AFHRL-TP-86-30

(12)

AIR FORCE

HUMAN RESOURCES

AD-A172 905

MAN-MACHINE INTERFACE CONCEPTS FOR AN ADVANCED INTEGRATED MAINTENANCE INFORMATION SYSTEM

Andrew P. Chenzoff
Debra C. Evans
Reid P. Joyce
J. Thomas Roth

Applied Science Associates, Incorporated
P.O. Box 1072
Butler, Pennsylvania 16003

LOGISTICS AND HUMAN FACTORS DIVISION
Wright-Patterson Air Force Base, Ohio 45433-6508

September 1986

Final Technical Paper for Period May 1985 - March 1986

Approved for public release; distribution is unlimited.

LABORATORY

DTIC FILE COPY

AIR FORCE SYSTEMS COMMAND
BROOKS AIR FORCE BASE, TEXAS 78235-5601

86 10 [illegible] 011

NOTICE

This paper has been reviewed and is approved for publication.

DONALD L. THOMAS
Contract Monitor

BERTRAM W. CREAM, Technical Director
Logistics and Human Factors Division

DONALD C. TETMEYER, Colonel, USAF
Chief, Logistics and Human Factors Division

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

ADA172905

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) AFHRL-TP-86-30
6a. NAME OF PERFORMING ORGANIZATION Applied Science Associates, Incorporated 6b. OFFICE SYMBOL *(If applicable)*	7a. NAME OF MONITORING ORGANIZATION Logistics and Human Factors Division
6c. ADDRESS (*City, State, and ZIP Code*) P.O. Box 1072 Butler, Pennsylvania 16003	7b. ADDRESS (*City, State, and ZIP Code*) Air Force Human Resources Laboratory Wright-Patterson Air Force Base, Ohio 45433-6508
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Air Force Human Resources Laboratory 8b. OFFICE SYMBOL *(If applicable)* HQ AFHRL	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N66001-83-D-0059
8c. ADDRESS (*City, State, and ZIP Code*) Brooks Air Force Base, Texas 78235-5601	10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.
63751F	2362	00	03

11. TITLE *(Include Security Classification)*
Man-Machine Interface Concepts for an Advanced Integrated Maintenance Information System

12. PERSONAL AUTHOR(S)
Chenzoff, A.P.; Evans, D.C.; Joyce, R.P.; Roth, J.T.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (*Year, Month, Day*)	15. PAGE COUNT
Final	FROM May 85 TO Mar 86	September 1986	106

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP
05	08	
05	09	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
battle damage assessment; computer-based performance aids; diagnostics; maintenance; maintenance information systems; troubleshooting

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*
The Air Force Human Resources Laboratory is developing the Integrated Maintenance Information System (IMIS). The purpose of this system is to consolidate all of the various information systems used in Air Force maintenance into a single set of hardware, software, and protocols. The IMIS will increase productivity by reducing the amount of training required for technicians to be able to access the full set of maintenance data and by making a wider range of data readily available to each technician.

This paper explores the man-machine interface requirements of a system such as IMIS. By extrapolating current hardware and software trends into the future, it predicts the types of user-friendly features that will probably be available when the IMIS is fully implemented. It catalogs the types of information that need to be contained in such a maintenance information system. It outlines the relationships among the individual informational elements. It describes the data-processing functional requirements that should be present to process information in the system. It describes display formats for presenting each class of data and illustrates each format. It describes the interactive capabilities that must be incorporated to permit the technician to operate the system easily and effectively.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Nancy A. Perrigo, Chief, STINFO Office	22b. TELEPHONE *(Include Area Code)* (512) 536-3877 22c. OFFICE SYMBOL AFHRL/TSR

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted.
All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

AFHRL Technical Paper 86-30 September 1986

MAN-MACHINE INTERFACE CONCEPTS FOR AN ADVANCED INTEGRATED MAINTENANCE INFORMATION SYSTEM

Andrew P. Chenzoff
Debra C. Evans
Reid P. Joyce
J. Thomas Roth

Applied Science Associates, Incorporated
P.O. Box 1072
Butler, Pennsylvania 16003

LOGISTICS AND HUMAN FACTORS DIVISION
Wright-Patterson Air Force Base, Ohio 45433-6508

Reviewed by

Donald L. Thomas
Combat Logistics Branch
Logistics and Human Factors Division

Submitted for publication by

Bertram W. Cream
Technical Director
Logistics and Human Factors Division

This publication is primarily a working paper. It is published solely to document work performed.

SUMMARY

This paper presents man-machine interface concepts that could be used in an advanced system that stores and retrieves all of the information required by maintenance technicians and by their managers and supervisors. The analysis began with certain hardware assumptions. Although conventional computer terminals will be used in part of the system, the primary focus of the analysis was upon ways to present information to a flightline or shop technician through a lightweight, battery-powered, portable computer. This computer was assumed to have high processing speed and power, high memory capacity, a built-in radio transmitter/receiver, and a large, legible, flat-panel display.

The software concept was formulated to adapt the man-machine interface to the individual user's needs and preferences. As conceived, the default presentations of content and format features will be based upon what the user has asked for in the past and how he/she has asked that the information be presented.

When using the device during troubleshooting, the user will have the option of either using his/her own troubleshooting strategy (with the computer providing the necessary information and keeping track of what is known about the "health" of various portions of the malfunctioning system) or asking the computer to suggest the next test to be performed.

Some of the output techniques discussed are: adaptive menus, graphics pyramiding, variable-scale windows, speech synthesis, dynamic illustrated parts breakdowns, "peel-away" illustrations, "suspect" lists, and multiple windows. Some of the input techniques discussed are: speech recognition, touch-screen, adaptive manual function keys, and structured natural language input.

PREFACE

This document presents man-machine interface concepts that could be used in the Integrated Maintenance Information System (IMIS) being developed by the Air Force Human Resources Laboratory (AFHRL). It is a "think piece" that extrapolates current trends into the future and provides a fresh perspective on the functional requirements of the IMIS, especially in the area of man-machine interface. The views presented in this paper are those of the contractor and do not uniformly represent the views of the AFHRL scientists working on the IMIS project. However, these views will be fully considered during IMIS development.

Preparation of this document was accomplished under Delivery Order 7J04 of Contract Number N66001-83-D-0059, as part of a joint Air Force/Navy effort to develop advanced man-machine interface concepts to meet the functional requirements of integrated, user-defined technical information delivery systems. The technical monitor for the Navy was Dr. Robert J. Smillie. The technical monitor for the Air Force was Dr. Donald L. Thomas.

The Project Director for the contractor was Mr. Reid P. Joyce. The Principal Scientist for the project was Dr. Andrew P. Chenzoff. Sharing responsibility for the preparation of this paper were Mr. Joyce, Dr. Chenzoff, Dr. Debra C. Evans, and Dr. J. Thomas Roth.

TABLE OF CONTENTS

TABLE OF CONTENTS
(continued)

LIST OF FIGURES

TABLE OF CONTENTS
(continued)

LIST OF FIGURES
(continued)

Figure Page

I. INTRODUCTION

Background

The computer-based information systems being introduced into Air Force maintenance operations have the potential for greatly improving the efficiency of maintenance. Some such systems have been implemented, some are under development, and others are planned. They store and retrieve information in areas such as technical data, training, diagnostics, management, scheduling, and the maintenance history of aircraft. A problem arises when each of these information systems requires its own hardware and software, and each requires the operator to learn a new protocol before it can be used. Significant economies and improvements in productivity can be achieved by developing a single system that will integrate all of the information systems used in maintenance. The Air Force Human Resources Laboratory (AFHRL) is presently engaged in an effort to develop such a system -- the Integrated Maintenance Information System (IMIS). It will provide a single set of hardware, a single set of software, a single set of protocols to use, and a single man-machine interface (MMI). By learning to use this one system, the technician will be able to access any of the information systems used in maintenance.

The design and development of the IMIS is a difficult and complex task, requiring the skills of many disciplines. Although a significant amount of preliminary work has been done by AFHRL personnel on the basic concepts and design of the system, an outside "view and analysis" of the problem was needed, to provide a fresh perspective and to ensure that all relevant factors are considered, especially with regard to the MMI. To this end, a contract was awarded to Applied Science Associates, Inc., to conduct a "think" task to examine the objectives of the IMIS program, to develop advanced man-machine techniques, to develop a description of the IMIS as they see it, and to make recommendations for the development of the IMIS. This document presents the results of this effort. It provides a detailed description of the ideal IMIS, as seen by the scientists on ASA's staff. The views presented here are those of the contractor and do not necessarily represent the views of AFHRL scientists working on the development of the IMIS. They are presented, essentially as submitted, as an additional source of ideas and information for personnel working on the design of IMIS or similar systems.

For quite a number of years, the military services have made advances in the presentation of technical information to support the performance of maintenance technicians. Although the format and content of this technical information have been improved, maintenance technicians still have to contend with some bulky, poorly indexed, poorly organized, and poorly written manuals. In most cases, the information they need is in the manuals, but it is too hard to extract or too hard to understand. In some cases, the information that would be the most helpful is absent.

Current maintenance documentation also gives insufficient attention to the professional development of the maintenance technicians. When they consult some types of manuals for troubleshooting aid, they are told what tests to perform, but neither their troubleshooting skills nor their understanding of equipment functioning is enhanced. In some theory of operation sections, they are told in great detail how the system was designed and what specifications it meets, instead of being told, in terms they can understand, how each functional unit accomplishes its functions, how various functional units interact, and how the system interacts with other systems that provide its inputs and utilize its outputs.

One reason for the failure of paper-based technical manuals to achieve the clarity and utility of which they are capable has been a lack of understanding of how people perform maintenance and how they use technical information on the job. This problem has been strongly addressed in the current effort. Earlier studies performed under the present work effort have:

1. Investigated the information needs of, and the technical information retrieval processes used by, North Atlantic Treaty Organization (NATO) Seasparrow Surface Missile System technicians.

2. Investigated the use of technical information by experienced and inexperienced civilian electronics maintenance personnel during the troubleshooting process.

3. Described the functional requirements for accessing technical information in future user-defined technical information systems, based upon the needs of both experienced and novice troubleshooters.

4. Described the decision processes used by Air Force maintenance technicians and battle damage assessors, listing the information needs that will have to be fulfilled by future technical information delivery systems.

The advent of personal computers (and, especially, lap-top portables) has made it feasible to consider providing each maintenance technician with a portable computer that contains all of the information needed to complete the jobs of that technician. By speeding up information access time, a computer-based technical information system will quickly pay back its development costs. But access time is not the only important dimension. It is also vital that the right information be presented at the right time, that this information be 100%

accurate, that it be clear and comprehensible, and that it increase the maintenance skills of the user. A computer-based technical information system is not a panacea; it will not cure all ills. It is just as easy to enter poorly written, poorly organized technical information into a computer database as it is to present it in the paper format. It is easy to make computer-based information difficult to retrieve. However, it is hoped that the mistakes of the past will not be repeated (or will be repeated with lower frequency). The rapid strides being made in computer hardware and software must be matched by advances in technical information display format, data access methods, frame content, and user interaction techniques. The latter factors, referred to as "advanced MMI concepts," are addressed in this paper.

Computer hardware and software technologies have developed to the point that an electronic performance aid system can be procured without further delay. Processing speed, which has been considerably advanced by the very high speed integrated circuits (VHSIC) program, is increasing at an accelerating rate. Memory capacity is growing at an incredible rate, and cost per unit of memory is dropping drastically. People who were working with 32 kilobytes of memory 4 or 5 years ago are now buying units with 30 megabytes of memory. It is presently possible to store over half a gigabyte of information (or the equivalent of an encyclopedia) on a compact-disc ROM (read-only memory).

The bottleneck hardware problem for lightweight portable computers is presently display contrast. As things now stand, a display that is highly readable consumes so much energy that it cannot be powered for very long by a battery that is portable. Until readable display media with lower power consumption or lightweight batteries with higher storage capacity are developed, the sale of lap-top computers will remain at its present low level. The computer industry is presently hard at work on this problem, and a breakthrough can be expected in the near future. Most present portable computers use a liquid crystal display (LCD) because this type of display is cheap, lightweight, and has low power requirements. There are LCD displays in the prototype stage that offer three times the contrast of traditional LCDs. Active-matrix LCD prototypes have even broken the color barrier. Work is also continuing on light-emitting displays, such as plasma, electroluminescent, and vacuum fluorescent displays. Their practical size is growing and their power requirements are shrinking. Perhaps the ultimate victor in the marketplace will be a hybrid of two or more technologies. By the time a full-scale procurement of a computer-based maintenance information system is initiated, it is almost certain that display contrast will have been improved to an acceptable level.

Integrated Diagnostics

As presently conceived, the portable computers in the hands of the technicians will be only part of a larger IMIS. The other major IMIS systems will be the Automated Technical Order System (ATOS), the Intelligent-Computer-Aided Instruction (ICAI) system, and the Core

Automated Maintenance System (CAMS). The ATOS will be a computer database of technical order information, together with the apparatus for creating, correcting, and updating the database. The ICAI system will contain on-the-job training, explanations of system modifications, advanced upgrade training, and systems training. It will be possible to present any of these training modules on the technician's portable computer. In addition, the technician will be able to suggest changes to the training or to the ATOS database through a feedback capability designed into the IMIS. The CAMS will store and process information related to work management, maintenance control, maintenance reporting, aircrew debriefing, and supply.

The portable computer will be used in a stand-alone mode for entering data into, and accessing data from, a set of plug-in memory modules. It will contain a radio system for voice communications with other base personnel, such as maintenance control and flightline expediters. Secondly, the technician will be able to connect the portable computer to an aircraft external maintenance panel, to interface with on-board systems. Thirdly, the technician will be able to connect with the ground-based subsystems (mentioned in the previous paragraph) through a maintenance workstation at the workcenter.

The portable computer will be as ubiquitous as the present paper-based technical orders. There are few maintenance activities that the computer will not support. The computer will be used for supporting corrective and preventive maintenance. It will be used in organizational, intermediate, and depot maintenance. It will be used during troubleshooting and repair tasks. It will be used for ordering spare parts and supplies, and for communicating the status of those orders. It will be used during debriefing and during bench testing. It will be used for documenting repair actions and deferred maintenance.

The support of battle damage assessment is an additional function to be served by the portable computers. Under the Aircraft Battle Damage Repair (ABDR) program of the Air Force, Combat Logistics Support Squadrons (CLSSs) are available for deployment at a moment's notice to any spot on the globe. The portable computers will be used by these CLSSs. In addition, organic units will use the portable computers to help them begin battle damage assessment before the arrival of a CLSS team. The battle damage assessment function is an ideal candidate for electronic aiding because of the time pressures under which assessors must work when performing their wartime mission and because the information they need is not retrieved frequently enough to be memorized.

Hardware Assumptions

Although the portable computer that is issued to the maintenance technician will be only a part of the total IMIS, the focus of this paper is on that device and the interface between that device and its user. In describing the MMI, the following basic hardware assumptions have been made about the properties of this portable computer:

1. The computer will weigh less than 10 pounds.

2. The computer will be able to run for extended periods on battery power or other available power sources. Eight hours is a minimum; 12 hours is a target.

3. It will have sufficient processing speed and power to display complex graphics and handle moderately complex artificial intelligence (AI) software.

4. It will be able to provide access to large amounts of technical data. It will have plug-in modules capable of holding over 10 megabytes of data, or it will be able to access compact-disc ROMs with over 500 megabytes of data.

5. It will have the capability to download information input by the technician, from the portable computer to some interfacing device, and ultimately to a mainframe computer.

6. It will have interfaces for interacting with the systems being maintained, for the purpose of conducting Built-In Test Equipment/Automated Test Equipment (BITE/ATE) tests.

7. It will contain a radio for voice communications with other base organizations.

8. The display will be easily legible under a wide variety of lighting conditions. It will be a flat panel, at least 10 inches in size, with at least 512 by 512 pixels resolution.

Other hardware capabilities, deriving from the ones listed above, will be mentioned in conjunction with the software requirements. No firm assumption has been made about the user input interface device. Although at some point it would be desirable for the user to be able to input data through a QWERTY keyboard, the computer can be an effective information retrieval device without such a keyboard. It should be possible to exercise all of its retrieval functions with a limited keyboard, with a touch screen, or with voice-activated commands. In addition, it should be possible to input alphanumeric data by steering a cursor through a menu of alphanumeric characters presented on a screen. All that is required is a means of steering the cursor, plus a function key. The only occasion when a full keyboard is a necessity is when the technician is inputting extensive alphanumeric data. This will probably be necessary only at the workcenter.

Purpose of the Present Paper

This paper presents MMI concepts for an advanced integrated maintenance information system, and covers the following topics:

1. Information Content Requirements

2. Data Relationships/Organization

3. Data Analysis/Processing Functional Requirements

4. Interactive Capabilities

5. Display Formats

The discussion will address the software and hardware functional capabilities that will have to be developed to smooth the transactions between the technician and the system. This system must meet certain fundamental design requirements. The system must be quick, easy, and pleasant to use, so that it meets with total user acceptance. It must consolidate into one channel the information that the technician heretofore has obtained from multiple sources. It must have standardized commands and displays. It must allow the user to integrate and analyze various types of information on a single display. This paper describes, in basic terms, some approaches that should be considered for fulfilling these requirements.

II. INFORMATION CONTENT REQUIREMENTS

Introduction

Before presenting the information content requirements, it is necessary to set forth a context in which the maintenance information will be provided. The maintenance information system will consist of:

1. At least three mainframe computers:

   Automated Technical Order System (ATOS) - Where all of the technical information will be generated and updated.

   Intelligent Computer-Aided Instruction (ICAI) System or Intelligent Tutor System (ITS) - Where all of the non-resident training will be created and updated, including on-the-job training (OJT), explanations of system mods, advanced upgrade training, and systems training.

Core Automated Maintenance System (CAMS) - Where all information related to work management, maintenance control, maintenance job reporting, aircrew debriefing, and supply will be stored and processed.

2. Desk-top computers that interface with the mainframe computers. The following organizations (in the Air Force) will probably have desk-top computers:

   The Deputy Commander for Maintenance (DCM) Staff

   Administration

   Programs and Mobility

   Training

   Production Analysis

   Quality Assurance/Quality Control (QA/QC)

   Plans and Scheduling

   Job Control

   Materiel Control

   Each shop.

3. Portable computers for each technician who needs one.

At the beginning of a shift of work, the technician will check out a portable computer that has been customized for his/her use. The computer will contain information relevant to the jobs that have been scheduled for that technician. It will have been customized to contain this technician's assignment information from CAMS, the required technical information from ATOS, and this technician's personal Notepad (see p. 16) and pattern of technical information use. If assignments are added or altered during the work shift, the technician will update the information in his/her portable computer by plugging it (or its memory module) into one of the desk-top computers (or into a flightline terminal) and entering his/her ID number. This operation will usually take less than a minute, and never more than 5 minutes. Alternatively, the user may be required to log into the system and to communicate features of the job and the subject equipment to the system. It is at this point that the system will decide whether the user is authorized to retrieve the requested information, and what kinds of data manipulation the user will be permitted to accomplish. Information that the system will request will include identification of the user, the job, and the tail number or serial number of the equipment. The job information could be entered by job control instead of the user; the system should allow either method.

Information to be Provided by the Computer

The discussion that follows assumes that the maintenance information is being used by a technician whose primary task is corrective maintenance -- whose aim is to restore a malfunctioning equipment system to an acceptable level of functioning. The following questions need to be answered by the portable computer:

I. What are my assignments for today?

A. What job numbers have I been assigned to?

B. What sort of equipment is involved?

II. What is already known about this malfunction?

A. What was the equipment doing when the malfunction was first observed?

B. What were the environmental conditions at the time of the malfunction?

C. Had the equipment been subjected to any unusual stresses?

D. What were the immediately observable symptoms of malfunction?

E. What were the results of any subsequent tests that may have been performed?

III. What is this equipment designed to do and how does it do it? (Although this topic will be stressed in training, the technician will need to be able to refer to theory of operation from time to time.)

A. What are the major stages and how does each stage work?

B. How do the stages work together to produce the desired system outputs?

IV. How is the equipment operated? (This type of reference information will be necessary only for novices.)

V. Given the current symptom pattern, what portion of the system is implicated?

VI. What tests could be performed to localize the malfunction?

A. What relevant tests are quick and easy to perform?

B. What relevant tests are the most diagnostic?

VII. How has this weapon system (system, subsystem, unit) failed in the past?

A. What is the recent maintenance history of this aircraft?

B. What is the reliability of the line replacement units (LRUs) in the "ambiguity group"?

C. When the current initial symptoms have been present in the past, what has been the source of the malfunction?

VIII. What are the procedures for performing the needed tests?

A. How do you gain access to the test points?

B. How do you connect and operate the test equipment?

IX. What are the nominal readings and tolerances for each test point?

X. What is our current state of knowledge about where the malfunction lies?

A. What tests have I already performed, and what were the results?

B. What is the current "ambiguity group"?

C. What was the previous test point and what were the "suspects" at that time?

XI. What are the procedures for repairing this equipment?

A. Assembly/Disassembly

B. Lubrication

C. Adjustment/Alignment/Calibration

XII. How do you perform inspection or preventive maintenance tasks on this equipment? (This information will not be provided unless required as part of a corrective maintenance task.)

XIII. How do you check out the operation of the equipment most efficiently?

XIV. What are the system input and output specifications?

XV. What must you do (or avoid doing) to keep from hurting yourself or harming the equipment?

XVI. What are some of the "tricks of the trade"?

XVII. What are the most common misconceptions about this equipment, and what are the most common pitfalls in troubleshooting it?

In order to answer the above questions, the computer will also have to contain a library of graphic support materials. Along with the text, it will have to be capable of presenting the following:

XVIII. Locator diagrams.

XIX. Physical illustrations of the equipment (at several levels of detail).

XX. Block diagrams and schematic diagrams (at several levels of detail), with and without the "suspects" highlighted.

XXI. Motion pictures.

In addition to answering the above questions, the maintenance information system will have to be able to accept, store, and process information supplied by the technician. (However, some of this information will not be entered into the system through the portable computer if the portable computer does not have a QWERTY keyboard. It will be entered instead at a desk-top computer or terminal.) The types of information that the system must be capable of accepting from the technician are the following:

XXII. What was the result of each test performed?

XXIII. What work was accomplished by the technician? (AFTO Form 349)[1]

XXIV. What repairs were made to the equipment? (AFTO Form 781A)[1]

XXV. How should the technical information be changed? (AFTO Form 22)[1]

XXVI. What spare parts are needed for repair?

XXVII. How should the training be changed?

These questions will require the following types of data in order to answer them: linear procedures, branching procedures, support information, supply information, graphics, and maintenance history. The system will also supply job information such as assignments and debriefing results.

---

[1] AFTO Form 349, Maintenance Data Collection Record.
AFTO Form 781A, Maintenance Discrepancy and Work Document.
AFTO Form 22, Technical Order System Publication Improvement Report and Reply.

Linear Procedures

Linear procedures consist of step-by-step guidance for the performance of specific tasks. The steps consist of text which may or may not be supported by illustrations, depending upon the level of detail to which the procedure has been written. The graphics include illustrations, at any of several levels of detail, and tabular presentations such as tables of values or checklists. A linear procedure will include input conditions, set-up instructions for both prime and test equipment, procedural steps presented via text and illustrations, necessary readings and tolerances, and any notes, cautions, or warnings that apply over the course of the procedure.

A linear procedure may also include information in tabular form if such a form is appropriate for the kind of information displayed, such as readings and tolerances, or specifications (torque, spark plug gaps, etc.). The procedure may also allow the option to gain direct access to generic procedures (soldering, safety-wiring, etc.), reference information (theory of operation), schedules for preventive maintenance, and schematic diagrams.

The need to access a linear procedure will typically arise at the beginning of a task. Entering the task name into the system should result in retrieval of the linear procedure for that task, or a menu listing two or more procedures related to the task. Because users may occasionally have to interrupt a task somewhere in the middle and return to it later, it should also be possible to go directly to a frame or paragraph (identified by name or alphanumeric identifier) within a given procedure.

Branching Procedures

Branching procedures have all of the user-interface elements of linear procedures, except that they require the user to make (and report) decisions over the course of the task. A branching procedure such as a troubleshooting or checkout task will go wherever the user's decisions lead, either by following a path in an existing "tree" structure, or by "firing" rules in an expert system that can generate the procedure on the fly. The technical information system must be able to capture a complete protocol for the task being performed, so that the user (or the user's supervisor) can reconstruct the exact decision path in case an error is suspected. This will permit retracing of the path, possible discovery of the error, and resumption of troubleshooting in the proper direction.

The most frequent need to access a branching procedure will be at the beginning of a troubleshooting task that is entered because of an equipment failure. The technician should be able to call a trouble-shooting routine directly, by naming the failed system and asking for troubleshooting information. Sometimes the need to troubleshoot is revealed during the course of a routine checkout or preventive maintenance task. The latter presentations within this system should simply send the user directly to the beginning of the appropriate branching (troubleshooting) procedure.

Support Information

This refers to any non-procedural information that the user might need, either in direct support of the task at hand, as a supplement to that information, or for his/her own general interest. If there were tables of specifications such as torque values, they would reside as part of the Reference Information collection. Also included here would be any generic procedures such as safety-wiring, specialty soldering, wire-wrapping, and any instructions on use of special test equipment that the authors elected not to embed in task-specific procedures. Theory-of-operation information would be classified as support information. Finally, this is where an overall index and glossary of nomenclature would reside.

