-Clear Lake.

Funding has been provided by the Tracking and Communications Division, within the Engineering Directorate, NASA/JSC through Cooperative Agreement NCC 9-16 between NASA Johnson Space Center and the University of Houston-Clear Lake. The NASA Technical Monitor for this activity is Oron Schmidt, Systems Techniques Section, Communications Performance and Integration Branch, NASA/JSC.

The views and conclusions contained in this report are those of the authors and should not be interpreted as representative of the official policies, either express or implied, of NASA or the United States Government.

RECEIVED

FEB 11 1987

RICIS

INTERIM REPORT

UNIVERSITY OF HOUSTON-CLEAR LAKE

RESEARCH INSTITUTE FOR THE COMPUTING

AND

INFORMATION SCIENCES

RESEARCH ACTIVITY AI-1

COMMUNICATIONS AND TRACKING EXPERT SYSTEMS STUDY

Prepared by:

T. F. Leibfried, Jr., Ph.D. -- Principal Investigator
T. Feagin, Ph.D. -- Research Area Director
D. Overland, B.S.M.E. -- Research Associate

_ .

In cooperation with: 0. Schmidt -- Technical Monitor/NASA-JSC 30 January 1987

.

University of Houston-Clear Lake Research Activity AI-1 Communications and Tracking Expert Systems Study Interim Report for Semester Period Ending 31 Dec 1986

EXECUTIVE SUMMARY

- The original objectives of the study consisted of five broad areas of investigation:
 - Criteria and issues for explanation of C & T system anomaly detection, isolation, and recovery;
 - 2. Data storage simplification issues for fault detection expert systems;
 - 3. Data selection procedures for decision tree pruning and optimization to enhance the abstraction of pertinent information for clear explanations;
 - 4. Criteria for establishing levels of explanation suited to needs;
 - 5. Analysis of expert system interaction and modularization.

Progress was made in areas 1, 2, 3 and 5, but to a lesser extent in area 4 during Phase 1.

Among the types of expert systems studied were those related to anomaly or fault detection, isolation and recovery. Specifically, the interim results Harris and TRW T&C expert system studies were examined and work supplementing them with explanation facilities was initiated.

An expert system which is rule based may be thought of as a sequence of if-(condition)-then-(action and fact(s)) statements in an endless repeating loop. A given statement or rule, when its condition has been satisfied and it executes, is said to have "fired". The rule usually asserts "facts" which may satisfy the condition(s) of other if-statements or rules which in turn can assert actions and/or other facts and so on. The beauty of these systems is that they are flexible, easily expanded or modified, succinct, and readily understood by humans. Their problems are a general lack of structure, modularity for groups of related rules, and therefore poor maintainability. Also, in complex rule-based systems there are problems controlling the order in which the rules are applied. That is, a "resolution strategy" must be provided. Most expert systems of this type are each essentially one big program where object-oriented design and information hiding are relatively absent. These latter concepts are essential for data integrity of software systems. This is a major issue for any project as big as the Space Station, where the total software system size may be measured in millions of lines of code. The potential benefits of expert systems are too great to reject them out of hand, especially since the modularity issue is probably tractable. Modularity in such systems is being investigated as part of area number 5 above.

1

Results to date indicate that modularity is possible especially in the case of predictable expert systems (i.e., systems where rules cannot make new rules). In fact, virtually any predictable expert system is capable of being rewritten in a procedural language. In some case that may be even desirable for part, if not all, of a given system. This was done in this study for a simple subsystem based upon a system simulator written by TRW. The simulator, written in C, was rewritten in Ada. Then, fault diagnosis and explanation facilities were implemented to investigate area number 1 (explanation facility) and area number 5 (interaction and modularization) described above. The source code for this system is in Appendix B as MAIN SIM MOD3.ADA. Results of this portion of the the study are shown in item D of the body of this report. Briefly, the results indicate that, an explanation facility is best structured as an integral part of the system rather than as an appendage. Object-oriented modularization promotes data integrity, and that the capability of retrieving and using old data is probably best achieved through a procedural language. Object-oriented design was first suggested by Parnas and implies that each object (e.g., variable) should be controlled by one module and the resources necessary to modify or change any of its characteristics should reside only within that module.

The areas number 1 (expert explanation), number 2 (storage simplification) and number 3 (decision tree issues) were investigated by the implementation of a simple engine diagnosis program. The source code and concomitant comments for DIAG4.ART are in Appendix A. The results are shown in item B of the body of the report. Briefly, the results show that rules have a taxonomy (e.g., explanation rules, bookkeeping rules, action rules), and that the time stamping of facts is necessary for explanations of any past expert system actions. The need for functional or domain partitioning was one of the discoveries of this investigation.

Not much implementation of techniques for pruning decision trees of irrelevant decision nodes (area number 3) was accomplished; however, these and other data compression concepts were discussed extensively and some paths of investigation and experimentation are indicated. So-called spine optimization of the decision tree may provide a capability for explaining in retrospect how and why a decision has been reached or even perhaps why a particular decision was not reached. (A spine is defined as the conjunction of decisions for which the retrieval is a single conclusion.)

In the relatively short time this study has been in progress, what were originally nebulous issues have become more clear. This is not to say that they have become more tractable, but at least some of the issues are better defined than they were four months ago. We expect that this trend will continue.

BODY OF THE REPORT

A. Introduction

The original statement of activity objectives consisted of several items summarized as follows:

 Examination of existing C&T systems configuration and monitoring problems--

The activity started with an examination of fault detection expert systems and historical database storage, which is one facet of the subject area of C&T system anomolous behaviour. An anomolous event may be defined as the situation when normal sytems software has been unable to operate according to plan. Some tentative conclusions for structuring these activities have been determined. This activity continues.

2. Audit trail simplification policies--

Some experimental work has been accomplished for this activity. In particular, audit trails for fault detection expert systems have been analyzed. An hypothesis has been formulated, based on preliminary results which indicates that shallow explanations may be able to use a limited audit database but deeper explanation facilities may require a parallel expert system, otherwise the audit trail database might become too cumbersome. The question of the effect of utilizing distributed expert systems on practical explanation facilities has not yet been considered, but that is a subject which will need to be addressed once the resources needed by such facilities have been identified.

3. Data selection procedures--

This activity is closely related to item 2 but has been modified to focus upon decision tree pruning and storage depth of unchosen paths. For explanation purposes one hypothesis to be explored is to store chosen path nodes and one node level below each chosen path node for all unchosen paths. This would, at a minimum, double the storage required for a chosen decision tree but it would obviate the necessity for reexercising a duplicate of parts of the original expert system for shallow explanations. No general rule with specifics has yet been established but some simple subsystem simulations, namely, DIAG4.ART and MAIN SIM MOD3.ADA, have given some insight into this problem. This activity continues.

4. Criteria development for user-expert system interfaces--This activity has been modified to focus upon identifying levels of explanation suited to various user's requirements. It will be assumed that user requirements will have already been identified. It will be assumed that short explanations will be supplied initially and that levels of depth may be requested at any point in the process. This activity has not seen much progress to date but will continue in the revised direction. 5. Expert system interaction--

Intercommunication and hierarchical decision priorities have been discussed but no definitive assertions have yet been determined. At first sight this issue seems related to partitioning and modularizing expert systems. This latter issue is one which is to be explored in the next phase of this study.

B. Explanation Facilities and Decision Trees

An explanation facility will be an important part of any system that is used to provide diagnostics for problems that may develop in the tracking and communication systems on the space station. Also, once the most probable source of a problem has been identified and corrective measures are begun, an explanation of why and how such actions have been taken will be necessary. It would appear that there are several levels of explanation that can be provided. At one extreme, a complete explanation with all the intricacies and details of all of the inferences used could be provided, such that the explanation would be effectively equivalent to providing a complete decision tree (in which only one path has been followed). At the other extreme, a brief, superficial explanation could be provided that would only indicate the most immediate reason why a particular path was selected. This might be of use in the case when a systems human monitor might have selected a different alternative than did the expert system and he seeks the reasons for the selection made by said expert system. In this latter case, it would be practical to store the path and the collection of decisions made along the path. However, simply quoting the path to the given conclusion will not, in general, provide a satisfactory explanation because there may be irrelevant decisions on the path and there may be a need for deeper explanations. It appears that the need for deeper explanations can only be satisfied if the complete reasoning process for reaching a given conclusion is available (i.e., either we provide access to all the rules that were used or we provide the complete decision tree). Such might be needed when a large scale manual intervention is planned for direct control of C&T systems and in that case irrelevant decisions would effectively be "noise" in the system. A procedure is available that would permit the removal of irrelevant decisions. It is called spine optimization, which consists of pruning the decision tree to obtain a "prime spine", and which prime spine is formed by delaying or removing irrelevant decision nodes in the tree. Most of the work so far on spines has been theoretical, so it is not clear that the benefits of such removal would justify the costs.

It may be, in some cases, that the option of providing the complete decision tree would not be practical in light of the large amount of storage that this would require for each of many points in the past. It is possible, however, that a few of the most recent decision processes could be stored in their entirety and that a number of older decision processes could be saved in an abbreviated form (perhaps with just a simple rundown on the decision path actually taken with only immediate explanations for taking a particular path provided). If more elaborate explanations were required for older decisions, then it would still be possible, albeit time-consuming, to reload another knowledge base with the facts that were true at the time of interest and then begin to execute some of the explanation rules over again, only this time with the user interrupting that expert system and asking questions about the decisions which were made.

One of the dilemmas encountered with after-the-fact explanation facilities is that the explanation facility itself must provide some constructive filtering. Indeed, in order to remove irrelevant decision information the explanation program must prune and perhaps redo, albeit with a different perspective, some of what the diagnostic facility did in the first place. The simplest thing to do would be to supply the complete decision tree, but this would be barely one cut above the Automate Reasoning Tool (ART) dribble file as far as user utility is concerned.

An hypothesis or question which probably is worth considering is, "Should the explanation facility 'anticipate' queries so that it would effectively run in parallel to the diagnostic expert facility but without encumbering it?" Another possibility is to initiate the building of a more extensive event data base whenever an anomolous condition occurs.

At this point, without making a definite statement, let us offer a conjecture. We are of the opinion that storing the entire decision tree may be a viable solution, after all. It's not really as bad as storing the entire state at several points in time -- which might be needed since some problems will be building up over time -- in order to be able to provide explanations. The decision tree information would be kept active for a minimal time, presumably until time had passed when an explanation might be required, and then archived. We think that the decision tree also has just those rules used/needed to provide the explanation. If the other approach is used, we would have to ferret out those rules that were relevant or reload the entire expert system.

- C. The ART Environment and the Attempt to Implement an Elementary Simulation, Diagnostic and Explanation System
 - The ART Environment

There are many advantages and some deficiencies to the Automated Reasoning Tool (ART) environment. The flexibility allows both forward and backward chaining with schema (which provides memory slots similar to frame-based languages), and viewpoints. The viewpoints may provide a way of recalling how a particular expert system operated at a given time in the past. This is something which would be useful for providing an explanation of a past event. Unfortunately, the first diagnostic and explanation example program did not use viewpoints and the lack of the availability of file input-output limited the scope of the program. Useful knowledge was obtained, nevertheless, and that should help the development of future explanation experiments.

Recommendations on Expert Systems Languages

A few observations have resulted from the first diagnosticexplanation system. The ART language coupled with the Symbolics system is a very versatile, albeit sometimes clumsy, approach to expert system development. The feasibility of developing an in-house (Ada-based) rule-based language should be investigated. This would allow data structures and I/O requirements to be tailored for the application.

This could also allow the development of subprocedure calls for both subprocedures written in sequential languages and those written in the development language (in other words, calling other expert system programs).

The DIAG4.ART Diagnostic-Explanation Program

Objectives

The goal of this program was to demonstrate at least a rudimentary explanation facility on a limited domain, mainly as a means of exploring the concepts involved, and also as a means of learning the ART language and Symbolics system.

Description

The program, written in ART, simulates the operation of a four-stroke, two-valve, single-piston internal-combustion engine. It also diagnoses failures in the ignition phase of the engine operation, and implements corrective action. It is then capable of explaining the diagnosis and the corrective action taken.

The program was also written so as to reflect some logical organization: The explanation rules are at the beginning of the program, followed by the bookkeeping rules (those responsible for updating the current parental and subgoals), followed by the initializing "split" rules, followed by the rules for action at each stroke of the cycle. The "split" rules were made necessary because the condition portion of a rule in ART does not have a provision to match on two OR'ed schema slots.

Program Outline

The engine state is modeled by a schema named CURRENT which is modified by the action of the stroke and ignition rules to reflect the operation of the engine. Each relevant component of the engine state is stored in a slot of the schema. At the same time, each slot in CURRENT is compared to a similar slot of the IDEAL engine state. Discrepancies, such as the spark plug not firing, cause error flags to be set which allow the diagnostic rules to fire. In this case, each flag triggers one diagnosis, but combinations of flags could also do this.

Diagnostic rules printout specific error messages and take corrective action (replacing the spark plug). They also query if an explanation is required. If one is, another flag is set.

The combination of the error and explanation flags allow the firing of explanation rules, of which there are two in this program.

Conclusions

The program served as a learning process, but ended up with serious deficiencies:

The fault tree was never more than two levels deep (both spark plugs fail) and did not demonstrate any combinatorial failures nor did it ever have to "guess". There couldn't be a wrong diagnosis.

All explanations were developed as the program "sequenced" through the simulation. No explanations could be given after the cycle continued: the program has no memory of what has already happened; it can only respond to the current state. This could be remedied by time-stamping all facts (saving the telemetry stream), but this would also require a new set of rules to react with past data in addition to those already existing which react with current data, effectively doubling the size of the program (at least).

All explanations either exist from the beginning or they could not be given. The drawback with programming in this style is that all possible faults and fault combinations must be figured in advance and coded into the program. This is trivial for a program this size, but is not practical for a large program. This increases the size of the program exponentially with the number of components. It also requires being able to simulate the "correct" functioning of the system at all times in order to detect discrepancies.

The source code for this program is shown in Appendix A.

Proposed Further Avenues of Exploration

The concept employed with this program would only allow explanations to be given "on the fly", that is, only after each diagnosis and/or fault correction. Explanation cannot be generated past that point in time; there is no memory of the transaction.

To avoid this fault then in order to generate explanations of past events either:

Save all explanations for possible future recall, or;
 It must be possible to generate explanations from saved data.

There are a number of unexplored avenues here, such as the optimum way of generating explanations, the best way of storing data, etc.

Another concept to be explored is that of partitioning the expert system. The program, as currently written, does not support this, but the idea of breaking the system into modules, either functionally, where a module would perform certain tasks, or by domain, where each module would service a system or subsystem of the overall domain. The ability to do so would have quite an effect on the memory requirements and speed of the system. D. Ada Implementation of a C&T Subsystem Simulator (TRW), Fault Detection and Explanation Facility with User Interaction

Motivation (objective)

This demonstration program was initiated for two reasons. First to demonstrate how a sequential software system could effectively duplicate the results of a simulation written in C and an expert system written in a specialized rule-based language such as ART (Automated Reasoning Tool by Inference Corp.). The second reason was to demonstrate that in contrast to conventional rule-based systems a sequential system can more easily store facts in a database and access them for an explanation facility. An ancillary reason was to examine how modularization could be used when implementing both diagnostic and explanation facilities.

Structure (program outline)

The program, written in Ada, consists of five modules called packages.

1. MAIN SIM MOD3

This is really not a module but rather the driver program for the demonstration system. It essentially structures the system by calling subprograms. A simplified algorithm for this program is given in the following enumerated steps.

