DOCUMENT RESUME

ED 066 888 EM 010 186

AUTHOR	Landa, Suzanne
TITLE	CATTS: Computer-Aided Training in Troubleshooting.
INSTITUTION	Rand Corp., Santa Monica, Calif.
SPONS AGENCY	Air Force Directorate of Operational Requirements and Development Plans, Washington, D.C.
REPORT NO	R-518-PR
PUB DATE	May 72
NOTE	108p.; Project Rand
EDRS PRICE	MF-$0.65 HC-$6.58
DESCRIPTORS	Aviation Technology; *Computer Assisted Instruction; *Computer Graphics; Computer Programs; *Military Training; *Technical Education; Training
IDENTIFIERS	CATTS; *Computer Aided Training in Troubleshooting; POGO; Programmer Oriented Graphics Operation

ABSTRACT

The Rand Corporation's Programmer-Oriented Graphics Operation (POGO) was used in the design, implementation and testing of a computer-assisted instruction course to train airmen in malfunction diagnosis--CATTS (Computer Aided Training in Troubleshooting). The design of the course attempted to reduce the problems of computer graphics for both instructor and student. The observations of an Air Force instructor using the system suggest that the system is workable. (RH)

R-518-PR

May 1972

CATTS: Computer-Aided Training in Troubleshooting

Suzanne Landa

A Report prepared for

UNITED STATES AIR FORCE PROJECT RAND

Rand
SANTA MONICA, CA 90406

This research is supported by the United States Air Force under Project Rand—Contract No. F44620-67-C-0045—Monitored by the Directorate of Operational Requirements and Development Plans, Deputy Chief of Staff, Research and Development, Hq USAF. Views or conclusions contained in this study should not be interpreted as representing the official opinion or policy of Rand or of the United States Air Force.

R-518-PR

May 1972

CATTS: Computer-Aided Training in Troubleshooting

Suzanne Landa

A Report prepared for

UNITED STATES AIR FORCE PROJECT RAND

Rand
SANTA MONICA, CA. 90406

Bibliographies of Selected Rand Publications

Rand maintains a number of special subject bibliographies containing abstracts of Rand publications in fields of wide current interest. The following bibliographies are available upon request:

Aerodynamics • Arms Control • China • Civil Defense
Communication Satellites • Communication Systems
Computer Simulation • Computing Technology
Decisionmaking • Game Theory • Maintenance • Middle East
Policy Sciences • Probability • Program Budgeting
SIMSCRIPT and Its Applications • Southeast Asia
Space Technology and Planning • Statistics • Systems Analysis
USSR/East Europe • Weapon Systems Acquisition
Weather Forecasting and Control

To obtain copies of these bibliographies, and to receive information on how to obtain copies of individual publications, write to: Publications Department, Rand, 1700 Main Street, Santa Monica, California 90406.

Published by The Rand Corporation

PREFACE

The Rand Corporation has within recent years developed a computer-based, "interactive" video-graphics system that permits a user to create, modify, and record for later transmission visual images on a television screen, or cathode ray tube, linked to his console. (See RM-5825-PR, *POGO: Programmer-Oriented Graphics Operation*, June 1969.) At the same time, research on new training methods by the Air Training Command has indicated that computer-aided instruction might be a useful and more economical way of teaching trainees than present techniques. This Report describes a video-graphics system for instructing airmen in malfunction diagnosis: Computer-Aided Training in Troubleshooting (CATTS).

The Report tests the hypothesis that video-graphics could eliminate several limitations on the application of computer-aided instruction to technical training. Troubleshooting was selected because of its practical significance and because it is not being taught effectively by current methods. Test observations of the system using an Air Force instructor tend to support this hypothesis.

Distribution will include supply and maintenance groups at the Air Force Logistics Command and in the field, as well as offices within the Air Training Command.

SUMMARY

The high cost of traditional methods of technical training has stimulated research by the Air Force on new educational methods. Of particular interest is the use of computer-aided instruction (CAI) as a more efficient and effective teaching medium.

This Report first examines costs of conventional Air Force training methods that CAI has the potential to reduce: (1) it could decrease the number of instructors needed; (2) it would provide a method of individualized instruction that may prove more useful than present group instruction; and (3) in the long run, it might cut equipment costs. Current capabilities and limitations of CAI systems are examined next. CAI languages include the facility to display pictures as well as text on a cathode ray tube (CRT), a great improvement over a line-at-a-time device such as a typewriter. However, CAI languages do not permit easy use of the CRT, either in the creation or in the execution of CAI courses. Not only is it difficult to use the graphical capabilities of the CRT through course author languages, but also the languages demand considerable time and attention from the user-instructor who wishes to modify a standard, packaged course.

It appears that several of these limitations could be eliminated using The Rand Corporation's Programmer-Oriented Graphics Operation (POGO), which was used in the design, implementation, and testing of a course described here called Computer-Aided Training in Troubleshooting (CATTS) for training airmen in malfunction diagnoses. POGO, which permits direct and easy creation and execution of graphic displays on a CRT, is intended to simplify the graphical programming process in two ways:

1. It relieves the programmer of the tedious and artificial process of specifying CRT displays by transcribing coordinates from layout paper and stringing together calls to graphic support subroutines.

2. It permits users to create interactive computer graphics programs without spending a great deal of time learning the intricacies of the graphic subroutine package.

Test observations supported the initial hypothesis that CAI systems design and execution could be improved by use of a graphics system such as POGO. An Air Force instructor who tested the system found the problem-creation, -insertion, and -execution facilities easy to use. In addition, he noted that the problem structure of the prototype course provides a convenient framework for classroom presentation.

Current CAI systems are necessarily limited by the few facilities that have been designed into them. To remove this limitation, it is recommended that future CAI systems be "coherently programmed" to permit interleaving of software as required by the user.

ACKNOWLEDGMENTS

The author wishes to express her appreciation to those at Rand who helped to make this research possible. Special thanks go to Dr. Barry Boehm for his constant guidance and encouragement throughout the project, and to Dr. John White for continued support during the writing of this Report. In addition, the generous contributions in time and effort of John Rieber, Polly Carpenter, and Don Cohen are gratefully acknowledged.

CONTENTS

I. INTRODUCTION

THE HIGH COST OF TECHNICAL TRAINING

During fiscal 1967, the Air Training Command (ATC) taught approximately 550,000 students in 3500 technical courses, which required 13,000 instructors and 4000 classrooms. As one of the world's largest training programs, the ATC is also one of the most expensive. Total training costs for 1967 were about one billion dollars [1]. Such high costs prompted examination of ATC training programs and procedures for possible improvements. Of particular interest, because of its size as well as its costs, is the resident training subprogram. Before suggesting possible improvements in the program, it was first necessary to identify those factors contributing to high costs.