Schedules or Assignment for Preventive Maintenance

Asking the technical information system to present preventive maintenance (PM) schedules for a particular equipment item (probably the one identified during the log-on sequence) should produce the schedules. The user should be able to designate (by selecting items from the schedule) those tasks for which he/she wants to see procedures. The procedures should then be automatically retrieved and presented in the proper order. It should also be possible for the user to simply enter the name of the PM sequence that he/she has been asked to perform, and have the system present all related procedures without going through the intermediate step of presenting the schedules.

History Information on Aircraft, System, or Part

There may be times when a technician simply wants to research the maintenance history of an aircraft, a system, a subsystem, or a particular part without necessarily tying the research activity to a particular task. By specifying the equipment item (if it is different from the one entered when the user logged on) and requesting history, the user should get a menu listing available classes of information and kinds of reports that can be assembled.

Given access to such information, some users might want to perform some simple manipulations of the data, such as cross-tabulations. In such a case, a menu of available options should be provided, through which the user can retrieve a procedure for performing such an operation and seeing the result.

To the extent that troubleshooting is supported by an "expert-system" approach, the system itself may access the maintenance-history database (as described in Section IV), with or without revealing this fact to the user. If, for example, the expert system uses historical failure data to calculate the recommended next test point, the user may not care to know exactly how the system arrived at its recommendation; therefore, the user would not be shown anything except the recommended test point unless he/she asked for a detailed explanation of the system's decision.

Supply Information

The needs for supply information will most often come at the end of a checkout or troubleshooting procedure when a defective component is identified, or at the beginning of a PM or time compliance technical order (TCTO) procedure when new components are scheduled to be installed. In either case, the system should name the needed components for the user and provide direct access to supply information about that part without the user's having to key in the identification.

This class of information contains everything that a maintenance person will need for identifying a hardware item, determining whether the item is authorized at the user's level of maintenance, and generating ordering information for it. Supply information includes the following features.

Illustrations. This feature includes the kinds of line graphics or photographs that are presently found in Illustrated Parts Breakdown (IPB) manuals, but may also include computer graphics that can be manipulated by the user (e.g., by selecting a rotated or exploded view).

Nomenclature. This term refers to the system of names by which equipment items are identified. The system should certainly include the "official" nomenclature scheme used by the logistics support system, but it should also allow the user to access supply information using approximate spellings and a certain amount of common but informal nomenclature.

Stock numbers. The system will also allow access through the National Stock Number (NSN) scheme.

Ordering information. The objective of providing ordering information is to allow the user, with little or no information entered through the keyboard, to create (using menus and prompts or touching the screen) an order that the computer can send directly to the supply system. When users select an option that moves them into an ordering mode, they should be asked to designate (from the illustrations in the IPB section, or by directly entering the nomenclature or NSN) the part that they want to order.

The user should then be given the equivalent of an order form to fill in, which requests that the user supply (via menu selection) all necessary variable information needed for a formal transaction with the supply system. If the system already has (as a result of the user's logging on or as part of a permanent dossier on each authorized user) enough information about the user's organization, the job or work order number, and so on, that information should be inserted in appropriate places by the computer. The user should be requested to review that information for completeness and accuracy in the context of the present supply transaction, but should not have to enter any information that is already in the computer's possession.

Availability information. The user should be able at any time to request availability status information on a part that he/she is considering ordering (or has already ordered). The technician should be able to determine when the part will be available, how many he/she will get if only one is ordered, and so on.

Cannibalization information. There should be a way for the user to initiate a request to cannibalize, to transmit the request to the proper authority, to receive authorization, and to track the process through to completion (which includes getting a new part through supply and having to replace the cannibalized part on the original aircraft).

Graphics

Illustrations

Illustrations such as line drawings will most often appear automatically as parts of procedural frames that support specific tasks. Users may also wish to access individual drawings directly. They should be able to do so by entering nomenclature, part or stock number, or other allowable descriptors, or (if they know where the part is but not what it is called) by stepping their way through a series of locator illustrations to the specific item they want. It should also be possible to get directly to any illustration from a function diagram that represents the object illustrated.

Function Diagrams

Users will most often wish to access function diagrams, such as schematics, during the performance of checkout and troubleshooting procedures. Experienced users may feel at some point in the checkout or troubleshooting process (whether or not they are already in the process of using a linear or branching procedure) that they might be able to identify a faulty component more efficiently on their own if they could see a schematic or logic diagram. It should be possible for the user to go directly from any point in any linear or branching procedure to the part of a schematic diagram that deals with the area of the devices involved (and to return to the procedure when finished examining the diagram). It should also be possible for the user to go directly to the beginning of such a diagram, by providing nomenclature, part or stock numbers, etc., as with illustrations. Finally, it should be possible to go directly from any illustration to the schematic that represents the object illustrated.

Maintenance History Information

This class of information will allow the user to call up information that may be useful in modifying or establishing a troubleshooting strategy that takes advantage of probabilistic information about the subject system's maintenance history. Such information includes the following features.

Entry of specific data. The user will identify some or all of the following items: tail number, LRU/assembly/subassembly number, and serial number of the device currently being worked on. If the log-on process already included this information, the system will ask the user to simply verify it rather than re-entering it.

Selection of history reports. The user will be able to retrieve history on a selected system and specific serial numbers for the entire service history of the subject system, or for a user-specified number of previous months/years. The user will be able to retrieve a summary of reliability information for the selected part over its history. The user will also be able to get a report that subdivides that history by aircraft type if the part is installed in more than one aircraft type.

The user will also be able to retrieve summaries of the history of selected systems in the present aircraft (tail number) and of selected parts in the present aircraft (tail number) and reviews of related symptoms reported on the present aircraft (tail number) and on all other aircraft of this type.

III. DATA RELATIONSHIPS/ORGANIZATION

The basic principle of data organization is that the user must be able to go from viewing one type of information to viewing any other type, and must be able to change the display from one type to another with a minimum number of actions. Thus, the technical information system must be extremely flexible with regard to data organization. The following paragraphs describe one system of organization that facilitates movement from one type of information to another.

First of all, a frame is defined as the screenful of data that the user is able to view at one time.

A frame may be filled with text, graphics, or motion pictures, or any combination of the above.

Text, graphics, and animation data will be stored in separate files, but some inflexible links will be established, such that a specific paragraph of text is always accompanied by a specific graphic, or a specific graphic is always accompanied by a specific text. For example, a text paragraph in a remove/replace procedure may always be presented with a specific locator diagram; or an IPB illustration may always be accompanied by a specific parts list. If two data items are inflexibly linked, it will also be possible for the user to fill the frame with either of the linked items alone. This organization will allow for default presentations of the data, but it also will allow the user to define the presentation of data.

Inflexible links will also be established between notes, cautions, and warnings and the procedural steps to which they apply. If a step has a note, caution, or warning associated with it, the technician should never be able to view the step without also having to view the note, caution, or warning. Also, any step that has a note, caution, or warning associated with it should be distinctively identified in some way (see p. 45).

The data relationships will be such that it will be possible for a frame to contain more than one class of data at one time -- or more than one instance of one class of data. Each such data item will be presented in a window, and the user will be able to scroll (or pan) within a window. The user will be able to alter the aspect ratio and position of any window.

Before importing new data into a frame, the user will be able to indicate that some portion of the present frame is to be retained on the screen. This is done by blocking or boxing the data to be retained. In a similar way, the user will be able to indicate that some portion of the frame is to be copied into the user's personal Notepad. This Notepad will be retrievable at any time.

The data organization will also support "bookmarking," another way of assuring rapid access to selected information. As the user scans through various frames of information, he/she will be able to "bookmark" one or more frames for immediate retrieval at a later time. There will be Priority 1, Priority 2, and Priority 3 bookmarks. Within each priority level, the information will be retrieved in reverse order of entry. In other words, the last item of information to be bookmarked will be the first to be retrieved. After retrieving a bookmarked frame, the user will have the option of cancelling or retaining the bookmark. The Notepad and the bookmarking capability are devices that accommodate the fact that some items of information are very frequently consulted, whereas other items are very rarely used. Paper manuals that have been in the field for some time always have pages that are soiled and dog-eared and pages that are clean. It is important to provide rapid access to frequently used information. Another use of bookmarks is to enable the technician to suspend a task for some period of time and to be able to return quickly to the same frame of technical information. For example, the technician may need to leave the terminal in the middle of a lengthy fixed procedure. The technician will be able to turn off the computer to conserve its batteries and, upon return, will quickly be able to access the same guidance that was displayed when the interruption occurred.

Data will be accessed through hierarchies of menus. Menus for a technical information storage and retrieval system can be designed in a large variety of reasonably good ways. Which way is best is, ultimately, an empirical question. The following list of options might appear in the Main Menu of one such system:

A. Job
B. History
C. Troubleshooting
D. Maintenance Instructions
E. Forms
F. Supply
G. Direct Access Data
H. Theory of Operation
I. Tutor
J. Battle Damage Assessment Data

Further amplification of these options appears in Figure A-1. Each of these Main Menu options will evoke a hierarchy of subordinate menus to define more specifically what the user wants to see. When the desired information is equipment specific, the computer will "make an educated guess" about the information items the user might want to see and will ask him/her to select one or to specify another. When the user selects the "other" option, another hierarchy of menus is invoked, to help in the specification of the desired information item. The hierarchy for these menus will be based on an equipment breakdown or some system of technical information organization such as the Maintenance Integrated Data Access System (MIDAS).

Direct-access data will not be called up in terms of a frame number. Direct-access data will be called up from a menu of information types. For example, the menu might consist of:

A. Schematic or Block Diagram
B. IPB
C. Locator Illustration
D. Readings and Tolerances
E. Specifications
F. Generic Procedures

The user will have the option to go directly from any frame to the Direct-Access Menu (DAM). When the user selects one of the options on the DAM, the computer will list the items of information most likely to be desired (based upon the previous frame shown), and it will ask the user to select one of these or to specify another.

Some data items will be linked to others in an unalterable serial order. For example, a screenful of data within a long remove/replace procedure will be linked to the procedure description that precedes it and the one that follows it. To enable the user to move from one frame to any other frame, the user will always have the following options available while viewing a frame of information:

A. Move one frame - Forward or backward
B. Change level of detail - More detail, less detail
C. Main menu - Return to Main Menu (see above)
D. Previous menu - Return to the last menu viewed
E. Direct-access menu - View direct-access menu (see above)
F. Format - Change the format of the frame or freeze some portion of the frame before importing additional information
G. Notepad - Retrieve and (possibly) add to it
H. Bookmark - Insert bookmark or retrieve bookmarked frames
I. Help - Invoked when the user is unable to retrieve the desired information

IV. DATA ANALYSIS AND PROCESSING REQUIREMENTS

Introduction

This section describes a set of functional characteristics or capabilities of the system that require special or unique data processing or analysis capabilities. These features represent either unique capabilities or more effective ways of implementing the storage and presentation of technical data. (Section V describes the functions that will allow the user to input, retrieve, and manipulate the data.) Each of the following six subsections describes one general functional capability and its component attributes or functions:

1. Fault diagnosis guidance and assistance for the maintainer;

2. Direct data gathering from the target (prime item) system under test;

3. Provision of maintenance history data for technician and system use;

4. Generative graphics implementation;

5. Interfaces with maintenance and logistics management functions; and

6. Embedded training for the maintenance technician.

Each of the subsections begins with a description of the characteristic or capability in terms of intended functions in the context of the maintenance information system. Each description is followed by a discussion of the rationale for including the characteristic in the system. Next, descriptions of the data analysis and/or processing requirements believed to be requisite to implementation are provided. As appropriate, the subsections are concluded with descriptions of user interaction, and with recommendations for implementation of the functional characteristic.

Fault Diagnosis Guidance and Assistance

This processing and analysis feature will be utilized by the technician whose objective is to isolate a particular fault in a system or subsystem that has failure symptoms which are not immediately traceable to a specific malfunction or fault. Guidance and assistance will be provided for all stages of fault isolation. These include: (a) symptom verification/duplication in the troubleshooting milieu, (b) symptom pattern completion, and (c) fault isolation to the lowest-level replaceable or repairable unit for the level of maintenance involved.

Fault diagnosis guidance as a general feature includes a number of incorporated or subordinate processing and analysis routines which may be performed independently (one at a time), under direction from software controlling the overall function of guidance or assistance or as invoked by the technician. These features may also be implemented in an integrated and coordinated manner. Specifically, these are:

1. Provision and presentation of quick-check procedures which utilize a minimum of testing (and test equipment) to rapidly isolate a fault. Many of these procedures will require only "front-panel" tests and checks to be performed by the technician. The system will utilize historical fault and symptom pattern data (for the particular system, systems of like type, subsystems within the prime system, or subsystems of like type) to generate and present suggested quick-check procedures to the technician. Based on the initial symptom pattern, the system will rapidly identify complete symptom patterns which incorporate the initial pattern, identify the checks which must be made to isolate each of the faults, construct an efficient test sequence for the technician, and present the test sequence for performance via the display device(s) of the system. Historical probabilities of occurrence of each of the faults which conform to the initial symptom pattern will be used by the system in constructing and ordering the test sequence. Tests which provide the most information toward symptom pattern completion will be ordered early in the quick-check procedure, under the assumption that all "front-panel" checks are equally resource-consuming. If a quick-check procedure fails to isolate a fault conclusively, the information gained will still be valuable to the extent that it reduces the size of the "ambiguity group."

2. Identification and suggestion of specific tests for later-stage troubleshooting and fault isolation. Utilizing a model of the system or subsystem or component on which troubleshooting is being performed, and historical fault probability data (if applicable and available), the job

aiding system will be able to select and suggest the most information- and cost-efficient tests for further isolating the fault, given what is already known about the symptoms of the malfunction. The technician, of course, need not follow the suggestions provided by the system if he/she believes a different strategy or approach is more effective. A discussion is presented later in this section which deals with the need to provide multiple adaptive strategies to accommodate the "styles" of troubleshooting for various classes of users.

3. Input ambiguity resolution. The process of using the system is viewed as a dialogue between the technician and the system. The technician requests information from the system and inputs data into the system to enable the system's processing functions to act on the data. The system, in return, provides the requested information and utilizes the input data in performing its functions. In many cases, the inputs from the technician will have ambiguous "meaning" to the processing functions. Ambiguity will be resolved by asking the technician to provide more specific inputs. The system must both identify the ambiguity of the input and gracefully elicit more explicit information from the technician. For example, in a fault isolation procedure, a technician's troubleshooting tests may be under guidance by the system, and a test point reading may have been requested to help the system's software reduce the ambiguity group. The system will have suggested the test point and test procedure to the technician (along with a nominal "good" value or range of the measurement) and requested that the measurement be made. The technician's response is "Voltage high." In this case, the specific value of the voltage (and not just that it is higher than the nominal range) is critical to the logic of reducing the ambiguity group. The system would then request that the specific value of measured voltage be input. (Note that a well-designed and thoroughly thought-out user interface minimizes the problem.) If this confuses the technician, or the technician does not understand the necessity for the additional information, an explanation facility (see below) will be available.

4. Comprehensive, adaptive, context-sensitive, on-demand presentation of information. With conventional technical data, a major problem in troubleshooting and fault isolation is the gathering of all information needed at one particular time. A technician may (for example) need to consult an electronic schematic (logical guidance), a locator diagram (to identify where test points are physically located in the equipment), a table (for correct test point parameters), and operating instructions for a piece of test equipment (to actually make a test) in order to perform a single test to further isolate a fault. With conventional technical data, this information may be distributed across several volumes of information. A critical feature of the job-aiding system is to be able to provide information needed for the maintenance or troubleshooting job at hand, the state of progress toward problem resolution, the individual technician performing the job, and his/her style of task performance and information utilization.

Rationale

Troubleshooting and fault isolation potentially constitute the most difficult and resource/time-consuming activity of a maintainer. The difficult aspects of this process are: remembering all the facts that have been learned, interpreting the meanings of those facts, making inferences from patterns of facts (e.g., symptoms), and generating a strategy and tactical approach to gather more information so as to eventually isolate the malfunction and restore the equipment to service. The first activity is to complete the symptom pattern. The main objective of the Fault Diagnosis Guidance and Assistance functions is to support the generation of logical approaches to fault isolation, to help the maintainer remember what has been tested and how the tests came out, and to suggest (based on the known outcomes of tests and the historical pattern of faults and symptom patterns) the next step in fault isolation. Although many maintainers may not require all of the capabilities afforded by these functions, practically all maintainers will find the functions helpful in performing troubleshooting and fault isolation.

Data and Processing Requirements

Since troubleshooting and fault isolation require a model of the target system, a comprehensive component and dependency model of the target equipment system is required. A complete system model is quite difficult to achieve, since all the functional dependencies, energy flows, feedbacks, and other system characteristics must be represented. An alternative might be symptom-cause information at a level of specificity sufficient for the particular job at hand, and algorithms or heuristics for defining or generating the most effective modes of testing to complete symptom patterns (at which time a fault would be considered isolated and a repair/restore action could be specified). For organization-level maintenance, the symptom-cause/algorithm/heuristic model would be relatively manageable in terms of size, assuming a partition of levels of the target system. Initial symptoms are usually sufficient to implicate one or more relatively specific systems or subsystems in a particular malfunction; thus, a representation of symptom-cause and inference rules for the entire system would seldom be necessary. Likewise, a symptom-cause/algorithm/heuristic model for particular LRUs at the intermediate maintenance level would be reasonably tractable, since the problem is bounded by the LRU partition. Both the "quick-check" and more detailed troubleshooting capabilities depend on the existence of some sort of fault isolation symptom-cause model at the very least, along with relationship models which link test or check actions to malfunction symptoms. A problem exists in such models in the case of multiple or induced faults, however, as well as in the case where the model is incomplete and does not permit the technician to fault-isolate to a very low level. In such cases, a technician will have to rely on self-generated troubleshooting strategies supported by the information presented by the system.

Historical fault data will be necessary for the invocation (if algorithmic models are used) or generation (by heuristics) of efficient troubleshooting strategies. Alternately, such information could be used in the development of fault isolation model algorithms or heuristics. However, with newly developed systems no such data would be available, and frequent updates of the fault isolation models would be required. Depending on the class of problem and the level of target system detail, several types of historical information might be necessary. These include: (a) system/subsystem repair/fault history for a tail number or specific system, (b) system/subsystem repair/fault history for an aircraft or system model or type, (c) subsystem repair/fault history for all units of a subsystem or component in the inventory, and (d) subsystem repair/fault history for a particular copy of a subsystem/component.

The primary data processing requirement will be for software to symbolically model the target system's component activity and dependency structure, and to utilize the model to infer the need for additional information to elaborate or to complete symptom patterns. An artificial intelligence/expert-system capability is envisioned which uses both algorithmic and heuristic inference to identify the group of potentially failed components within the target system and to infer the most efficient testing approaches and procedures. The efficacy of this capability depends almost entirely upon either: (a) a dynamic representation (model) of the system, which is used to determine the group of potential faults based on the known symptoms; or (b) a static "pre-modeled" representation of highly comprehensive, complete symptom-cause patterns to which the "model" software can "match" current symptom pattern information. In the first case, a very significant software overhead is required to dynamically model the system. This overhead may not be acceptable. In the second case, a fault-symptom model is utilized, which may be much more tractable from an implementation standpoint, although a thorough front-end analysis of failure modes and symptoms will be necessary to develop the symptom-cause model.

Coupled with the software "models" just discussed will be analysis capabilities to tie the historical failure data for the equipment under investigation to the symptom-cause models for testing and strategy formulation. Assuming that a dynamic update capability for historical failure and repair data is available, these data could interface with either of the two approaches to "modeling" the system for fault isolation described above. With the dynamic model approach, new or updated information would be used to alter the probabilistic failure structure of the model and would thus generate different fault isolation strategies if the repair/failure history of the target equipment changed significantly. Likewise, with the symptom-cause model, changed failure probabilities could alter the relative weights of different probable failure "outcomes" within consistent (incomplete) symptom patterns. This would also alter the suggested strategies by influencing the weight or value of various possible tests to reduce the group of potentially faulty components.

An additional processing capability will be required for identifying the value and cost of gathering additional information to localize a fault. This function will almost certainly have to be an adaptable, dynamic model capable of interpreting the state of progress toward fault isolation, identifying potential tests to gather more information, and selecting the test with the greatest information value per unit resource requirement to gather the information. As human troubleshooters must, this function must be able to analyze the group of possible faults from a functional and relationship standpoint, identify a population of potential tests to reduce the size of the group of possibilities, assess the effort/cost for each test, and determine the optimal course of action based on these data. (Note that human troubleshooters utilize both general knowledge and domain-specific knowledge about the problem at hand or the equipment being worked on, to generate strategies. It is likely that the software to generate suggested strategies will have to do so as well.) If the troubleshooting support software is to be maximally general in application, it will probably have to support a number of generic troubleshooting approaches to suit the "style" of approach of a variety of types of users. These styles are not currently well defined.

Information Utilization and Presentation

All available types of fault isolation support information should be provided by the system. This information includes various forms of graphic representations of the system (locators, block diagrams, schematics, interconnection diagrams, single-function diagrams, wiring diagrams, and assembly illustrations), tabular information (test points and nominal values, repair history data, component substitution information, etc.), and text information consisting of both fault isolation support instructions and general reference information (including instructions on the use of test equipment, theory of operation of the system/subsystem, and instructions for performing common procedures). The technician will have to perform all non-inferential portions of the fault isolation task (e.g., accessing test points, operating test equipment, making front-panel readings, etc.), and may prefer to perform part or all of the inferential portions of the task (generating test strategies and tactics, etc.). The system must fully support the technician's involvement by making available all information that the technician might use if he/she were performing the fault-isolation task unaided by the system's inference capabilities. The availability of the inference and suggestion capabilities must not prevent the technician's performing the entire task or any part of the task, if this approach is desired. The system should also have the capability to "learn" the types of information typically utilized by particular technicians, and to provide that type of information as a default (when that individual performs a task), unless requested to do otherwise.

Requirements for Information Display

Variables that influence the type and options of presentation of troubleshooting and fault location information are:

1. User information utilization preferences and patterns

2. Completeness of symptom patterns and historical or probabilistic data for fault probability estimation

3. User stylistic approach to troubleshooting

4. User level of experience with troubleshooting

The system should be able to present both textual and graphic information to the user:

Textual - Identify the "best next check" as a test point reference; provide the capability to explain the reason why the suggested check is the best one (explanation should include the rationale based on the state of progress of the troubleshooting problem and the information/data/parameters which support the recommendation for the check); supply locator and test-equipment-use information for the suggested test point when the user wishes it; provide nominal test point indications for user reference when requested to do so (also, possibly, what various non-nominal values may mean in terms of the troubleshooting problem at hand).

Graphic - On block diagrams or schematics, highlight the test point or location of the recommended test (this may best be done when a graphic which indicates the ambiguity group is already displayed). Explanation, location of test points data, test equipment use instructions, and nominal values information should be available on user demand, as described above.

Windowing of graphics or additional graphics presented on a separate screen should be possible, so that users who wish to simultaneously consult schematics/locators/block diagrams can do so.

Direct Data Gathering from Target System

As a second functional capability, the system will have the ability to interface directly with Built-In Test (BIT) equipment incorporated in prime target systems, and possibly with General Purpose Test Equipment (GPTE) for performing tests or interrogating the prime target system to determine equipment parameters. The effectiveness and generality of use of this characteristic depends largely on the standardization of data interfaces within target systems; however, with the increasingly common provision of the MIL-STD-1553A database (especially in aircraft systems), this interface capability is becoming feasible.

It is envisioned that the system might interrogate the target system BIT equipment in one of two circumstances. The first is the verification of symptoms reported by the BIT equipment, as the first step in fault isolation. The second is the gathering of data on specific, measurable parameters of the prime target system, to aid in symptom pattern completion, either as directed by the technician or as part of an automated troubleshooting capability. The maintenance information system should be able to selectively query the BIT equipment of the target system to obtain information on a set of parameters or indicators, or a single parameter or indicator.

In order to streamline the processes of data collection for such tasks as adjustment, alignment, and fault isolation, it may also be desirable for the technical information system to interface with common GPTE, such as oscilloscopes, volt-ohm-milliammeters (VOMs), and component testers. With this capability, the technician will not have to mediate the transfer of information from the test equipment displays to the technical information system. This will reduce testing time and will eliminate the possibility of errors in reading test equipment displays and entering data into the technical information system. Ideally the system should also be able to control the GPTE control settings. This would require specially configured GPTE.

Rationale

The combination of the capability to derive logical troubleshooting strategies and the ability to directly collect information from the target system to support tests for troubleshooting has the potential to reduce fault isolation time and errors to a large extent. These capabilities, in effect, would make the technical information system into a highly capable and flexible "external BITE" device for use with many systems.

There is some doubt about the wisdom of removing the control of the test situation from the technician (i.e., having the system do all the work and essentially leaving the technician as a "backup"). This could prove to be very demotivating, and could foster the decay of skills required to perform fault isolation and other difficult maintenance tasks. This would be undesirable, in that there are likely to be situations in which an automated system will be unavailable. Detailed consideration will have to be given to the merits and risks of "usurping" technician functions and responsibilities, although the potential overall performance gains appear large at this point.

Data and Processing Requirements

This feature will interact intimately with the troubleshooting strategy generation software described earlier. When the troubleshooting strategy software identifies the need for a particular piece of information on a system parameter, it might obtain the information directly. The available interfaces with the system under test would be interrogated. For this function to work well, the maintenance information system will need data on the composition (both functional and logical) of the target system, so that the correct parameters required by the test at hand can be identified. One very important functional requirement of this feature will be the capability to compare subtly different parameters in order to determine whether the parameters exhibited by the target system are within limits. For many parameters (DC voltage, amperage, resistance, continuity), this is a very straightforward numerical process. However, for AC voltages, especially complex waveforms, the comparison and judgment process is not as straightforward. In general, humans have been demonstrated to have higher success rates than computers in identifying subtle differences in waveforms; it may, therefore, be more efficient to have the technician perform these sorts of judgments. A tradeoff study, considering all the other processing requirements for the system as well as those for this function, will be required to determine whether and how this function should be implemented.