- (1) Initialize the communications system database;
- (2) Check for inconsistencies in the data, and if there are any, call a subprogram to ask the human monitor to correct the data;
- (3) Call the equipment emulator program;
- (4) Based upon the measured equipment output and the status of the switches, determine the condition of the equipment (i.e., diagnose the probable cause of failure, if any);
- (5) Call a subprogram to display the status of the equipment to the human monitor;
- (6) Call subprograms to provide an explanation of the diagnosis if requested;
- (7) Call a subprogram to interact with the human monitor to see if another simulation is to be run;
- (8) If the human monitor wishes to stop the program then terminate else call a subprogram to ask the monitor to update the simulation parameters and then go to step (2).

2. Y_OUT_MOD

This package contains the equipment emulator subprogram. It effectively simulates the action of the hardware given the parameters in the equipment status database.

 FAULT-ANALYZE This package contains the subprograms which measure the observable parameters and determines the probable fault, if any.

- 4. EXPLAIN DIAGNOSIS This package presents an analysis of the reasons behind the fault diagnosis when requested by the human monitor.
- 5. I O SIM MOD This package is the one which accesses and updates the simulated equipment parameters.

The source code for this system is shown in Appendix B.

Observations Based Upon Results

Not too much can be asserted with certainty but there are a few points which the work suggests. Among those are:

To explain even a moderately complex fault analysis decision 1. it may be simpler to parallel a part of the decision process rather than try to filter the information from information written into the data base. Again, a trade-off exists between the classical performance parameters of execution time versus memory (primary and secondary storage). For example, in the diagnostic package called FAULT_ANALYZE the principal program DIAGNOSIS tests the output power EQUIP Y OUT LEVEL and if it is less than -145 dbm it calls a "low level" program to examine the on-off switches in SWITCH STATUS. If the main power switch is "ON" then the oscillator switches are tested. If the selected oscillator is "ON" then the program "reasons" that the selected oscillator is inoperative. Now when an explanation of this event is requested the EXPLAIN DIAGNOSIS package is activated. Examining the code for procedure EXPLAIN DIAGNOSIS we see that it parallels the reasoning of procedures DIAGNOSIS and SWITCH STATUS in package FAULT ANALYZE, that is to say, it has the same nested "if" structure. The only thing it adds is the diagnostic message, "We found the selected oscillator switch to be ON, the power output was in the noise level, so potentially we had a catastrophic oscillator failure." If the DIAGNOSIS and SWITCH_STATUS programs had been more cooperative they could have selected the appropriate literal value for an enumeration variable and stored this "hook" in the data base for access by the EXPLAIN DIAGNOSIS package. Then all the EXPLAIN DIAGNOSIS procedure would have to do is examine this variable by virtue of a simple "case" statement and supply the quoted explanation, "We found..etc."

Observations Based Upon Results - continued

Such a "type" definition for the desired variable might be:

type Switch_Permutation_Type is
 (POWER_OFF,
 POWER_ON_OSC_OFF,
 POWER_ON_OSC_ON);

A variable of this type could be set equal to one of the enumeration literals by procedure SWITCH_STATUS in package FAULT ANALYZE.

- 2. Communicating with a user and writing facts to files and/or ephemeral data storage which facts are useful for explanation facilities may require interfacing a given expert system to procedural language I/O routines.
- 3. It seems that it should be possible to modularize a given expert system to some extent. For example, a FAULT ANALYZE subsystem could in large measure be separate from an EXPLAIN DIAGNOSIS subsystem. There would probably be some shared data and possibly even some shared utility routines, but the main thread of each subsystem could be separate.
- 4. One of the deficiencies of this implementation is that explanations are done with the same database as the fault analysis system, and before any recovery corrections are made. Thus, the database is unchanged when the explanations are made. The programming system could easily be altered to allow changes in the data and still allow an explanation after the fact. The technique employed would be to create two variables for each entity, one for the old value and one for the current value. This is easily done in Ada or in any procedural language but is more difficult in a so-called expert systems language. This may indicate that any production system would require its expert systems language to provide an interface to procedural language subprograms.

Future Direction

The next task which may be proposed is to expand this demonstration program to include more realistic fault detection with more realistic "hook" data generation. Then the explanation subsystem could be restructured to use these improvements. A parallel system in ART will also be implemented (if feasible).

SUMMARY OF OVERALL RESULTS AND RECOMMENDATIONS

- A. The issue of storing the complete decision tree versus only storing the knowledge base for an explanation facility has not been resolved, but a few alternatives to be explored have been identified.
 - 1. Store the decision tree in its entirety with perhaps one branch node for paths not taken at each node in the decision tree.
 - 2. Build and store a complete knowledge data base whenever an anomaly occurs. (This could be any undesirable outcome such as the software system displaying unanticipated behavior. This could be signalled by the astronaut monitor or the expert system itself such as when a system failure is detected by the low level systems.)
 - 3. Store only the facts with appropriate time stamps, and when an explanation is required, load a system containing rules similar to the original expert system so as to effectively parallel the operation of the original expert system but this time allowing the user to interrupt this parallel system to ask for appropriate explanations about paths not taken.

These alternatives are not mutually exclusive but all should be examined in future work.

- B. The issue of the features of an expert systems language which language is appropriate for development and perhaps implementation of software systems which meet C & T functional requirements needs to be addressed. This recommendation is based upon the experience acquired by the structuring of a simple simulation and diagnostic-explanation program written in ART, (DIAG4.ART in Appendix A), the program had to be all in one module. In addition, there were the file I/O deficiencies of ART and the difficulty of storing more than one value for a given parameter. As a minimum any such language should be capable of interfacing with a compiler language program in a straightforward manner.
- C. It is recommended that the issues of object-oriented design for expert systems be raised and investigated. In addition, the concept of supplying adequate "hooks" for explanation should be addressed early. Explanation facilities are best implemented when they are built in and not just "added" as a separate entity. If this is not done the explanation facility requires additional resources and must parallel some previously "paths" already trodden by other systems. The issues of object-oriented design, modularization, and so-called information hiding may not be just academic but a necessity for implementation of any large and perhaps distributed control and monitoring system, be it a procedural and/or rule-based software system. The topic of objectoriented design for expert systems addressing anomaly detection, recovery and explanation was examined by structuring a procedural software system in Ada. The system consisted of a driver and hardware status simulator, fault diagnosis, and fault explanation modules for a small radio frequency communication subsystem. Implementing expert system functions in Ada indicated that modularization and objectoriented design indeed are feasible without compromising effectiveness.

APPENDIX A

Source Code for DIAG4.ART

an Engine Diagnostic/Explanation Program

۰.

_

MAX:>overland>diag4.art.23

```
;;; -*- Mode: ART; Base: 10.; Package: ART-User -*-
(defschema engine-state "state of engine - beginning of intake-stroke"
           (carburation good)
           (piston-direction descending)
           (sparkplug1 good)
           (sparkplug2 standby))
(defschema current
           (instance-of engine-state))
(defschema ideal
           (instance-of engine-state))
(deffacts state
          (state-name intake))
(defrule compare-ok "compares current and ideal if same"
         (schema current (carburation ?state3))
(schema ideal (carburation ?state3))
          (schema current (piston-direction ?state4))
          (schema ideal (piston-direction ?state4))
~ >
          (printout t t "compared ok" t t))
(defrule compare-not-carburation
          (schema current (carburation ?state3))
          (schema ideal (carburation ~?state3))
= >
          (printout t t "carburation compared not ok" t t)
          (assert (error-trap carburation)))
(defrule compare-not-direction
          (schema current (piston-direction ?state4))
          (schema ideal (piston-direction ~?state4))
=>
          (printout t t "piston-direction compared not ok" t t)
          (assert (error-trap direction)))
{defrule no-ignition
          (state-name power)
          ?ignition <- (ignition fail)
≈>
          (printout t t "sparkplug did not fire" t t)
          (printout t t ?ignition t t)
          (assert (error-trap ignition))
          (retract ?ignition))
(defrule intake-stroke
          ?state-name <- (state-name intake)</pre>
          (schema current (carburation ?state3))
          (schema ideal (carburation ?state3))
          (schema current (piston-direction ?state4))
                          (piston-direction ?state4))
          (schema ideal
=>
          (printout t t "1st Stroke - Intake Completed" t t)
          (modify
                 (schema current
                          (piston-direction ascending)))
          (modify
                 (schema ideal
                         (piston-direction ascending)))
          (retract ?state-name)
          (assert (state-name compression)))
 (defrule compression-stroke
          ?state-name <- (state-name compression)</pre>
          (schema current (carburation ?state3))
          (schema ideal (carburation ?state3))
          (schema current (piston-direction ?state4))
          (schema ideal (piston-direction ?state4))
 =>
          (printout t t "2nd Stroke - Compression" t t)
          (modify
                  (schema current
                          (piston-direction descending)))
          (modify
```

ORIGINAL FAGE IS OF POOR QUALITY

MAX:>overland>diag4.art.23

12/01/86 13:08:16 Page 2

```
(schema ideal
                         (piston-direction descending)))
         (retract ?state-name)
         (assert (state-name power)))
         (printout t t "Did sparkplug : fire or fail?" t t)
;
         (assert (ignition = (read))))
(defrule power-stroke
         ?state-name <- (state-name power)</pre>
         ?ignition <- (ignition fire)</pre>
         (schema current (carburation ?state3))
         (schema ideal (carburation ?state3))
         (schema current (piston-direction ?state4))
         (schema ideal (piston-direction ?state4))
= >
         (printout t t "3rd Stroke - Power" t t)
         (modify
                 (schema current
                        (piston-direction ascending)))
         (modify
                 (schema ideal
                        (piston-direction ascending)))
         (retract ?state-name)
         (assert (state-name exhaust))
         (retract ?ignition))
(defrule exhaust-stroke
         ?state-name <- (state-name exhaust)</pre>
         (schema current (carburation ?state3))
         (schema ideal (carburation ?state3))
         (schema current (piston-direction ?state4))
         (schema ideal (piston-direction ?state4))
=>
         (printout t t "4th Stroke - Exhaust" t t)
         (modify
                 (schema current
                         (piston-direction descending)))
         (modify
                 (schema ideal
                        (piston-direction descending)))
         (retract ?state-name)
         (assert (state-name intake)))
(defrule switch-plugs
         ?error-flag <- (error-trap ignition)</pre>
         ?state-name <- (state-name ?name)</pre>
         (schema current (sparkplug2 standby))
= >
         (modify
                 (schema current
                         (sparkplug1 fail)
                         (sparkplug2 good)))
;
         (retract ?error-flag)
         (retract ?state-name)
         (assert (state-name compression))
         (printout t t "Sparkplug1 set to Failed" t t)
         (printout t t "Sparkplug2 reset from standby to good" t t)
          (printout t t "Is an explanation desired? yes or no" t t)
         (assert (explanation-flag =(read))))
(defrule no-plugs
         ?error-flag <- (error-trap ignition)</pre>
         ?state-name <- (state-name ?name)</pre>
         (schema current (sparkplug2 good))
         (schema current (sparkplug1 fail))
=>
          (modify
                 (schema current
                         (sparkplug2 fail)))
         (retract ?error-flag)
;
         (retract ?name)
          (assert (state-name compression))
         (printout t t "Sparkplug2 set to Failed" t t)
         (printout t t "Is an explanation desired? yes or no" t t)
          (assert (explanation-flag = (read))))
```

OF POOR QUALITY

12/01/86 13:08:16 Page 3

MAX:>overland>diag4.art.23

(defrule ignition-fire (state-name power) (split ((sparkplug1 current good) ≃>) ((sparkplug2 current good) =>)) - > (printout t t "Did sparkplug : fire or fail?" t t) (assert (ignition = (read)))) (defrule explanation-1 ?error-flag <- (error-trap ignition)</pre> ?ignition <- (ignition ?fire) ; (schema current (sparkplug1 fail)) (schema current (sparkplug2 good)) ?expl-flag <- (explanation-flag yes)</pre> =" **>** (printout t t "Sparkplug 1 did not fire. Therefore it was replaced by the backup." t t) (retract ?error-flag) (retract ?expl-flag)) (defrule explanation-2 ?error-flag <- (error-trap ignition)
?ignition <- (ignition ?fire)</pre> ; (schema current (sparkplug1 fail)) (schema current (sparkplug2 fail)) ?expl-flag <- (explanation-flag yes)</pre> : > (printout t t "Sparkplug 2 did not fire. Since it is the backup sparkplug, (sparkplug 1 has already been considered failed) there is no remedy. However, since it is unusual to have both sparkplugs failed, the problem may reside elsewhere." t t) (retract ?error-flag) (retract ?expl-flag))

APPENDIX B

Source Code for System MAIN_SIM_MOD3.ADA

a Communication Amplifier Diagnostic/Explanation Software System

{ TYP_DEF 01	/ WIS_	1	~	~	-	-	-	}	<pre></pre>
F		package TY	PEF	SIM is					
01 M 4 M 60		type SW type FA type SC type CO	VITCH 1 NULT 15 NURCE 1 NNTINUE	s (ON,OFF); (ACTIVE,IN) s (A,B); is (YES,NO	CTIVE);				
8 1		end TYP_Di	EF_SIM;						
PSECT 1	(AP								
Psect 0	Hex Size 00000037 000000037	Dec Size 55 12	TY TY TY	P_DEF_SIM_\$ P_DEF_SIM_\$	CODE CONSTANT	_			

%ADAC-I-CL_ADDED, Package specification TYP_DEF_SIM added to library Replaces older version compiled 17-Nov-1986_00:53