A major factor is the low ratio of students to instructors. Because an instructor can supervise only a limited number of students during work on complicated equipment, individual classes average ten students and, in some instances, the student-instructor ratio is as low as one to one. Small classes, of course, require more teachers; in addition, more facilities are required which, in turn, necessitate a greater investment outlay.

Required investment is further escalated by the cost of specialized labs and equipment. Most courses make heavy use of labs and simulators. In some cases the simulated equipment and systems cost more than real-world facilities. In addition, increasing technological advances result in a high rate of equipment obsolescence.

Another major cost factor is low reenlistment rates. For example, only 16 percent of first-term airmen reenlist [1]. The void left by the other 84 percent must be refilled by new trainees from Air Force schools. Such a turnover rate results in a high ratio of training costs to effective work time on the job.

This high ratio is aggravated by the fact that evaluation of on-the-job performance after graduation from ATC classes indicates that airmen are not learning basic skills in the classroom. A possible cause of the program's ineffectiveness is that courses are organized in traditional, lock-step blocks. Since student learning rates vary, a self-paced approach might improve results.

A POSSIBLE SOLUTION

Computer-aided instruction (CAI) is a medium with the potential to supply individualized, controlled instruction for the military trainee. With the computer's immense capabilities to accept and store information rapidly, execute complex rules, and output results, it becomes possible to adapt instruction to the needs of each individual trainee without the constant attention of an instructor. In addition, to the extent that the quality of instructional programs can be controlled more readily than the quality of lectures, for example, CAI has the potential to increase instructional effectiveness. More individualized, effective instruction will reduce time-in-training and improve on-the-job performance, thereby reducing the ratio of training costs to effective work time on the job. Although a CAI system requires a large investment outlay itself, it has the potential to reduce or eliminate significantly the investment in systems and components for teaching purposes through simulation of the actual equipment. Computer simulations differ from physical simulators in that (1) the former are not physical replicas, and (2) one computer can simulate many machines and processes, but a physical simulator can usually replicate only one machine or system.

Systems simulation can further reduce training costs and increase training effectiveness in two ways:

1. By eliminating the obsolescence factor inherent in the use of actual equipment; i.e., simulation programs can be altered or replaced at less cost than real equipment.
2. By presenting through simulation common system malfunctions that damage or ruin real equipment.

Although arguments for using CAI in technical training are presented here, it is not within the scope of this Report to pursue and establish the potential value of CAI to technical training. Rather, this discussion is presented to establish the reasons for Air Force interest in CAI research. It is the author's purpose, having accepted CAI as a viable alternative, to specify its current limitations; to describe a non-CAI computer system that has the potential to eliminate some of these limitations; and to develop, implement, and test a prototype course utilizing the recommended system.

ORGANIZATION OF THE REPORT

Section II discusses CAI; the graphics system applicable to CAI; and the area of instructional application, which is troubleshooting. Both the instructor and student modes of the system developed to solve troubleshooting problems are described in Sec. III. A teacher from Chanute Air Force Base tested the instructor mode, and his reactions are presented in Sec. IV. Section V offers some conclusions and recommendations. The Report includes an appendix that details the purpose of standard displays and the required instructor's actions.

II. APPLICATION OF COMPUTER-AIDED INSTRUCTION

CAI: CURRENT CAPABILITIES AND LIMITATIONS

A computer-aided instructional system must provide a course author with the means to store text and graphic displays, and a means to retrieve the displays for the student in some specified order. The student requires a means to interact with the displays. In addition, both author and student may need to retrieve information or use computational facilities (e.g., matrix routines) that are not part of the CAI package. The way these functional requirements are met is determined by the interface devices (hardware) and the CAI languages (software) that are available.

Interface devices incorporated into current CAI systems, either alone or in combination, are typewriters and cathode ray tubes (CRT).* Though commercial research has found that most CAI systems use a typewriter as the interface device, the inadequacy of simple typewriter devices for handling such essential teaching materials as maps, pictures, and diagrams has been stressed repeatedly [2]. The Navy, in response to similar Department of Defense pronouncements on the inadequacy of typewriter devices, observed:

> The terminal is perhaps the area in greatest need of improvement if we are to realize the full potential of CAI. The author input requirements, mostly through a keyboard, force present-day CAI systems to work under severe handicaps. The student must also be freed from the keyboard to as great an extent as possible [3].

As part of Project IMPACT, in 1969 the Army critically examined the nature of both the CRT and the typewriter to gain a clear understanding of their fundamental characteristics and differences as interface devices. They noted that a typewriter is strictly a serial device that forces a sequential presentation of text, no matter what ordering the logic of the course might dictate. Another critical factor is that

*Cathode ray tube: a computer-driven display device.

the number of characters transmitted is a function of the display capability of the terminal device. A single transmission either to or from a CRT terminal may contain many more characters than a single transmission from a typewriter. Then, too, data displayed on a CRT can take on an additional structure (e.g., pages and pictures), which is not possible when a line-at-a-time device such as a typewriter is used [4].

Because of the advantages a CRT offers as an interface device in an educational environment, CAI languages have been developed that include the facility to display text and pictures on a CRT. These languages are of two types. One type, which includes such languages as LYRIC and CAL, allows pictures to be generated by strategically placing selected characters. This facility is not graphic, but rather tabular; it is a formatting capability, and its application can be very time-consuming. The other type includes languages with graphics facilities. But, although graphics can be displayed (to the benefit of the student), the graphic insertion process by the instructor may require special devices and unnatural and tedious steps. For example, the CATO language accommodates the overlay of an image with a computer-generated display, but special hardware is required to accomplish this [5]. INFORM, a CAI language prepared by Philco-Ford, requires the author to prepare the display, correct answer region, and so forth, off-line (i.e., not at the computer terminal), in the form it is to appear on the screen. An assistant must next punch this information line by line on cards. Finally, a special translator must process the cards before the computer can interpret and display the graphics [6]. Close attention, then, must be paid to the amount of work involved in preparing computer-based pictures. Displays that originate in free-hand drawings that are automatically processed obviously require far less effort from the course author [5].

Current CAI software has another drawback in that both course designers and instructors must learn a computer language. Although a team of CAI experts may develop a course, the instructor must also learn their course design language if he is to retain the much-desired right to modify or tailor-make the "standard package." Although designers of course author languages have made considerable progress toward the goal of natural language interaction between man and machine, the instructor-user today must essentially learn a computer language. This requirement decreases the attractiveness of CAI as a teaching medium for many instructors who have neither the time nor the desire to learn a computer language.

In summary, the CRT as an interface device has the potential to satisfy the functional requirements of the CAI user. However, CAI languages have not been developed that permit easy use of the CRT, either in the creation or execution of CAI courses. Not only is it difficult to use the graphical capabilities of the CRT through course author languages, but also the languages demand considerable time and attention from the user-instructor who wishes to modify a standard, packaged course.