Display Requirements

Since the function for directly gathering system data is largely transparent to the technician, display requirements are apparently minimal. In view of the man-machine tradeoff variables discussed above, however, it may be necessary to provide continuous status information to keep the technician "in the loop" or "in the problem" so that the effects of the system's automated capabilities on technician morale and motivation are mitigated. This might be accomplished by having the system only suggest next checks, measurements, etc., to the technician and only perform tests or provide guidance upon technician request. This would allow the user to determine how much or how little help is to be provided by the system. Some technicians may prefer to let the system run the problem to the extent to which it is capable; others may want no assistance other than information to support their task performance. There are also many intermediate modes of use which might be acceptable to particular individuals. The exact amount of system autonomy to be supported will have to be determined by an analysis of technician needs and preferences, and by the desirable level of tradeoff between training the technician to be a technician and training the technician to utilize the system's automated capabilities.

Maintenance History Data

A third capability of the system will be to provide information on the maintenance history of the target system and subsystems for which maintenance is to be performed. Maintenance history data will include listings of previous maintenance actions and faults for the target system and its subsystems and components, and statistical summaries or probability indices associated with the repair history and reliability of the target system. This information will be presented to the technician to provide clues as to the probability of particular malfunctions, based on the historical fault record of a system or subsystem. The maintenance history data will also be used by logical fault-isolation guidance software in the maintenance information system, to assist in developing suggested troubleshooting strategies for the technician. Maintenance history data will be presented to the technician in a number of ways, both as raw data and analyses. These will include the following:

1. Time history of faults and/or maintenance actions for the system, subsystem, or component under consideration. A list of maintenance actions performed and faults discovered in the system element under consideration, in reverse chronological order, will be available to provide possible guidance to the technician in terms of recently appearing or recurrent problems which may be likely candidates in fault isolation. Data for both specific copies of the target system and the population of like-type systems will be available. This will be raw data which the technician can use as desired.

2. Reliability summaries (statistical or probabilistic). Summaries will be included for specific, selected components (life history of the component in a particular system model, or over all systems in which the component is installed), for the target system as a whole (both a particular copy of the target system and all like-type systems), and for particular subsystems or elements within the prime item system. Data may include information on failure probabilities associated with given symptom patterns (for components and subsystems) and statistical indices such as experienced mean time between failures (MTBF). Fault isolation and repair times (Mean Time To Repair or similar indices) may also be included, to allow the technician to weigh the costs and payoffs of particular fault isolation approaches.

Rationale

Historical fault and repair data can provide very valuable clues to isolate malfunctions. Inherently unreliable components can be identified from such data, and a fault isolation strategy can be developed to test, early in the troubleshooting task, parts that have high likelihood of failure. Troubleshooting time and effort can frequently be minimized by such a strategy. Presently, data to compose useful maintenance histories are difficult or impossible to access in technical information, and most "fault histories" are based on the experience of individual maintainers (or sometimes on informally collected institutional data). Incorporation of accurate historical fault and failure data can enhance the development of efficient and effective troubleshooting strategies (which can in turn reduce repair times and replacement of nonfaulted components) and result in more effective utilization of the maintenance workforce. The fault isolation guidance software in the system can also make use of statistical reliability and repair data in suggesting appropriate strategies and tests to the technician in the troubleshooting task, as mentioned earlier.

Data and Processing Requirements

Maintenance history data are what is primarily needed to implement this capability. In order to promote use of the data, the data should be fully current and highly reliable, and should be essentially complete for the entire system. (Incomplete data will lead to technician frustration, which may result in disuse of this feature.) The maintenance history database probably should be organized as a complete fault history of the target system, since probabilistic data can be derived from a historical listing by software.

Processing requirements will include functions to develop probabilistic indices from the chronological maintenance history and functions to interface the maintenance history data with the fault isolation guidance functions. Display and formatting software for the data will also be required, to support the user display modes envisioned for these data.

Display Requirements

Maintenance history data should be displayed to the technician in one of three ways, depending on the individual's use of other types of data and on user personal preference. One method of display will be tabular; i.e., listings of chronologically or probabilistically sorted faults or likelihóods of failure, either those that are consistent with current symptom patterns, or technician-specified subsets of the data. The second method will be as enhancements to graphic displays such as schematic diagrams or functional diagrams. Components having high failure probabilities, given the currently known symptom pattern, will be indicated in some fashion on the diagrams (highlighting, color, or other unique indicators), to emphasize the greater likelihood of those components being implicated in the current problem. A third way of expressing reliability could be a likelihood index based on a 1- to 10-point scale, or the probability figure to two decimal places.

Generative Graphics

Yet another capability of the system will be to generate its own graphics for user display from a comprehensive, unitary graphics database. Instead of having many discrete graphic representations of the target system and its components, stored and retrieved separately, the maintenance information system will be able to construct and present graphic representations of the target system or any of its components. The information system will be able to generate graphics that show the complete target, any subsystems or subassemblies, and any parts, from any useful perspective and in a wide variety of levels of detail. A single database will be used as a master source for generating graphic representations of the target. The database will likely be based on techniques of solid (or shape) graphics modeling, which are presently becoming technologically mature.

Specifications for particular representations of the target or its components will guide construction of graphics. These specifications will be developed during front-end analysis and data development. Specific graphic representations will be provided for at least the following types of graphics:

1. Diagrams -- Schematic diagrams, functional block diagrams, single-function diagrams, wiring and interconnection diagrams, assembly diagrams, etc. Diagrams will be supported at all needed levels of detail, from representations of the entire system to diagrams of single components, as required to support technician job performance.

2. Pictorial representations -- Component and test point locator graphics, IPB graphics, assembly/disassembly representations of the system and its components, peel-away representations of the system for battle damage repair, etc. Again, various levels of detail will be supported, both to suit individual technicians' preferences and to provide precisely the level of detail and amount of information appropriate for each situation.

3. Limited animation presentations -- To support the performance of tasks where instructions are more effectively conveyed by this approach. Although such tasks are relatively uncommon, some intricate manipulative tasks such as assembly, disassembly, and rigging are extremely difficult to describe, but can readily be conveyed by a motion picture approach. A non-generated approach, such as videodisc, will be used to provide animation presentations when they are required. This approach will minimize graphic processing needs and will simplify the graphics database.

One way in which graphics will be handled is by pyramiding. In graphics pyramiding, graphics are generated according to a hierarchical structure of graphic displays. With this concept, moving from one level of detail to another results in the generation of a different graphic representation of the system. For example, if a technician were viewing a functional block diagram of a subsystem and wanted to look in more detail at a portion of the subsystem, a new graphic, containing only details of the portion of the subsystem desired by the user would be generated. The usual approach is the development of a single graphic representation of the entire subsystem and sluing a window space across that (virtual) graphic, as is done in many conventional digital graphics systems. Pyramiding offers an opportunity both to reduce the graphics manipulation requirements within the maintenance information system and to implement a user-oriented approach to graphics presentation. Only the portion of the system in which the user has current interest will appear on any graphic displayed by the system. Other, more conventional types of graphics formats are described in Section VI.

Rationale

Two major advantages will accrue from the use of a single graphics database and the generation of graphics from the database according to specifications embedded in the technical information. First, this approach has the potential to reduce greatly the information storage requirements for graphic information. Storage of complete images is not required; only the graphics database and the composition or generation rules need to be stored. Although the graphics models of the system may be fairly large, the generation rules for specific graphics should be relatively short and not very numerous, thus producing a beneficial tradeoff in storage requirements relative to storing vector- or bit-images of each graphic.

Second, the use of constructed or generated graphics supports many dynamic graphics processes that would be very difficult or impossible to implement if discrete images were to be stored. For example, the implementation of peel-away graphics for use in battle damage assessment would require an immense number of discrete graphics if the "individual-picture" approach were used. With a generative database (which incorporates information about the geometric structure of the target system), graphics will be constructed "on the fly" to represent any useful perspective or desired level of penetration into the target system. Another example of the utility of constructed graphics is to have the system automatically produce graphics representing the user's point of view. This principle is well accepted for job aids in general; the use of graphic manipulation on constructed images, based on a unified data model of the system, facilitates the implementation of this principle. The data designer will have to identify the elements of the target system to be included in a graphic, have the graphic constructed, and then adjust the perspective to match the user's view of the equipment during task performance. Level of background detail for context enhancement of a graphic could be selected automatically by the graphics authoring system, based on the system model and general rules for graphics development.

Data and Processing Requirements

Two types of data are required to implement this function. The first is the overall graphics database of the target system. As suggested above, part of this database may be an exemplar of the now-maturing solid-graphics-model concept for graphics representation in digital form. The entire target system must be represented in the graphics database, in order to support generation of all required graphic and pictorial representations of the system. Several types of information, other than structural (solid graphics model) data, will be required in the graphics database. These include basic data for developing graphic diagrams (schematics, functional diagrams, etc.), and rules for transitioning from one level of representational graphics to another (e.g., in pyramiding) and for manipulating displays as discussed in Section V. The second type of data will be the rules for generating individual or generic types of graphics using the various graphics data available. These rules will be developed by an authoring system which post-processes graphics created in the front-end data development process on a Computer-Assisted Design (CAD) system. The authoring system will generate links or tags to other related technical information (e.g., text with which the graphic may be presented) and will develop efficient, compact composition rules for the particular graphic at hand. This information will be included in the graphics database to support generation and presentation of the graphics.

The graphics software will operate on data from the graphics database, manipulate the data according to the generative rules for each particular graphic, and compose the graphic portions of the user display. It is possible that the highest-level graphics software will be responsible for composing all user displays, including the addition of text to graphics (when necessary) or the composition of all-text displays for the user.

Management Interfaces

The maintenance information system must also have the capability to interface with numerous external management systems for a variety of purposes. Systems with which interfaces will be possible include the following:

1. Supply system interface -- parts status information and parts ordering will be possible on-line, using a communications link and appropriate interface software. The technician will be able to query the supply system computers to inquire about the availability of particular parts and to order parts which are required for the particular repair at hand. The system will automatically provide appropriate data and other required information to the supply system computers, once the technician has confirmed the need for particular parts to complete a job. If back-ordering of parts is required for particular jobs, the order and availability status of ordered parts (for jobs for which the technician is responsible) can be accessed at any time, to allow for planning of future work.

2. Job control/maintenance control interface -- communication between the technician and job control/maintenance control will be possible either by means of a communications link and appropriate interface software or via a radio transmission and reception capability. The technician, or the system, will be able to report job status information, request assistance, or respond to queries via this link. The system will have the capability to autonomously update job control/maintenance control computers when interrogated, and to advise the user that the update or response has been made. This capability will make dynamic scheduling and tracking of maintenance work more manageable, and may improve manpower utilization through continuous monitoring of the availability and activities of maintenance personnel.

3. Automatic report and forms generation -- the system will be capable of generating maintenance forms such as AFTO 349s and AFTO 781s[2], or OPNAV 4790-2Ks[3], which are used to record the results of maintenance actions. The technician will supply whatever particulars are needed to complete the form (in addition to information already known to, or generated by, the system), and the actual form will be output, either in hardcopy or electronically (or both) when the system is interfaced with the next-level computer in the system.

4. Scheduling assistance -- the system will be able to send and receive messages from maintenance support computers, to facilitate work scheduling, including the performance of periodic or phased inspections, Quality Assurance/Quality Control inspection of work performed, TCTO compliance, and periodic or preventive maintenance tasks. The scheduling features will be very helpful to work center supervisors and other senior personnel.

5. Automated maintenance action updates -- the system will interface with master maintenance information databases in order to periodically update the maintenance history files for maintained target systems. At a minimum, the maintenance information system will report the results of fault isolation, repair actions, and maintenance job time for all target systems. This information will be generated and communicated to the master database automatically when the terminal is re-connected to the shop master computer or other next-level device from which information is uploaded and downloaded.

The management interface features of the system will be restricted, via a password scheme, to only those individuals authorized to perform each particular function. Few 3-level technicians will have access to any of the management functions except spare parts status inquiry and forms generation and transmission. However, 5-level or 7-level personnel would have access to all the functions, except possibly some of the management interface capabilities, which might be restricted to workcenter supervisors or other designated positions.

Rationale

Present-day interfaces between active, working maintainers and managers of the maintenance process are functionally adequate, but often slow, clumsy, and prone to error in information exchange. The provision of automated or semi-automated communications between the working technicians' data systems and various management entities will improve the timeliness and comprehensiveness of the information transfer process.

---

[2] AFTO 349, Maintenance Data Collection Record.
AFTO 781, Aerospace Vehicle Flight Data Document.
[3] OPNAV 4790-2K, Maintenance Action Form.

Data and Processing Requirements

For most of the management functions, the system will serve as a semi-automated inquiry or response terminal. Therefore, the system must have data reflecting the data structures and interface formats of the computer systems that support the functions which are served by the management interfaces. For maintenance forms reporting, a specific format and protocol for composing and transmitting information to the support system will be required. The information system will also require software or functions to prompt the technician for needed information for forms completion. For parts status querying and ordering, information about the software interface of the supply system computer(s) will be required in order to support effective query and message generation. Similar information will be needed for supporting the interface between the information system and the maintenance control function.

Processing requirements that support the management interface functions will include message generation, transmission, and receipt; interpretation of information transmitted from other systems; and effective display of information. Specific capabilities for managing the on-line data communications which some of the functions will utilize will also have to be incorporated, including device control, encoding and decoding of messages, and information formatting to suit the requirements of other computer systems with which communications are made.

User Interfaces

Many of the management interface functions of the system will be semi-transparent to the technician who uses the system, except when he/she must invoke a function and provide some information to support the performance of the function. Display of information for the management functions may be in the form of dialogue queries or, in the case of information interchange with other computer systems or functional entities, direct display of information from other persons or systems. Invocation of the management interface functions should be consistent with the general approach of using adaptive, context-sensitive menus. However, in the case of restricted access functions, passwords or other access control sequences must be provided for.

Embedded Training for Technicians

Finally, the maintenance information system will have the capability to provide both on-line embedded training (ET) in all aspects of the maintenance tasks and underlying knowledge required for effective job performance, with the exception of hands-on manipulative tasks. Two training modes will be provided. The first will be a Computer-Assisted Instruction (CAI) mode for presentation of knowledge and factual information; the second will be a task simulation (TS) mode. The

CAI mode will have the capability to present practically any information in the maintenance information system database to support the learning of such topics as system and subsystem theory of operation, test equipment utilization and operation, basic theory of technologies (electronics, hydraulics, etc.), and maintenance information system utilization. The TS mode will present troubleshooting and fault isolation practice scenarios. Efficient and effective learning strategies will be used in the presentation of information to technicians/trainees and in the use of troubleshooting scenarios. Instructional management facilities will be provided for testing technicians' mastery of the presented material, for instructional record-keeping, and for sequencing the instruction and remediation. If possible, the training modes will be adaptive to the individual user's needs, current knowledge state, and learning style. However, adaptive CAI systems, referred to as ITS or ICAI, require the use of technologies (such as computer modeling of the students' thought processes and of their interactions with teaching or training methods) whose development will not be completed for several years.

As the complexity of target systems increases, and the use of advanced technologies in those systems proliferates, resident technical training for maintenance specialties becomes less cost-effective. Because of the sheer amount of information that must be mastered to effectively learn the intricacies of a given system (especially aircraft systems and subsystems), and the consequent reduced emphasis on basic understanding of technology in resident training, the "training gap" will continue to widen. While many existing training deficiencies can be mitigated by such means as fault-tolerant design, BITE, ATE, and *effective technical information systems*, there will continue to be a need for genuinely knowledgeable and skilled technicians at all levels of all maintenance specialties. The training gap in maintenance specialties can be spanned by providing effective on-the-job training, utilizing the maintenance information system routinely used by the technician on a daily basis. With the extraordinary capabilities of the system for storing, processing, and presenting data, it is efficient to utilize the same device for instruction.

Data and Processing Requirements

Relatively little additional data will be needed to implement an effective and efficient instructional capability as ET within the maintenance information system. The same database that is utilized for presentation of job-aiding information can be utilized for most of the instructional presentations. Some additional processing functions to present the information in the form of CAI or ICAI lessonware and to develop and present TS-mode scenarios will be required. The functions for presenting CAI- and TS-mode information are not essentially different from the functions of an instructional authoring system, of which many are in current use. It is potentially feasible to adapt existing instructional software for use in the maintenance information system. Also, currently available instructional management software and approaches may prove adaptable to developing an ET management capability for the system. For efficient support of the TS ET capability, the fault isolation software must have the capability to function in parallel with the ET software, to provide models of effective fault isolation with which to compare students' performance and remediate their strategies and approaches to troubleshooting. When it becomes technologically feasible to do so, the system may be converted to an ITS.

Display

It is anticipated that, except for lessonware that concerns actual use of the maintenance information system itself, the presentation of information in the ET CAI mode will differ markedly from the usual type of presentation for job aiding. User information display in this mode will essentially resemble the content in conventional CAI systems. In the TS ET mode, information presentation and utilization schemes may be very similar to those of ordinary job aiding, since patterns of information access and use will likely be parallel to those used in actual troubleshooting. An exception to this will be suggestions to correct student logic or strategy defects, which will rely heavily on the concurrent explanation capability of the troubleshooting and fault isolation guidance software of the system. In an expert-system-like manner (e.g., MYCIN), the system will be able to detect mistakes in student reasoning and logic by comparison with the generated "ideal" logic of the system, explain the mistake to the student by example and reconstruction of the logic, and suggest more efficient approaches. This reliance on a built-in expert system will be the first step in producing an ITS.

V. INTERACTIVE CAPABILITIES

Introduction

As has been alluded to, the user will interact extensively with the maintenance information system. The user will retrieve, manipulate, and input data. These actions will occur within various contexts (for example, retrieval of IPB data). However, these interactive capabilities may be described in a context-free manner. They are so discussed within this section, although elaborations concerning many of these capabilities are presented in other sections of this document.

Data Retrieval

The user will have several methods for data retrieval: interactive adaptive menus, voice interaction, alphanumeric entry, and call-up through interaction with a graphic.

Dynamic, Interactive, Adaptive User Control Interface via Menus

The user will be able to access all information from the system via the selection of options from a dynamically constructed, user-adaptive, context-sensitive menu display. The menu of the currently accessible options for information retrieval or processing will be either always on the screen or immediately available for display. Most menus will not be predefined, except as generalized list structures with links available to be filled in as appropriate. One exception to the dynamic nature of menus will be that at least one menu entry will consistently appear in all menus. This entry will provide access to a top-level menu which can be used for accessing any information available via the system.

As with other features of the system, menus will be adaptive to particular users' patterns of information use and styles of job performance. Before the system has gained experience with a particular user, default menu structures (predefined) will be utilized to present available options. As experience with a user's pattern of accessing and utilizing information in various contexts accumulates, the system will modify particular menu displays to present options that are consistent with the user's historical pattern of requesting information in particular contexts. For example, in performing troubleshooting tasks, a particular user may tend to request that a block diagram of the malfunctioning subsystem or system be displayed, along with a single-function diagram (schematic) with the "suspects" highlighted. This user tends not to request information on the use of tools or test equipment, and only occasionally requests assistance with identifying the "best next test" in troubleshooting. The system would display only menu choices that are consistent with this technician's pattern of information use while troubleshooting. For other types of procedures or tasks, menus would be similarly tailored to user preference. If the user needs to call up rarely consulted information, it will always be possible to branch to a higher-level menu to select the desired information type.

Menu entries will be developed in a manner that is sensitive to the type of maintenance activity in which the user is engaged. For example, if an adjustment procedure is being supported by the system, procedural information on how the adjustment should be performed and the criteria for correct adjustment would normally be presented as procedural text and graphics. Other potential user needs for information might be instructions for operating test equipment to determine the values of the adjustment parameters, information regarding locations of the components to be manipulated to perform the adjustment, and (possibly) theory-of-operation information to assist the maintainer to understand what is being adjusted and its importance in the context of the system. These types of information would be listed as options on the current menu displayed during the procedure. Also, procedural options, such as moving to the next step in the procedural information display, or going back to the last branch point in the procedure, would be displayed. The various categories of information that may be appropriate to particular contexts are suggested in the discussion of information content requirements elsewhere in this paper. The use of menus for user access of information minimizes the need for the user to remember the structure and organization of the data contained in the system. Also, menus provide a flexible and comprehensive means of indicating to the user what information is immediately available for retrieval and display.

Three processing issues will have to be addressed in implementing this feature: menu composition, user adaptation, and context sensitivity. Menu composition and context sensitivity may not be distinct issues, since they are highly interdependent. User adaptation will require implementing a means for recording or summarizing actions taken by the user to access information in various contexts and types of tasks, and using the patterns identified by this process to dynamically regulate the composition of information access menus. Menu composition will require identifying the data available to support the user, given the task context (e.g., procedure, fault isolation); preparing menu entries to reflect the available information and procedural options; and generating the menu text to display the available options. Context sensitivity requires the identification of the task at hand and the information available to support the user at particular points in the task. Database structuring and information identifier "links" will provide a means for implementing context sensitivity.

Voice Interactive Capabilities

The technical information system will have the capability to recognize and generate (synthesize) speech to a limited extent. Speech recognition will be of the personalized, speaker-dependent type, with a limited vocabulary capability of perhaps 200 to 300 words. While a more restricted vocabulary might be acceptable, a vocabulary of this size allows a greater degree of flexibility in user interaction. The voice recognition capability will largely support menu-driven access to information and system control. It is envisioned that a voice interaction scheme will be particularly effective in conjunction with the dynamic, context-dependent menu access structure described above. Technicians will make choices from the menus by speaking a code corresponding to one of the menu alternatives (codes will always be displayed with the alternatives), and then speaking an action word to "enter" the choice (e.g., "SCHEMATIC -- GO" or some such). Access to some information or menus should be provided by "master" or "override" action words, to permit rapid context switching or accessing some types of information more rapidly than with the general menu structure (e.g., a "MASTER MENU" command). The effectiveness of this type of speech-based control of the information system is critically dependent on the extent to which the menu control structure is genuinely adaptive and context-sensitive, and can adapt to user preferences and styles for the use of technical information. More rigid control structures (e.g., predefined, non-adaptable menus) will probably be intrinsically less satisfying to use under most circumstances. Voice input for control should be optional with particular technicians. A keypad or keyboard, or a touch-sensitive input device overlaying the display, should be provided for manual input as an alternative to speech control. Also, a noise-cancelling microphone should be provided for use in noisy environments.

Maintenance technicians frequently need to consult new information while using both hands to perform a task. Voice input relieves the technician from having to have a second individual available to manipulate the system. As argued above, however, a manual input system, in addition to the voice input capability, should be provided to suit individual preferences in interacting with the device.

Speech synthesis (e.g., phonemic synthesis or perhaps text-to-speech) will have more restricted applicability, especially in typically noisy environments such as machinery spaces on ships, or flightlines. Synthetic speech, if provided for at all, should perhaps be restricted to a few special purposes, such as acknowledging voice inputs or providing limited readout of parameters specifically requested by the technician, and such as providing the nominal voltage expected at a single test point. Given the limited voice quality of present-day speech synthesis (many voice synthesis systems either sound harshly mechanical or inflect in very odd ways), no attempt should be made to have the system output large quantities of connected text (e.g., procedural instructions).

Software to implement the voice recognition (and synthesis, if included) function will be required. Much of the seminal development work performed by the Air Force and Navy in speech recognition may be useful in developing an implementation for the maintenance information system.

Alphanumeric Entry

As stated previously, there may be situations in which the user may need to type in the same requests that he/she might make verbally, or to supply alphanumeric data that the computer may require. A full keyboard will probably not be available on the flightline. Two solutions to this problem are most attractive. One is to provide a small, grease-resistant, ruggedized keyboard in which a small number of keys may be used to represent the full set of alphanumeric symbols. This is accomplished by allowing each key to evoke one of several possible symbols, depending upon which shift-key depression has preceded the activation of that key. The other solution is to represent the alphanumeric set on the computer screen and allow the user to select each symbol by touching the screen (or steering a cursor to the desired symbol) and pressing a function key.

Interaction with Graphics

When information can be closely tied to an illustration, such as in the case of information for ordering parts, it will be useful to have that information preceded by a graphic. First, the user would be presented with a graphic relevant to the desired information. The user would indicate to the system the information being requested. The information system would then supply the requested data. For example, if a user wishes to retrieve fault history information concerning a particular component while he/she is currently viewing an isometric drawing of part of the target system, he/she could select the component from the illustration (by touching the screen with a finger or stylus or by steering a cursor) and indicate via a menu the type of data needed. The system would then present all the requisite information.

The maintenance information system will be designed to present data in alternative modes. Accessing from a graphic allows users to recognize the component within a target system without needing to remember the exact nomenclature of the part. The user can select this method of data access if it suits his/her needs at a particular time.

Data Manipulation

There are several ways in which a user might wish to manipulate data appearing on the screen. Each method serves a particular purpose in aiding the user to perform a maintenance task optimally. For example, the user may use the zoom feature to focus on the details of a graphic. For the system as it is envisioned, the following modes of data manipulation are foreseen:

1. select and highlight
2. scroll
3. zoom
4. shrink and expand
5. move
6. iconize
7. overlay
8. tag (bookmark)
9. notepad

Each of these methods of data manipulation will be discussed in turn. They will be defined and described in relation to the context(s) in which they will be used and the system requirements for data manipulation.