Psect Hex Size 0 00000037 1 00000000

There are no uses of potentially non-portable constructs

PORTABILITY SUMMARY

 Punction Pun	 Punction Punction Punction Punction Punction Punction Punction Propertion and measurements of a regio frequency from the proceedure "optimization personness" Propertion and parameters are read from the antipart optimization personness of proceedure "optimization personness" Processing P	1	Main program for Equipment Simulator/Diagnose
<pre>is in place. There is the place.</pre>	<pre>1 is in place. 1 is in place. 1 Paur 1 Paur 1</pre>	1	Function To simulate operation and measurements of a radio frequency power source and analize any anomalies. An explanation facility
 Indiant of teasers is analytical parameters are read from a file partial parameter or standard interactively via the percendent roused of tease Fouranty in parameter are entred interactively via the exponential parameter or parameter or parameter or parameter or parameter are entred interactively via the exponential proceeding are checked for validity for the area or properly in the proceeding are checked for validity by the event of the standard proceeding are explored are portable or provide are are entred interactively via the proceeding are checked for validity by the event of the standard proceeding are checked for validity by the event of the standard proceeding are event. Proceeding are explored are proceeding and by proceedings are distributed and the parameter are and by proceeding a standard proceeding and a proceeding are are are are are are are are are are	<pre>runt. runt</pre>		is in place.
 Py procedure 1 © ATT in parameters are read for an file Py procedure 1 © ATT in parameters are noticed interactively via the hyboroid. Py procedure 1 © ATT in parameters are noticed interactively via the hyboroid. Proceeding 1 © ATT in parameters are noticed interactively via the hyboroid. Proceeding 7 OF 900 Hyboroid. Proceeding 2 OF 900 Hyboroid. Proceeding 3 OF 900 Hyboroid. Proceeding 4 OF 900 Hyboroid.<td> Dittail for source standards by the activation of from a file styber. Procedure 1_0_STTT in percept or 0_STT the context of a standard stype stype of the activation of the strandard strandard of the strandard strandard</td><td>1 1</td><td>Input</td>	 Dittail for source standards by the activation of from a file styber. Procedure 1_0_STTT in percept or 0_STT the context of a standard stype stype of the activation of the strandard strandard of the strandard strandard	1 1	Input
 by proceedures of these parameters are anticed interactively via the exceedence of the simulator values of these parameters are anticed interactively via the simulator the own as property that interactions are fed to the simulator the positions are fed to the simulator the position is required interactively via the simulator the position is required interactively via the simulator the position is required interactively the simulator the simulator the simulator is required in the sector of the simulator is required by the sector of the disposition. cuptuation is required in a simulator by communicated by the of the CFR and the sector of the disposition is communicated by the of the disposition is required in the sector of the disposition is required to the simulator by reading the equipaent parameters from a file. cuptuation is required to the disposition of the simulator by reading the sector is a simulator by reading the equipaent parameters from a file. cuptuation is required to the simulator by reading the sector is a simulator by reading the sector is the simulator by the motion is the simulator by reading the simulator by reading the sector is the simulator by reading the sector is the simulator by reading the sector is the simulator by reading the sector is the simulator by reading to sectio is the s	 by proceeding a summary and a suffered interaction for validity exponent. Proceeding presents the antired interaction for validity into a submary of these presents only construct a supervised on any proceeding the present only construct a supervised for validity from the antired mark proceeding to only construct a supervised to the standard on the construction of the proceeding. Proceeding and the present only construct a for construct a supervised for the standard on the construction of the construction of proceeding. Proceeding a supervised are presented only construct a for construction. Proceeding a supervised are presented by the supervised for the standard on the standard data of the standard for the standard on the constructed by the standard data of the standard for the st	1	Initial rf source simulation parameters are read from a file
<pre>propriet. Processing The input presenter organization survey besting and survey and survey and survey of proprist, the input presenter and survey and survey of proprist and survey and survey and survey and survey provide the propriet and survey provide the angle of the disposite and by proceedures the survey and survey provide the disposite and by proceedures the survey and s</pre>	<pre>propriet. provide array property, the input proceedures are checked for validity the input proceedure size only construct and procedure statistics. the nonstruction for the input procedures the article are assumption processing. Torm NOO waits present are a force the statistics. the nonstruction for the disposite and by procedures Pag. SWITCH, PLANO0515. Proceeding and the disposite and by procedures Pag. SWITCH, STATUS and output device areas. are monitoring to disposite and by procedures Pag. SWITCH, STATUS and output device areas. Apportune MAIL STA MOD is communicated by them to the CAT apportune MAIL STA MOD in a algorithm word for the driver program is: I fulfialize the simulator by reading the equipment parameters from a file. While the human monitor desires to continue remain the human monitor desires to continue remain the simulator in parkeds are surpled on and if so force a to file. 1. Algorithm are from any statistic by reading the squipment parameters from a file. 3. While the simulator desires to continue remain the simulator in package from and if so force a context. 3. The simulator in package from and if so force a to file. 4. All suffichers are from and if so force a the antitics are are are are to be and the solution: 1. All suffichers are from and the solution in the bold. All are from a simulator in package from the antick is off or if on if bold. 1. All suffichers are in the off position: 1. All suffichers are in the off position: 1. All suffichers are are and the caused by contlater of the solution is block to be and the a the area drive the simulator output and the fractor is a thread to area and the at a matter area the and the caused by contlater in presentably the the at all suffichers are in the off position: 1. All the output is area area that caused by and and the caused by carcit and the application output is at position than the failure 1. All the output is a positive is a position. 1. All the output is a positis are output and area and</pre>	1	by procedure I O INIT in package I O SIM ; thereafter revised
Processing The year of properly, the input primaters are fead to the simulator then, when set properly, the input primaters are fead to the simulator protector from this termination primaters are fead to the simulator upon return from this termination on the conduct approximation to any explored the fragmention of the conduct protector by the transplantion is forgonia and by proceedures PML SWICH STATUS and transplantion is forgonia and by proceedures PML SWICH STATUS and the angle davies are for the driver proceedures PML SWICH STATUS and over issuing a strenge 1 or SML MOD is communicated by them to the CMT appoints must set the driver program is: the algorithm ward for the driver program is: the algorithm used for the driver algorithm to the algorithm is and the simulator the algorithm for the alfernate oscillator: the applifier on driver the monitor to suich is off or if or if or if the is the applifier and driver the is allowed to near the algorithm for the alfernate oscillator: the applifier and driver the is allowed to near the is allowed to orbit and the output curved by an output curve drive the is allowed to near the output curve the applifier and output to reased by an output or the alfernate oscillator: the applifier is driver the monitor to suich by endicing the output curved by any orbit or is allowed to orbit tor and by and output tor the alfernate oscillator: t	 Processing <		kavboard (citate gained contraction of the contrac
 Processing Proving an experience of the present of the clock a statistic the character structur from the new possible of the input presenters are fact the statistic of the monitored lass upon the resent only consist of procedure building. Proving a solution is requered, same is supplied via KIPAAN_DIMENTY_O. Output the statistic structur from the resent of the output presents and the structure from structure structure and the resent of the dispositions. Output the statistic structure of the disposition are considered by Them to the CHT Algorithm MAIN_SIM_MOD Algorithm used for the driver program is: Algorithm used for the driver program is: I miticalize strom. I miticalize the simulator by reading the equipment parameters from the the human monitor dustres are switched on and if so force a filter of the simulator in present strom strome stromes. I miticalize the simulator in profession of the driver program is: I miticalize the simulator by reading the equipment parameters from the the bulk of the simulator by reading the summer strome structure strome strome strome strome structure strome strome structure structure strome strome strome structure structure strome structure structure structure structure structure strome strome structure structure strome structure s	 Processing Provide the input primeters are fact to the sublact the input the input branches series for the sublact the input primeters are fact to the sublact the privation is reprised by then set provide a position. Prove an explanation is requered, same is supplied via UTFLART_INDIANS'. Output an explanation is requered, same is supplied via UTFLART_INDIANS'. Output and the adapting a tentary diagonis is and the set of the sublact set of the sublact set of the sublact set of the standard set of	;	
 The input presenter or solitator switch positions are feat for the available. Then, when set property, the input present only consist of proceedure state of a position is requere the outprocense are feat for an application is requere the outprocense are are are are are are are are are ar	 The input primare oscillator satich positions are chocked to variaty providency of the position are properly than the input primares as is applianted with a simulator of the other present only consists of procedure spin surfaces. Utput are application is request, and survey consists of procedure spin survasts. Output are application is requested, such as is applied via KTRMARIE parkameters. Output are applied and a signation of the cost of the co		Processing
 then, when set projections the output parameters are fad point. then, when set projections the output power and solicity positions are applicable of the disposition is requested, state is supplied via EUTPUL Plantation is requested, state is supplied via EUTPUL Plantations. OUDT results of the disposition and by procedures PNR SWITCH STATUS PLAGNOSIS. OUDT results of the disposition and by procedures PNR SWITCH STATUS PLAGNOSIS. OUDT results of the disposition and by procedures PNR SWITCH STATUS PLAGNOSIS. Algorithm MAIN SIM MOD is communicated by Them to Tether EUTP and the adjoint parameter for a signification by reading the equipment parameters from a file. Algorithm used for the diver programis: Algorithm MAIN SIM MOD Algorithm MAIN SIM MOD Algorithm Bund for the diver programis: Algorithm MAIN SIM MOD Algorithm Walk SIM MAIN SIM Algorithm the simulator in package VUT MAIN SIM All subtributes are proper and output is greater than noise but less choice in the subjust of fourties four holds: Algorithm that caused by an output is greater than noise but less the subjust of the subjust of solid side statication four the statust of the subjust of the subjust of solid side statication of the subjust of the subjust of solid side statication of solid side state state of the subjust of solid side statication of solid side state state of the subjust of the subjust of solid side statication of the subjust of the subjust and subjust and subjust	 then, when set projective, the input parameters are red to the statustor of the statustor of the state of the descent is noticed as a supplication is requested, same is supplied via UTEALN DIANNESS. ners manifored. Base upon the redding the advected parameter state output sets an application is requested, same is supplied via UTEALN DIANNESS. output sets an explanation is requested, same is supplied via UTEALN DIANNESS. output sets of the degrees and by procedures phy SWITCH STATUS and output sets and the state of the output sets. output sets of the degrees and by procedures phy SWITCH STATUS and output sets are supplied via UTEALN DIANNESS. output sets of the degrees and by procedures phy SWITCH STATUS and output sets are included by them to the state of the output set of the output is output to set of the output is output to set of the output is output to set of the output set of the output is output to set of the output is output to set of the output is output to set of the output set of the output is the output to set of the output to set		The input parameter oscillator switch positions are checked for validity
 processy for yob which at present only consists of proceedure SUCHAMARY.". transplanation is requered, same is supplied with ETEALME_DIAGNOSIS. transplanation is requered, same is supplied with ETEALME_DIAGNOSIS. transplanation is requered, same is supplied with ETEALME_DIAGNOSIS. output from the sequence is an is supplied with a ETEALME_DIAGNOSIS. and the results of the diagnosis and by proceedures PWR_SWITCH STANDS and the certain and switch presents. algorithm while STM_MOD is communicated by Them to the CRT the algorithm used for the driver program is: algorithm used for the driver program is: rimitalize the simulator by reading the equipment parameters from a file. while the human monitor desires to continue. transpace the simulator by reading the equipment parameters from a file. while the human monitor desires to continue. while the human monitor desires to continue. transpace the simulator is proceed are switched on and if so force a colore: the same to the simulator the state of the side of the site of the continue. the simulator in parkage Y_OUT_MOD: the sublick the same are switched on and if so force a colore: the sublick the same are switched on and if so force a colore: the sublick the same are switched on and if so force a colore: the sublick the same are switched on and if so force a colore: the sublick the same are switched on and if so force a colore: the sublick the same are switched on and if so force a colore: the sublick the same are switched on and if so force a colore: the sublick the same are switched on and if so force a colore: the sublick the same are	 package to OVT MOCH which at present only consists of precedure should with at protein the standard of the standa	1	then, when set properly, the input parameters are fed to the simulator
 Term the state of the output power and service parameters are monitored. Here the output for the state of the output power and service parameters are subjusted. The results of the diagnosis made by proceedures NWE SWITCH STATUS and the results of the diagnosis made by proceedures NWE SWITCH STATUS and the results of the diagnosis made by proceedures NWE SWITCH STATUS and the results of the diagnosis made by proceedures NWE SWITCH STATUS and the results of the diagnosis made by proceedures NWE SWITCH STATUS and the results of the diagnosis made by proceedures NWE SWITCH STATUS and the results of the diagnosis made by proceedures NWE SWITCH STATUS and the results of the diagnosis made by them to the city of the results of the diagnosis made by the equipment parameters from a file. Mile the human monitor desires to continue respect 2 thru 9; Test if both A is B sources are switched on and if so force a continue respect 2 thru 9; Test the results of the simulator in parkage PULT ANAITXED; Run the simulator in parkage Y OUT 900; Run the simulator in parkage of OUT 900; Test the results of the simulator is posted witch is of or if on if both notify the matter statich is discribed in and the failure the notif or the situated to excite the notif or the output is greater than the output is greater than the failure than the notifue results of the simulator to switch to the alternate oscillator; Test the output is greater than that for normal operation than the failure of the output is greater than the source and the subjust of the normal operation is the the norisit of the output is greater than the source of the situated to excite to subjust the output is greater than the source of the source of the normal operation then the failure of the output is greater than the failure of the source of the normal operation then the failure of the output is greater than that for normal operation then the failure of the output is greater than that for normal oper	 The naviplantion is requested, sum is supplied wis EXPLATE DIAGONSIS. The naviplantion is requested, sum is supplied wis EXPLATE DIAGONSIS. The request of the diagnosis made by procedures PMR SWITH DIAGONSIS. Algorithm WAIN SIM (A) is constant and by the tip is the CR and supplied wis EXPLATE DIAGONSIS. Algorithm WAIN SIM (A) is constant and by the tip is the CR and supplied wis EXPLATE DIAGONSIS. Algorithm WAIN SIM (A) is communicated by the tip is the CR and supplied wis EXPLATE DIAGONSIS. Algorithm WAIN SIM (A) is communicated by the tip is the CR and supplied with EXPLATE DIAGONSIS. Algorithm WAIN SIM (A) is communicated by the tip is the CR and supplied with EXPLATE DIAGONSIS. Algorithm WAIN SIM (A) is a subject by reading the equipment parameters from a file. While the heat man monitor desires to continue. The tip is allocing in package Y (A) (A) (A) (A) (A) (A) (A) (A) (A) (A)	1	package Y_OUT_MOD which at present only consists of procedure EQUIPMENT_Y_O.
 If an explanation is requested, same is supplied via EXTLAN DIAGNOSTS. OUTPUT INTERNIES of the diagraph of the reachest of supplied via EXTLAN DIAGNOSTS. OUTPUTENTEL IN proceeding and by proceedures PWE SWATCH STATUS and OUTPUTENTS. Algoritha MAIN_SIM_MOD Algoritha MAIN_SIM_MOD Algoritha MAIN_SIM_MOD Initialize the simulator by reading the equipment parameters from a file. Initialize the simulator by reading the equipment parameters from a file. The algoritha work for the driver program is: Initialize the simulator by reading the equipment parameters from a file. The algoritha work of the desires to continue Sector 2 thru 9; Test if both A t B sources are switched on and if so force a choice; Test the results of the simulator (package FAULT AMAIZE); Test the results of the simulator (package FAULT AMAIZE); If the notput is package Y_OUT_MOD); Test the results of the simulator (package FAULT AMAIZE); If the notput is package and output is greater than noise and in for if on if both the A t B suttches are in the off position; If and the output caused by an output is greater than noise to excite the alternet oscillator output is greater than presumably the sale test of the simulator of system is found to excite the output is greater than noise to excite the output is greater than noise to evolve the alternet oscillator; If output is greater that that contoural operation then the failure output is greater than the output is greater than noise to evolve the alternet oscillator output is greater than that for normal operation then the failure 	 If an applanation is requested, same is supplied via EXPLAN_DIAGNOSTS. Output Output The results of the diagonis made by procedures PMR_SWITCH STATUS and output device screen. Algorithm MANR_STM_MOD Algorithm MANR_STM_MOD Algorithm WANR_STM_MOD Anst is both A is sources are switched on and if so force a colore. Anst is both A is sources are switched on and if so force a colore. Anst is both A is sources are switched on and if so force a colore. A the he submits in one of position: A the simulator in package Y_OUT_MOD; A the simulator is noise (1:0145.0 dbm) than Cales which was and the A is switches are output is greater than noise but is solved. A subtreas are proper and output is greater than noise but is both the color source sources are suitched to a strain the children source. A subtreas are proper and output is greater than noise but is solved. A subtreas are proper and output is greater than noise but is solved. A subtreas are proper and output is greater than noise but is solved. A subtreas are proper and output is greater than noise but is solved. A subtreas are proper and output is greater than noise but as a solved. A subtreas are	1	Upon return from this procedure the output power and Swirch positions
If an explanation is requested, such that is supplied with any and output device screen	If an explanation is requested, same is supplied via extract	1	are monitored. Base upon the readings a tentarized diagnosis is made.
 Output Output The results of the diagnosis made by procedures PWR SWITCH STATUS and The results of the diagnosis made by procedures PWR SWITCH STATUS and output device screen. Algorithm MAIN_SIM_MOD Algorithm MAIN_SIM_MOD The algorithm used for the driver program is: nitialise the simulator by reading the equipment parameters from a file. Mhile the human monitor desires to continue repeat 2 thru 9; Minise the simulator by reading the equipment parameters from choice: Test if both A & B sources are switched on and if so force a choice. Test if both A & B sources are switched on and if so force a choice. Test is both A & B sources are switched on and if so force a choice. Test is both A & B sources are switched on and if so force a choice. Test is both A & B sources are switched on and if so force a that the application states (in the off position); Test is the output is assultator (passite han the switch has off). Test the output caused by no upput is greater than noise but less than the output caused by an output is greater than noise but less than the output caused by no upput is greater than noise but less than the output caused by no upput is greater than noise but less that the output caused by no upput is greater than noise but less that the output caused by no upput is photow the threshold needed to excite selecter denotice is and output is below the threshold needed to excite the auguitier to avaite the anonicor is writch to the alternate oscillator; Te output is greater than that cor normal operation then the failure Te output is greater than that second by oscillator; 	 Output Output The results of the diagnosis ande by procedures by "them to the current results of the diagnosis and by procedures by "them to the current part in package r_0_sim_OD is communicated by them to the current and part davics areas	ł	If an explanation is requested, same is supplied via EXFLAIN_DIAGNOSIS.
 Output: The results of the diagnosis made by procedures PWR_SWITCH_STATUS and output device screem. Algorithm MANR_SIM_MOD Algorithm used for the driver program is: Initialise the simulator by reading the equipment parameters from a file. While the human monitor desires to continue While the human monitor desires to continue While the human monitor desires to continue Test if both A is monitor desires to continue Test if both A is monitor desires to continue Test if both A is monitor desires to continue Test if both A is monitor desires to continue Test if both A is monitor desires to continue Test if both A is monitor desires and if so force a choice? Test the results of the simulator (package FAULT_ANALYZE): To the amulator in package Y_OUT_MOD: Test the results of the simulator (package FAULT_ANALYZE): Test the results of the simulator (package FAULT_ANALYZE): Test the neutror if the master switchis of or if on if both the attribute than proise but less than the output caused by an output applicit fish unce the post output to shold the section of the statement of statement of statement of the statement of t	 Orughu: Are results of the diagnosis made by procedures PWR SWITCH STATUS and results of the diagnosis made by procedures PWR SWITCH STATUS and our LIVEL: In paraleters 1_0.51M MOD is communicated by them to the CRT our LIVEL: In paraleters 1_0.51M MOD is communicated by them to the CRT our LIVEL: In paraleters 1_0.51M MOD is communicated by them to the CRT in algorithm warm group the symphone parameters from a sille. Algorithm HAIN SIM MOD Initialise the simulator by reading the equipment parameters from a sille. Initialise the simulator by reading the equipment parameters from a sille. The human monitor desires to continue respect 2 thru 9; Test if both A & B sources are switched on and if so force a choice; Run the simulator in package Y_OUT_MOD; Fost the cutput is noise i.e145(0 dam) there from rise but less that a partiches are proper and output is greater than noise but less than the output is noise i.e145(0 dam) the check sath of the output is noise a chick in a splifier filture than noise but less than the output is greater than that caused by socillator is ubstander of output is greater than the contput is place with the the output is noise a chick of the output is greater than that caused by oscillator substandard of the output is greater than that caused by oscillator substandard of the output is a tormal level the result of the applifier filture than the failure of the output is a drived by oscillator substandard of the output is a tormal level the resolution than the result of the same main the result of the same substandard of the output is a tormal level thermation than the failure of the output is a tormal level the resolution than the failure of the output is a tormal level thermation than the failure of the output is a tormal level to caused by oscillator the same substandard of the output is a tormal level the resolution than the failure of the output is a tormal level to caused by oscillator the same substandard of	1	
 The results of the diagnosis mode by proceedues PW_winter_ statute and output device screen. Algorithm MAIN_SIM_MOD Initialize the simulator by reading the equipment parameters from a file. Initialize the simulator by reading the equipment parameters from a file. Initialize the buman monitor desires to continue While the human monitor desires to continue The still beth A is Bources are switched on and if so force a concerning the simulator in package Y_OUT_MOD; Run the simulator in package Y_OUT_MOD; Run the simulator in package Y_OUT_MOD; Test the results noise (is15(2) mode; To file another is noise (is15(2) mode; If all suitches are in the off position; If all suitches are in the off position; If output is greater than that caused by another threshold mode to cortie site and not the splitter than holds. If output is greater than that caused by another the shlure than noise but less the amplitier substandard output is specied output substandard output is specied output is specied to substandard output is position; If output is greater than that caused by another to splitter than presumably the splitter so dyna than cortice of the subject is solved over the splitter is splitter to subject a subject of solved is position; 	 The results of the diagnosis make by proceedures PW_SWITC_SILPORT The results of the diagnosis make by proceedures PW_SWITC_SILPORT The algorithm warm gin more serven. Algorithm MAIN gin MOO The algorithm used for the driver program is: Initialize the simulator by reading the equipment parameters from a file. Initialize the simulator by reading the equipment parameters from a file. While the human monitor desires to continue repeat 2 thru 9. While the human monitor desires to continue repeat 2 thru 9. The simulator in package Y_OUT_MOD: In the simulator in package Y_OUT_MOD: If the output is noise (it to end the simulator substanded to accurded the substanded to accurded solution then the failure of the output is at normal level to action of the output is at normal level to contal operation th	•	Output
 OUT INTE IN PACEND. Algorithm MAIN_SIM MOD is communicated by them to the data output device screen. Algorithm WAIN_SIM MOD Initialize the simulator by reading the equipment parameters from a file. Initialize the simulator by reading the equipment parameters from a file. While the human monitor desires to continue Rine if both A is B sources are switched on and if so force a choice; The simulator in package Y_OUT_MOD; Run the simulator in package Y_OUT_MOD; Run the simulator in package Y_OUT_MOD; Fast the results of the simulator (package FAULT_AMAIZE); Fast the results of the simulator (package FAULT_AMAIZE); In the aventus of the simulator (package FAULT_AMAIZE); If all subtreas are in the off position; If all subtreas are in the off position; If all subtreas are in the off position; If output is greater than that caused by an output sing frastree than presumably the spectra ballifier so advise the splitter than presumably the state output subtuce subtor to switch to the alternate oscillator of the subtor to switch the threshold needed to section the output is greater than that caused by oscillator substandard If output is greater than that caused by oscillator substandard If output is greater than that caused by oscillator substandard If output is greater than that caused by oscillator substandard If output is greater than that caused by oscillator substandard If output is greater than that caused by oscillator substandard If output is greater than that caused by oscillator substandard If output is greater than that caused by oscillator substandard If output is greater than that caused by oscillator substandard If output is greater than that caused by oscillator substandard 	 OUT_INTEL IN package I_0_INFNOD is communicated by them to the two output device screen. Algorithm MMIN_SIM_MOD Initialize the simulator by reading the equipment parameters from a file. Initialize the buman monitor desires to continue While the human monitor desires to continue Test if both A & B sources are switched on and if so force a concerce? Run the simulator in package Y_OUT_MOD: Run the simulator in package FAUT_AMAIZED: Frest the events of the simulator (package FAUT_AMAIZED: Frest the events of the simulator (package FAUT_AMAIZED: Frest the events of the simulator (package FAUT_AMAIZED: Frest the events of the source is and output is greater than noise but less than the output is greater than that caused by ontilitier to shift of the alternate oscillator: If output is greater than that caused by oscillator substandard of the output is greater than the contract os stich is subjutee. If the output is sthormal level then report normal operation If the output is a normal level then report normal operation 	1	The results of the diagnosis made by procedures PWR SWITCH STATUS and
 Algorithm MAIN_SIM_MOD Algorithm used for the driver program is: The algorithm used for the driver program is: Initialize the simulator by reading the equipment parameters from a file. while the human monitor desires to continue while the human monitor desires to continue Test if both A & B sources are switched on and if so force a choice; Test if both A & B sources are switched on and if so force a choice; Test the results of the matuator (package FAULT_AMALTZE); Test the results of the matuator (package FAULT_AMALTZE); Test the results of the matuator (package FAULT_AMALTZE); Test the results are proper and output is off or if on if both the human the simulator subtainding the sources are subtained to the alternate oscillator; If output is greater than that caused by oscillator substandard output is greater than noise but less the output caused by oscillator substandard output substandard output substandard output substandard output substandard output caused by accurred; A couptup to subject and that caused by oscillator substandard output substandard output substandard output substandard output substandard output caused by oscillator substandard output caused by accurred; 	 output device screen. Algorithm MAIN_SIM_MOD The algorithm used for the driver program is: Initialize the simulator by reading the equipment parameters from 	1	OUT_LEVEL in package I_O_SIM_MOD is communicated by them to the LKI