In the next section, an interactive graphics system is examined that, although developed for other purposes, seems to provide the facilities necessary for CAI applications utilizing a CRT. This system can be used to develop packaged courses that instructors may subsequently alter without reprogramming in a computer language.

POGO: A GRAPHICS SYSTEM APPLICABLE TO CAI

The Rand Corporation has developed an interactive graphics system called Programmer-Oriented Graphics Operation (POGO), which permits direct and easy creation and execution of graphic displays on a CRT [7]. POGO is intended to simplify the graphical programming process in two ways:

1. It relieves the programmer of the tedious and artificial process of specifying CRT displays by transcribing coordinates from layout paper and stringing together calls to graphic support subroutines.
2. It permits users to create interactive computer graphics programs without spending a great deal of time learning the intricacies of the graphic subroutine package.

POGO facilities consist of two parts:

1. A DESIGN program for drawing interactive computer graphics displays directly on the face of a CRT, using the Rand Tablet.*
2. A set of routines for interacting with this display.

DESIGN is a complete program that enables a user to draw lines and other figures with the Tablet stylus on the CRT and to type in characters with the typewriter keyboard. Parts of a display can be moved around on the screen or erased. Sensitive areas can be set up and given code numbers. Areas to display variables can be defined, and indicators can be set to determine which variables to display. Finally, all the information can be stored on a magnetic disk device or punched out automatically on cards. The DESIGN program makes extensive use of function keys, which designate the type of figure to be drawn on the screen or the type of action to be taken with the Tablet stylus; e.g., draw boxes or circles, move or delete displays. Table 1 describes the DESIGN functions available.

Interaction with the displays created by DESIGN is accomplished by executing a program written in FORTRAN IV with calls to special POGO routines. These routines perform several basic functions which the user would previously have had to recode for every new program. Table 2 presents POGO execution functions. Most users in a CAI situation, however, need not be exposed to even this level of programming. That is, a team of course development experts can create a standard CAI course by using POGO DESIGN facilities in combination with some simplified FORTRAN programming. The user-instructor may then modify this package through the use of the DESIGN facilities without resorting to any computer programming.

POGO seems to offer a solution to the graphical input limitations of current CAI systems. The user may enter text and graphic information directly onto the CRT without resorting to layout pages and tedious,

---

*A graphic input device that allows a computer user to communicate with a computer by drawing freehand on an electronic "tablet" surface with a special stylus.

Table 1

DESIGN FUNCTIONS

Function	Description	Purpose	User Action: Keyboard	User Action: CRT
Boxes	Plain	Display; sensitive area	B	Indicate 2 points on screen
Boxes	Fancy (center dot)	Display; sensitive area	X	Indicate 2 points on screen
Boxes	Invisible	Sensitive non-box displays	I	Indicate 2 points on screen
Lines	Horizontal	Display	H	Indicate 2 points on screen
Lines	Vertical	Display	V	Indicate 2 points on screen
Lines	JOINED (polygon)	Display	J	Indicate corner points
Circle		Display	C	Indicate 2 points (radius)
Characters	Small	Display	S	Enter from keyboard
Characters	Large	Display	L	Enter from keyboard
Codes	5-digit numbers	Identify sensitive areas	W	Enter with stylus
Values	Location	Identify place to display a variable	N	Indicate point on screen
Erase		Erase entire screen	E	
Delete		Erase specific display	D	Indicate display with stylus
Move		Move displays on screen	M	Touch display and move to new position with stylus
Touch up		To change displayed characters	T	Write over with stylus
Reset		Permits current display to remain on screen as a ghost (underlay display)	R	
File Control	Index	Provides user with index to displays already stored and a means of storing current displays on-line	F	Indicate display to recall/store
Punch		Punch out on cards display on screen	P	Enter name of display from keyboard
Kill		Terminate DESIGN	K	

Table 2

EXECUTE FUNCTIONS

Function	Action
Open	Opens a display file for use by the program.
Recall	Using data in file recreates desired display on screen, setting up sensitive areas and displaying values of selected variables.
Action	Registers interrupts from any of the graphic devices.
Check	Moves a check mark to any designated box of a set of boxes.
Saval	Tests values of variables on the screen to see if they have been changed since the last such test.

time-consuming reprogramming for each type of information to be displayed. The following section describes the instructional area of application chosen for testing these POGO facilities in a CAI situation.

TROUBLESHOOTING: INSTRUCTIONAL AREA OF APPLICATION

To evaluate POGO's facilities for CAI application, the area malfunction diagnosis, or "troubleshooting," was chosen. Troubleshooting was selected because it is not being taught effectively by current methods, and might benefit from the application of interactive computer graphics.

The traditional classroom methods used to teach troubleshooting are lectures and workbooks. Few mock-ups or simulators are available because of their high cost. If a malfunctioning plane happens to be in the vicinity of the classroom, an instructor might use the craft to demonstrate the troubleshooting process. The result is that few airmen get any practice applying procedures and concepts they have discussed in class until they are on the job. In the training context, conceptual knowledge is insufficient. Practice in applying these concepts to specific problem situations is essential [8].

Because a CAI system can be used to simulate problem situations and can be time-shared* by many students simultaneously, it might be used effectively for troubleshooting practice. A study of the feasibility of teaching troubleshooting using electronic equipment on-line** with a CAI system has been conducted by the Navy [9]. Their goal was to teach decision behaviors involved in effective troubleshooting by using a troubleshooting "problem tree," a network of contingent action-choices. For a typical path through the problem-tree, the instructor had to code 200 to 300 lines of instructions in the author language, COURSEWRITER. Further compounding the problem of program complexity was the addition of on-line equipment to the instructional loop (student-computer-equipment). In spite of these obstacles, two gratifying conclusions were reached.

1. Computer-guided practice in following trouble isolation sequences can facilitate effective troubleshooting performance; even a few hours of such practice show performance improvement.
2. Research on the application of CAI to troubleshooting should have fairly immediate practical significance.

A final recommendation was to do away with the physical presence of the equipment altogether. If a video display of the basic circuit schematic were provided, the student could then synthetically change component values to see what would happen to the circuit; or he could follow alternative paths through a troubleshooting tree, observing the effects immediately. As mentioned in Sec. II, this alternative CAI configuration could result in considerable savings in the capital cost of training hardware. Furthermore, it would eliminate the problem of keeping equipment in a good state of repair and peaked-up for successive students; it would permit simulation of faults that normally damage equipment.

---

*The use of a device for two or more purposes during the same overall time interval.

**Pertaining to equipment or devices under direct control of the central processing unit of the computer.

In addition to cost saving factors, simulation of the troubleshooting process utilizing schematics would force airmen to become familiar with and use aircraft diagrams. The Air Force emphasizes the use of schematics as an aid to malfunction diagnosis, but apparently repairmen rarely use the available schematics on the job. This guessing approach can be time-consuming and expensive. With graphic displays, the airman can be taught a methodological approach dependent upon aircraft schematics.