Select and Highlight

The user will be able to select a section of information which he/she wishes to indicate as important for the task at hand. The system will then highlight that particular section for the user. The user may use one of several methods for indicating the data of interest: finger, stylus, light pen, cursor, etc. The system would highlight the information in one of two ways: reversal of background and foreground colors, or display in some color other than the foreground or background colors.

A user would employ the select-and-highlight feature when a certain piece of information, either text or graphic, must be set off due to its relevance to the task. For example, when working with a piece of circuitry, a technician may wish to look at a hardware schematic. After accessing the schematic, the technician may decide that only one particular input line and its subsequent outputs are important for the task. The technician would then select the relevant input and output(s), and the lines connecting them would be highlighted. If desired, the user could also highlight sections of text, such as steps that need to be completed before continuing with the task.

In order for the maintenance information system to accommodate the select-and-highlight feature, certain hardware and software must be included. The selection of a method whereby a user can pick out information for highlighting will be dependent on the system hardware (touch-screen, mouse, cursor keys, etc.). The selection of highlighting mode will depend on both hardware and software constraints. A monochrome display will limit highlighting to a reversal of foreground and background, whereas a color display will allow for a tri-color presentation (background, foreground, and highlight).

Scroll

The system will have a method whereby a user can drag a virtual window "across" a display. This function is sometimes called "pan." The user will be able to move the window in any direction unless the display is such that only two directions are all that are necessary (as in the case of single-function diagrams, which scroll only in the vertical direction).

The system will include the scroll facility because a user will often need to look at a diagram that is too large to appear within the frame all at one time. These graphics, if they were printed on paper and laid out flat, would extend both horizontally and vertically beyond the area circumscribed by the size and shape of a computer screen. Thus, the user needs a way to access the continuation of the graphic. Pyramiding of graphics will alleviate the problem of large graphics, but there will always be some information that will not fit completely within one frame. Scrolling permits the user to remain aware of the continuity between elements in a display.

The necessary system software for scrolling already exists. Thus, the implementation of this function presents no difficulty.

The combined use of scrolling with the select-and-highlight capability offers a great deal of utility to the technician. For example, if a signal path needs to be traced through a large, complex schematic diagram, the technician could highlight the signal path on the portion being displayed, and then scroll the display in the direction of the signal path until the highlighted path is barely visible at one edge. Then he/she could extend the highlighting to the other (visible) end of the signal path. By repeating this process, the technician could extend the highlighting from one extreme of the underlying schematic to the other. When this process is too laborious, the technician will have another option. After placing a cursor on a signal path, the technician could request that the downstream (or upstream) continuation of that signal path be highlighted throughout the schematic. If the execution of this command results in highlighting more than was intended, the technician will have the option to remove any segment of the highlighting by placing the cursor on some branch of the signal path and requesting that the highlighting be removed from the downstream or upstream continuation of the signal. Of course, it will be important to have a clear indication of signal flow direction throughout the diagram.

Shrink and Expand

The user will be able to shrink and expand the display of data. When presented with data in its default format, the user will be able to specify the aspect ratio of a virtual frame around the data and have the data placed within a frame of that designated size. Several problems need to be solved before this function can be implemented. Neither graphics nor text should be distorted when moved to a frame with a different aspect ratio. There are also legibility limits beyond which a frame must not be shrunk. Some of the constraints and problems with the implementation of this function are discussed in Section VI.

The option for shrinking and expanding a display will allow a user to have several pieces of data on the screen simultaneously, thus making it possible to refer to multiple information sources.

Move

Along with the capability for shrinking and expanding the area in which data are presented, a user will be able to move data to any location upon the screen, thus enabling the user to arrange information in a preferred format.

Iconize

The user will be able to select from a menu of symbols a small icon to represent any piece of information, such as a graphic, for easy access at some later time. The user would then be able to place the icon anywhere on the screen. To re-access the data, the user would indicate the correct icon (possibly by moving a cursor to the appropriate icon), and confirm.

The utility of the iconize function is that the user will be able to store several data items concurrently while looking at other information in such a way that data stored as icons will be easily retrievable when needed.

Overlay

The user will be able to overlay windows containing data, in a method that is analogous to laying or stacking pieces of paper on top of each other. In the default mode, only the title or header of the bottom "piece of paper" would be visible. These overlaid windows could be stacked such that information of interest would be adjacent. The technician could then make comparisons of, say, two full-sized graphics representing the same target system.

Notepad

The Notepad function, as previously described, will allow the user to make notes concerning the task at hand and save them for later access. The user will be able to call up a window which will overlay the existing window (or copy the existing window), enter his/her notes, and either manipulate said window with any of the previously described functions (shrink, enlarge, move, and/or iconize) or save the contained notes to be re-accessed at some later time.

The Notepad will allow a user to maintain the continuity of a task when required to leave the task for any length of time. This utility will also be useful for complex situations in which there is a lot of information to be remembered and/or organized. The Notepad will serve as a memory aid for the technician in this latter situation. At the end of each shift, the technician's Notepad will be saved in a computer at the workcenter. Before the technician's next shift, the Notepad will be transferred to the computer he/she will be issued (as part of the "customization" process).

Tag (Bookmarking)

As mentioned in Section III, the user will be able to tag any full screen of data for later retrieval, in a way analogous to using a bookmark. If, at some later date, the user wishes to re-access those data, it will simply be a matter of entering the bookmark number.

The bookmarking function will be useful to a technician who wants access to data that were previously selected and arranged in a preferred format. At any point, the user should be able to retrieve a listing of the information on the bookmarked screens, so he/she knows what has been marked for later retrieval. Bookmarking does not supplement direct access. It is merely another means for rapid access to frequently used technical information.

Data Input

The methods for data input have been discussed within the context of other man-machine interactions. However, the types of input and their modes of input should be summarized.

Types of Input

There are two primary types of input to the system: alphanumeric and binary confirmatory. Alphanumeric input will be utilized whenever the user is required to supply part names, part numbers, a menu selection, readings made, etc. The user will supply binary confirmation input, by making a yes/no decision or a "that one" choice with a cursor or finger, or by making a "that one" choice and pressing a function key.

Modes of Input

There are several ways in which the user will input data. Alphanumeric data will be input vocally or through a keyboard. Binary data may be entered via finger pressure on a touch screen, cursor position with confirmation, or through any method appropriate to alphanumeric data.

VI. DISPLAY FORMATS

In the design of general display formats for the system, there are several considerations: the content of the data to be displayed, the type of data (text, graphics, or both), the screen shape upon which the data will be displayed, and the availability of display options such as color. These factors have been considered in conceptualizing the formats suggested in this section. However, final decisions on format should not be made until various alternative formats have been experimentally evaluated.

Data Content

Data content dictates the types of text and graphics formats that are chosen. For example, a presentation of the tools available for a task obviously should be structured as a list. The data content also affects the organization of data within a frame (for example, information that identifies the presented frame should be placed at the top of the frame). Many of the constraints on display formats related to data content have been discussed in previous sections.

Text Formats

A limited set of text format types is likely to be used for the presentation of information on the system. These include: paragraphs, warnings, cautions, notes, lists, procedural step formats, queries, tables, titles, labels, headers, and footers.

Paragraphs

Paragraphs will be used primarily to present theory-of-operation and tutorial information. Paragraphs should meet the following constraints:

1. A paragraph should contain one main idea, with some elaborations upon that point. The user must not be overloaded with too much information within one paragraph.

2. The information contained within a paragraph should be authored at a level appropriate to the reading capabilities and technical background of the users.

3. Information in paragraph format can be presented across frames, as long as the user is aware that there are multiple frames and as long as no paragraph is continued on the next frame. If a paragraph were to be continued across frames, the user might, in the process of advancing to the next frame, lose the concept presented within the paragraph.

4. A line should be skipped between paragraphs, to enhance the readability of the material.

5. A series of paragraphs presenting information on a single topic should be titled at the beginning of the sequence. If the information is presented on multiple frames, each frame should contain an abbreviated title within the header information. Such titling will help users to keep track of their position within the data.

6. A series of paragraphs presenting information on a single topic and requiring multiple frames for the presentation of the information should include (at the top of each frame) header information which indicates the number of frames in the sequence and the number of the current frame within the sequence.

An example of information presented in a paragraph format is shown in Figure A-2.

The author of such information should also take into consideration that the mode of presentation (i.e., the computer screen) will radically constrain the amount of information that may be presented to the user at one time. Thus, the user may have to read several frames of material in order to acquire all the information needed on a topic. With the presentation of multiple frames comes the problem of information loss as the user goes from one frame to the next. The computer user will have less contextual material to help integrate several concepts into a total understanding of the material. Thus, paragraphs need to contain references to graphics or to other frames to help the user integrate the material across frames.

Recent work by Kieras (1985) on the simplification of technical writing suggests some other constraints that could guide the authoring of this type of information: continuity of use of references across paragraphs, amount of information presented within a single sentence and a single paragraph, and organization of material presented.

Warnings, Notes, and Cautions

Warnings, notes, and cautions are data types which, although paragraphs, require special formatting considerations. Each type requires a way of calling attention to itself in order to ensure that a user will read the information. There are several ways to draw attention to the information:

1. Using bold headings to announce that the information following is either a caution, note, or warning.

2. Using reverse highlighting of the information.

3. Using color; for example, red for warnings, yellow for cautions, and orange for notes.

4. Using different types of boundary lines around the heading or the information itself.

5. Using a tone or bell when this information is being presented on the screen.

The concurrent use of several of these devices will increase the likelihood that the user will attend to the information. When screen space permits, the note, caution, or warning and the step to which it applies should occupy the same frame. No other steps should occupy this frame. When this is not possible, attention-getting techniques such as those above will be used. Examples of appropriate formats for cautions, notes, and warnings are presented in Figures A-3 through A-5.

Lists

Several types of data should be presented in a standardized list format. These data types include input conditions, schedules and assignments for corrective or preventive maintenance, symptom lists, and packaging information. There should also be special list formats for the overall system index and glossary of nomenclature, and tables of contents (menus).

List items can be single items, labeled sentences, or labeled paragraphs depending on the data being presented. For example, each entry in a table of contents (or menu) is a separate item. On the other hand, a glossary will contain nomenclature and associated definitions, with each name and definition pair becoming an item. Thus, an item may require more than one line for presentation.

The list format for the presentation of information on a computer should conform to the following rules:

1. A list should have a title identifying its contents. If the list appears on multiple sequential frames, an abbreviated title should appear in the header information. The reason for this titling is the same as for paragraph presentations.

2. The items in a list should be sequentially numbered or lettered. This is especially important in the case of lists used as menus for the selection of data to be presented. The user will then be able to enter his/her choice with single key or by touching a spot on the screen.

3. A line should be skipped between items. This will improve the readability of the list.

4. The item numbers (or letters) should be vertically aligned in the left margin, unless the items are no longer than five words each. In the latter case, the list may be centered in the frame. This will help the user to scan the list and find the needed information.

5. The number of items presented in a frame at one time should be limited to a maximum of eight, with six items a preferable number. If there are more than eight items to be included on the list, or if there are fewer than eight items but not all of the items will fit in the frame at once, sequential multiple frames (or scrolling down the list) may be used.

6. As in the case with paragraphs, if multiple frames are used in presenting a list, each frame should have an indicator that informs the user that there are multiple frames and identifies the frame being viewed as to its place in the sequence.

7. If multiple frames are used to present a list, the last item in each frame should be complete and not continued on the next frame. Thus, the user will have all the information concerning that item available at one time.

An example of information presented in a list format is shown in Figure A-6.

The glossary and index should conform to the rules for standard lists with the following exception:

The list items should not be sequentially numbered or lettered. Instead, the items should be arranged alphabetically. This standard format for indexes and glossaries in a paper-based medium is also appropriate within the context of an electronic medium. Keeping some formats similar to those used in paper-based systems may help users to adapt more quickly to the computerized system.

An example of a glossary is shown in Figure A-7; an example of an index is shown in Figure A-8.

Menus

Although menus are lists (of options), they have properties that require special consideration. With a paper-based information system, the user selects a data choice from the list that makes up the Table of Contents and uses the associated page number to access the information. In a computer-based system, the user selects a data preference from a menu list and accesses it by using the associated number or letter. Thus, the roles of the two data types are comparable within the two systems.

The presentation of items in a menu should conform to the previously stated rules for lists. In the case of full-screen menus, menu items usually will be centered within the frame in a non-abbreviated form unless the user defines another location for the text window presenting the menu. These windows may exist in several forms, definable by the user. The user will have control over the placement on the screen, the size, and the internal format of the menu display. The user will also be able to determine whether the menu will overlap currently presented data or appear within a frame empty of other data.

Some menus will appear only when the user requests them. Other menus will accompany particular data types, to show the user what options are available from the frame being viewed. If the menu is of the latter "continuous-prompt" type, certain constraints must be followed:

1. A border must separate the menu window from the rest of the screen. This will aid in the differentiation of the menu from the rest of the display.

2. Suggested alternatives for default menu placement are: (a) the four corners of the display (square or rectangular menu format), (b) top and bottom "strips" of the display (horizontal format), or (c) the right or left side of the display (vertical format).

3. An option to suppress the menu display might also be considered, where full-display graphics may be presented; if this alternative is available, a master command to restore the menu window should be provided, so that the user will not have to remember or guess at the available alternatives.

4. The user should have the option of displaying the full text of menu choices or having meaningful abbreviations or mnemonics of the choices displayed.

5. Menu manipulations should be independent of the alternatives presented in the menu (e.g., "Move Menu" or "Size Menu" commands should always be available).

Full-screen menus require other important considerations:

1. If the default size for a menu window is full screen, when a user redefines that menu window to be less than the full screen, the text occurring in that window must also be redefined or else some of the text may be omitted from the window. If the change results in a deformation of the text characters greater than 25% of the standard 4:3 ratio (height: width) in either or both directions, then the system will have to either reorganize the presentation of the text or change the font size such that the text both optimally uses the defined space and maintains clarity, and so that the information content in the window remains constant.

2. If desired, instead of completely removing a menu after use, the user should be able to convert to an icon which has mnemonic value (such as a miniature restaurant menu) and move this icon to any selected location on the screen. The user can then re-access the menu by some method, such as moving a cursor to the icon or by typing a special character, etc. The menu would be re-presented in its last user-defined location and size.

An example of a single-frame menu is presented in Figure A-9. In Figure A-10, a multiple-frame menu is shown.

Procedures

In general, tasks that are presented as procedural steps can be divided into maintenance and troubleshooting tasks. For maintenance tasks, two types of formats seem the most useful for the presentation of procedural steps and may be used as the default formats. These are: a checklist format and a job-guide-type format. The checklist format may be useful for presenting information to expert technicians who have no need to refer to many pictorial aids as they perform a task. The second format, the job guide type, is derived from the fully proceduralized job guide format used for the paper-based presentation of technical information. The modified job guide format, with illustrations fully integrated with the text, may provide less expert users the guidance they need to perform the task at hand. These are suggested only as default formats, since it is envisioned that the system will have the flexibility to allow a user to integrate graphics and text as desired.

Checklist. A checklist consists of a list of procedural steps and input conditions, as well as notes, cautions, and warnings. The system will determine the order of presentation of information, guided by input from the user. Many of the writing requirements for text which have been developed for the production of paper-based job guides are applicable to both the checklist and the job guide format. These rules need very little modification, other than to take into consideration some points that result from the computerized presentation of this material:

1. Each step should indicate one action.

2. The steps should be sequentially numbered or lettered. Thus, the user will always know where he/she is in the procedure.

3. A line should be skipped between steps. This will improve the readability of the information on the screen.

4. If there is a branching step within a sequence, it should occur as the last step within a frame.

5. A step that requires involvement in the task in such a way that the user is unable to operate the computer should not be presented as the last step in a frame. This will ensure that the technician will be able to advance to the next step. If the system includes voice input capabilities, this consideration may not be necessary.

6. The first frame of a checklist will contain a title naming the task and the target equipment. All subsequent frames should include an abbreviated title within the header information, so that the technician will know that he/she is looking at the appropriate information.

It may be appropriate to present a few illustrations or diagrams with a checklist. Illustrations can apply to either one step or to several. In the checklist format, illustrations or diagrams should appear either to the right of the steps contained within the frame or below them, depending on the shape of the display screen. In the case of a horizontal rectangular screen, the illustration should appear to the right of the text unless there is sufficient room below the text. If the screen is a vertical rectangular one, the illustration will best fit below the text. When the checklist is presented on a square screen, the illustration should be placed in the location (either to the right of the text or below it) that allows for the greatest clarity of detail. Examples of the integration of illustrations/diagrams and text for the checklist format for different screen shapes are shown in Figures A-11 through A-13.

As previously mentioned, the user will have the option to redefine the placement of the illustration and/or its size. If there are illustrations that pertain to the steps being presented but are not part of the default presentation for the checklist (for example, illustrations produced for parallel steps presented in job guide format), the user will be able to access these other illustrations without having to access the specific procedures to which they pertain. While within a frame presenting procedures, the user will be able to call up any other type of information desired and display it concurrently with the steps and illustration(s) currently being viewed.

Job Guide. In job guide format, steps and illustrations are fully integrated; i.e., more steps have pictorial representations of information in the job guide format than in checklist format. The steps themselves should follow the same constraints as the steps in checklist format, but the integration of illustrations will affect the overall organization of the information within the frame. In job guide format, illustrations that pertain to one step only should appear either directly below that step for a vertical rectangular screen or to the right of the step in the case of a horizontal rectangular screen. If the screen is square, either position is appropriate, as long as clarity is maintained. If an illustration applies to several steps, its placement should be the same as for checklist formatting. Figures A-14 through A-16 present procedural steps in job guide format for different screen shapes.

As in the case of checklists, procedures presented in job guide format should offer the user the option of changing the size and location of the presented illustrations. The user should also have access to all other data types and should be able to view them within the same frame as the procedural steps being followed.

Tables

Tables are used to present tolerance values and similar numeric data. Historical data about the target system or subsystem may be presented as a table of text. Tables should be constructed such that the information needed can be most easily extracted. Thus, tables should conform to some simple rules that aid the process by which the pertinent data are retrieved:

1. A table should consist of labeled rows and columns. In most cases, the entries in rows and columns will be numeric values.

2. Entries should be vertically aligned under column labels and horizontally aligned with their row labels.

3. Entries consisting of numeric values should all be expressed to the same number of decimal places.

4. Entries consisting of numeric values should be vertically aligned either according to their final digit or according to the decimal point.

5. Items to be compared should appear in adjacent columns or in adjacent rows. Furthermore, it is easier to compare data in adjacent rows than in adjacent columns.

6. A blank line should be skipped between rows of values.

7. As a default placement, a table should be centered in the frame.

Figure A-17 shows an example of a table with numeric values; Figure A-18 presents a table of text.

Queries

Queries ask the user to input some type of information. The information requested may include equipment readings, yes/no responses, management interfacing data, information to aid the system in understanding previous ambiguous input, or choices for data presentation. As such, they are embedded within other text formats, such as job guides or checklists. The system often uses the technician's responses to such queries to select the next piece of information to be presented. Thus, answers to queries are important in directing the flow of information to the user. Queries should be clearly worded, leaving no doubt as to the type of information requested. Each query should be expressed as a question and terminated by a question mark, unless the query is an explicit direction to input information (e.g., "enter your choice from the menu"). In the latter case, the query may be terminated with a period. A query should be placed such that it is the last item read by the user before reading the footer information. Some indicator should be placed after the query to indicate that input is expected. An example of an indicator is a box bounding a space large enough for the requested input and containing a cursor. The system will then echo, within the box, the user's input.

Titles

Titles play a very important role in the display of information. They keep the user from forgetting the nature of the task at hand, the information currently in use, and how that information relates to other data that might be accessed. Thus, every frame which presents a new type of material should contain a title that is descriptive of the type of information that is to follow (e.g., Theory of Operation or Procedures for Disassembly) and the component of the system to which it pertains (e.g., for an oscilloscope). The main title should appear centered directly below the header information. Each major word in the title should be capitalized. A line should be skipped between the title and the information to which it pertains. If the information to which the title applies is presented in multiple frames, then each frame should have an abbreviated title placed with the header information.

Header and Footer Information

Most frames will contain header and footer information. Header information should include a frame number if data are to be directly accessed in this way. Other header information should include an abbreviated title of the presented information and, if the frame is one of a sequence, a statement that indicates how many frames are within the sequence and where the presented information fits within the sequence. Header information, as the name suggests, should appear at the top of a frame.

Footer information appears at the bottom of a frame. Footer information could consist of the options that a user has when looking at data.

Other Text Formats

There are other text formats to be used that do not pertain to the technical data but are related to the system itself, such as the format of the opening frame when the system is first turned on or the format for the command requesting users to enter their password.

Graphics

The system will present two major types of graphics: illustrations and diagrams. The category of illustrations includes all pictorial representations that accompany procedures, and the exploded views presented in the IPB. In a paper-based system, these representations are line drawings, usually in isometric perspective, portraying the technician's view of the component. In a computer environment, these illustrations can be much more than line drawings. For example, when an illustration is presented, the user should have the option of seeing it from angles other than the one initially presented. So, an illustration could either appear to rotate in real time or appear immediately in the desired new position. Animation also may be incorporated when verbal descriptions of the required action are difficult to understand without such a visual aid.

In general, the authoring rules for individual illustrations will be very similar to those for paper-based presentations. However, there will be certain constraints placed on their development due to the presentation system. These constraints include overall size of the illustration and placement in relation to the text. Illustrations must be sufficiently small to fit the available space, while still maintaining clarity. Thus, an illustration must contain only as much detail as is necessary to support task performance. In order to meet this criterion and to produce displays which allow the technician to easily locate the part upon which he/she is to perform a task, it is often useful to have a locator illustration in which the technician can zoom to the necessary part or step down through a series of more and more detailed illustrations, to the level appropriate to the task.

Limited animation may be provided by sequencing through a set of graphics, with each successive graphic depicting the progressive stages of task completion. The pacing of the frames will be under user control. For a small number of tasks, there will be no performance aid more useful than a video disc presentation, as described on page 29. Because of its limited use, the video disc unit will be a "strap-on" module for the portable computer.

Illustrated Parts Breakdown (IPB)

Another innovation to which the computerization of illustrations can lead is interactive IPBs. The IPB could consist of an unexploded illustration of a component, regarding which the user would have several options. The user could use such illustrations for ordering parts or for accessing the repair and troubleshooting histories of a system or component. Since the computer system will be linked to the supply system, no paperwork by the technician will be necessary for ordering parts. All the user will have to do when looking at an IPB illustration is exercise the following options:

1. Explode the illustration.

2. Indicate the needed part (by moving a cursor to the required part, touching the required part on the screen, etc.).

3. Select a Part Ordering option (presented as an item within a menu of options presented with the illustration).

Given this process, the technician will not even need to know the name of the part being ordered, although as a safety measure, the name of the part and its NSN should be displayed directly above the footer information. The supply system should also transmit back to the technician information concerning the availability of the requisitioned item. This information could appear next to the part name and NSN. An example of an exploded IPB and the menu choices which would accompany the illustration appear in Figure A-19.

If a part is needed that is not directly visible in the exploded IPB illustration currently on the screen, the user should be able to move a cursor or touch the screen to indicate the part of the exploded view in which the target item is embedded. The user could then select an option from the presented menu that will replace the current illustration with an exploded illustration (or some other appropriate representation) of the selected part. The second illustration could be utilized for ordering.

In order to use the IPB to access part or system histories, the user would follow the same process as for ordering, except for the final step. Instead, he/she would choose an option from the menu which would present the needed history data.

It may be useful to have an IPB present other views of the component in order to aid user recognition. The same rotational possibilities could exist for IPB illustrations as for illustrations that accompany procedures.

In preparing the IPB illustration, the same guidelines should be followed as in preparing other illustrations, with one extra rule. At all times the user must be kept informed of any additional IPB information that may be available in this mode.

Peel-Away Graphics

Another type of illustration that may be needed is a peel-away graphic. This illustration initially presents the technician with an external view of a target system. The technician then indicates how many "layers" he/she wants to be removed in order to reveal the internal structure. These layers are then "peeled away" to reveal the desired view. A layer is defined by what equipment can be seen at a given time, not by a specific depth from the outer surface. The skin on an aircraft is the first layer. Removing the components revealed by removing the skin constitutes removal of the second layer. The production of frames with peel-away graphics should conform to the following rules:

1. Each illustration used to present a layer should be constructed in the same way as other illustrations.

2. The transformation from one layer to another should occur within the same frame.

3. The header information should indicate the number of layers available for viewing and the number of the currently viewed layer, in addition to all of the other necessary header information.

4. The footer menu presented with the graphic should include the following options:

   (a) return to the next higher layer

   (b) return to the top layer

   (c) go down X number of layers

   (d) go up X number of layers

   (e) return to the procedure or menu being used prior to peel-away

   (f) access to other data types through a direct access mode

(g) access to peel-aways for other systems or other parts of the present system

(h) access to other data types through a menu

Examples of peel-away graphics are presented in Figure A-20.

Diagrams

The second major type of graphic information consists of diagrams. Diagrams that a technician may wish to use may consist of function diagrams, such as schematics, waveform diagrams, logic diagrams, functional block diagrams, and single-function diagrams, or they may be diagrams such as charts and graphs.

Function diagrams will be available to assist the technician in understanding a target system and in troubleshooting a system when the on-line troubleshooting aids do not provide sufficient information. Some technicians may prefer to utilize function diagrams to augment on-line troubleshooting procedures. Some of these diagrams may also be presented with procedures as an automatic supplementation of information.