 Algorithm MAIN_SIM_MOD The algorithm used for the driver program is: Initialize the simulator by reading the equipment parameters from file. While the human monitor desires to continue while the human monitor desires to continue respeat 2 thru 9; rest if both A & B sources are switched on and if so force a choice; The neurly of the simulator in package Y_OUT_MOD; Test the output is noise (i.e145.0 dbm) then check switches and rest the results of the simulator (package FAULT_AMALYZE); Test the output is noise (i.e145.0 dbm) then check switches and noticy the monitor if the master switch is off or if on if both than the output is noise (i.e145.0 dbm) then check switches and 1 If the output is noise (i.e145.0 dbm) then check switches and a stiches are proper and output is greater than noise but less than the output caused by an output so off or if on if both than the output caused by an output so off or if on if both 3 If output is greater than that caused by oscillator substander output but less than that for normal operation then the failure of the output sapilities any have occurred; 	 Algorithm MAIN_SIM_MOD The algorithm used for the driver program is: Initialize the simulator by reading the equipment parameters from a file. Mhile the human monitor desires to continue repeat 2 thru 9; Test if both A & B sources are switched on and if so force a concect. Test the results of the simulator (package FAULT AMALYZE); Run the simulator in package Y_OUT_MOD; Test the results of the simulator (package FAULT AMALYZE); If the output is noise (i.e15:0 dbm) than "heck switches and notify the monitor if the mase eff position; If all switches are proper and output is greater than noise but less the amplifier so advise the monitor to switch to the alternate oscillator; If output is greater than tht for normal operation then the failure of the output is at normal presention then the failure of the output is at normal presention then the failure 		output device screen.
The algorithm used for the driver program is: Thitialize the simulator by reading the equipment parameters from file. mula the human monitor desires to continue repeat 2 thru 9; rest if both A & B sources are switched on and if so force a repeat 2 thru 9; rest if both A & B sources are switched on and if so force a rout the simulator in package Y_OUT_MOD; Tut the simulator in package Y_OUT_MOD; Tut the simulator in package Y_OUT_MOD; Test the output is noise (i.e145.0 dbm) than check switches and notify the sonitor if the matter suich is off or if on if both the A & B suitches are in the off position; If the neuticut of the simulator (package FAULT_MALYEE); Test the anticut if the matter suich is off or if on if both the A & B suitches are in the off position; If than the output caused by no uptut is position; If output is greater than that caused by oscillator; If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure output a publicite may have occurred; 	The algorithm used for the driver program is: Thiridize the simulator by reading the equipment parameters from file. this the human monitor desires to continue rest if both A & B sources are switched on and if so force a choice; Test the results of the simulator (package FAULT ANALYZE); Test the output is noise (i.e150 ddm) then -back switches and	ļ	Algorithm MAIN_SIM_MOD
 The algorithm used for the driver program is: Initialize the simulator by reading the equipment parameters from Mille the human monitor desires to continue while the human monitor desires to continue While the human monitor desires to continue Tepeat 2 thru 9; Test if both A t B sources are switched on and if so force a choice; Test the results of the simulator (package FAULT ANALYZE): If the output is noise (is 145:0 dbm) than check switches and noify the monitor if the master switch is off or if on if both If all switches are proper and output is greater than noise but less If all switches are proper and output is greater than noise but less If output is greater than that caused by oscillator is the monitor to switch to the alternate oscillator; If output is greater than that coursed by oscillator substandard output us applifier may have occurred; 	 The algorithm used for the driver program is: Initialize the simulator by reading the equipment parameters from a file. Mule the human monitor desires to continue respect 2 thru 9; Test if both A & B sources are switched on and if so force a concice; Test the results of the simulator (package FAULT_AMALYZE): Test the results of the master switch is off or if on if both neight neight the for the master suit and if so for or if on if both the for the result applifier failure then prosumably the splaced oscillator output is greater than noise but less the neight the splifier to advise the monitor to switch to the alternate oscillator; If the output is dreased by oscillator substandard output is greater than the failure then prosumably the splifier appoint the to switch to the alternate oscillator; If the output is dreased by oscillator substandard or switch to the alternate oscillator; If the output is at normal level then report normal operation 	1	
 Initialize the simulator by reading the equipment parameters from a file. While the human monitor desires to continue repeat 2 thru 9; Test if both A & B sources are switched on and if so force a choice; Test the simulator in package Y_OUT_MOD; A Run the simulator in package Y_OUT_MOD; Srest the results of the simulator (package FAULT_AMALYZE): Test the output is noise (i.e145:0 dbm) then check switches and If all switches are proper and output is greater than noise but less the A & B sutches are proper and output is greater than noise but less that the output caused by an output is greater than noise but less than the output is below the threshold needed to accide that the suplifier so advise the monitor to switch to the alternate oscillator; 3 If output is greater than that concreal operation then the failure the amplifier so advise the monitor to switch to the alternate oscillator; of the output is preset than that concreal operation then the failure the amplifier so advise the monitor to switch to the failure the amplifier so advise the monitor to switch to the failure the amplifier so advise the monitor to switch then the failure output is greater than that caused by oscillator substander of the output to avoit the monitor to switch then the failure of the output to avoit the monitor to switch then the failure of the output to avoit the monitor to switch then the failure of the output to avoit the monitor the monitor then the failure of the output to avoit the monitor to switch then the failure the monitor failure the	 Initialize the simulator by reading the equipment parameters from a file. While the human monitor desires to continue repeat 2 thru 9; Test if both A & B sources are switched on and if so force a conice; Test if both A & B sources are switched on and if so force a conice; I if the output is noise (i.e145.0 dbm) than check switches and notify the monitor if the master switch is off or if on if both that A & B suitches are in the off position; I if all switches are in the off position; I if output is greater than noise but less the amplifier so advise the monitor to switch to the alternate oscillator; I if output is greater than that for normal operation then the failure of the output is at normal operation then the failure of the output is at normal operation then the failure of the output is at normal operation then the failure 	1	The algorithm used for the driver program is:
 Initialize the simulator by reading the equipment parameters from a file. ifile. ifile the human monitor desires to continue repart 2 thru 9; Test if both A & B sources are switched on and if so force a choice; a. Test the neuron in package Y_OUT_MOD; 4 Run the simulator in package Y_OUT_MOD; 5. Test the neuron if the master switches of on if both the A & B suitches are off or if on if both the A & B suitches are output as greater than noise but less than the A & B suitches are output as greater than noise but less the results of a simulator to switch is off or if on if both the applifier so advise the monitor to switch to the alternate oscillator; 3. If output is greater than that caused by oscillator substandard output is greater than the failure the the here applifier and by the selected by an output as greater than noise but less the applifier so advise the monitor to switch to the alternate oscillator; 3. If output is greater than that caused by oscillator substandard output is greater than the failure ended to satilator; 3. If output is greater than that for normal one the failure off. 	 Initialize the simulator by reading the equipment parameters from a file. A file. A mile the human monitor desires to continue respeat 2 thru 9; Test if both A £ B sources are switched on and if so force a choice; A mu the simulator in package Y_OUT_MOD; Fost the results of the simulator (package FAULT ANALYZE): Test the output is noise (i.e145.0 dbm) then check switches and noify the sources are svitch is off or if on if both (i.e. 11 svitches are proper and output is greater than noise but less than the output caused by an output is greater than noise but less than the output as peaker and output is packed to escillator; I ff output less than that for normal operation then the failure of the output is greater than the for than the output is below the threshold needed to escillator; I ff the output is output is peaker than the caused by oscillator substandard of the output is greater than the for normal operation then the failure of the output is greater than the for normal operation then the failure of the output is greater than the results of the number of the number of the number of the number of the for normal operation then the failure of the output is greater than the report normal operation then the failure 	1	
 While the human monitor desires to continue respent 2 thru 9; respent 3 thrushoft a simulator (package FAULT ANALYZE); respent the results of the simulator (package FAULT ANALYZE); respent the results of the matter subject (1.e145.0 dbm) than check subtches and notify the monitor if the matter switch is off or if on if both the A t B switches are in the off position; respect descriptions are in the off position; respect descriptions of the monitor to switch to the alternate oscillator; respect than that coursed by oscillator substandard output is placer than the subject is specified to excite the moleculator output is below the threshold needed to excite the moleculator; respect the subject is ray have oscillator substandard of the output camplifier failure to switch to the alternate oscillator; respect the output cample to respect on the needed to excite the output subject to switch to the alternate oscillator; 	 While the human monitor desires to continue While the human monitor desires to continue respect 2 thru 9; Test if both A E B sources are switched on and if so force a choice; Run the simulator in package Y_OUT_MOD; Run the simulator in package Y_OUT_MOD; Test the output is noise (i.e145:0 dbm) than check switches and noiffy the monitor if the master switch is off or if on if both the A E B switches are in the off position; If an the output caused by an output suplifier failure than noise but less than that cutout as blow the threshold needed to excite selected oscillator output is bound to the alternate oscillator; If the output less than that for normal operation than the failure of the output is at normal operation than the failure of the uptut is at normal level than result or remain the attinte of the nutput to a dupt the solution to suitch to the alternate oscillator; If the output is at normal level than report normal operation 	1	1. Initialize the simulator by reading the equipment parameters from
 While the human monitor desires to continue repeat 2 thru 9; Test if both A is B sources are switched on and if so force a choice; Test the results of the simulator (package FAULT ANALYZE): If the output is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both the A is B switches are in the off position; If all switches are in the off put is greater than noise but less than the output caused by an output is greater than noise but less than the output is below the threshold needed to accide the applifier so advise the monitor to switch to the alternate oscillator; If output is greater than that caused by oscillator substandard output be less than that for normal operation then the failure output less than that for normal operation then the failure 	 While the human monitor desires to continue repeat 2 thru 9; Test if both A & B sources are switched on and if so force a choice; Test the results of the simulator (package FAUTT AMALYZE): Test the results of the simulator (package FAUTT AMALYZE): Test the nutput is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both the A B switches are in the off position? If all switches are proper and output is greater than noise but less than the output caused by an output is greater than noise but less than the output caused by an output is greater than noise but less than the output us below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; If the output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the output is at normal level then report normal operation 	!	a lite.
 2. While the Numan Monitor desires to continue 3. Test if both A & B sources are switched on and if so force a choice; 3. Test if both A & B sources are switched on and if so force a choice; 4. Run the simulator in package Y_OUT_MOD; 5. Test the results of the simulator (package FAULT_ANALYZE); 5. Test the results of the simulator (package FAULT_ANALYZE); 6. Test the results of the simulator (package FAULT_ANALYZE); 7. Test the results of the simulator (package FAULT_ANALYZE); 6. Test the results of the simulator (package FAULT_ANALYZE); 7. Test the results of the simulator (package FAULT_ANALYZE); 7. Test the results of the simulator (package FAULT_ANALYZE); 7. Test the results of the simulator (package FAULT_ANALYZE); 7. Test the results of the simulator (package FAULT_ANALYZE); 8. Test the results of the simulator (package FAULT_ANALYZE); 9. Test the results of the simulator (package FAULT_ANALYZE); 9. Test the results of the moster switch is off or if on if both the site than here and output amplifier fact on output 1 is greater than noise but less that the cutput caused by oscillator substandard oft output us greater than that caused by oscillator substandard oft the output amplifier may have occurred; normal oneration then the failure 	 2. While the Numan Monitor desires to continue 7 Test if both A is B sources are switched on and if so force a 2. Test if both A is B sources are switched on and if so force a 3. Test the results of the simulator (package FAULT ANALYZE): 4. Run the simulator in package Y_OUT_MOD; 5. Test the results of the simulator (package FAULT ANALYZE): 5. Test the results of the simulator (package FAULT ANALYZE): 5. Test the results of the simulator (package FAULT ANALYZE): 5. Test the results of the simulator (package FAULT ANALYZE): 5. Test the results of the master switch is off or if on if both the A is B switches are in the off position; 2. If all switches are proper and output is greater than noise but less than the output is below the threshold needed to ascimator; 3. If output is greater than that caused by oscillator substandard output is greater then the failure of the output is greater then the for normal operation then the failure of the output is greater than report normal operation for the failure of the output is greater than report normal operation then the failure 4. If the output is a normal level then report normal operation 	1	
 repeat 2 thru 9; 3. Test if both A t B sources are switched on and if so force a choice; 3. Test if both A t B sources are switched on and if so force a choice; 4 Run the simulator in package Y_OUT_MOD; 5. Test the results of the simulator (package FAUT_AMALYZE); 5. Test the results of the simulator (package FAUT_AMALYZE); 5. Test the output is noise (i.e145.0 dbm) then_check switches and notify the moster switch is off or if on if both notify the monitor if the master switch is off or if on if both the A t B switches are proper and output is greater than noise but less than the output caused by an output applifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; 3 If output less than that for normal operation then the failure output camplifier may have occurred; 	 repeat 2 thru 9; Test if both A & B sources are switched on and if so force a choice; Run the simulator in package Y_OUT_MOD; Run the simulator in package Y_OUT_MOD; rest the results of the simulator (package FAULT_ANALYZE): rest the results of the master switch is off or if on if both notify the monitor if the master switch is off or if on if both the A & B switches are in the off position; rest than the output caused by an output is greater than noise but less than the output caused by an output is greater than noise but less the amplifier so advise the monitor to switch to the alternate oscillator; rest the output is greater than that caused by oscillator substandard of the output is greater than the failure of the output is greater than the failure of the output is greater than noise but less than that for normal operation then the failure of the output is greater than noise but less than that for normal operation then the failure of the output is greater than report normal operation 	1	2. While the human monitor desires to continue
 3. Test if both A & B sources are switched on and if so force a choice; a Run the simulator in package Y_OUT_MOD; 5. Test the results of the simulator (package FAULT_ANALYZE): 5. Test the results of the simulator (package FAULT_ANALYZE): 5. Test the results of the simulator (package FAULT_ANALYZE): 6. Tast the output is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both the A & B switches are in the off position; 2. If all switches are proper and output is greater than noise but less than the output caused by an output is greater than noise but less the amplifier so advise the monitor to switch to the alternate oscillator; 3. If output is greater than that for normal operation then the failure oscillator; 6. The output caused by an output is period to accide to accide the applifier so advise the monitor to switch to the alternate oscillator; 	 3. Test if both A & B sources are switched on and if so force a choice; 4. Run the simulator in package Y_OUT_MOD; 5. Test the results of the simulator (package FAULT ANALYZE); 5. Test the results of the simulator (package FAULT ANALYZE); 5. Test the results of the master switch is off or if on if both neity the monitor if the master switch is off or if on if both than A & B switches are in the off position; 2. If all switches are proper and output is greater than noise but less than the output caused by an output is failure then presumably the selected or advise the monitor to switch to the alternate oscillator; 3. If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the output is at normal level then result for normal operation. 4. If the output is at normal level then report normal operation 	1	repeat 2 thru 9;
 3. Test if both A & B sources are switched on and if so the choice; 4 Run the simulator in package Y_OUT_MOD; 5. Test the results of the simulator (package FAULT_ANALYZE): 5. Test the results of the simulator (package FAULT_ANALYZE): 5. Test the results of the master switch is off or if on if both the A B switches are proper and output is greater than noise but less that the output caused by an output is greater than noise but less that the aughlifier so advise the monitor to switch to the alternate oscillator; 3. If output is greater than that for normal operation then the failure then presumably the selected oscillator output is below the threshold needed to excite output is greater than noise but less the amplifier so advise the monitor to switch to the alternate oscillator; 3. If output is greater than that for normal operation then the failure of the output amplifier may have occurred; 	 3. Test if both A t B sources are switched on and it sources are switches are subject in package Y_OUT_ANALYZE): 4. Run the simulator in package Y_OUT_ANALYZE): 5. Test the results of the simulator (package FAULT_ANALYZE): 5. Test the soutput is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both the A t B switches are in the off position; 2. If all switches are in the off position; 3. If output caused by an output amplifier failure than noise but less than applifier so advise the monitor to switch to the alternate oscillator; 3. If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the output is at normal level then report normal operation. 	1	
 4 Run the simulator in package Y_OUT_MOD; 5. Test the results of the simulator (package FAULT_ANALYZE): 5. Test the results of the simulator (package FAULT_ANALYZE): 5. Test the noutput is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both the A & B switches are proper and output is greater than noise but less that the output caused by an output is greater than noise but less selected oscillator output is below the threshold needed to excite the amplifier failure the presumably the selected oscillator output is below the threshold needed to excite the amplifier far than the output is greater than the sillator substandard output is greater than the for normal operation then the failure occillator: 3 If output but less than that for normal operation then the failure occillator: 	 Run the simulator in package Y_OUT_MOD; Fest the results of the simulator (package FAULT_ANALYZE): I If the output is noise (i.e145.0 dbm) then_check switches and notify the monitor if the master switch is off or if on if both the A & B switches are in the off position; If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumbly the the amplifier so advise the monitor to switch to the alternate oscillator; If output is greater than that caused by oscillator substandard output us potention then the failure betweed to excite the output is greater than the for normal operation then the failure. If output is greater than that caused by oscillator substandard output is at normal level then report normal operation then the failure If the output is at normal level then report normal operation 	1	3. Test II Doth A & B sources are switched on and it so rouch a
 4 Run the simulator in package Y_OUT_MOD; 5. Test the results of the simulator (package FAULT_ANALYZE): 5. Test the results of the simulator (package FAULT_ANALYZE): 5. Test the results of the master switch is off or if on if both 1. If the output is noise (i.e145.0 dbm) then check switches and 1. If the monitor if the master switch is off or if on if both 2. If all switches are in the off position; 2. If all switches are proper and output is greater than noise but less 2. If all switches are proper and output is greater than noise but less 3. If output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite 3. If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure 3. If output less than that for normal operation then the failure 3. Output but less than that for normal operation then the failure 4. Output but less than that for normal operation then the failure 	 4 Run the simulator in package Y_OUT_MOD; 5. Test the results of the simulator (package FAULT ANALYZE): 5. Test the output is noise (i.e145.0 dbm) then check switches and notify the anomitor if the master switch is off or if on if both the A & B switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excitle the amplifier so advise than that caused by oscillator substandard output but less than that for normal operation then the failure of the internate oscillator; 3 If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the output is at normal level then remain operation. 	1	CHOICE
 4 Run the statuator in package r_ou_not. 5. Test the results of the simulator (package FAULT_ANALYZE): 5. Test the output is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both If all switches are in the off position: 2 If all switches are proper and output is greater than noise but less selected oscillator output amplifier failure then presumably the the amplifier so advise the monitor to switch to the alternate oscillator: .3 If output is greater than that for normal operation then the failure of the output amplifier may have occurred: 	 4 Kun the simulator in parkage r_our_nut. 5. Test the results of the simulator (parkage FAULT_ANALYZE): 5. Test the notput is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both 1. If the output is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both 2. If all switches are in the off position; 2. If all switches are proper and output is greater than noise but less than the output caused by an output is greater than noise but less 2. If output is greater the monitor to switch to the alternate oscillator; 3. If output is greater than that caused by oscillator substandard output ut less than that for normal operation then the failure of the output applifier may have occurred; 4. If the output is at normal level then report normal operation 	ł	
 5. Test the results of the simulator (package FAULT ANALYZE): .1 If the output is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both the A & B switches are in the off position; 2 If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; .3 If output is greater than that caused by oscillator substandard output but less than the output amplifier may have occurred; 	 5. Test the results of the simulator (package FAULT ANALYZE): 1. If the output is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both the A & B switches are in the off position; 2. If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the oscillator; 3. If output is greater than that caused by oscillator substandard of the output amplifier may have occurred; 4. If the output is at normal level then report normal operation 	1	
 5. Test the results of the simulator (package fAULT ANALISE): 1. If the output is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both 1. If all switches are in the off position; 2. If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; 3. If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure 	 5. Test the results of the simulator (package rAUL: ANALISC): 1. If the output is noise (i.e145.0 dbm) then check switches and 1. If the antion if the master switch is off or if on if both 1. If all switches are in the off position; 2. If all switches are proper and output is greater than noise but less 2. If all switches are proper and output is greater than noise but less 3. If output is greater than the the edged to excitlency; 3. If output is greater than that caused by oscillator to switch to the alternate oscillator; 4. If the output is at normal level then report normal operation 	1	
 If the output is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure output but less than that for normal operation then the failure 	 I If the output is noise (i.e145.0 dbm) then check switches and notify the monitor if the master switch is off or if on if both the A & B switches are in the off position; If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure If the output is at normal level then report normal operation 	•	5. Test the results of the simulator (package FAULT_ANALIZE):
<pre>notify the monitor if the master switch is off or if on if both the A & B switches are in the off position;</pre>	<pre>notify the monitor if the master switch is off or if on if both the A & B switches are in the off position; that the output caused by an output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; if output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure if the output is at normal level then report normal operation</pre>	1	.1 If the output is noise (i.e145.0 dbm) then check switches and
<pre>the A t B switches are in the off position; If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; all foutput is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the output camplifier may have occurred; </pre>	the A is B switches are in the off position; If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; output but less than that for normal operation then the failure of the output amplifier may have occurred; If the output is at normal level then report normal operation	1	notify the monitor if the master switch is off or if on if both
 If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the output amplifier may have occurred; 	 If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the output amplifier may have occurred; If the output is at normal level then report normal operation 		the 2 the evitthes are in the off position;
 If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; If output is greater than that for normal operation then the failure of the output less than that for normal operation then the failure of the output camplifier may have occurred; 	 If all switches are proper and output is greater than noise but less than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the output is at normal level then report normal operation If the output is at normal level then report normal operation 		
than the output caused by an output amplifier failure then presumably the selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; 	 that the output caused by an output amplifier failure then presumably the that the output caused by an output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure output is applifier may have occurred; If the output is at normal level then report normal operation 		o re all cuitches are proper and output is greater than noise but less
 selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the output amplifier may have occurred; 	 selected oscillator output is below the threshold needed to excite selected oscillator output is below the threshold needed to excite the amplifier so advise the monitor to switch to the alternate oscillator; a If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the output is at normal level then report normal operation 	1	. It all settores are proper and over a setting the failure then Dresumably the transmission of transmi
<pre> selected oscillator output is below the monitor to switch to the alternate oscillator; 3 If output is greater than that caused by oscillator substandard 0utput but less than that for normal operation then the failure </pre>	 selected oscillator output is below the monitor to switch to the alternate oscillator; i.3 If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the output is at normal level then report normal operation 	1	chair the output caused by an output whether threads the particular of the control of the contro
 the amplifier so advise the monitor to switch to the attendary operation. If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the outputc amplifier may have occurred; 	 the amplifier so advise the monitor to switch to the attendary optimized optimized	1	Selected Oscillateroi otribut to betoe the thread of the second of the s
 If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the outputc amplifier may have occurred; 	 .3 If output is greater than that caused by oscillator substandard .0 utput but less than that for normal operation then the failure of the outputc amplifier may have occurred; .4 If the output is at normal level then report normal operation 	1	the amplifier so advise the monitor to switch to the attenuate votation.
 If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the outputc amplifier may have occurred; 	 .3 If output is greater than that caused by oscillator substandard output but less than that for normal operation then the failure of the outputc amplifier may have occurred; .4 If the output is at normal level then report normal operation 	1	
output but less than that for normal operation then the failure of the outputc amplifier may have occurred;	 output but less than that for normal operation then the failure of the outputc amplifier may have occurred; .4 If the output is at normal level then report normal operation 	1	.3 If output is greater than that caused by oscillator substandard
of the outputc amplifier may have occurred; 	of the outputc amplifier may have occurred; 4 If the output is at normal level then report normal operation	;	output but less than that for normal operation then the failure
et de cecestre artener au les contraction et	.4 If the output is at normal level then report normal operation		of the cutourty and if it are have occurred:
· · · · · · · · · · · · · · · · · · ·	4 If the output is at normal level then report normal operation		
		1	· · · · · · · · · · · · · · · · · · ·