With the Air Force's requirements and the Navy's results and recommendations in mind, a CAI course was developed to simulate the troubleshooting process with interactive displays created easily and rapidly with POGO.

III. COMPUTER-AIDED TRAINING IN TROUBLESHOOTING

INTRODUCTION

Computer-Aided Training in Troubleshooting (CATTS) is the system developed with POGO to create and exercise troubleshooting problems easily. CATTS is based on the requirements and procedures the Air Force set forth as those contributing to expertise in corrective maintenance [10]. To become a troubleshooting expert, an airman not only must have a thorough understanding of aircraft systems but also must be able to do the following:

1. Study and understand schematic diagrams of the systems.
2. Know and use pertinent technical orders.
3. Follow the basic steps of the logical troubleshooting sequence.

Schematics play an essential part in the troubleshooting process because it is easier and more efficient to trace out a system on a schematic than to try to follow it through on the aircraft. Furthermore, because system configurations on different aircraft may vary, the airman should check the aircraft's schematics for its particular layout.

Technical orders include schematics as well as the many flight forms that record the aircraft's history. The flight manual is available on the aircraft and may provide information that will help locate the defect. Therefore, the airman should know the purpose of each flight form and the meaning of the information recorded on it.

The Air Force emphasizes the use of a job plan for attacking troubleshooting problems. This plan consists of six basic steps that incorporate the use of schematics and technical orders to diagnose a malfunction. CATTS, designed to follow these recommended procedures, permits the student to devise a plan and follow it during the EXECUTE phase. The instructor provides the problem information to the program during a DESIGN phase.

1. DESIGN (author mode): An application of POGO's design program for drawing interactive computer graphic displays directly on the face of a CRT, using the Rand Tablet. It includes a special file of standard displays used to create new troubleshooting problems.
2. EXECUTE (student mode): A program allowing the user to interact with newly created displays (troubleshooting problems), written in FORTRAN IV with calls to POGO EXECUTION routines.

Because CATTS relies so heavily on POGO, it might seem more efficient to use POGO directly. But this approach requires recreation of basic displays and repetitious programming for each problem. CATTS eliminates these unnecessary steps by supplying the standard displays and the program for logically troubleshooting a malfunctioning aircraft system.

The student mode of CATTS is described next to demonstrate the system's problem presentation and execution capability. This discussion is followed by a description of the author mode or problem creation stage.

EXECUTE: STUDENT MODE

CATTS in the student mode gives the airman an opportunity to practice the troubleshooting procedures specified by the Air Force in CDC-42152-04-0269 on a number of problems. In addition to becoming familiar with the procedure, CATTS aids the student in developing relationships between previously learned knowledge such as the location, construction and operating characteristics of all components in the system and a class of problems.

Instructions for Use

To use EXECUTE, the student should be aware of the prerequisites required of him, his objective in executing the program, the types of actions required of him, and the types of responses to his actions that he may expect to receive.

Some previous knowledge of the system to be studied is required. The student must know what units comprise the system, and the location of all the units within the system. He should also know something about their construction and operating characteristics, and how the operation of one component affects the operation of another. Finally, he should know how the system functions normally (correct gage readings, etc.). CATTS is not designed to teach the student about a system, the purpose of CATTS is to teach him how to bring together pieces of knowledge about a system into an effective and efficient troubleshooting process, producing the specific result of malfunction diagnosis.

The student's goal is to learn efficient and effective malfunction diagnosis through study and practice. In using CATTS as a means to this end, his immediate objective is to maximize his score, while minimizing problem-solving time. He will be penalized, however, for taking a "hit and miss" approach. In this way, the program tests his ability to recognize malfunctions, analyze them, and determine specific causes.

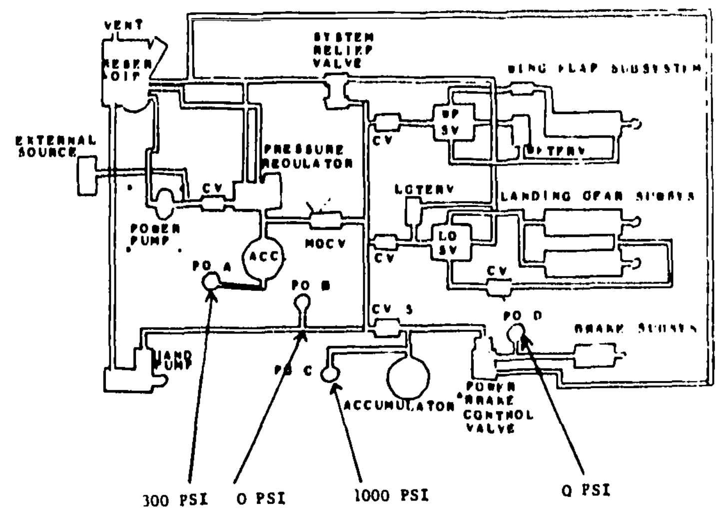
The student proceeds both passively and actively through the problem. Passive participation involves reading displayed information. For example, the Trouble display (Fig. 6, page 18) requires the student to read the problem on an AFTO 781A form, just as he would on the job. The study step also provides him with information to [illegible].

Most steps, however, require the student to make selections with a stylus. For displays with multiple selection lists (Fig. 7, page 20), this simply means touching the box preceding the alternative selected. For displays with schematics or forms requiring action (Fig. 9, page 21), the student touches the stylus to the component or area selected.

After taking an action, the student may expect to see an immediate response on the screen, either in text form (identified by ***) or in graphic form (identified by arrows). The graphic form is used specifically to present gage readings (Fig. 1). Depending on the instructor's use of the program, the text response may present a simple Yes or No (Fig. 2), a corrective statement (Fig. 3), or a helpful hint (Fig. 4).

* SYSTEM INSPECTION *

FINALLY YOU MUST VERIFY THAT THE SYSTEM IS MALFUNCTIONING AS REPORTED ON THE SCHEMATIC. ACTIVATE THE COMPONENT(S) THAT WILL DEMONSTRATE THE CONDITION AND DETERMINE IF IT EXISTS. WHEN YOU WANT A SYSTEM RESPONSE TO YOUR ACTIONS TOUCH [RESPONSE]



WHEN FINISHED TOUCH CONTINUE

| QUIT | RETURN | BACK-UP | CONTINUE |

Fig. 1 — Gage reading response

* REMEDIES *

NOW THAT YOU HAVE FOUND THE DEFECTIVE COMPONENT, HOW WOULD YOU REPAIR IT? WHEN FINISHED TOUCH [DONE]

- [·] REPLENISH THE COMPONENT.
- [X] REPLACE THE COMPONENT.
- [·] CALL AN ELECTRICIAN.
- [·] REPAIR AND RETURN THE COMPONENT.
- [·] REPOSITION THE COMPONENT.