Function diagrams require considerable care in their preparation. In a paper-based medium, these graphics are often large, multiple-page or fold-out displays. Translating such displays to a small screen requires certain deviations from the usual method for representing this information. For example, a schematic or a logic diagram may be represented as a set of single-function diagrams (SFD). Each SFD traces a separate function portrayed within the schematic or logic diagram in a linear way, with nodes indicating branches that integrate with other functions. The user can select the function that he/she wishes to examine. The first frame of the SFD would contain a title identifying the function it represents, the type of diagram from which it is taken, and the hardware to which it pertains. This type of diagram will require multiple frames or scrolling in order for the technician to see the whole display, but the movement would occur only in the vertical dimension. Thus, the user would be less likely to become disoriented within the display, a possibility that might occur if the diagram required scrolling in both horizontal and vertical directions.

There are three other methods by which function diagrams may be made more amenable to a computer presentation. Each method has its good and bad points and will be discussed in turn.

First, the size and level of detail of large graphics could be reduced so that initially the full graphic would fit on the screen. Then the user would have a function whereby he/she could point to a portion of the graphic and zoom it up so that the detail in the graphic becomes clear. This method has the advantage of giving the user a sense of the overall coherency of the diagram. However, it is important to preserve the legibility of the major functional labels on the overall view, so the user is able to select the correct portion of the diagram to enlarge. Also, there must be some method whereby the user can indicate the portion of the display to be enlarged (this problem also exists in the case of user-defined windows for illustrations and menus). A device such as a mouse might be incorporated into the presentation system to handle this problem. There is also the question of whether or not the user-defined window should overlay or replace the original display. Overlaying the display may produce contextual confusion for the user unless the overlay is produced in such a way that the user can easily differentiate between the two diagrams, while still maintaining a sense of the integration between the two. If the enlarged section of the diagram replaces the original, then the user may have difficulty remembering the context from which the enlarged section was taken. An example of a schematic presented in this type of format is shown in Figure A-21.

A second way in which large diagrams may be presented is to incorporate a "moving window" or scroll into the information presentation system. In this case, the graphic must be constructed so that it is is legible when presented upon the screen. In order to look at the entire diagram, the user would be able to drag a "window" (defined as the space available within the frame which is not used for footer and header information) across the graphic. The major problem with this solution is that it is difficult for the user to maintain a sense of how the section being examined fits into the total diagram. An example of a logic diagram using this type of solution is shown in Figure A-22.

A third way to handle the large diagram problem is, in practice, similar to the zoom solution, but produces a conceptually different outcome. The user is initially presented with a conceptually more abstract version of the diagram that he/she wants to access. For example, a schematic might be represented by a functional block diagram. The technician could then indicate with a cursor (or by touching the screen) the part of the diagram that he/she wishes to access. The requested section would then appear on the screen. A scaled-down version of the original diagram would also appear within the display, above the larger diagram showing the selected section. The reduced abstract diagram would include some type of indicator to designate which section the user is viewing. Within this solution, it still might be necessary to incorporate some type of window or scroll so that the user would be able to view the section of the diagram he/she has selected. However, in this situation, as compared to the zoom or scroll situations, the user would be less likely to become disoriented with regard to the relationship between the diagram being viewed and the whole diagram from which it was taken. A diagram utilizing this type of display is presented in Figure A-23.

The production standards of diagrams differ from those of illustrations only in the type of information contained within the graphic. Diagrams differ from other graphics primarily with regard to complexity and amount of presented information. Diagrams may require many frames or need to be scrolled, whereas illustrations rarely will require more than a single frame or need to be scrolled (IPB illustrations will be the exceptions). The nomenclature which is currently used for labeling diagrams will be sufficient for future purposes. Labeling within diagrams, however, probably should read from left to right, rather than in any other direction (i.e., bottom to top), since it is not reasonable to expect that the technician will manipulate his terminal to read labeling in the same way that he/she would rotate a paper foldout.

Manipulation of Graphics

Shrink and Move. As with menus, the user will have the option of shrinking and moving graphics within the screen. In this way, the user will be able to access several graphics at one time. Graphics initially would be presented in their optimal orientation and size. The user would then be able to manipulate the size and location of the graphic. However, such user freedom in defining the size and shape of graphics could result in large distortional problems. It may be determined that it is better to give the user control over the size and shape of graphic window only, rather than control over both the window and its contents. This last option, unfortunately, would prevent the user from having a sense of the relationships between all of the elements within the graphic after its window size or shape has been altered. The user might then have difficulty in comparing graphics, unless he/she specifies that the window be placed over the section in each graphic that needs to be compared. The technician might also scroll the graphics within their individual windows until he/she finds the sections needed for comparison.

Overlay. The user will also have the option to overlay graphics so that comparisons can be made between them. However, the layers of graphics should have borders that will aid the user in differentiating among them. The ability to overlay and shrink graphics should also be available when a user is within a series of procedural steps. In this way the user can define his/her own method for text and graphic integration as options to the default presentations.

Iconize. Whenever a user has the option to call a graphic, there will be an icon associated with the graphic, representing its type. Thus, when it is anticipated that future access to a graphic may be required in the course of the task, the user should be able to assign the correct icon to the graphic, have the icon take the place of the graphic, move the icon to some unobtrusive part of the screen, and then, when ready to re-access the graphic, place a cursor (or finger) on the icon, and have the graphic re-presented in its last defined size, shape, and location.

Select and Highlight. If the system includes the capability for selecting and highlighting, then certain issues for the display of such path tracing must be addressed:

First of all, the user would need a method for selecting the path to trace. A moveable cursor would be the most useful selector display (and would be used for other display functions, also). The cursor should be large enough to be easily seen against the background of the graphic. The cursor must also be small enough to indicate the beginning and ending points of the path without obscuring any of the surrounding content. (In this situation, the user's finger would probably be a poor instrument to use as a path selector, since the user would have to be very careful not to touch any of the surrounding lines or nodes in order to prevent them from being read as the beginning or end points.) For example, a good cursor might be a solid isosceles triangle with a base of no more than 1/16 inch.

The question of how the selected path is to be traced must also be considered. If color is incorporated into the information system, the tracing could be presented in a color that is easily distinguishable from the diagram and background colors. However, if the system includes only a monochrome display, path tracing should be indicated by a reverse highlight function. The reverse highlight should present the path in the background color, and the area on both sides of the path for the distance of 1/8 inch should be the color of the remainder of the graphic. Any labeling which appears within the section to be traced should also be reverse highlighted.

Charts and Graphs

The innovations in the design of charts and graphs for an advanced computer system will primarily concern the presentation of data representing three or four variables. The representation of three variables may be more easily presented as three spatial dimensions (x, y, and z axes) on a computer screen than on paper because a computer screen gives a user the perception of translucence of the medium. Thus, a user may be able to more easily grasp the patterns in data given a three-dimensional representation. The computer also allows for the introduction of motion for the representation of a fourth variable. Motion would also be useful for representing change over time or for presenting correlational relationships between pairs of variables (for example, if two variables have an inverse relationship, a display could show a bar representing one variable increasing in height while a bar representing the second variable decreases in height). However, the efficacy of a computer presentation of graphical information in the ways suggested above still needs further assessment.

Although the maintenance information system may allow for certain new methods for the presentation of chart and graphic material, most charts and graphs that will be presented are of the types familiar to everyone: bar charts and line graphs. The rules for the paper-based presentation of these displays have been standardized (but not empirically assessed) to a certain extent. However, the rules for the translation of "good" charts and graphs from paper to computer screen have not yet been established. Until further research has been conducted on the cognitive and perceptual factors affecting the way individuals comprehend abstract ideas presented in charts and graphs, it is probably safest to present such material according to the rules established for paper-based media.

VII. CONCLUSION

With the growth in number and complexity of maintainable hardware systems used by the military, paper-based methods for recording and presenting maintenance information to technicians have become uneconomical with regard to updating and managing material. Along with the increase in the volume of documentation needed to represent increasingly complex weapon systems, come problems associated with use of this documentation. Job performance time and likelihood of error increase as a function of the number of separate documents a user must access in performing a task. It is felt that a computerized information system, when developed to reflect its potential, will overcome both the economic and task problems associated with a paper-based system. A computerized information system will have its own problems, however. These will be surmountable, given the rate of advancement in relevant technologies and in knowledge about the interactions between information content, format, presentation, and usage.

Thus, it is foreseen that a computerized system for the presentation of maintenance information will ultimately replace the paper-based systems currently in use. This document was an attempt to put forth an idealized system for the presentation of information for technicians performing maintenance, damage asssesment, and repair. A computer-based system offering the user maximum flexibility as to the availability and formatting of information was described. The ways in which a user can access and manipulate the information were also discussed.

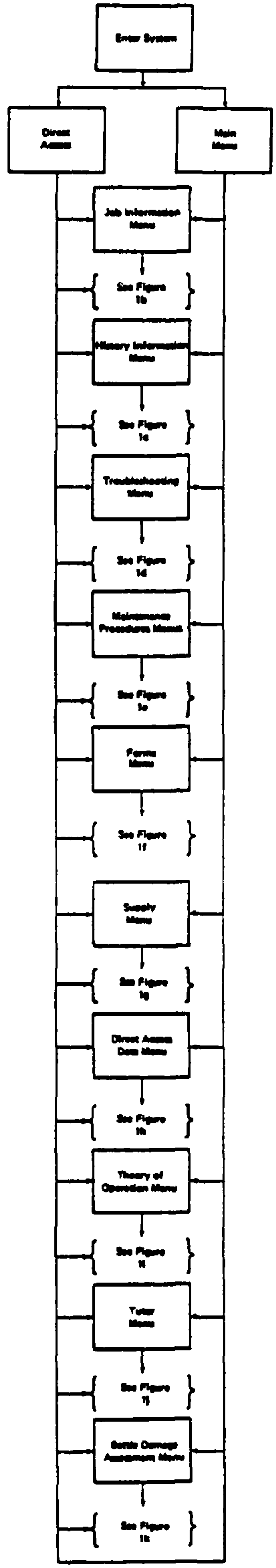
The system described herein does not currently exist, and is several years from full implementation. Some of the technologies suggested as applicable are still in the experimental stages. These technologies include speech synthesis, full ITS capabilities, and flexible modeling of weapon systems for an expert troubleshooting function. On the other hand, many of the components of this advanced system are already available or will be in the very near future. Ongoing projects such as the Computer-Based Maintenance Aids System (CMAS) sponsored by the Air Force Human Resources Laboratory, and the Navy Technical Information Presentation System (NTIPS) are integrating the existing technologies into information presentation systems which are leading to refinements of concepts concerning ways to present technical data. The findings from the existing projects will inform the planning, development, and implementation of the comprehensive systems which, in the future, maintenance technicians will utilize in their day-to-day activities. The systems of the future are within our vision today.

REFERENCES

Kieras, D. (1985). Improving the comprehensibility of a simulated technical manual (Technical Report No. TR-85/ONR-20). Ann Arbor, MI: University of Michigan.

APPENDIX A

FIGURES



<u>Figure A-1a</u>. Overall Data Structure for System.

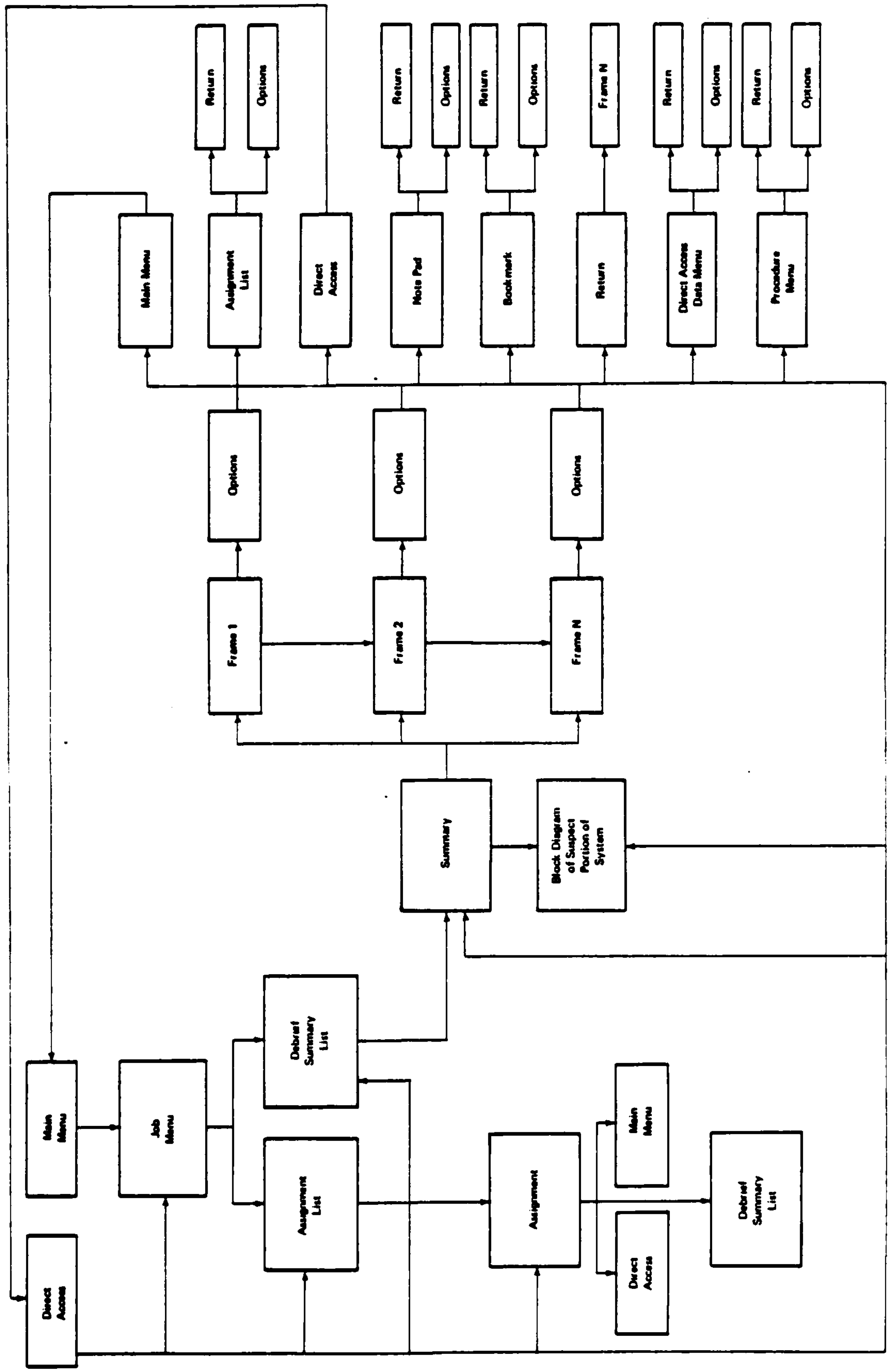


Figure A-1b. Data Structure - Job Information.

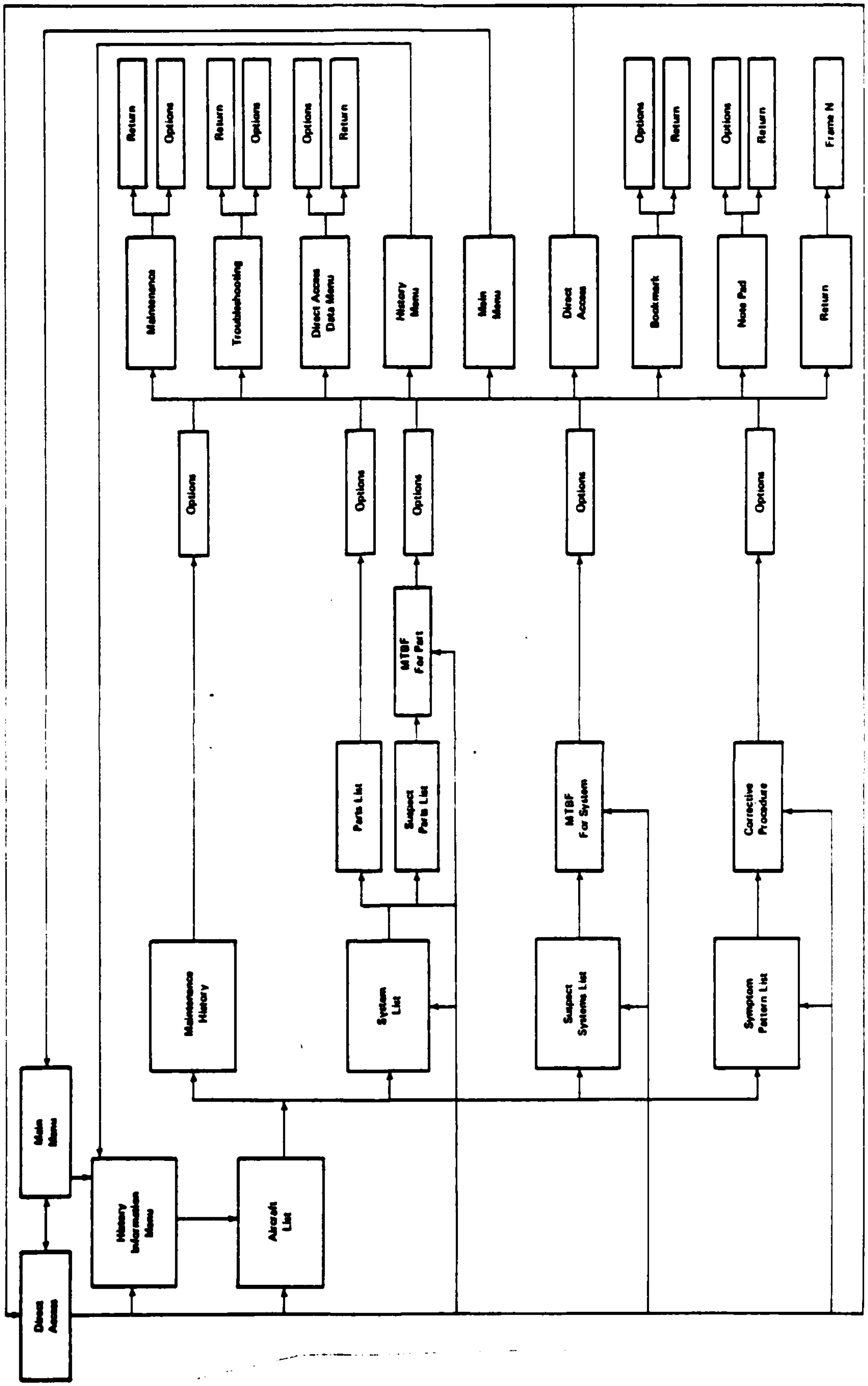


Figure A-1c. Data Structure - History Information.

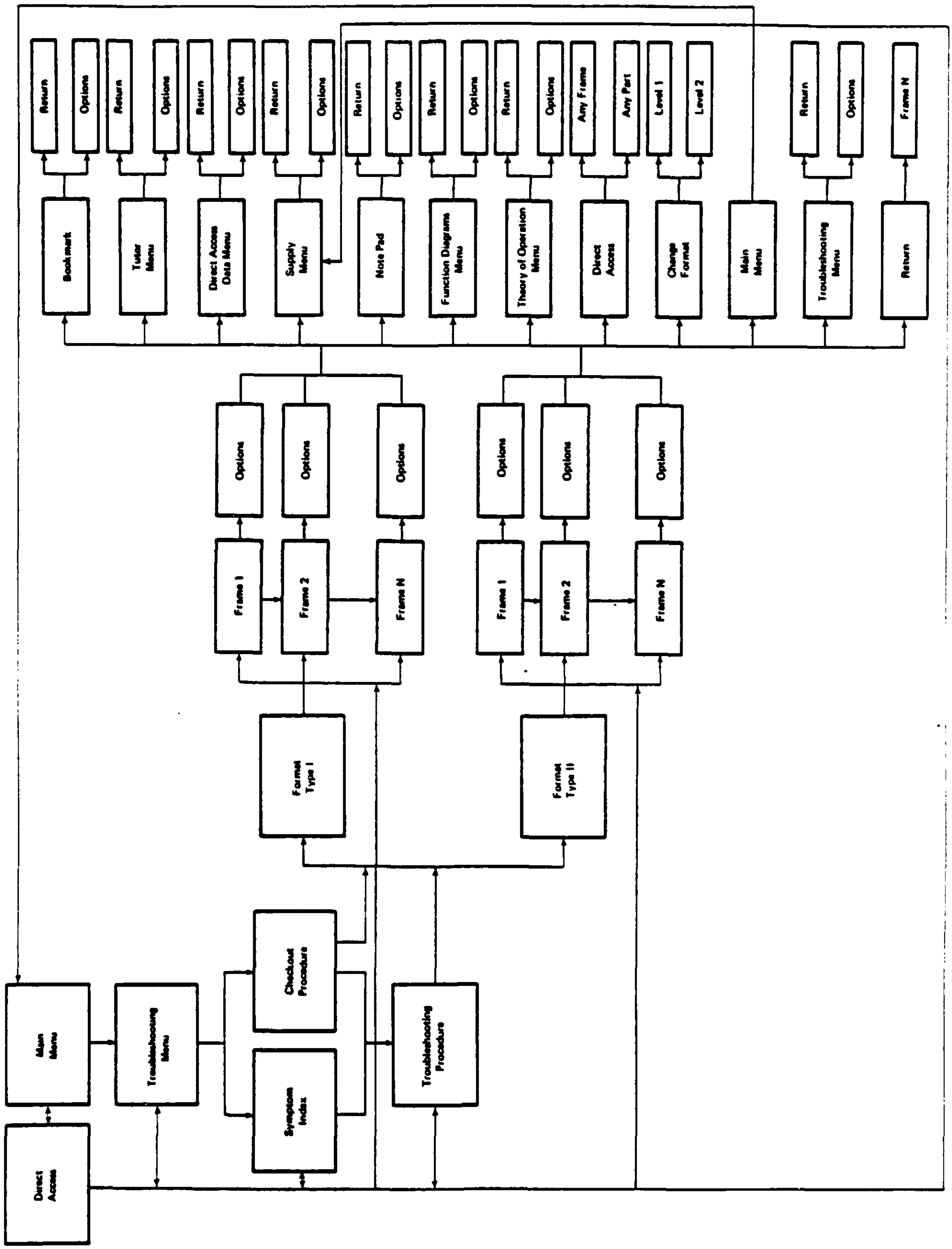


Figure A-1d. Data Structure - Troubleshooting.

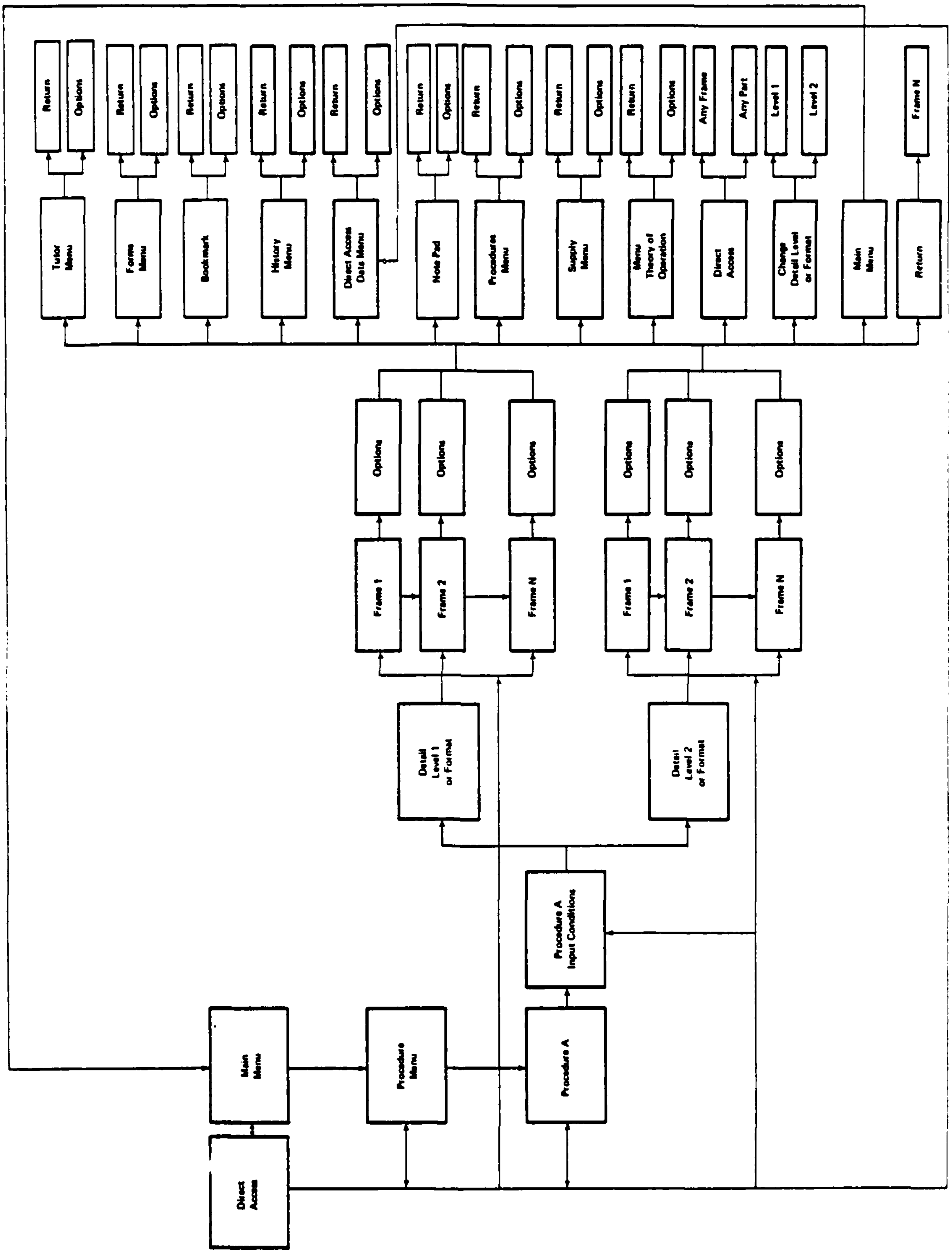


Figure A-1e. Data Structure - Maintenance.