	Write out the equipment status to the CRT (procedure OUT_PARAMS in package I_O_SIM_MOD). Interactively ask the monitor whether an explanation of the diagnosis is desired. If so then supply same. Interactively ask the monitor if the simulation is to be continued. If the monitor chooses not to continue then exit. If the choice is to run another simulation the ask what parameters are to be changed (i.e. call NEW_PARAMS in package I_O_SIM_MOD). Go back to item 2. Igorithm MAIN_SIM_MOD	DEF SIM; with T_OUT_MOD; with I_O_SIM_MOD; with FAULT_ANALYZE; AIN DIAGNOSIS; EF_SIM; use Y_OUT_MOD; MAIN_SIM_MOD3 is Continue:= YES; : continue:= YES; : co	IM_MOD.I_O_INIT(EQPT_Y,FREQ_SRC_A,FREQ_SRC_B,EQPT_Y_OUT_LEVEL_FAIL, REF_FREQ_LEVEL_A,REF_FREQ_LEVEL_B, EQUITP_Y_OUT_LEVEL_SIM,EQUITP_X_OUT_LEVEL); AGAIN = YES op if FREQ_SRC_A=ON and FREQ_SRC_B=ON then I_O_SIM_MOD.I_O_SWITCH(FREQ_SRC_A,FREQ_SRC_B); end if; If both TON" force a choice between A or B "ON" I_O_SIM_MOD.NEWPAGE;
	 6. Write out the equipmer in package I_0_SIM_MOI 7. Interactively ask the 7. Interactively ask the 8. Interactively ask the 9. If the monitor choose: 9. If the choice is to riar 10. If the choice is to riar 11. Go back to item 2. end algorithm MAIN_SIM_MOD 	With TYP DEF SIM; With Y OUT MO With EXPLAIN DIAGNOSIS; USE TYP DEF SIM; USE Y OUT MOD; Procedure MAIN_SIM_MOD3 is AGAIN : CONTINUE:= YES; EXPLAIN : CONTINUE:= YES; EXPLAIN : CONTINUE:= YES; EXPLAIN : CONTINUE:= YES; FREQ_SRC A, FREQ_SRC A, FREQ_SRC B : SWITCH; FREQ_SRC B : SWITCH; FREQ_SRC B : SWITCH; FREQ_SRC B : SWITCH; FREQ_SRC A, FREQ_SRC A, FREQUET FROUT LEVEL SIM, EQUIP Y OUT LEVEL FAIL : FAULT EQUIP Y OUT LEVEL SIM,	begin I_O_SIM_MOD.I_O_INIT(EQPT_Y REF_FR EQUIF_ While AGAIN = YES While AGAIN = YES if FREQ SRC A=ON and F then I_O_SIM_MOD.I O S end if; If both TO I_O_SIM_MOD.NEWPAGE;
N SIN 10	5 5 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7	アア島路路路路路路路路路路ののののののののの、ころまで、「「」の「」」の「」」の「」」」の「」」」」」」」」」」」」」」」」」」」」	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