*** THAT'S RIGHT.

| QUIT | RETURN | BACK-UP | CONTINUE |

Fig. 2 — Standard response

* SYSTEM INSPECTION *

BEFORE DIAGNOSING A MALFUNCTIONING SYSTEM, IT IS NECESSARY TO PERFORM THE FOLLOWING IN ORDER

1) A VISUAL INSPECTION,
2) AN OPERATIONAL INSPECTION,
3) A CONDITION VERIFICATION

WHICH OF THE FOLLOWING WOULD YOU VISUALLY INSPECT?

WHEN FINISHED TOUCH [DONE]

- [X] RESERVOIR FLUID LEVEL
- [·] INTERNAL LEAKAGE
- [X] MECHANICAL LINKAGES
- [X] ACCUMULATOR GAGE
- [X] ELECTRICAL CONNECTIONS
- [●] SYSTEM RELIEF VALVE
- [·] EXTERNAL LEAKAGE
- [·] POWER PUMP

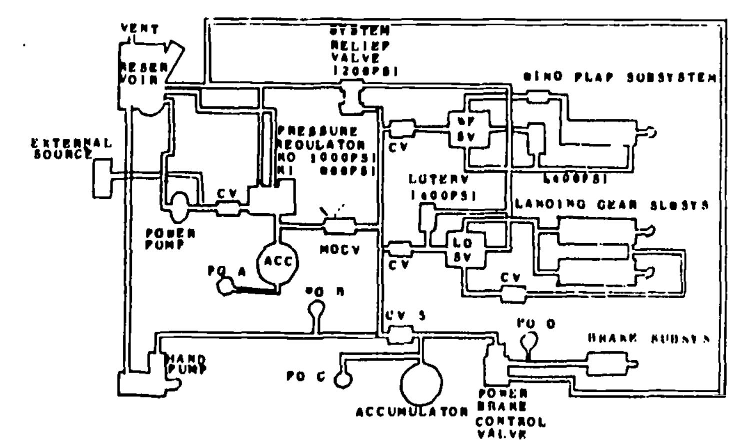
*** NO. SYSTEM RELIEF VALVE DEFECTS CANNOT BE DIRECTLY DETERMINED BY VISUAL INSPECTION.

| QUIT | RETURN | BACK-UP | CONTINUE |

Fig. 3 — Instructor-stored response

* PROBABLE CAUSES *

INDICATE ON THE SCHEMATIC WHICH COMPONENTS, IF DEFECTIVE COULD CAUSE THE TROUBLE. WHEN FINISHED TOUCH [DONE]



*** NO, YOU'RE GUESSING. THE COMPONENTS, IF DEFECTIVE, THAT CAN CAUSE A ZERO PRESSURE INDICATION ON THE SYSTEM PRESSURE GAGE ARE THE POWER PUMP, THE PRESSURE REGULATOR, AND THE SYSTEM PRESSURE GAGE. LET'S GO ON.

| QUIT | RETURN | BACK-UP | CONTINUE |

Fig. 4 — Response with hint

System Description

EXECUTE presents troubleshooting problems to the student for practice organized according to standard troubleshooting procedures. These steps are initiated and executed using a control page and control boxes. Although the troubleshooting steps are initially presented in order, the student may branch to any step he chooses. At the completion of the exercise, CATTS evaluates the student's problem-solving ability through a score analysis. Each step and its implementation in CATTS is described in the following paragraphs and illustrated by the appropriate displays.

Control Page and Boxes. The first display to appear on the screen during execution is the control page (Fig. 5), which allows the student to branch among the various troubleshooting steps. The normal sequence begins with Trouble and proceeds clockwise around the circle. The student may return to the control page at any time, however, and branch to another step. This flexibility simulates the same freedom the airman has on the job.

Within steps, though, the procedure is more structured because various actions within a step are usually contingent upon other actions and results. For example, CATTS will not permit the student to skip the checking actions and, by guessing, specify the cause of the malfunction within the Checking and Analysis step. But if the airman wishes to examine or reexamine the flight manual forms during the checking step, he may do so by branching back to that step. Actually both types of freedom are available on the job, but the first, the "hit and miss" method of problem-solving, leads to inefficient troubleshooting and has been excluded as an acceptable approach.

Control boxes in each display provide the user with the means to return to the control page, to branch among displays within the step, or to quit. Located at the bottom of each display, these boxes are labeled QUIT, RETURN, BACK-UP, CONTINUE. The CONTINUE box is used to proceed directly through the problem. Two other control boxes, RESPONSE and DONE, appear in some displays. Their use is explained in the displays in which they appear.