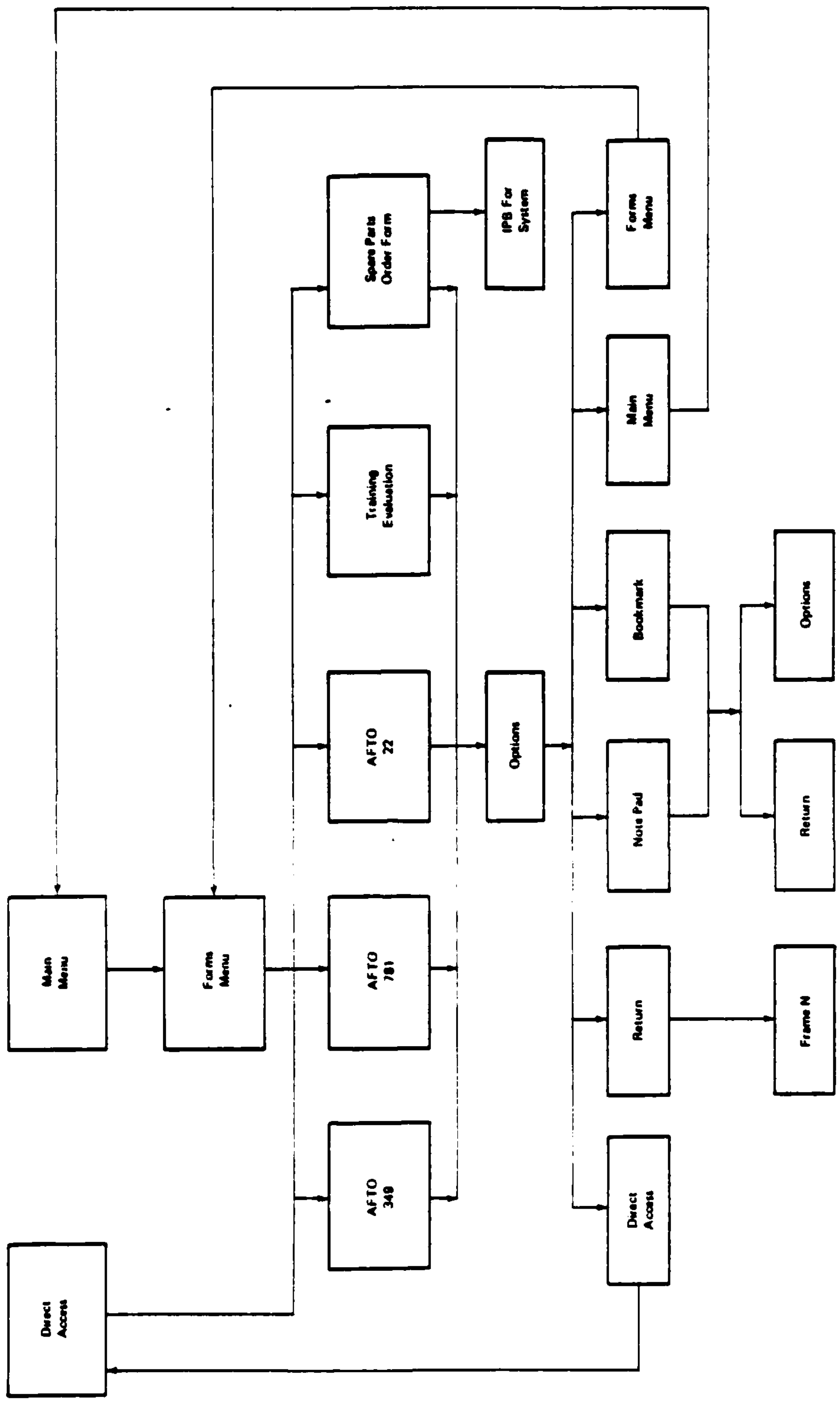


Figure A-1f. Data Structure - Forms.

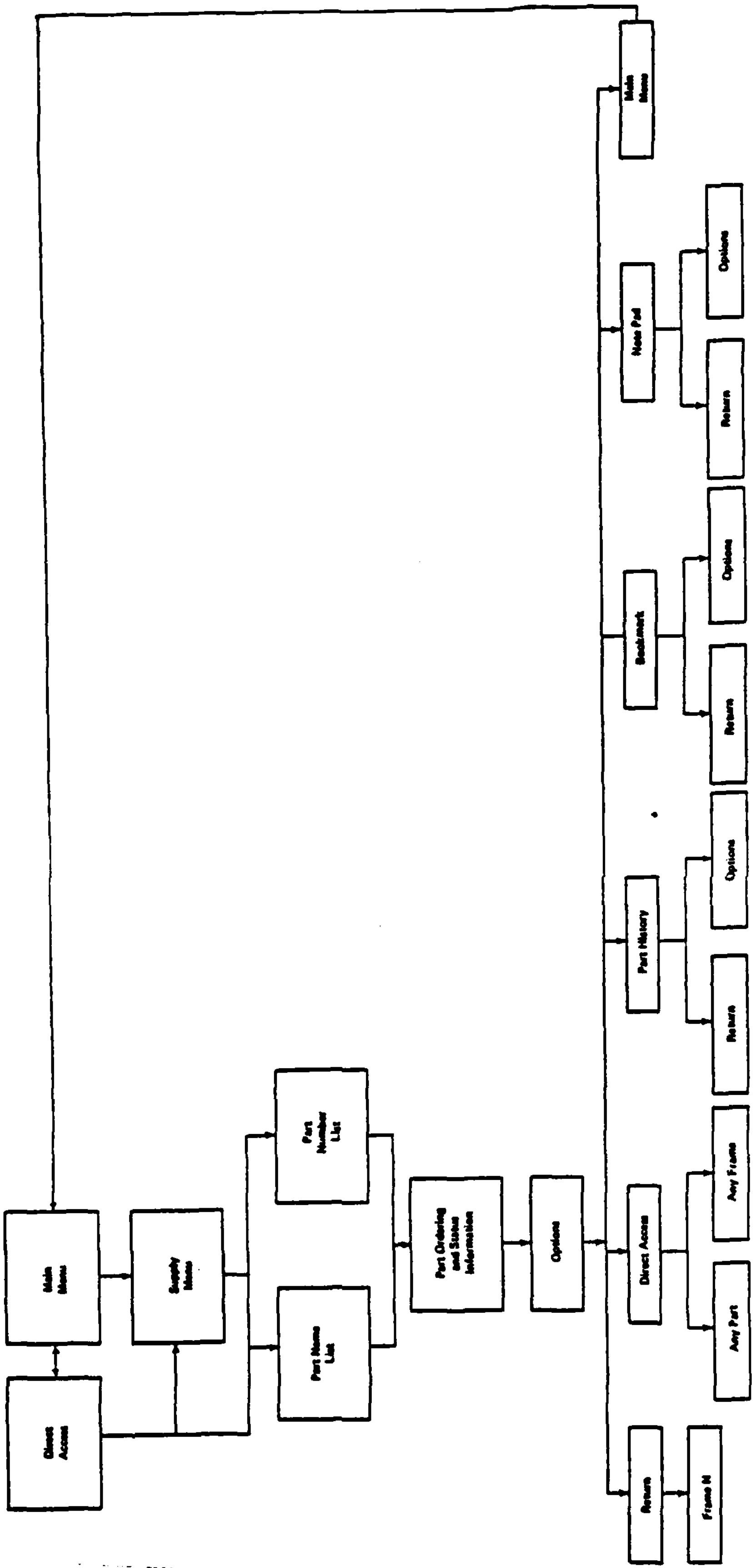


Figure A-1g. Data Structure - Supply Information.

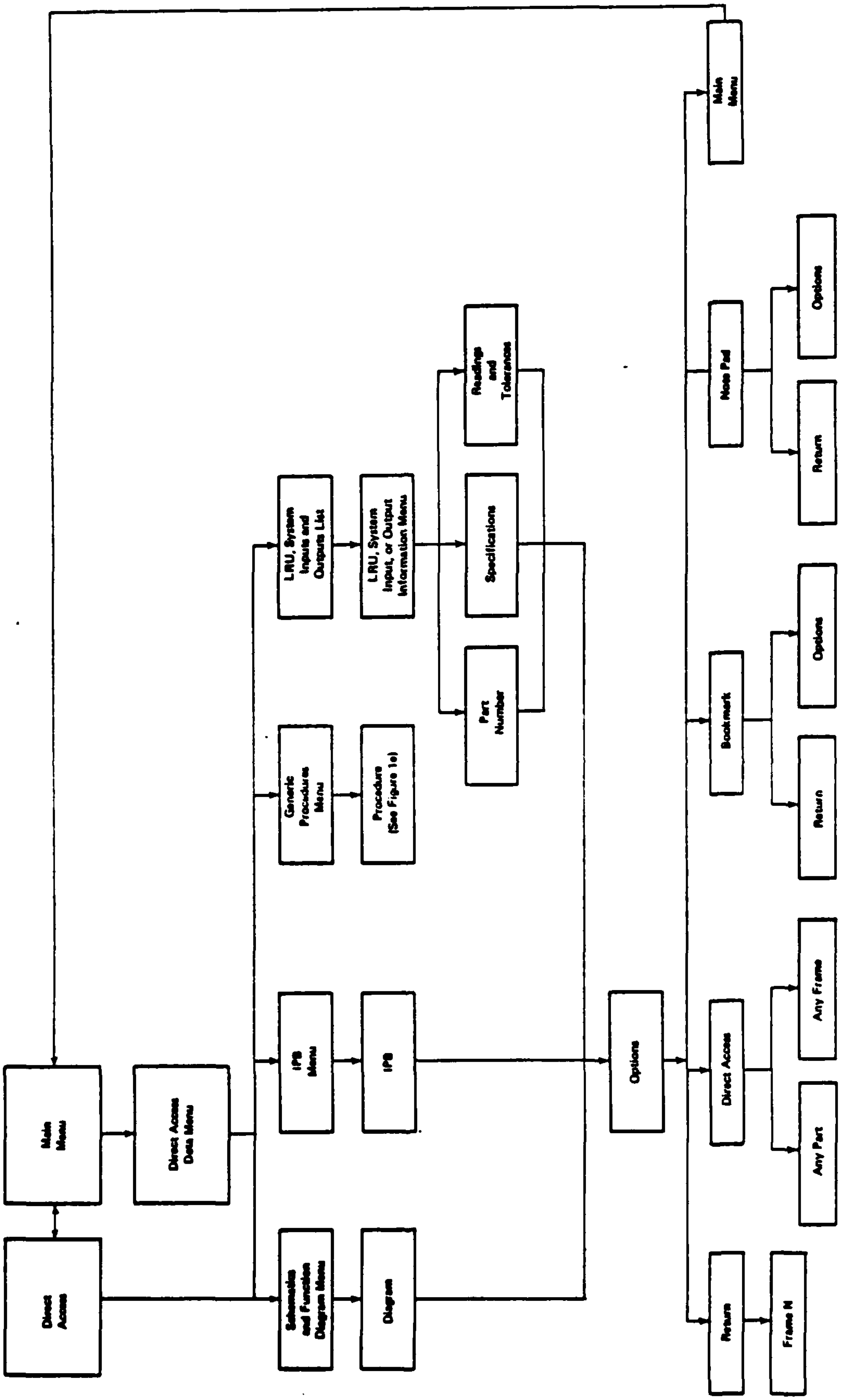


Figure A-1h. Data Structure - Direct Access Data.

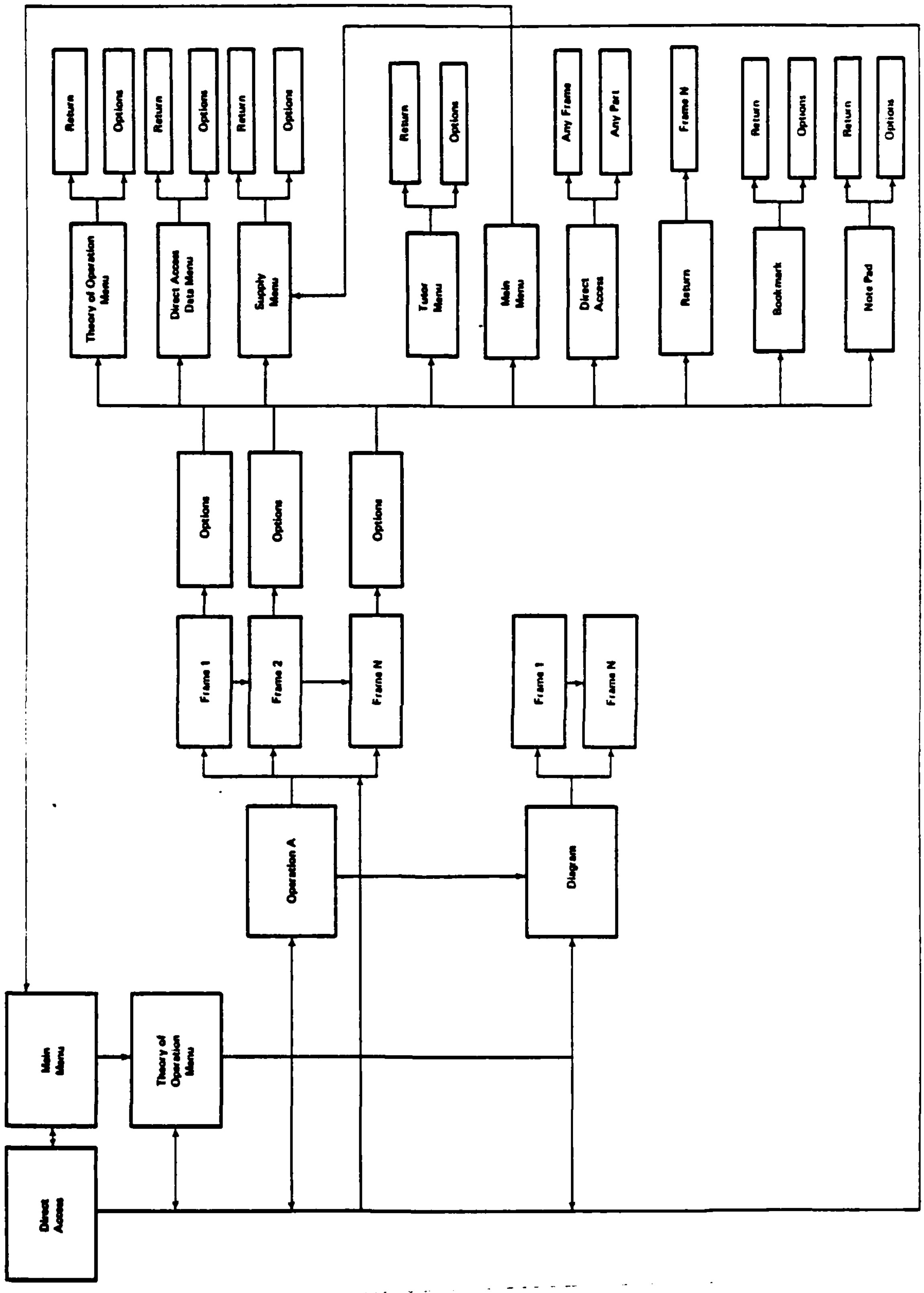


Figure A-1i. Data Structure - Theory of Operations.

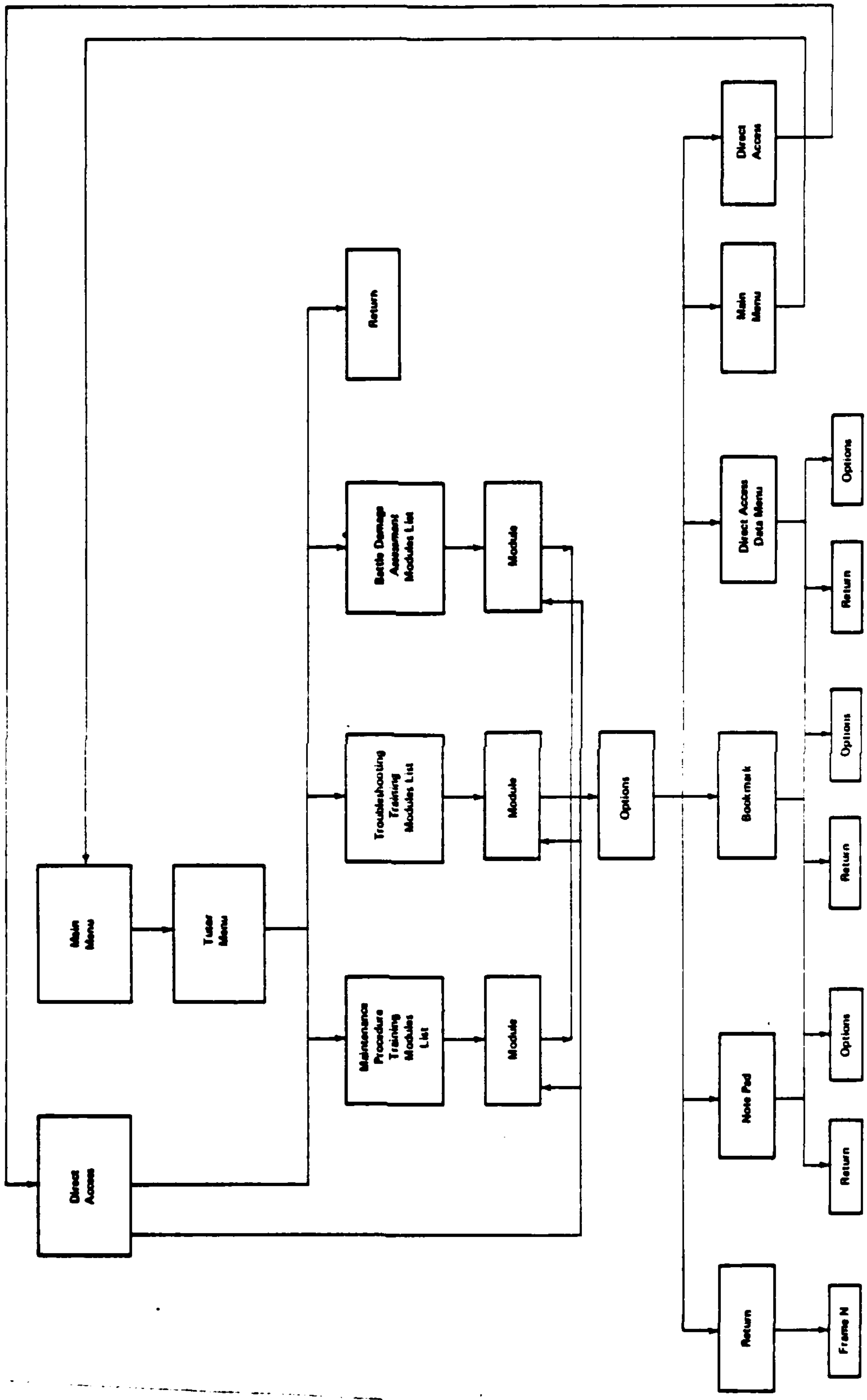


Figure A-1j. Data Structure - Tutor.

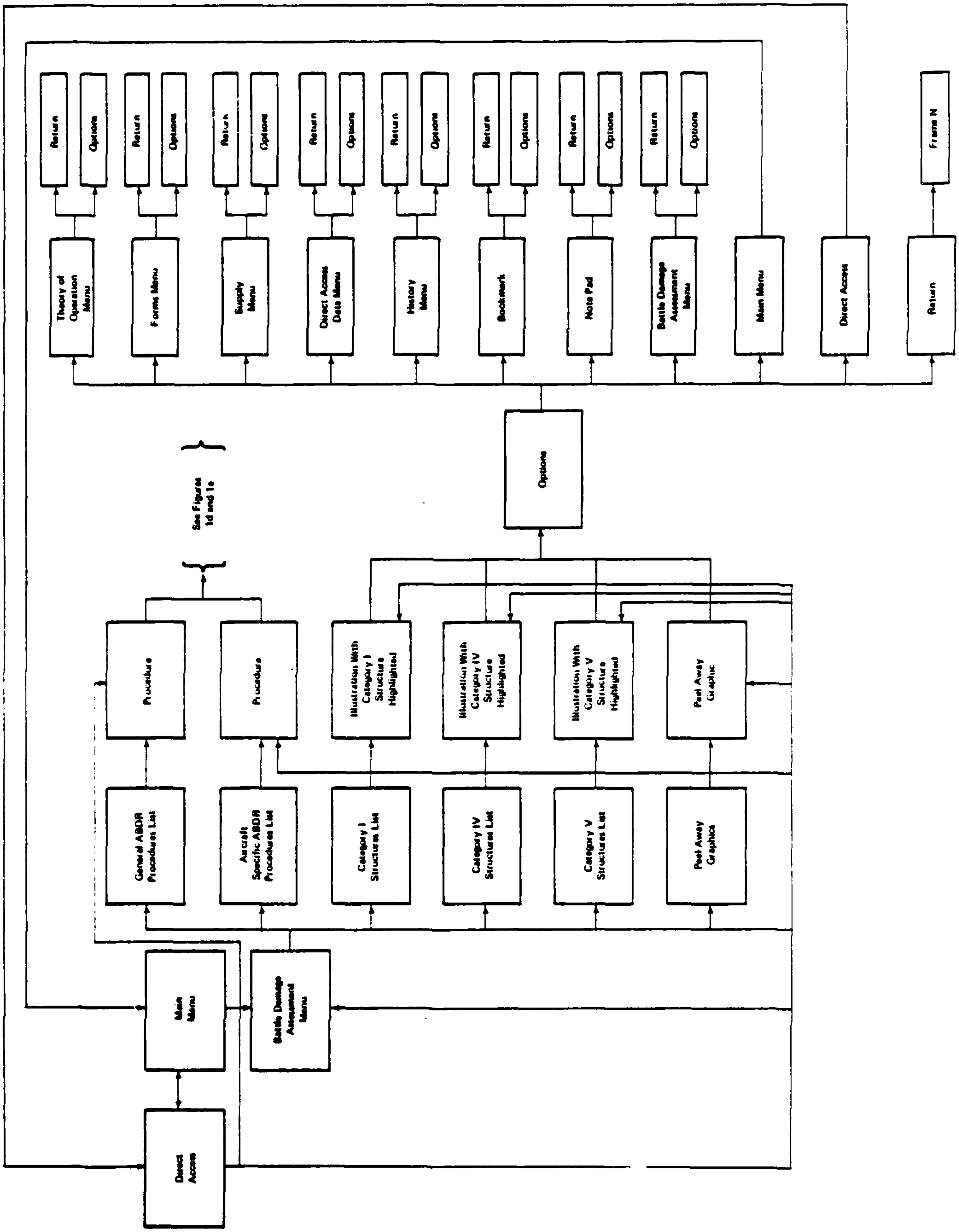
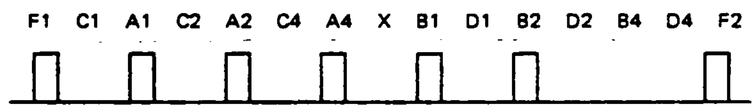


Figure A-1k. Data Structure - Battle Damage Assessment.

12P4-2APX64-2 THEORY OF OPERATION 1.2.6T

Mode 1 codes are designated by code numbers 00 through 73. Only groups A and B are used in this mode and pulse B4 is not used. There are only 32 possible combinations in Mode 1 operation.

PULSES USED IN MODE 1 OPERATION



NEXT = SPACE BAR BACK = B OPTIONS = O RETURN = R

Figure A-2. Example of a Paragraph Presenting Theory of Operation.

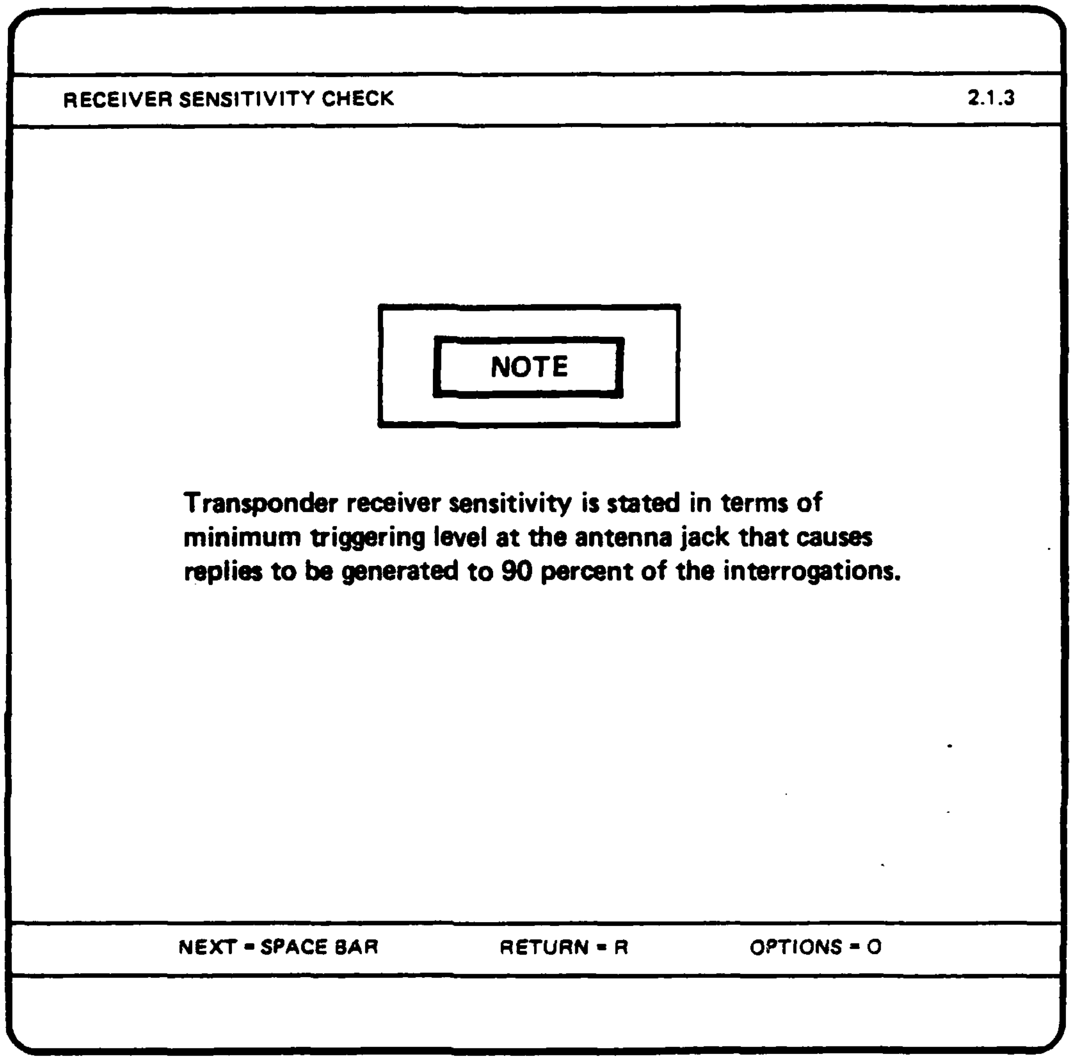


Figure A-3. Example of a Note in a Square Frame.