щзтмј ³ {	-Jart 7 27 28 X A 1.37 31 SIM MOD3.ADA:3 18-Jan-1987 22:31:20 USER2:[TFL.ADADIR]MAIN_SIM_MOD3.ADA:3	E T
114 115 116 117	EQUIPMENT_Y_O(EQPT_Y,FREQ_SRC_A,FREQ_SRC_B, REF_FREQ_LEVEL_A,REF_FREQ_LEVEL_B,EQPT_Y_OUT_LEVEL_FAIL, EQUIP_X_OUT_LEVEL,EQUIP_Y_OUT_LEVEL_SIM, EQUIP_Y_OUT_LEVEL) ; Run the Simulator	
118 119 120 121	FAULT_ANALYZE.DIAGNOSIS(EQPT_Y,FREQ_SRC_A,FREQ_SRC_B, EQUIP_Y OUT_LEVEL_SIM, EQUIP_Y_OUT_LEVEL,EQUIP_X_OUT_LEVEL); Diagnose fault	
122 123 124 125	I_O_SIM_MOD.OUT_PARAMS(EQPT_Y,FREQ_SRC_A,FREQ_SRC_B,EQPT_Y_OUT_LEVEL_FAIL, 	
126	I_O_SIM_MOD.QUERY_EXPLAIN(EXPLAIN); Does monitor want an explanation	
128 139 133 133 133 133	If EXPLAIN =YES then EXPLAIN_DIAGNOSIS.EXPLAIN_DIAGNOSIS(EQPT_Y,FREQ_SRC_A,FREQ_SRC_B,REF_FREQ_LEVEL_A,REF_FREQ_LEVEL_B, EQUIP_Y_OUT_LEVEL,EXPLAIN]; If explanation desired end if;	
135	I_O_SIM_MOD.AGAIN_I_O(AGAIN); Ask human if wants to repeat simulation	
137 138	exit when AGAIN = NO;	
139 140 141	I_O_SIM_MOD.NEW_PARAMS(EQPT_Y,FREQ_SRC_A,FREQ_SRC_B,EQPT_Y_OUT_LEVEL_FAIL, Ref_Freq_Level_A,ref_Freq_Level_B, Equip_Y_OUT_LEVEL_SIM,EQUIP_X_OUT_LEVEL);	
143 144	end loop;	
145 146	end MAIN_SIM_MOD3;	
PSECT MAP		
Psect Hex Size 0 000001F3	Dec Size Name 499 MAIN_SIM_MOD3.\$CODE	
\$ADAC-I-CL_ADDED,), Procedure body MAIN_SIM_MOD3 added to library	
PORTABILITY SUMMA	.ARY	
There are I	no uses of potentially non-portable constructs	
LIBRARY SUMMARY		
USER2:[TFL.ADADI]	[R.ADALIB]	
Unit name	Nodes Percent Blocks Unit kind read read	

.....

	 If the simulation is to run again this procedure is called to read in new values procedure NEW PARAMS(EQPT Y, FREQ_SRC A, FREQ_SRC B : in out SWITCH; Equipmt sw's procedure NEW PARAMS(EQPT Y OUT LEVEL FAIL : in out FAULT; HW failure REF FREQ LEVEL A, REF FREQ LEVEL B, OSC OUTPUT EQUIT Y OUT LEVEL SIM, HW failure output level EQUIT Y OUT LEVEL : in out FLOAT); Normal OUT 	Read from the keyboard whether eqpt monitor wants to continue the simulation or not procedure AGAIN_I_0(AGAIN : out CONTINUE); 3	procedure PARAM_OUT(PWR_LEVEL : in FLOAT);	EQUIP X OUT LEVEL, Normal Output Power EQUIP_Y_OUT_LEVEL: in FLOAT); Actual Output	REFFREQ LEVEL A, REFFREQ LEVEL B, OSC OUTPUT EQUT LEVEL A, REFFREQ LEVEL B, OSC OUTPUT EQUTP Y OUT LEVEL SIM, HW failure OUtput level	Present output parameters all at once procedure OUT_PARAMS(EQPT Y, FREQ SRC A,FREQ SRC B : in SWITCH; Equipmt sw's	Present Explanation if requested procedure EXPLAIN_NSG_LIST(MESSAGE_NO : in INTEGER);	Ask if explanation desired procedure QUERY_EXPLAIN(EXPLAIN : out CONTINUE);	Test and report procedure FAULT_MSG_LIST(MESSAGE_NO : in INTEGER);	Obvious what this does procedure NEWPAGE;	Procedure to test if both or none of ref oscillators are on procedure I_O_SWITCH(FREQ_SRC_A,FREQ_SRC_B : in out SWITCH); SW off B or A	Procedure to initialize the simulator procedure I_O_INIT(EQPT_Y, FREQ_SRC_A,FREQ_SRC_B : out FWULT; Equipmt sw's EQPT_Y_OUT_LEVEL_FAIL : out_FAULT; HW failure REF_FREQ_LEVEL_A, REF_FREQ_LEVEL_B, Osc_Output EQUTP_Y_OUT_LEVEL_SIM, HW failure output level EQUTP_X_OUT_LEVEL : out FLOAT); Normal Out	Package I_0_SIM_MOD is	with TYP_DEF_SIM; use TYP_DEF_SIM;	<pre>with TTP_DEF_SIM: use TTP_DEF_SIM: Deckage I_0_SIM_MOD is = Procedure to initiate the similate: Procedure to initiate the similate: Procedure to initiate the similate: SUFT_YOT_LATE.TERPELSIM BW FARLO SECH PW FALLOPE SUFT_YOT_LATE.TERPELSIM BW FALLOPE SUFT_YOT_LATE.TERPELSIM BW FALLOPE SUFT_YOT_LATE.SECH PW FALLOPE Procedure NEWRAGE. Procedure NEWRAGE.</pre>
--	---	---	--	---	---	---	--	--	---	--	--	--	------------------------	------------------------------------	---

uv# ^H NIS		י
	with TEXT_IO; use TEXT_IO;	
	Package body I_O_SIM_MOD is	
1 0 0 8 7 6 7 4	Package SWC_ENUM_IO is new ENUMER Package SRC_ENUM_IO is new ENUMER Package FAULT_ENUM_IO is new ENUM Package AGAIN_IO is new ENUMERATI Package PARAM_IO is new FLOAT_IO(I INITIAL : FILE_TYPE; File Of i	TION IO(SWITCH); TION IO(SOURCE); RATION IO(FAULT); N IO(CONTINUE); LOAT); itial parameters for the eqpt simulator
11 12	This package contains initializatio	and I/O programs
13 14 15 17 18	procedure I_O_INIT(EQPT_Y, FREQ_S EQPT_Y OUT_LEV REF FREQ LEVEL EQUIP_Y_OUT_LE EQUIP_X_OUT_LE	C_A,FREQ_SRC_B : out SWITCH; Equipmt sw's L_FAIL : out FAULT; HW failure A, REF PREQ_LEVEL_B, Osc Output EL_SIM, HW failure output level FL_: out FLOAT) is Normal Output
19 20	begin Get initial p	rameter values for the simulator
21 222 255 226 209 8 209 8	OPEN(INITIAL, IN FILE SWC_ENUM IO.GET(INIT SWC_ENUM IO.GET(INIT SWC_ENUM IO.GET(INIT FAULT ENUM IO.GET(INITIAL PARAM IO.GET(INITIAL PARAM IO.GET(INITIAL PARAM IO.GET(INITIAL PARAM IO.GET(INITIAL) CLOSE(INITIAL);	"INITIAL SIM.DAT"); "AL, EQPT Y); SKIP LINE(INITIAL); AL, FREQ_SRC A); SKIP LINE(INITIAL); AL, FREQ_SRC B); SKIP LINE(INITIAL); TIAL, EQPT Y OUT LEVEL FAIL); SKIP LINE(INITIAL); REF FREQ LEVEL B); SKIP LINE(INITIAL); , REF FREQ LEVEL B); SKIP LINE(INITIAL); , REV TOUT LEVEL SIM); SKIP LINE(INITIAL); , EQUIP Y OUT LEVEL);
32 33	end I_0_INIT;	
34 35	procedure I_O_SWITCH(FREQ_SRC_A,F	REQ_SRC_B : in out SWITCH) is
36 37	FREQ_SRC_SEL : SOURCE;	A of B
38 40 40	This procedut begin This procedut	e is called from MAIN SIM e only called when both A & B switches ar ON
44444444 2000 80100 80100	PUT(" Both frequenc) NEW LINE; PUT("Whic SRC_ENUM IO.GET(FNE) if FREQ_SRC_SEL = A then FREQ_SRC_B := (else FREQ_SRC_A := (sources A & B are 'ON' & inconsistent"); h one do you want 'ON'? (type 'A' or 'B')"); _sRC_sEL); sKIP_LINE;NEW_LINE; FF; FF;
50 51 51	end I_O_SWITCH; Thi: FRE(procedure shouldn't have logic but pass
1 0 0 0 0 0 0 1 0 4 0 0 C	procedure NEWPAGE is begin NEW PAGE; end NEWPAGE;	

ر ، ب

-Jat 7 0 34 NX A 1.34 . 20-Jan-1987 00:51:15 USER2:[TFL.ADADIR]I_O_SIM_MOD.ADA;19	procedure FAULT MSG LIST(MESSAGE NO : in INTEGER) is Test all on/off switch stati, then decide upon begin what diagnosis to transmit to equipment monitor	Case MESSAGE_NO is	when 1 => NEW LINE;PUT(" Equipment output power is down in the noise level."); i.e -125 dbm NEW_LINE;	when 2 => NEW LINE;PUT(" The main switch is 'ON' so an internal failure occurred"); NEW_LINE;	when 3 => NEW LINE;PUT("Both oscillators A and B are OFF so no wonder !!"); NEW LINE;PUT(" Next pass try turning on either oscillator."); NEW LINE;	when 4 => NEW LINE;PUT("The selected oscillator may have completely failed"); NEW LINE;PUT(" Next pass try turning on the other oscillator"); NEW_LINE;	when 5 => NEW LINE;PUT("The Main switch is 'OFF' so no wonder the output is noise"); NEW_LINE;	when 6 => NEW_LINE;PUT("The OPERATIONAL DIAGNOSIS is"); NEW_LINE;PUT("Since the power output is "); NEW_LINE;	when 7 => NEW LINE; PUT(" The selected oscillator level is probably below threshold "); NEW_LINE; PUT(" so next pass switch to the other oscillator."); NEW_LINE;	when 8 =>> NEW_LINE; PUT(" A degradation of output has occurred "); NEW_LINE; PUT(" probably NOT caused by an oscillator failure "); NEW_LINE; PUT(" switch to backup unit if it's available"); NEW_LINE;	when 9 => NEW_LINE; PUT("Normal operation with normal output."); NEW_LINE;	when others => NEW_LINE; PUT(" ERROR in value of MESSAGE_NO");NEW_LINE; NEW_LINE; PUT(" ERROR in value of MESSAGE_NO");NEW_LINE;	end case;	end FAULT_MSG_LIST;
<u> ;гм_</u> 01;гм_	3 5 7 0 8 8 7 9 7 0 8 7 9 7 0 8 7 9 9 9 8 7 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9	65 65	66 68 69 89	70 71 72	74 75 75 75 75	7 8 8 8 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9	885 865 7	80800H00	m 4 13 19 17 1 5 5 5 5 5 5 1	98 99 101 102 103	104 105 107	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0		

ORIGINAL PACE IS OF POOR QUALITY

•

1 (1)