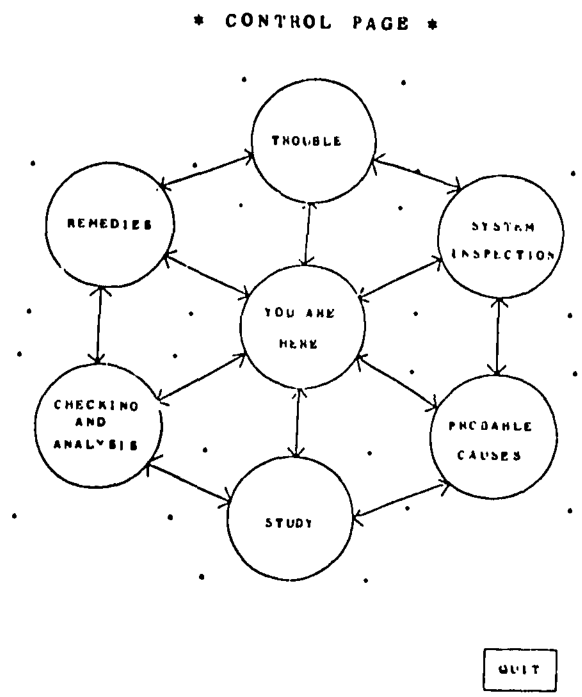


Fig. 5 — Control page

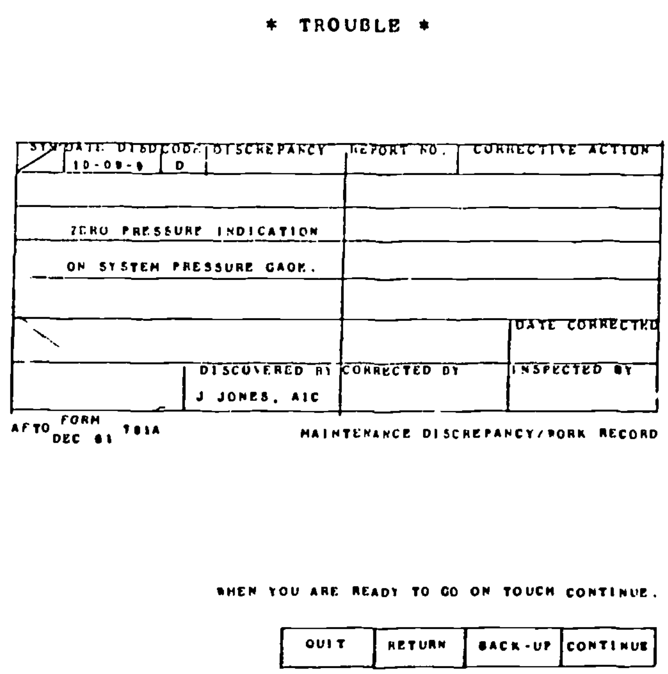


Fig. 6 — Trouble

Trouble. On the job an airman is officially notified of a malfunction by receipt of an AFTO 781A form that discusses the trouble. CATTS similarly presents the malfunction by displaying an AFTO 781A (Fig. 6). The airman is expected to understand the information (statements, codes, numbers) on the form.

System Inspection. System inspection involves the following substeps:

1. Visual inspection of the system for any obvious defects (Fig. 7).
2. Operational inspection to determine if the system is operating as it should expect for the malfunction (Fig. 8).
3. Verification that the malfunction exists as it is reported (Fig. 9).

By means of multiple selection questions, CATTS tests the student on his knowledge of which system components to check visually and what to look or listen for in an operational check.

To verify the malfunction, however, the student must interact with the schematic displayed. He uses the light pen to indicate which component(s) must be activated to verify the condition and then interprets the system response that follows. For example, to verify that a gage has malfunctioned, he must activate (touch) the power pump and note the gage responses. This interaction is meant to simulate on-the-job actions, which require reading a diagram, activating components, and interpreting system responses.

Probable Causes. On the job, an airman must be able to determine which system components can cause the given symptom(s). This requires an understanding of how the operation of one component affects the operation of another. The repairman is expected to use schematics to clarify these relationships. CATTS displays a schematic of the malfunctioning system and directs the student to touch the components that are probable causes (Fig. 10). CATTS reacts to the student's actions by displaying responses, either standard or instructor-supplied.