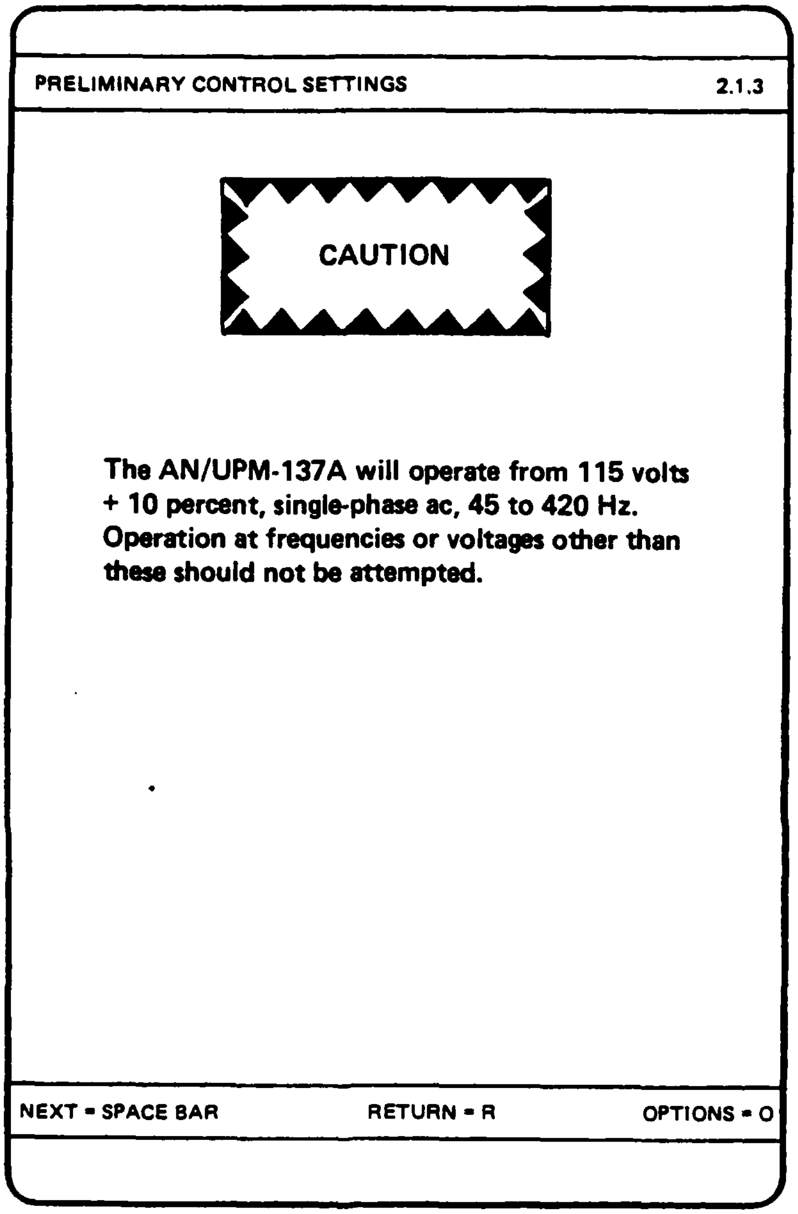


Figure A-4. Example of a Caution in a Vertical Rectangular Frame.

12P4-2APX64-2 PRELIMINARY CONTROL SETTINGS AND CONNECTIONS 2.1.20

WARNING

To prevent the possibility of electric shock when simultaneously touching the test unit and a ground object, establish a connection between the case and ground before applying power to the AN/UPM-137A.

NEXT = SPACE BAR BACK = B OPTIONS = O

Figure A-5. Example of a Warning in a Horizontal Rectangular Frame.

CHECKOUT 7634

INPUT CONDITIONS: CHECKOUT
RADIO RECEIVER-TRANSMITTER

a. Applicable Serial Nos: All

b. Supplies: None

c. Personnel Required: One

d. Special Tools and Test Equipment:

AN/UPM-137A Radar Test Set
AN/APM-239A Transponder Test Set
AN/APM-245 Mode 4, Signal Generator Simulator
Fault Isolation Meter

e. Safety Summary:
See following frames
— 7635
— 7636

NEXT = SPACE BAR BACK = B OPTIONS = O

Figure A-6. Example of List Format.

GLOSSARY: RADIO RECEIVER-TRANSMITTER 7684

Glossary of Nomenclature
For < System Name >

< First Item > : < Item Definition . . . . .
. . . . >

< Second Item > : < Item Definition . . . . . .
. . . .
. . . >

•
•
•
•
•

< Nth Item > : < Item Definition >

To locate an item, scroll up or down to it using the cursor control keys;
or input either the first letter of the item or all of it and Press RETURN ▲

BACK = B RETURN = R OPTIONS = O

Figure A-7. Example of the Format for a Glossary.

Index For Radio Receiver-Transmitter 36874
RT-728/APX-64(V)

Entry	Frame Number
Antenna, Diagrams	
Functional Block	683
Logic	684
Single Function	
<SFD 1 Name >	685
:	:
<SFD N Name >	:
Schematic	689
Procedures	
Maintenance Checklist	
< Maintenance Procedure 1 >	FN
:	FN
< Maintenance Procedure N >	FN
Maintenance Job Guide	
<Maintenance Procedure 1 >	FN
:	FN
< Maintenance Procedure N >	FN
Operation, Checklist	FN
Operation, Job Guide	FN

To Locate Desired Entry, Input Entry Name
To View Information, Input Frame Number ▲

↕ SCROLL * = OPTIONS

Figure A-8. Example of an Index.

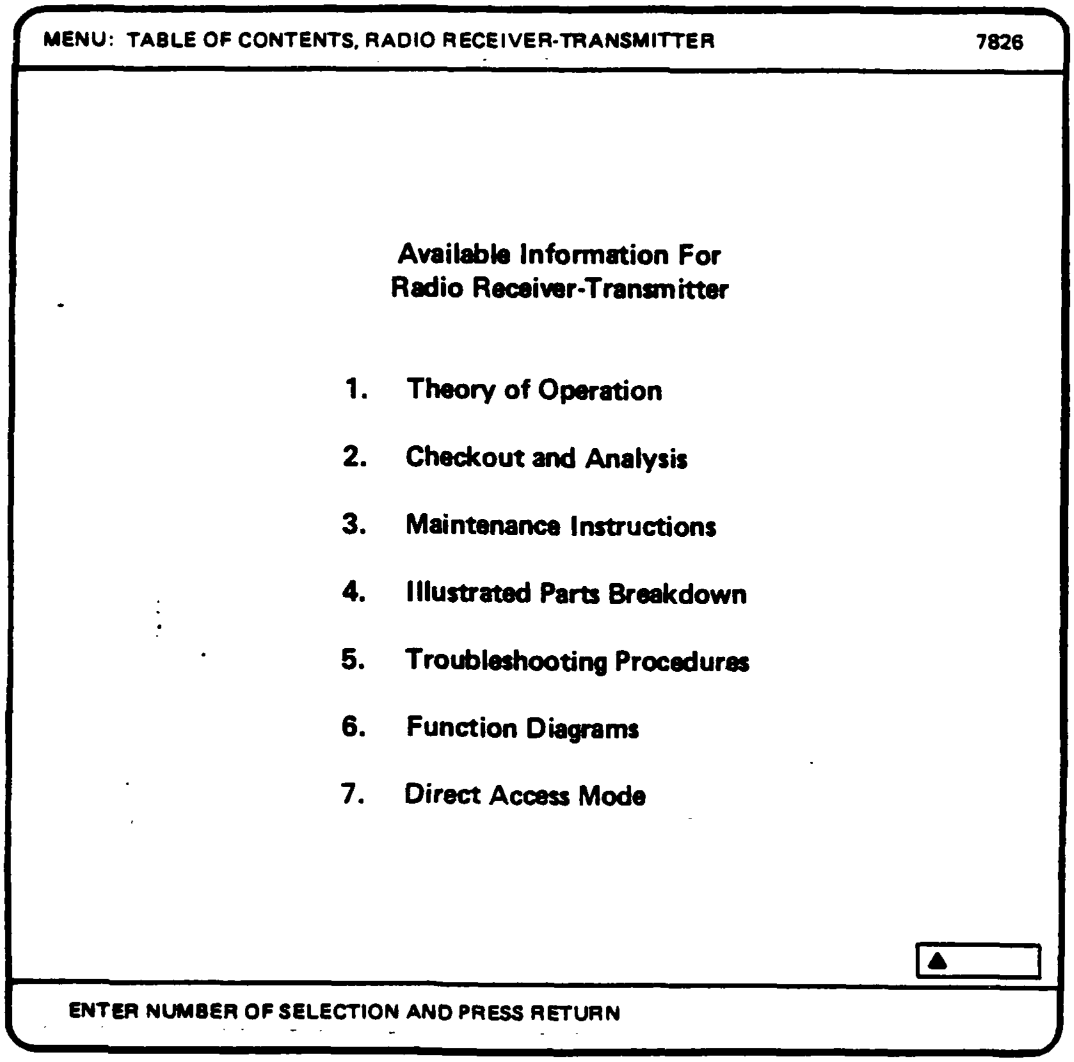


Figure A-9. Example of a Single Frame Menu.

MENU: SCHEMATICS – RADIO RECEIVER-TRANSMITTER (1 of 2) 6378

Schematic Diagrams For
Radio Receiver-Transmitter RT-728/APX-64(V)

1. RF Module – A1
2. IF Amplifier Module – A2
3. Decoder Module – A3
4. Delay Line Module – A4
5. Coder Module – A5

Select Choice and Press Return ▲

NEXT = SPACE BAR OPTIONS = O

FRAME 1

MENU: SCHEMATICS – RADIO RECEIVER-TRANSMITTER (2 of 2) 6379

6. Reference Signal Generator – A6
7. Transmitter Module – A7
8. Power Supply Module – A8
9. Test Module – A9
10. Table of Contents Menu

Select Choice and Press Return ▲

BACK = B OPTIONS = O

FRAME 2

Figure A-10. Example of a Multi-Frame Menu.

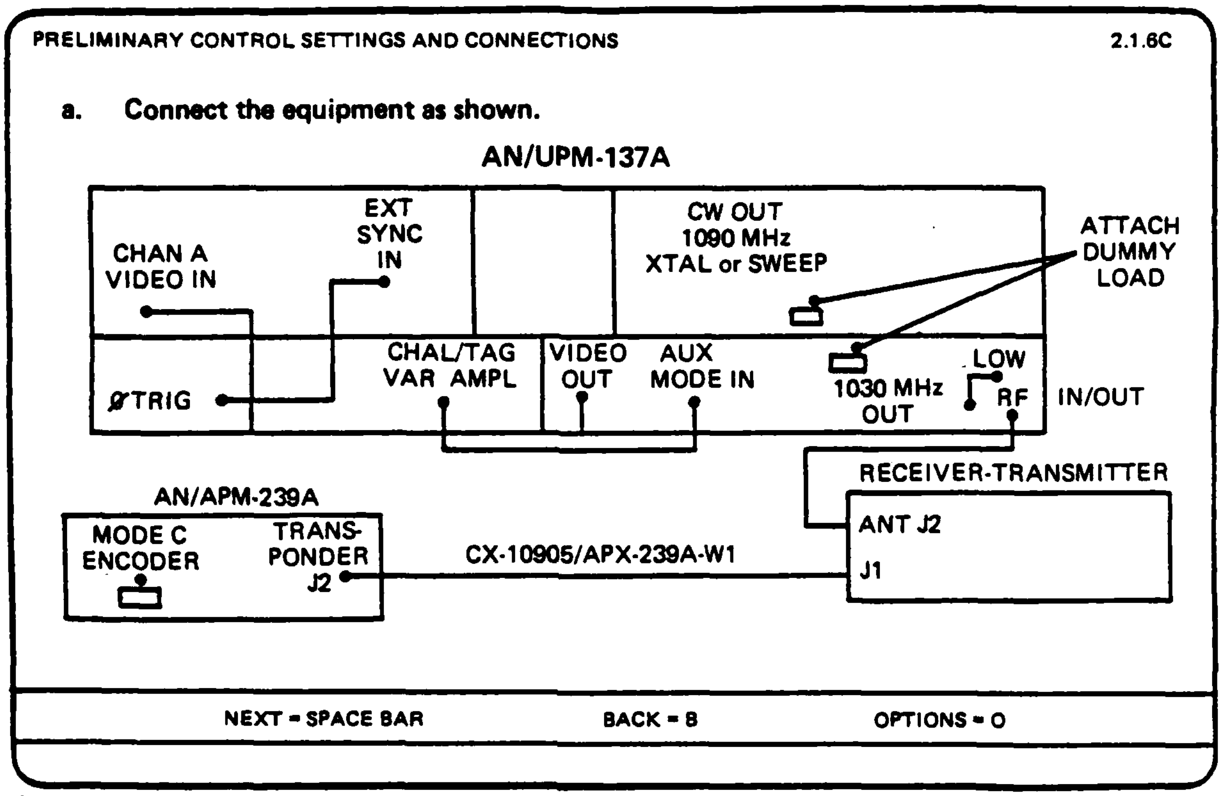


Figure A-11. Example of Checklist with Graphic in a Horizontal Rectangular Frame.

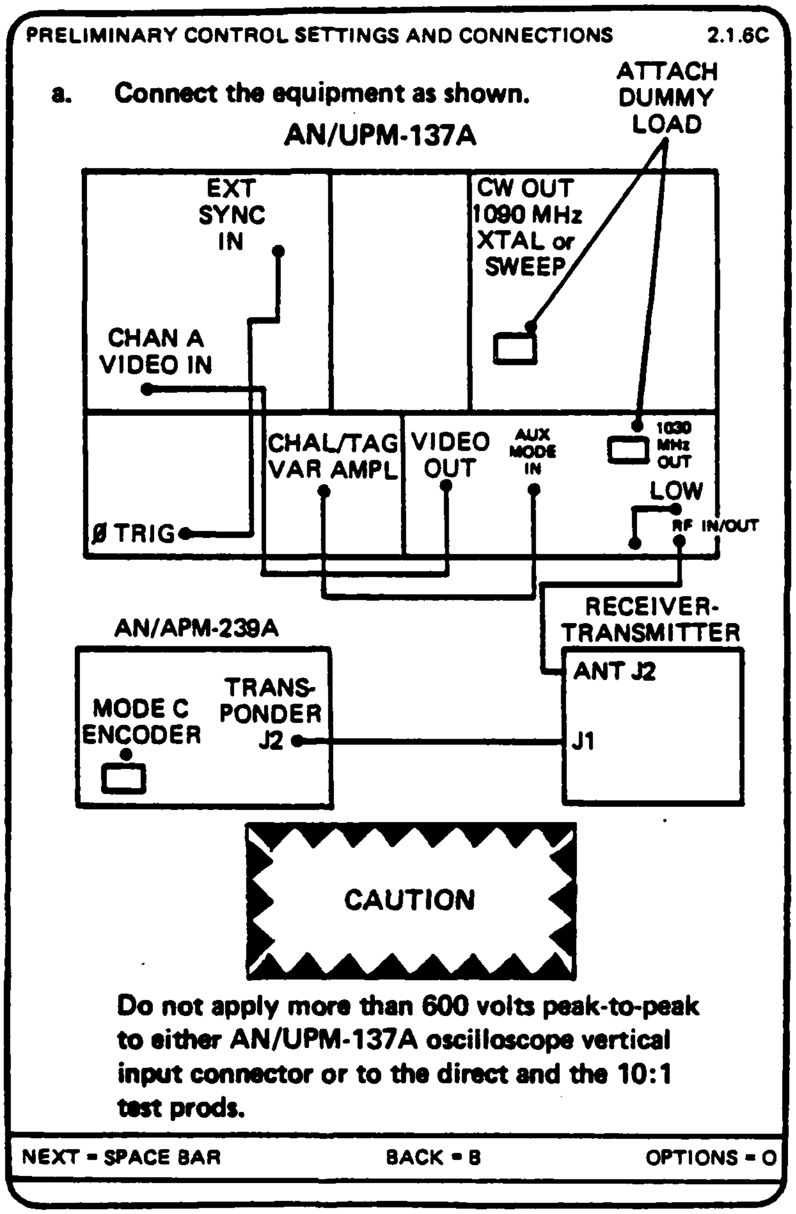


Figure A-12. Example of a Checklist with Graphic in a Vertical Rectangular Frame.

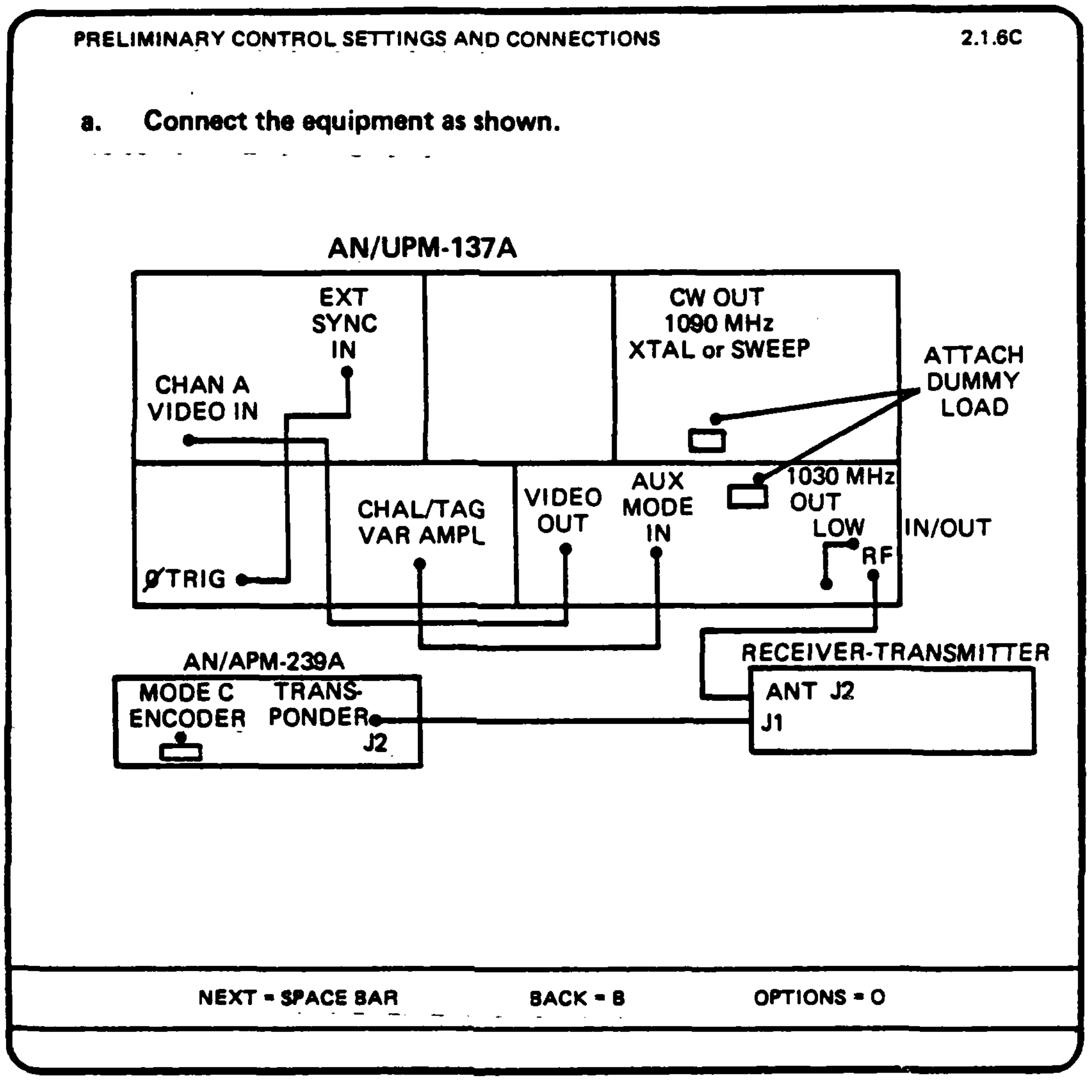
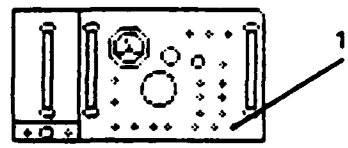


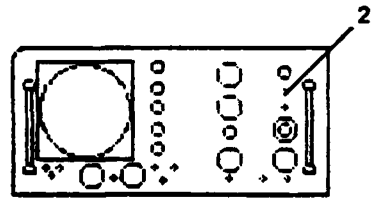
Figure A-13. Example of a Checklist with Graphic in a Square Frame.

PRELIMINARY CONTROL SETTINGS AND CONNECTIONS 2.1.25.1C



UPM-137 SIS

cc. Disconnect cable from AN/UPM-137A interrogator signal simulator PRF COUNTER IN jack (1) and connect cable to oscilloscope EXT SYNC IN jack (2).



Oscilloscope

NEXT = SPACE BAR BACK = B OPTIONS = O

Figure A-14. Example of Job Guide Procedure in a Horizontal Rectangular Frame.

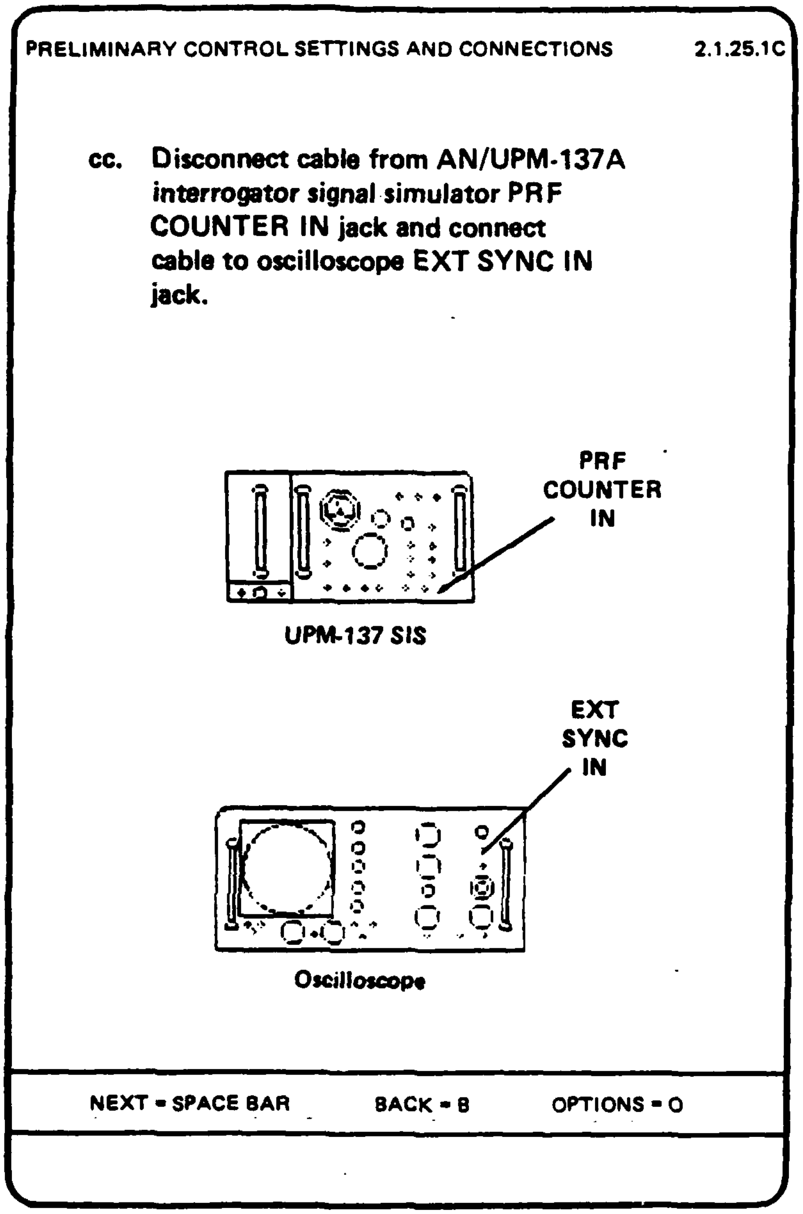
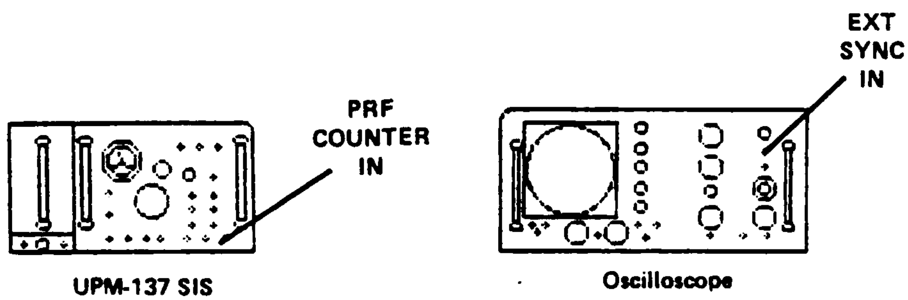


Figure A-15. Example of Job Guide Procedure in a Vertical Rectangular Frame.