$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 20 - Jan - 1987 00:51:15 & USER2:[TFL.ADAULK]I_O_SIM_MOU.AUA;19 \end{bmatrix} \begin{bmatrix} 1 & 3 \\ 2 & 0 \end{bmatrix}$	procedure QUERY_EXPLAIN(EXPLAIN : out CONTINUE) is	begin This communicates with the human monitor asking if an	explanation desired PUT(" Do you want to know the reasoning behind the diagnostic decision? (YES/NO)"); AGAIN_IO.GET(EXPLAIN);NEW_LINE(2);	end QUERY_EXPLAIN;	procedure Explain_MSG_LIST(MESSAGE_NO : in INTEGER) is	begin This procedure contains some of the diagnostic logic for analizing a substandard power output	case MESSAGE_NO is	when 10 =>	NEW_LINE;PUT(" The OPERATIONAL DIAGNOSIS EXPLANATION is:"); New_LINE;	when 11 =>	NEW_LINE;PUT(" The output measured was very low and noisy"); NEW_LINE;PUT(" yet the master switch was 'ON',"); NEW_LINE;PUT(" so the failure was something else.");	when 12 =>	NEW_LINE;PUT(" We then found that both the oscillator switches were 'OFF',"); NEW_LINE;PUT("so the malfunction was a set-up error."); NEW_LINE;	when 13 =>	NEW_LINE;PUT(" We found the selected oscillator switch to be 'ON',"); NEW_LINE;PUT(" so,since output is noise,potentially we had a catastrophic oscillator failure"); NEW_LINE;	when 14 =>	NEW_LINE;PUT(" We found the master switch in the 'OFF' position so "); NEW_LINE;PUT(" the malfunction was a set-up error. "); NEW_LINE;	when 15 =>	NEW LINE; PUT(" The output was between -145.0 dbm and -90.0 dbm. Thus the output"); NEW LINE; PUT(" device was probably OK but the oscillator was below threshold."); NEW LINE; PUT(" The oscillator output was insufficient to drive the output device."); NEW LINE; PUT(" The logical action is to select the other oscillator on the next pass."); NEW LINE; PUT(" The logical action is to select the other oscillator on the next pass."); NEW LINE; PUT(" The logical action is to select the other oscillator on the next pass.");	
TWT TO	115 116 117	118 119	120 121 122	124 124 125	126	129	131	1.3.4	135 136 137	138	140 141 143	145	8 4 4 1 8 4 4 1 8 6 6 4 1 8 6 7 1 9 7 1 1 9 7 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	151	152 153 154 155	150	158 159 161	163 163 (SSSSSSS ORIGINAL PAC F FOOR QUA	Έ IS LITY

<pre>procedure PARAM_OUT(PWR_LEVEL : in FLOAT) is</pre>	
procedure AGAIN_I_O(AGAIN: out CONTINUE) is	
begin Ask the equipment monitor whether or not to continue	
NEW LINE; PUT(" Do you wish to change some parameters and try again ? "); PUT("(type in YES/NO) ");AGAIN_IO.GET(AGAIN); SKIP_LINE; NEW LINE(2); end AGAIN_IO;	
procedure NEW_PARAMS(EQPT_Y, FREQ_SRC_A,FREQ_SRC_B : in out SWITCH; Equipmt sw's EQPT_Y OUT_LEVEL_FAIL : in Out FAULT; HW failure REF_FREQ_LEVEL A, REF_FREQ_LEVEL B, Osc output EQUTP_Y OUT_LEVEL_SIM, HW failure output level EQUTP_X_OUT_LEVEL_: in out FLOAT) is Normal out	
SubType INPUT STRING is STRING(19); STRLEN : NATURAL ; Length of the string read from the keyboard buffer LOCAL STRING: INPUT STRING; Declaration of max length to be read CONVLEN : POSITIVE ; Dummy local var containing len of enum attrib	
begin Display present parameter values and solicit new ones	
$pur("Enter new values and or just to let value stand as is");NEW_{-}$	LINE(2);
Show master switch position (ON/OFF) and ask for new value	
PUT("Equipment switch is ");SWC_ENUM_IO.PUT(EQPT_Y);NEW_LINE; PUT("Enter new value (ON/OFF)"); GET_LINE(LOCAL_STRING,STRLEN); Get new input if STRLEN /= INPUT_STRING'FIRST-1 then SWC_ENUM_IO.GET(LOCAL_STRING,EQPT_Y,CONVLEN); then enter into DBMS then SWC_ENUM_IO.GET(LOCAL_STRING,EQPT_Y,CONVLEN); then enter into DBMS end if; LOCAL_STRING := " ";NEW_LINE; reinitialize string object	< CR >)
Show Frequency sources switch positions and ask for new values	
<pre>PUT("Freq source A is ");SWC_ENUM IO.PUT(FREQ SRC_A);NEW_LINE; PUT("Enter new value (ON/OFF)");GET_LINE(LOCAL_STRING,STRLEN); if STRLEN /= INPUT_STRING'FIRST-1</pre>	as a string its enumeration t
then SWC ENUM IO.GET(LOCAL STRING,FREQ SKC B, CONVERN), end if; Local String := " "; NEW LINE;	
Present the simulated failure status (ACTIVE/INACTIVE) ask which?	
PUT("The GREMLIN is ");FAULT_ENUM_IO.PUT(EQPT_Y_OUT_LEVEL_FAIL); #54 1185.	

.

<pre>1</pre>

_

ORIGINAL PAGE IS OF POOR QUALITY

↓ x_our_mob 01	_	-	27-Nov	יביביביביביביביביביביביביביביביביביביב	MODADA;3 {aye 1
1 2 3 4 丘 S 7 8 9 0 ・ 1 ・	with TYP DEF SIM;use Package T_OUT_MOD is procedure_EQUIPMENT_Y	TYP_DEF_S: - 0(EQPT Y REF FRE REF FRE REF FRE EQUIP X EQUIP Y	<pre>EM; : in SWITCH; Equipa c A,FREQ SRC_B: in SWIT D_LEVEL A, D_LEVEL B : in FLOAT; OUT_LEVEL FALL : in FAL OUT_LEVEL SIM : in FLOAT] OUT_LEVEL : out FLOAT]</pre>	aent Switch TCH; Osc Sw's Osc Outputs in dbm JLT; Fail Flag output > 20.0 AT; Fail Out Pwr < 20.0); Equip Pwr Out (actual)	
12 13	end Y_OUT_MOD;				
Line 2		0000	Y OUT MOD ŞELAB:		
	55 DC CO	000 0000	entry subl2	Y OUT MOD_\$ELAB,^m <dv,iv> #I2.sp</dv,iv>	
	5C 00000000 EF DE	0005		SCONSTANT, ap	
	50 60 DE 51 51 DD	0000	BOVAL	(ap),ap ad.ad	
	F4 AD 5C D0	0012	Bovl	ap,-12(fp)	
	5C 00000000 ± EF 7E	0016	movad.	ADA\$HANDLER, ap	
	6D 5C D0	0100		ap,(Ip) Sconstant an	
		2200	C e a c al	ACOUSTANT, ar 4 (ad), ad	
		1200 1200			
	F4 AD 5C D0	0025	Bovl	ap,-12(fp)	
Line 13	F4 AD 00 DC	0032	movl	#0,-12(fp)	
		0036	Y OUT MOD \$ELAB\$RTN	LBL25:	
	04	0036	ret		
PSECT MAP					
Psect Hex Size 0 00000037 1 0000000C	Dec Size Name 55 YourMC 12 YourMC	D\$CODE	ANT		
\$ADAC-I-CL_ADDE Replaces ol	D, Package specificatic der version compiled 2	n Y OUT N 7-NoV-1986	IOD added to library i 00:15		
PORTABILITY SUM	IMARY				
There are	no uses of potentially	r non-port	able constructs:		
удужинэ улуадтт					

LIBRARY SUMMARY

USER2:[TFL.ADADIR.ADALIB]

		V OUT MOD is	
	Package Dogy procedure EQ	UIPMENT Y O(EQPT Y : in SWITCH; Equipment Switch FREQ_ERC_A,FREQ_ERC_B: in SWITCH; Osc Sw's REF_FREQ_LEVEL_A, REF_FREQ_LEVEL_B : in FLOAT; Osc Output in dbm EQPT Y OUT LEVEL A: in FAULT; Fail Flag EQUIP X OUT LEVEL Normal Output > 20.0 EQUIP Y OUT LEVEL SIM : in FLOAT; Fail Out Pwr EQUIP Y OUT LEVEL : OUT FLOAT) is Equip Pwr Out (actual)	
	Ref_freg_lev	el, Equip_Y_output_internal: FLOAT ; Internal variables	
	begin tf EQPT Y then if F then else else	= ON REQ_SRC_A = ON if FREQ_Ievel := REF_FREQ_LEVEL_A; if FREQ_SRC_B = ON then Ref_freq_level := REF_FREQ_LEVEL_B; else Ref_freq_level := -145.0;	
	11.00 11.000	REQ SRC A = ON) and (REF FREQ LEVEL A <= -120.0) REQ SRC A = ON) and (REF FREQ LEVEL A <= -120.0) if (FREQ SRC B = ON) and (REF FREQ LEVEL B <= -120.0) then Ref freq level := -145.0; end if;	
	9 t i f 9 t i f 9 t i f 9 t i f	<pre>if freq level <= -145.0 if Ref Freq level <= -145.0; if Ref Freq level < -5.0 then Equip Y output internal := -90.0; then Equip Y output internal := -90.0; else if EQPT Y OUT LEVEL FAIL = ACTIVE then Equip Y output internal := EQUIP Y OUT LEVEL SIM; end if for a serie free series of the ser</pre>	
	end EQU else EQU end if; end EQUIPME; end Y OUT MC	end if; if; if; P Y OUT LEVEL := Equip Y output_internal; IT Y OUT_LEVEL := -145.0; IT Y O; D;	
(AP	ł		
lex Size 0000108 0000000 0000000	Dec Size 267 12 1	Name Y OUT MOD.\$CODE Y OUT MOD.\$CONSTANT Y OUT MOD.\$DATA	

Laur_ANALLE 01	_			_	_	18-Jan- 18-Jan-		vax aua v1.3⊥. USER2:[TFL.ADADIR]FAULT_ANALYZE.ADA;3 (1)
-1 N M 4 1	with TYP with I_0_{ package_FAI	DEF_SIM; SIM_MOD; ULT_ANALY	use TYP_I ZE is	JEF_SIM;				
n vo r	procedure	e SWITCH	STATUS (EC	? PT_Y, FR	EQ_SRC_A,	FREQ_SRC_	3 : in SWITCH) ;	
~ \$\$ 5, O ;	procedure	e out_lev	EL (EQUIP EQUIP EQUIP	Y OUT LE	EVEL_SIM Vel, Vel : in	, Float) ;		
1 1 1 1 1 1 1 0 2 8 4 5 9	procedur	• DIAGNOS	EQUIP	Y, FREQ S T OUT LE T OUT LE T OUT LE	RC_A,FRE(VEL_SIM, VEL, VEL : IN	2_SRC_B : FLOAT) ;	in SWITCH;	
17 18	end FAULT	ANALYZE;						
PSECT MAP								
Psect Hex Size 0 00000020 1 00000000	Dec Size 32 12	Name FAULT FAULT	ANALYZE	.\$CODE .\$CONSTA	TN			
%ADAC-I-CL_ADDED Replaces old	, Package s er version	pecificat compiled	tion FAUL 18-Jan-1	Т АМАLYZ 987 21:2	E added 1	to library		
PORTABILITY SUMM	ARY							
There are	no uses of	potential	d-uou All	ortable	construc	t s		
LIBRARY SUMMARY								
USER2:[TFL.ADADI	R.ADALIB]							
Unit name		Nodes	Percer	it B.	locks	read	Uni	: kind
TYP_DEF_SIM I_0_SIM_MOD			41 41 7	1	5	3 5 4	Package specif Package specif	ication

ORIGINAL PAGE IS OF POOR QUALITY

(1)												
Image: Notice of the state	package body FAULT_ANALYZE is MESSAGE_NO : INTEGER;	procedure SWITCH_STATUS(EQPT_Y,FREQ_SRC_A,FREQ_SRC_B : in SWITCH) is	Called by MAIN SIM Test all on∕off switch stati, then decide upon begin what diagnosis to transmit to equipment monitor	MESSAGE_NO := 1 ; PUT(" Equipment output power is down in the noise level.") I O SIM_MOD.FAULT_MSG_LIST(MESSAGE_NO); if EQPT_Y = ON then	MESSAGE_NO := 2 ; Main switch is on so must be other failure I O SIM MOD.FAULT MSG LIST(MESSAGE NO); if (FREQ_SRC_A=OFF) and (FREQ_SRC_B=OFF) then	MESSAGE NO := 3; PUT("Both oscillators A and B are OFF so no wonder !!"); PUT(" Next pass try turning on either oscillator."); I O_SIM_MOD.FAULT_MSG_LIST(MESSAGE_NO);	MESSAGE_NO := 4; Selected oscillator must Bbe inoperative so MESSAGE_NO := 4; Rext time select the other oscillator I O SIM_MOD.FAULT_MSG_LIST(MESSAGE_NO); end_if;	<pre>MESSAGE NO := 5; Main Switch is 'OFF' that explains the poor output I O SIM_MOD.FAULT_MSG_LIST(MESSAGE_NO); end if; end;</pre>	procedure OUT_LEVEL(EQUIP_Y_OUT_LEVEL_SIM, EQUIP_Y_OUT_LEVEL, EQUIP_X_OUT_LEVEL : Tin FLOAT) is	<pre>begin This procedure contains some of the diagnostic logic for analizing a substandard power output MESSAGE_NO := 6; PUT(" The OPERATIONAL DIAGNOSIS is:"); PUT("The power output is "); I_O_SIM_MOD.FARAM_OUT(EQUIP Y OUT_LEVEL); if FQUIP Y_OUT_LEVEL <= -90.0 MESSAGE NO := 7;</pre>	PUT(" The reference oscillator level is below threshold ") PUT(" so next pass choose the other oscillator. "); I_O_SIM_MOD.FAULT_MSG_LIST(MESSAGE_NO); else	if EQUIP Y_OUT_LEVEL <= EQUIP_Y_OUT_LEVEL_SIM then MESSAGE NO := 8; FUT("A degradation of output has occurred,"); FUT("caused by an output device Failure.");
01 ⁻	10 7 2 2 0 7 3 2 7 0 7 8 7 0 7 9 7 10 7 9 7 10 10 7 9 7 10 10 10 10 10 10 10 10 10 10 10 10 10	5 2 7 2 7	7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7	9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9	990 97 97 97 97 97 97 97 97 97 97 97 97 97	0 0 1 7 1 0 0 0 1 7 1 0	1 4 4 4 4 4 4 1 4 10 0 C a	4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 10 10 1 2 4 10 1	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	6 9 6 9 7 0	27222