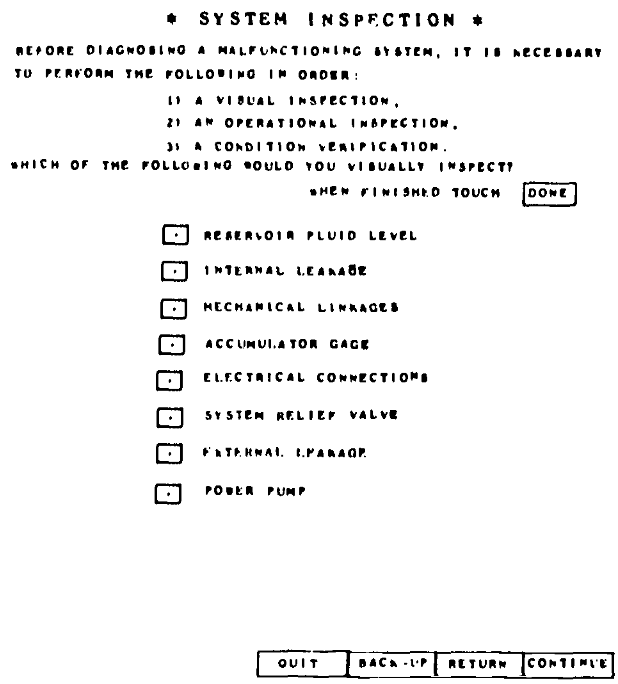


Fig. 7 — System inspection: visual

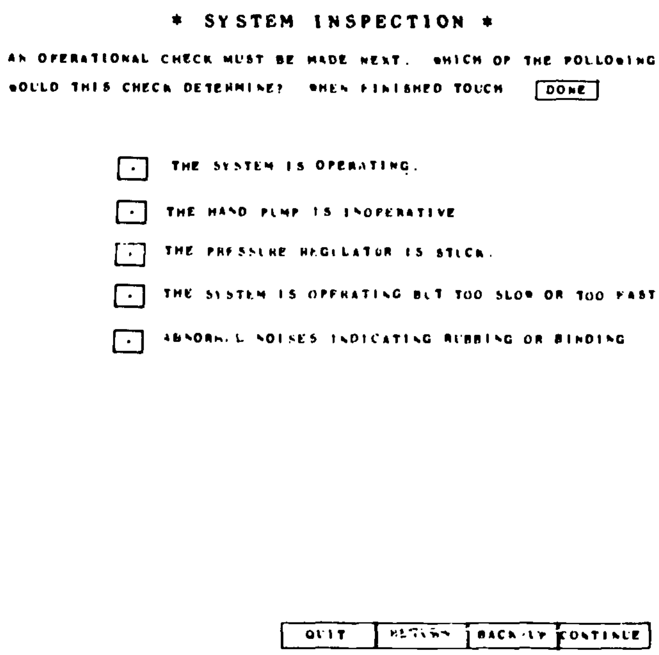


Fig. 8 — System inspection: operational

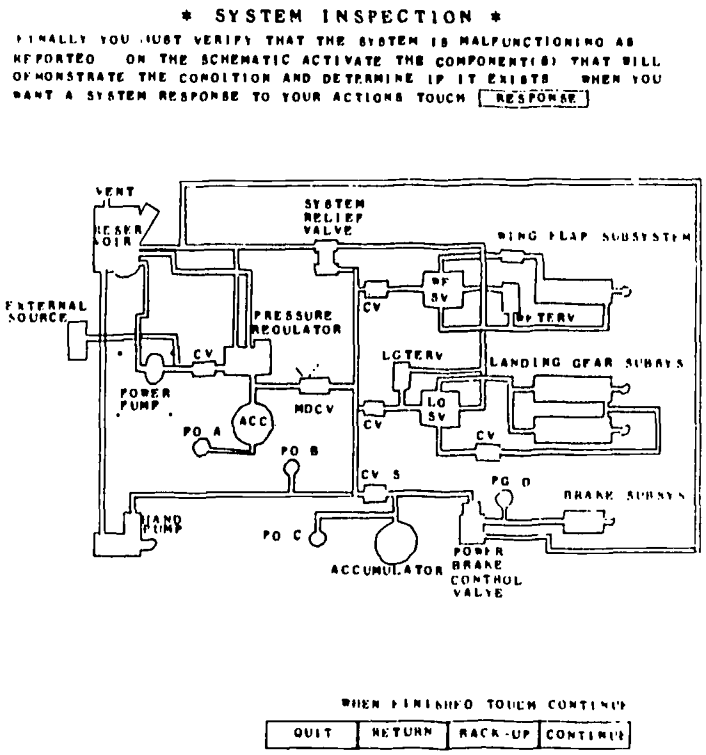


Fig. 9 — System inspection: verification

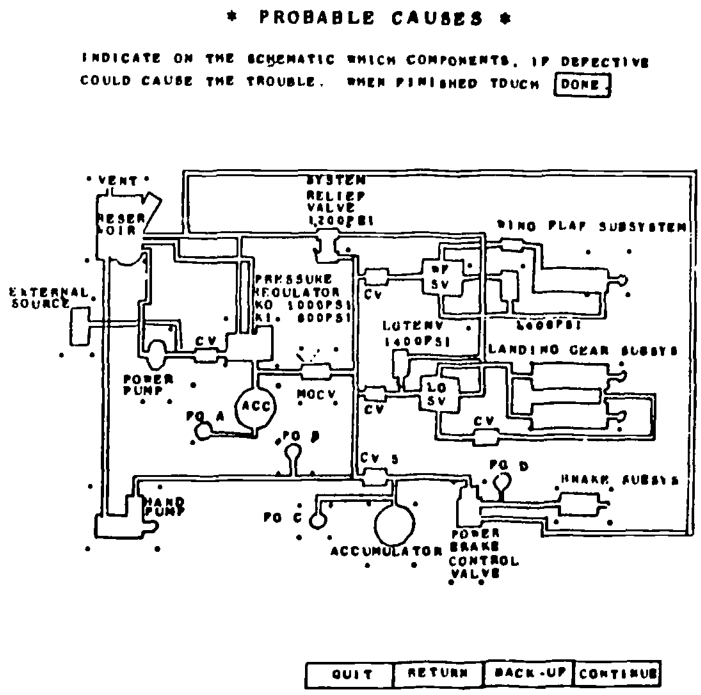


Fig. 10 — Probable causes

Study. Because there are usually a number of probable causes for any given malfunction, it is important to examine the aircraft's past history as recorded in the flight manual to determine if there is any information that may pinpoint the cause or reduce the number of possible causes. In reading the forms, for example, the airman would look for answers to the following questions:

1. Do the 781As show any inspection or maintenance overdue?
2. Has maintenance been performed recently on any unit, or in such close proximity that a unit had to be removed for easier access? (Someone not thoroughly familiar with the equipment may have unknowingly disturbed a critical adjustment.)
3. Has any particular unit given repeated trouble on this aircraft? Have there been similar malfunctions on other aircraft that would establish a pattern?

CATTS displays a list of available forms from which the student may select one or several that he wishes to examine (Fig. 11). CATTS then displays the form(s) for the student to read (Figs. 12-14).

Checking and Analysis. Perhaps the answer to one of the above questions will indicate the optimum way to check out the probable causes. If examination of the flight manual did not provide pertinent information, all checkout steps should be ordered to minimize time and cost.

In CATTS the student checks out the probable causes by activating (touching) components in the schematic displayed in their proper sequence, in the search mode (Fig. 15). When the sequence for a particular probable cause has been completed, the student requests a system reaction; noting the response, he then selects the next probable cause to check. This procedure continues until the defective component is located, in the solution mode (Fig. 16).

Remedies. When the defective part has been found, the airman must know how to repair it. After making the repair, he must record in the