PRELIMINARY CONTROL SETTINGS AND CONNECTIONS 2.1.25.1C

cc. Disconnect cable from AN/UPM-137A interrogator signal simulator PRF COUNTER IN jack (1) and connect cable to oscilloscope EXT SYNC IN jack (2).



NEXT = SPACE BAR BACK = B OPTIONS = O

Figure A-16. Example of Job Guide Procedure in a Square Frame.

TABLE: SETTINGS FOR RADIO RECEIVER-TRANSMITTER 7477

Table of Value Settings For Radio Receiver-Transmitter

	Dial Number			
	741	742	743	744
Low Value	7.3	14.6	3.2	1.0
High Value	10.1	20.2	5.0	5.0

BACK = B RETURN = R OPTIONS = O

Figure A-17. Example of a Table Containing Numeric Values.

PARTS HISTORY SYSTEMS 463

Parts History For
[SYSTEM]

Component	Number/System	Fault Symptom Rank Ordered By Frequency	Fault Possibilities Rank Ordered By Frequency
Component 1	N/S	Symptom 1	Fault 1 Fault 2 ⋮ Fault N
		Symptom 2	Fault
Component 2	N/S	Symptom	Fault 1 Fault 2
Component 3	N/S	Symptom 1 ⋮ Symptom N	Fault 1

To see subcomponent histories, select component number of
Desired Component [▲]

BACK = B NEXT = SPACE BAR RETURN = R OPTIONS = O

Figure A-18. Example of a Text Table.

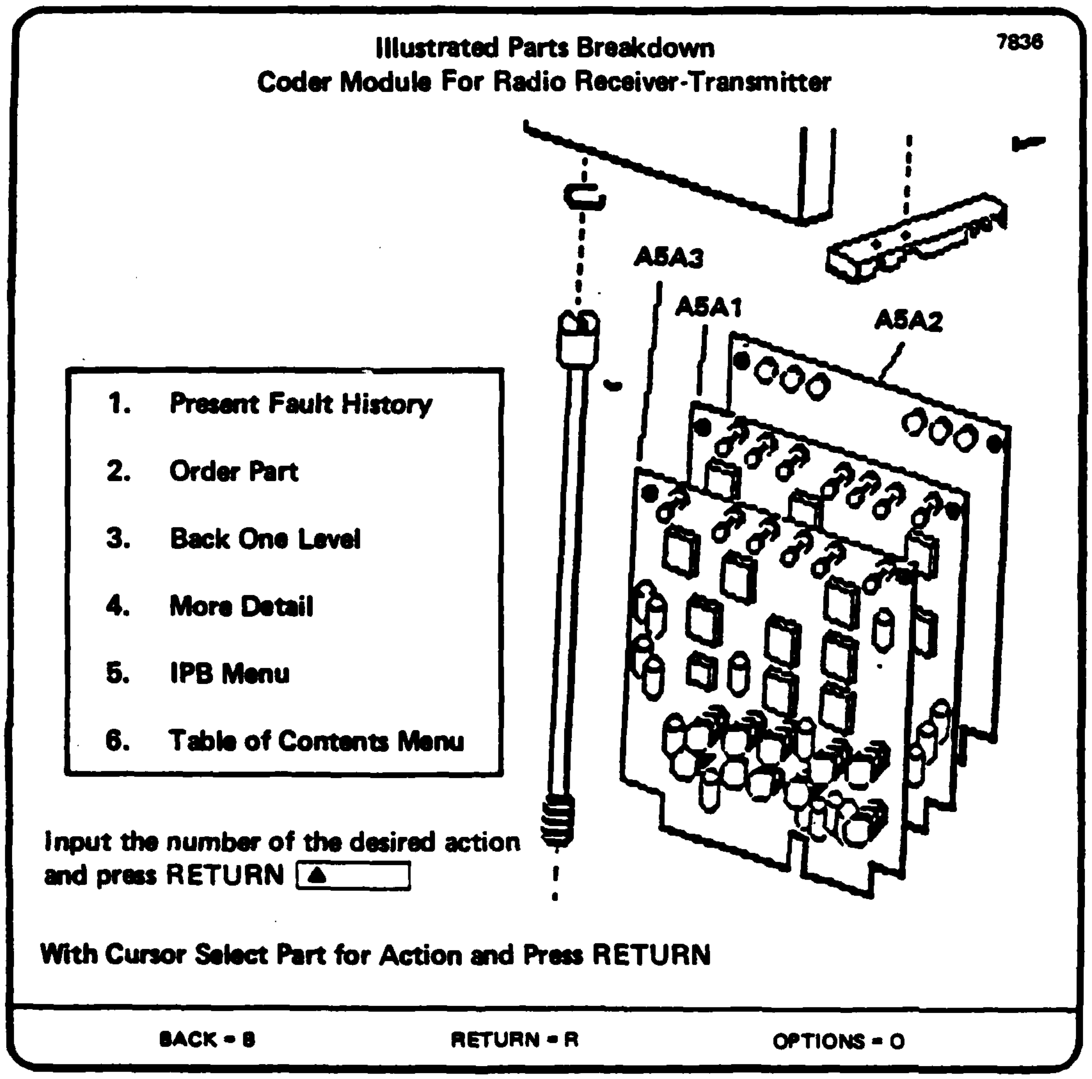


Figure A-19. Example of the Display of Illustrated Parts Breakdown Information.

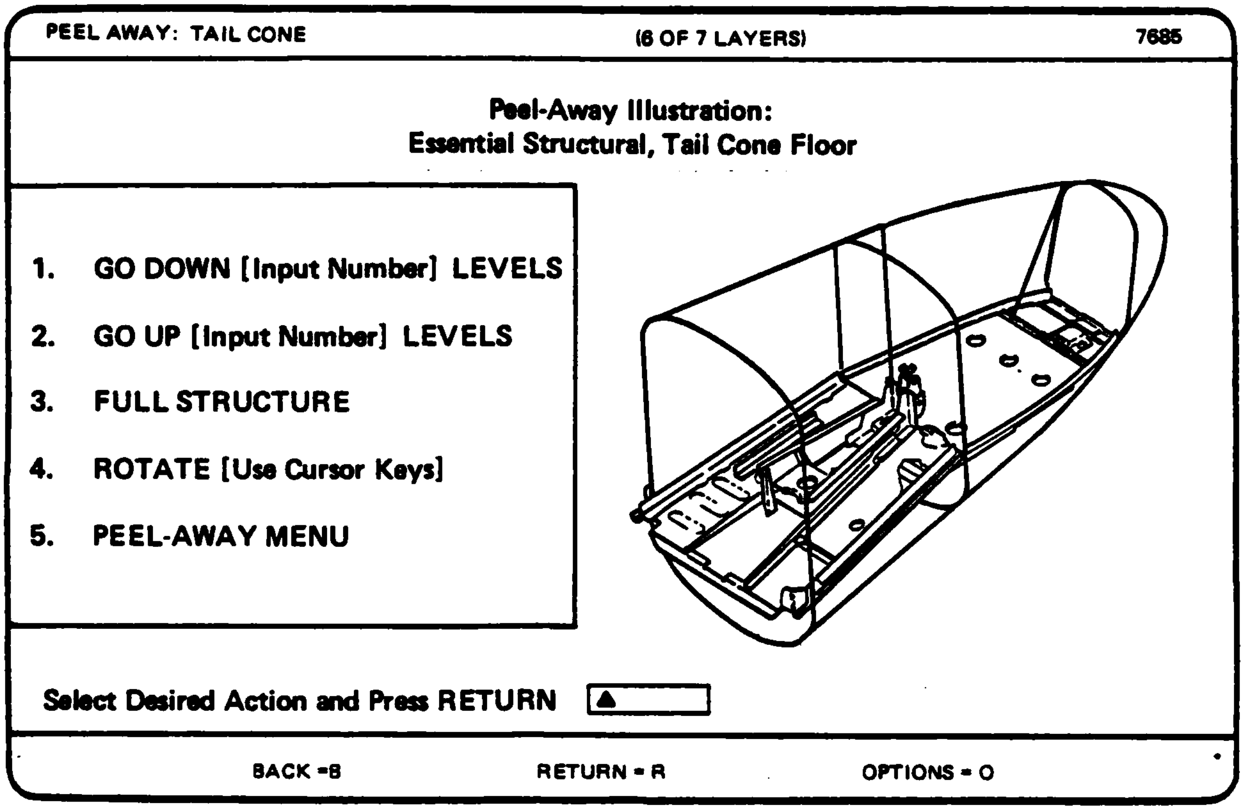
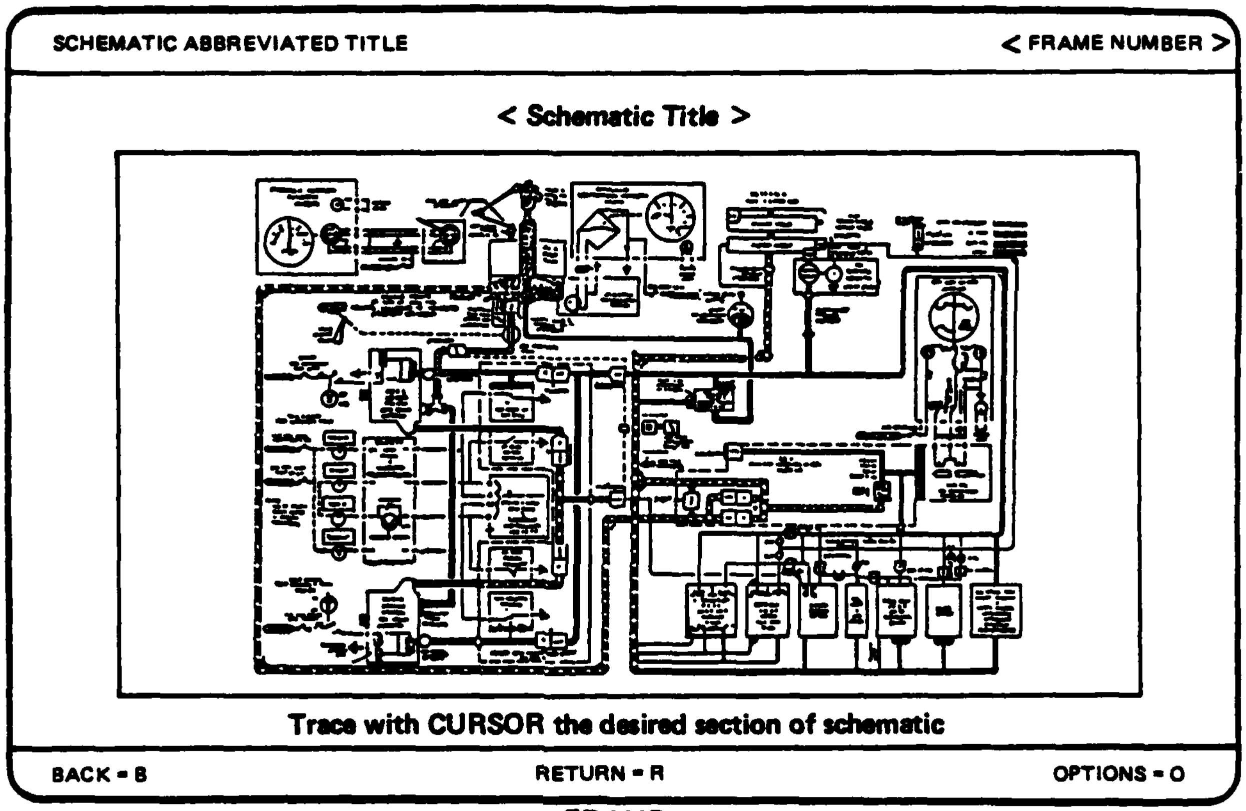
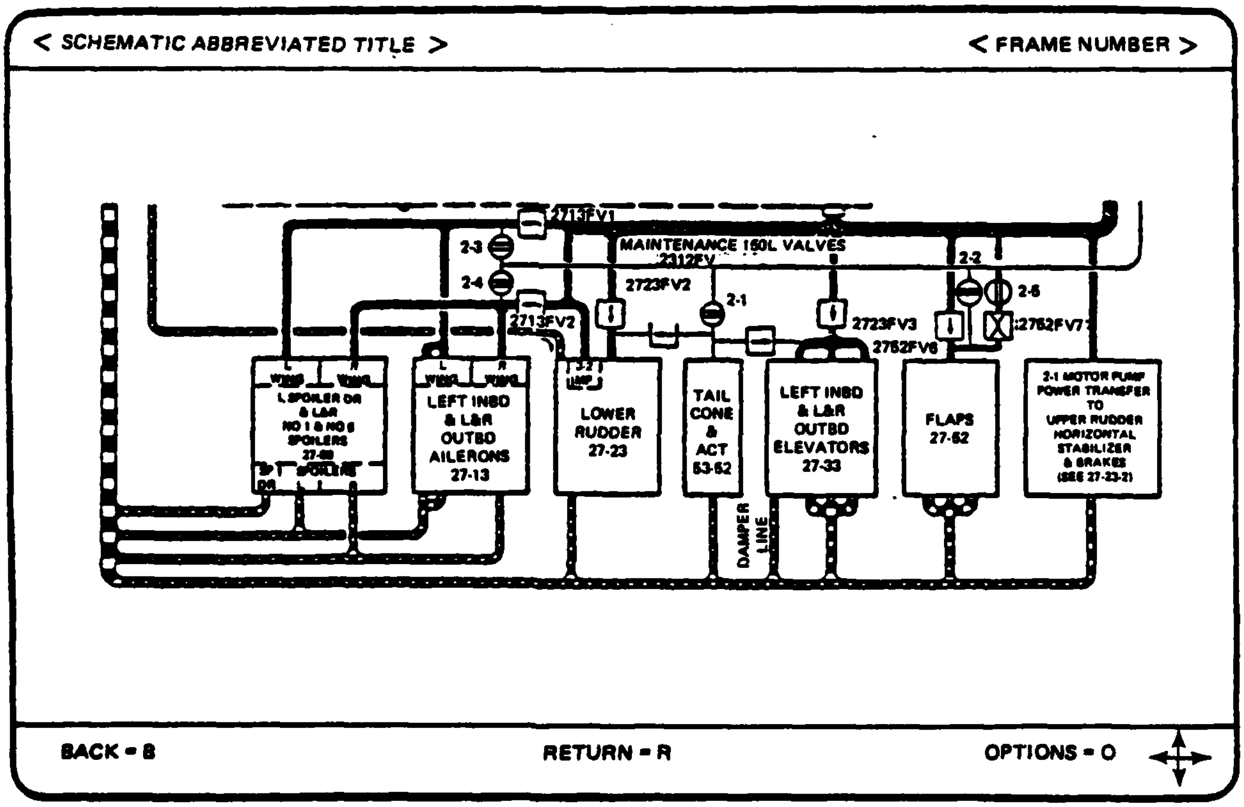


Figure A-20. Example of Format for a Peel-Away Graphic.



FRAME 1



FRAME 2

Figure A-21. Example of a Schematic with Zoomed Section.

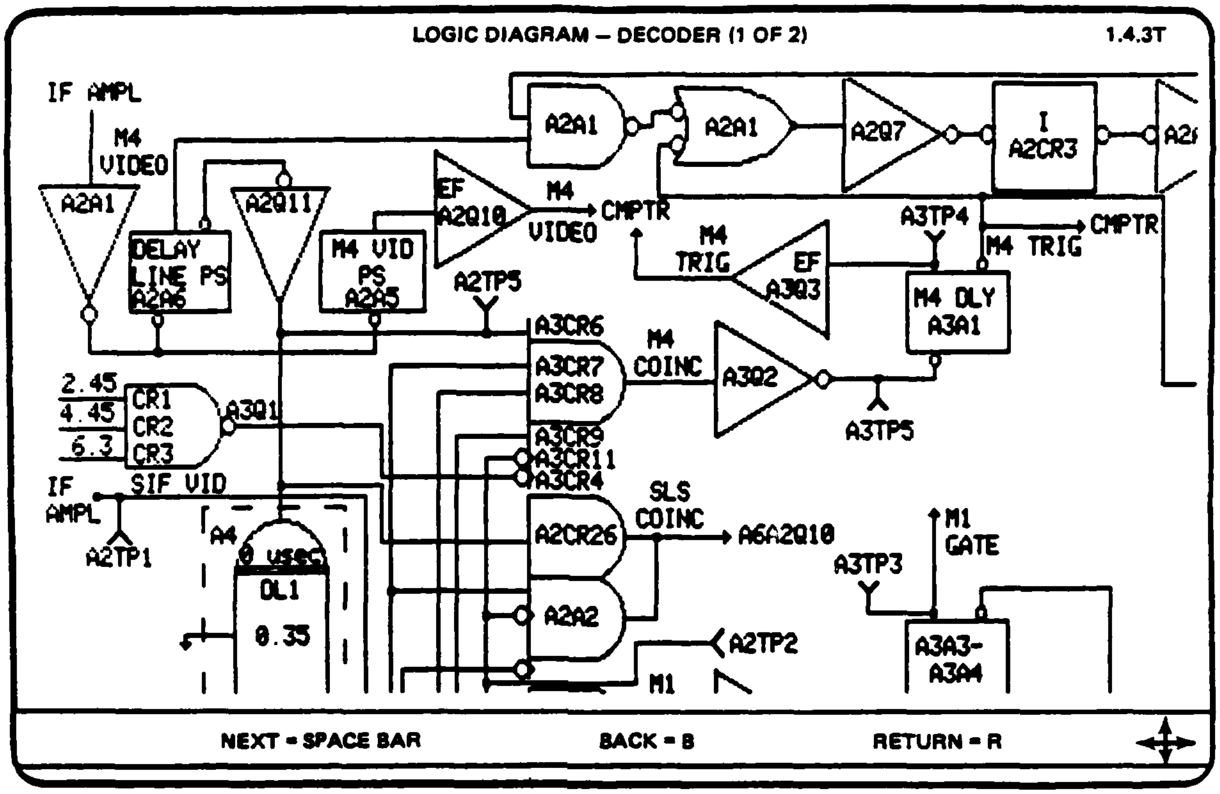
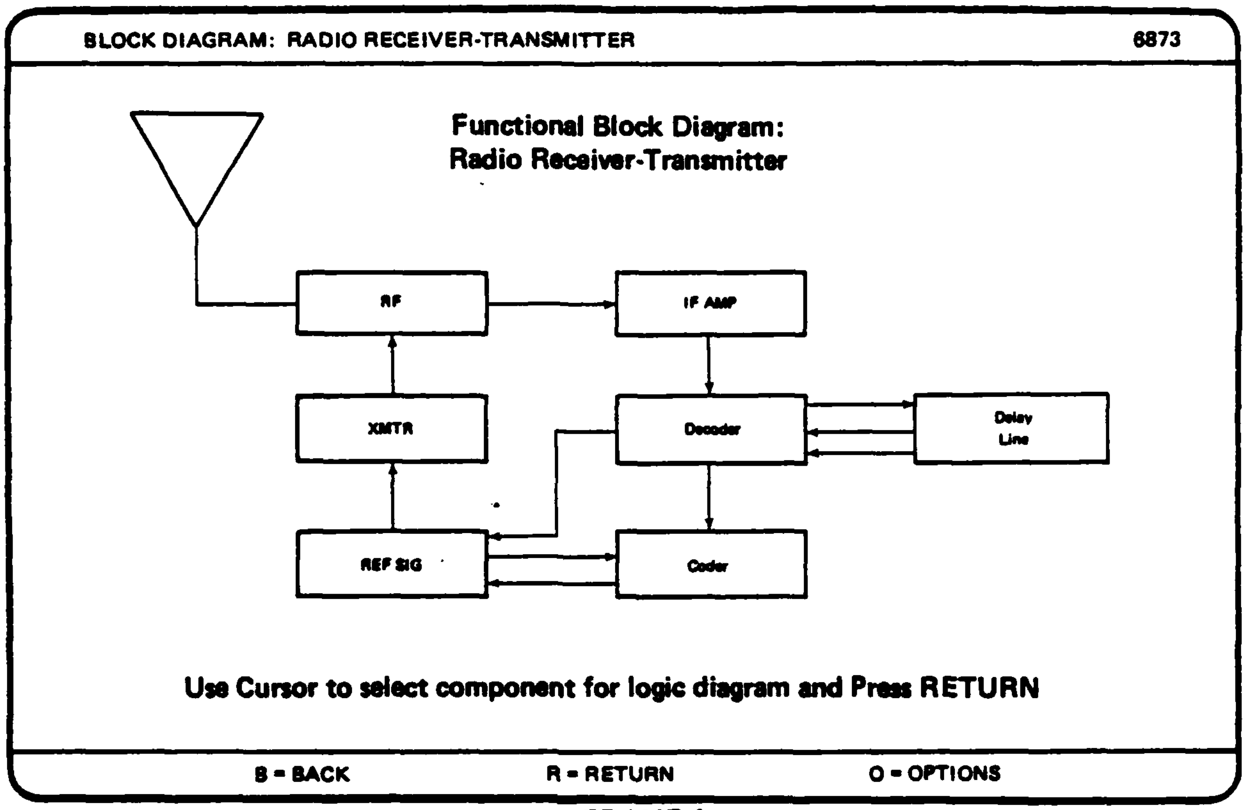
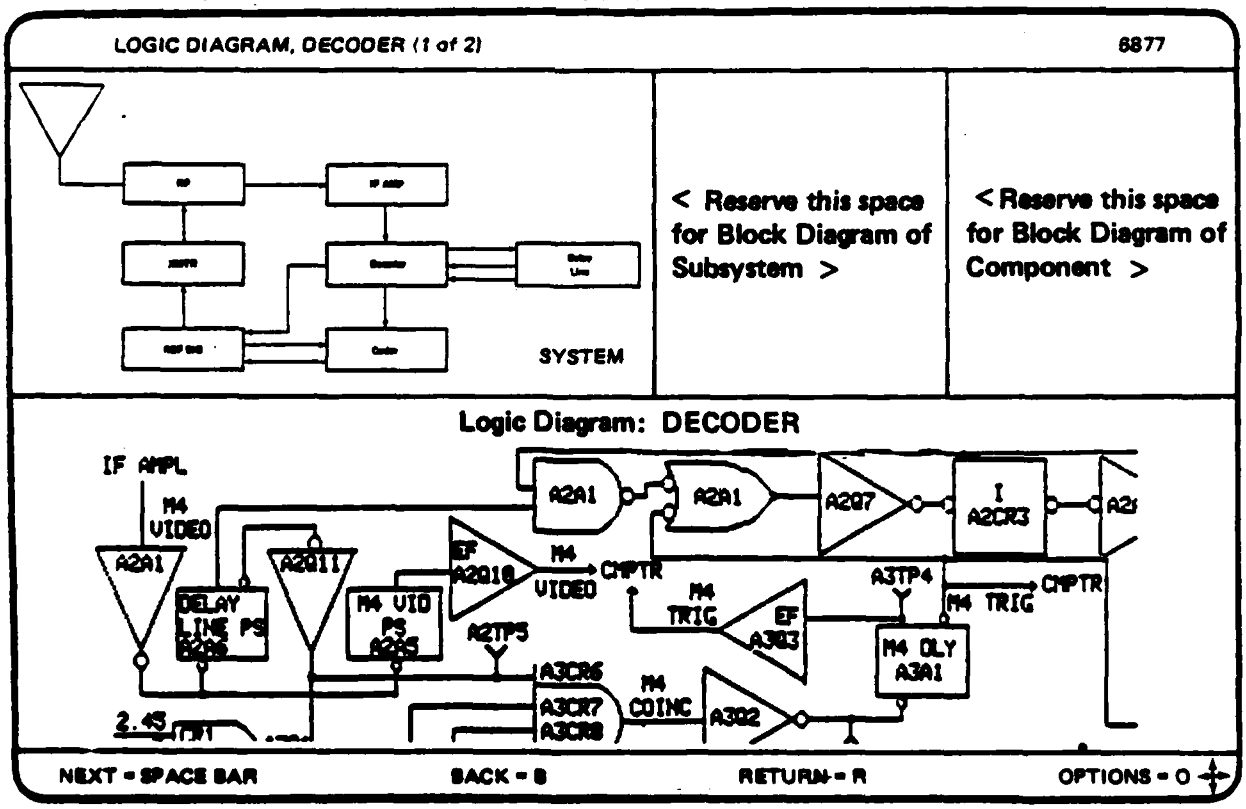


Figure A-22. Example of a Logic Diagram Presented with A Scroll Function.



FRAME 1



FRAME 2

Figure A-23. Example of Pyramiding Diagrams.