OFEGINAL PAGE IS OF FOOR QUALITY

-

```
-ТХ А¦' -Т.3-7
USER2': ["FLL. АДА⊔ІК] FAULI_ _ АИАГ' 446 - ADA; ] 3
 -1.37.7
    . -
                                                                                 --PUT("'Normal operation with normal output ");
I O SIM_MOD.FAULT_MSG_LIST(MESSAGE_NO);
end_if;
 --Jan --7 21 -- 57
18-Jan-1987 21:22:46
                                                                                                                                                                                                                                                                                                           else out_level(equip_y_out_level,sim, equip_y_out_level,
equip_x_out_level);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          %ADAC-I-CL_ADDED, Package body FAULT_ANALYZE added to library
Corresponds to package specification FAULT_ANALYZE compiled 18-Jan-1987 21:22
                                                                                                                                                                                          procedure DIAGNOSIS(EQPT_Y,FREQ_SRC_A,FREQ_SRC_B : in SWITCH;
EQUIP_Y_OUT_LEVEL,
EQUIP_Y_OUT_LEVEL,
EQUIP_X_OUT_LEVEL : in FLOAT) is
                                                                                                                                                                                                                                                                                       then SWITCH_STATUS(EQPT_Y,FREQ_SRC_B);
                                                            if EQUIP_Y_OUT_LEVEL <= EQUIP_Y_OUT_LEVEL
                                      I_O_SIM_MOD.FAULT_MSG_LIST(MESSAGE_NO);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                There are no uses of potentially non-portable constructs
        ------
                                                                                                                                                                                                                                                                 if EQUIP_Y_OUT_LEVEL <= -145.0
                                                                                                                                                                                                                                                                                                                                                                                                                                                                   FAULT ANALYZE.$CODE
FAULT ANALYZE.$CONSTANT
FAULT ANALYZE.$CONSTANT
                                                                                                                                                              end oUT_LEVEL;
                                                                                                                                       end if;
                                                                              then
                                                                                                                                                   end if:
                                                                                                                                                                                                                                                                                                                                                                                                  ••
                                                                                                                                                                                                                                                                                                                                                                           end DIAGNOSIS;
                                                      else
                                                                                                                                                                                                                                                                                                                                                                                                 end FAULT_ANALYZE
                                                                                                                                                                                                                                                                                                                                                                                                                                                            Name
                                                                                                                                                                                                                                                                                                                                                     end if;
                                                                                                                                                                                                                                                                                                                                                                                                                                                            Dec Size
617
                                                                                                                                                                                                                                               begin
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  12
8
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                USER2:[TFL.ADADIR.ADALIB]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             PORTABILITY SUMMARY
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         LIBRARY SUMMARY
                                                                                                                                                                                                                                                                                                                                                                                                                                                             Hex Size
00000269
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   00000000
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             00000008
          ы
                                                                                                                                                                                                                                                                                                                                                                                                                                       PSECT MAP
          C_AN
                                                                                                                                                                                                                                                                                                                                                                                                                                                             Psect
                                                                                                                                                                                                                                                                                                                                                                                                                                                                          0
                                                                                                                79
                                                                                           80
                                                                                                       81
                                              75
           -10
```

ORIGINAL PAGE IS OF POOR QUALITY

{ Explain_diagnos	LS IS	-	-	_			Jan-1987 Jan-1987	02:29:51 02:29:37	VAX A	V1.3 [{] 23 ADIR]EXPLAIN_DIAGN) JSIS.ADA;1	 rage 1 0 (1)	
- 0 -	with TYP_DE with I_0_SII	F_SIM; us M_MOD;	ie TYP_DE	KIS_3									
יז ע י ע	Package EXP	LAIN_DIAG	NOSIS is										
1978801	proced)	ure EXPLA	VIN_DIAGN REF EQUI EQUI EQUI	FREQ LE FREQ LE FREQ LE FREQ UE FREQ UE FREQ UE FREQ UE FREQ UE FREQ UE	PT Y, FRI VEL A, RI LEVEL S: LEVEL S: LEVEL ; LEVEL ; n out COI	EC_SRC EF_FREQ IM, 1 in flou	A,FREQ_S LEVEL_B IW failu tmal out AT;	RCB: in , Osc (re out lev Power	SWITCH; but rel	Equipmt sw's			
12 13	end EXPLAIN	DIAGNOSI	(S;										
PSECT MAP													
Psect Hex Size 0 00000020 1 0000000	Dec Size 32 12	Name Explain Explain	"_DIAGNO!	sis\$cc sis\$cc)DE NSTANT								
\$ADAC-I-CL_ADDE Replaces ol	D, Package sp der version c	ecificat: ompiled 3	ion EXPLU 11-Jan-19	AIN_DIAC 987_02:2	inosis ad 22	ded to	library						
PORTABILITY SUM	MARY												
Тһөгө аге	no uses of p	otential]	ly non-p	ortable	construc	ts							
LIBRARY SUMMARY													
USER2:[TFL.ADAD	IR.ADALIB]												
Unit name		Nodes	Percer read	l l	locks	read		пIJ	it kind				
TYP_DEF_SIM I_o_SIM_MOD			10 58		د م		Pac	kage speci kage speci	fication fication				
OPTON													

(4-<u>2-</u>

ORIGINAL PAGE IS OF PCCR QUALITY

it sor	-	_	0		_	· · · 	Jan 7 04 56 / ^X Jan 1987 02:29:37 [TF	A L.34 L.ADADIR EXPLAIN_DIAGNOSIS.ADA;10	(1 (1
			else if E E O U MES S S S S S S S S S S S S S S S S S S	IP_Y_OUT AGE_NO : (A_degr (-90.0 (This in (since (but th SIM_MOD.	LEVEL = 16; adation dbm but dicates some sig ere was EXPLAIN_	<pre><= EQUII of out that th nal was probabl MSG_LIS'</pre>	POUT_LEVEL_SIM had occurred, the out wa: 11 less than or equal to e selected oscillator wa: 'leaking' through the c y an out device Failure. T(MESSAGE_NO);	<pre>s above); the out failure dbm.); s OK,); ircuitry;););</pre>	
			else if E then ME ME ME d	QUIP_Y_O SSAGE_NO (Norma (sinc o_SIM_MO	UT_LEVEL = 17; al opera e the ou D.EXPLAI	<= EQU tion wi tis ab M_MSG_L	IP_X_OUT_LEVEL th normal or near normal ove the out failure leve IST(MESSAGE_NO);	out); 1.)	
	end EX end Explai	end PLAIN_DI n_Diagno	end if; end if; l if; AGNOSIS; sis;						
1128 0059 000C	Dec Size 249 12 1	Name Explai Explai Explai	in Diagno in Diagno in Diagno	sis.\$COI sis.\$CON sis.\$DA1	JE HSTANT FA				
ADDED s olde oonds t s SUMM	, Package bo ar version c to package s ARY	dy EXPLJ compiled specific	AIN DIAGN 11-Jan-J ation EXI	10515 add 987 02:3 1.AIN_D14	ded to li 22 Agnosis c	ibrary compileo	i 11-Jan-1987 02:29		
2 2 2 2	ao uses of F	potentia	lly non-E	ortable	construc	t s			
MARY	R.ADALIB]								
name AGNOSI	N	Nodes	Perce rea 16 100 16 99	B	slocks 9 8	read	Unit kin Package specificati Package specificati Package specificati	д оп оп	

GREAD ME PARE IS GE PEOR QUALITY

ORIGINAL PAGE IS OF POCR QUALITY

| 1 |_1 | 0FF -60.0 ____ ----____ ~ ----

INITIAL_SIM.DAT

[]MAIN_SIM_MOD3	-	_	-	 20-JAN-198	 7 00:52	 vax-11 Li	inker V04-00	{ Page 1	
			: Object Modu	.e Synopsis : 					
Module Name 	Ident 	Bytes 	File 			Creation Date	Creator		
ADA\$ELAB_MAIN_S]	CM_MOD3 01 11	82 [TF 67 TYP	L.ADADIR]MAIN DEF_SIMOBJ	SIM_MOD3.08J;1 22	1.2	0-Jan-1987 00:52 7-NOV-1986 15:34	VAX Ada V1.3-23 VAX Ada V1.3-23 vi 0-7		
TO EXCEPTIONS	01	6 [SY 67 [TF	SLIB ADALIB]I L.ADADIR.ADAL 	D EXCEPTIONS .01 IBJY OUT MOD .01	BJ;1 BJ;2 2 1:1 1	6-MAR-1983 12:40 7-NOV-1986 00:25 8-JAN-1987 22:18	VAX Ada V1.3-23 VAX Ada V1.3-23 VAX Ada V1.3-23		
Y OUT MOD TEXT IO	01	280 [TF 31 [SY 184 [SY	L.AUAUIK.AUAU SLIB.ADALIB]T SLIB.ADALIB]T	EXT IO .0BJ;1 EXT IO.0BJ;1	1	6-MAR-1985 12:42 6-MAR-1985 12:43	VAX Ada V1.0-7 VAX Ada V1.0-7		
TEXT IO I O SIM MOD I O SIM MOD	01 01	14608 100	SIM MOD OBJ	;514	- N	1-JAN-1987 00:13 0-JAN-1987 00:51	VAX Ada VI.3-23 VAX Ada VI.3-23		
EXPLAIN_DIAGNOS		44 EXP	LAIN_DIAGNOSI	s0BJ;5	1	1-JAN-1987 02:29	VAX Ada V1.3-23		
EXPLAIN_DIAGNOS FAULT_ANALYZE FAULT_ANALYZE	IS 01 01 01	262 EXP 44 FAU 637 FAU	LAIN DIAGNOSI LIT ANALYZE OB	s.0BJ;3 BJ;3 J;1 J:1		1-JAN-1987 02:29 8-JAN-1987 21:22 8-JAN-1987 21:22 8-JAN-1987 21:22 8-JAN-1987 22:31	VAX Ada VI.3-23 VAX Ada VI.3-23 VAX Ada VI.3-23 VAX Ada VI.3-23		
MAIN_SIM_MOD3	01		Program Sect	ion Synopsis					
	Modulo Name	Base Er	1 I	,ength	Align	_	Attributes		
					FONG	PIC, USR, CON, RI	EL, LCL, SHR, NOEXE,	RD, NOWRT, NOVEC	
\$ CONSTANT	TYP_DEF_SIM_		0020B 000000000000000000000000000000000	12.)	FONG	~ ~			
			00223 000000000000000000000000000000000	12.	DNO1 (2			
	I O SIM MOD	00000230 000	01117 00000EE	3 (3816.	DNOT (2			
	EXPLAIN DIAGNOS	15 00001118 000	01123 0000000	c (12.) LONG	7			
	EXPLAIN DIAGNOS	IS 00001124 000	0112F 000000	c (12.) LONG	5			
	FAULT ANALYZE FAULT ANALYZE	00001130 000 0000113C 000	0113B 0000000 01147 0000000	c (12.	FONG	2 2 DIC USR OVR B	EL.LCL, SHR,NOEXE,	RD, NOWRT, NOVEC	
\$ ZERO		00001200 000	01207 0000000 01207 00000000	8 (8 -) PAGE	6			
		00001200 000	01200 0000000 0123B 0000003	1 (1. 4 (52.) PAGE) LONG	9 2 NOPIC,USR,CON,R	EL, GBL, NOSHR, NOEXE,	RD, NOWRT, NOVEC	
77777777777777777777777777777777777777	ADA\$ELAB_MAIN_S	IM MOD3 0001208 000	0123B 000003	4 (52.) LONG	2		BD WRT NOVEC	
SDATA		00001400 000	0144B 0000004	c (76.) LONG	2 PIC,USR,CON,K	EL, LCL, NUSAR, NUEAL,		
	Y OUT MOD TEXT IO	00001400 000 00001404 000	01400 000000000000000000000000000000000	т (9 (FONG	1 7 9			
	I O SIM MOD	0000140C 000	0143F 0000003	4 (52.) LONG	7			
	EXPLAIN_UANU	00001440 000		1 (DNOT (2 ,			
	FAULT_ANALYZE	00001444 000	01448 000000	· ,		4			

Vaun2: [TDA	DIIN_L			-	-JAI	1 01	(VAX	H inke	4-0			م م
Psect Name	Module Name	Base	End	Length		Align		Attı	tibutes			
0 0 0 C 0 0 C 0 0	ADA\$ELAB_MAIN_S TYP_DEF_SIM TYP_DEF_SIM TV_OUT_MOD TEXT_TO TEXT_TO TEXT_TO TEXT_TO TEXT_TO TEXT_TO TEXPLAIN_DIAGNOS EXPLAIN_DIAGNOS EXPLAIN_DIAGNOS EXPLAIN_DIAGNOS FAULT_ANALYZE FAULT_ANALYZE MAIN_SIM_MOD3	00001600 IM MOD3 000016160 00001655 00001655 00001658 00001658 00001792 00001792 00001792 15 00001792 00001792 15 00001792 15 00001792 00004279 15 00004279 00004392 00004618	00000480D 00001654 00001654 00001654 00001691 00001691 00001691 00001691 00001691 00001865 00001865 00001865 00004298 00004391 00004391 00004301 00004301 00004301	0000320E (000001E (0000006 (0000006 (00000037 (00000037 (00000017 (00000017 (00000017 (00000017 (00000017 (00000017 (00000017 (00000020 (00000020 (00000020 (00000020 (0000017 3 (00000020 (000000020 (000000020 (000000020 (00000020 (00000020 (00000020 (00000020 (000000020 (00000020 (0000000000	12814.) 30.) 55.) 66.) 267.) 178.) 178.) 178.) 178.) 178.) 178.) 178.) 178.) 178.) 178.) 178.) 178.) 178.) 178.) 178.) 10739.) 499.)	LONG 2 BYTE 0 BYTE 0	PIC, USR, C	ON, REL, LO	2.T. , SHR ,	εxε,	RD, NOW	RT, NOVEC
Symbol		Value	-+ [odmay2		+	Value	Symb	01			Val	au
ADA\$ELAB_MAIN	SIM_MOD3	0001000	- 			-		1			1	1

ORIGINAL PAGE IS OF POOR QUALITY

Key for special characters above: +------+ : * - Undefined : : U - Universal : : R - Relocatable : : X - External : : V - Vectored :

÷

U	- JA	Ni 7 od	AX- nke	4 – 0 C	·	Pa(
	++ ! Image Synopsis : +++					
Virtual memory allocated: Stack size: Image header virtual block limits: Image binary virtual block limits: Image name and identification:	00000200 00004BFF 0 20. pages 1. 1. 1. 2. 37. MAIN SIM MOD3 01	0004A00 (18944. (1. block) (36. blocks)	bytes, 37. pages))			
Number of files: Number of modules: Number of program sections: Number of global symbols: Number of image servions:	19. 19. 11.					
User transfer address: Debugger transfer address: Number of code references to shareable images: OTS transfer address to shareable images:	15. 00001600 7FFEDF68 24.					
oro cranster address - LLBYINITIALIZE: Image type: Map format: Estimated map length:	0000480E EXECUTABLE. DEFAULT in file USEM 83. blocks	R2:[TFL.ADADIR]M	AIN_SIM_MOD3.MAP;	m		
+ +	Link Run Statistics !					
Performance Indicators	Page Faults	CPU Time Ela	apsed Time			
Command processing: Pace 1.	145	00:00:00.16	00:00:00.	18		
Allocation/Relocation:	417 31	00:00:02.33 00:00:00.09	00:00:04.	05 46		
Man data after chick militations.	63	00:00:01.21	00:00:02.	59		
war wata atter upject module synopsis: Symbol table output:	4 O	00:00:00.10	00:00:00	10		
Total run values:	660	00:00:03.91	00:00:00:07.	14 52		

---m

Using a working set limited to 1350 pages and 686 pages of data storage (excluding image)

Total number object records read (both passes): 944 of which 41 were in libraries and 211 were DEBUG data records containing 26139 bytes 2835 bytes of DEBUG data were written,starting at VBN 38 with 6 blocks allocated

0

11 Number of modules extracted explicitly with 2 extracted to resolve undefined symbols O library searches were for symbols not in the library searched

A total of 0 global symbol table records was written

LINK/MAP=[]MAIN_SIM_MOD3/EXE=[]MAIN_SIM_MOD3 SYS\$INPUT:/OPTIONS