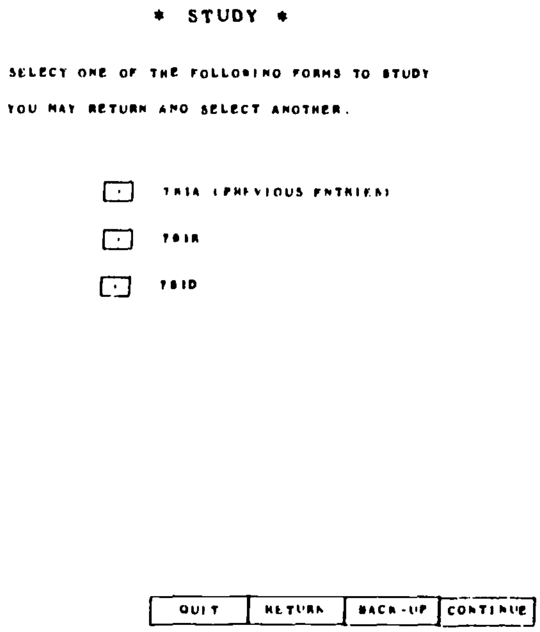


Fig. 11 — Forms available for study

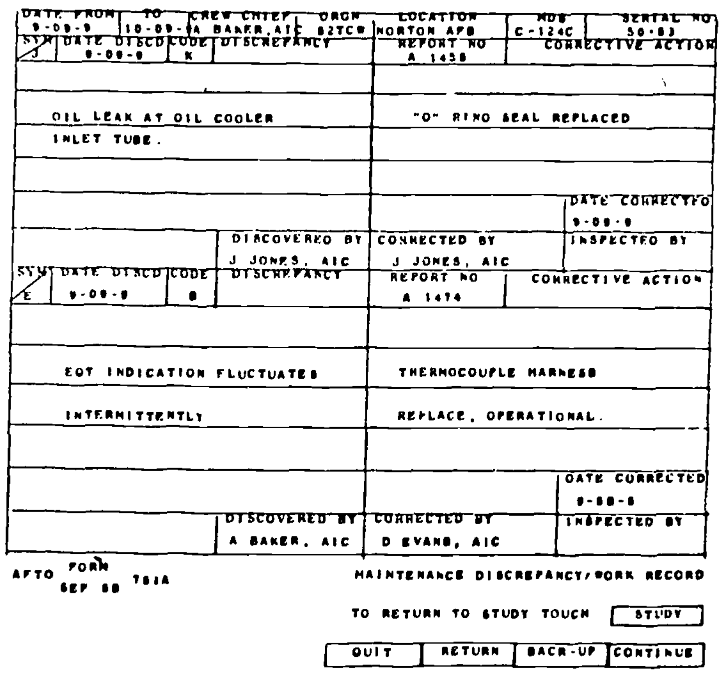


Fig. 12 - AFTO form 781A

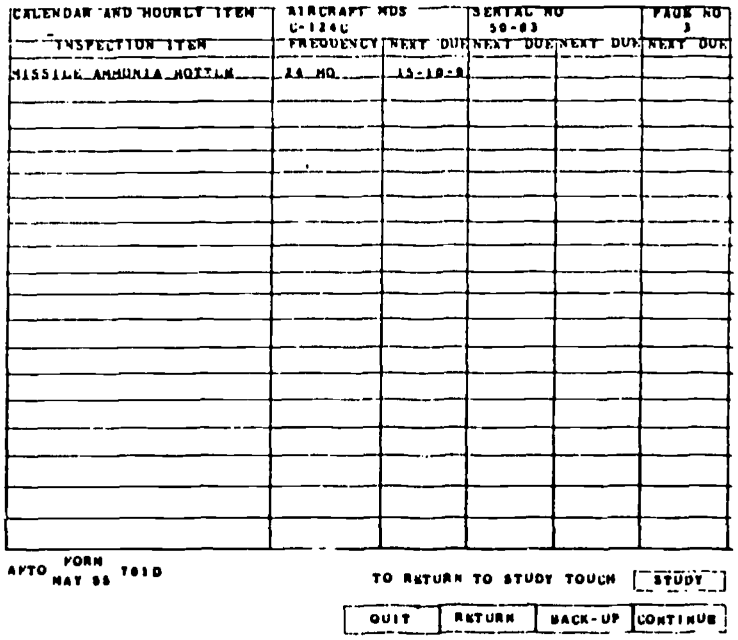


Fig. 13 — AFTO form 781D

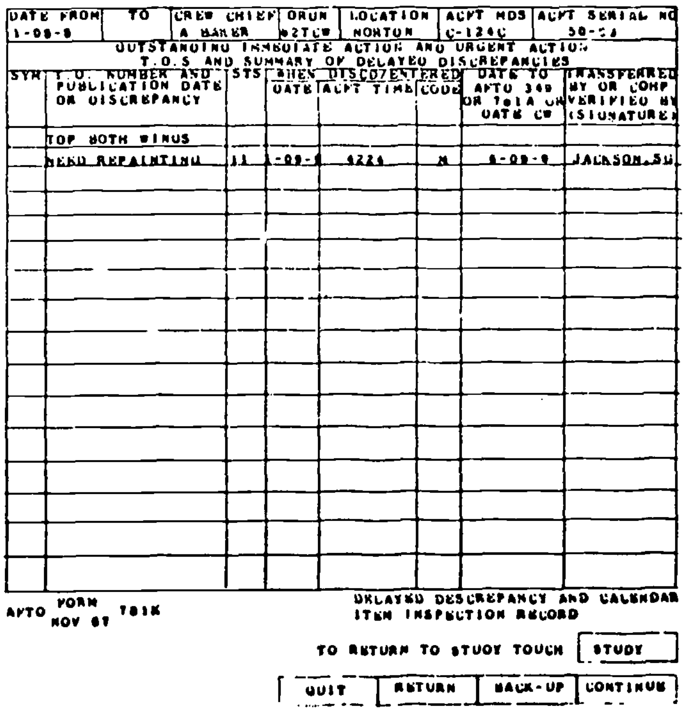


Fig. 14 — AFTO form 781K

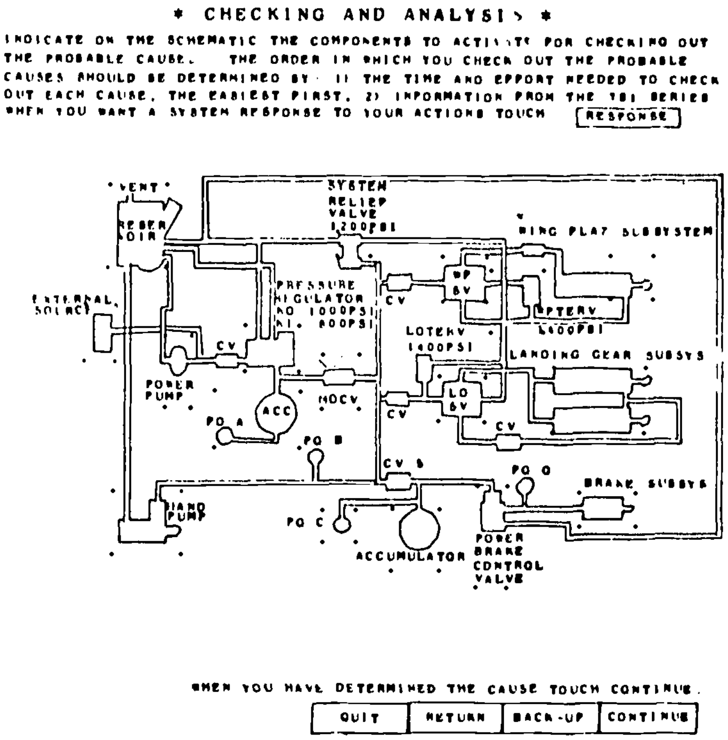


Fig. 15 — Checking and analysis: search

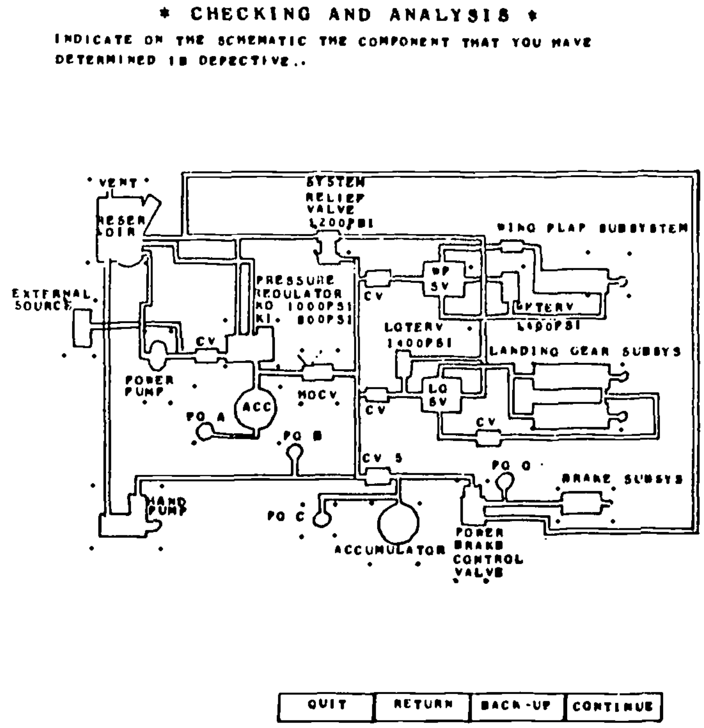


Fig. 16 — Checking and analysis: solution

flight manual the corrective action taken. A plane is not allowed off the ground until this last step is completed correctly. CATTS displays a list of possible remedies and from these the student selects one (or more) that is appropriate for the defect (Fig. 17). Finally he indicates with the light pen how he would complete AFTO 781A on which the corrective action and other vital information must be recorded (Fig. 18).

Score Analysis. To aid the student in pinpointing his weaknesses in handling troubleshooting problems, CATTS grades him on each step and analyzes the results (Figs. 19-20). For example, a low score on Probable Causes would indicate the need to review causes of basic malfunctions on the type of system under study. A high score on Checking but a low score on Analysis (sequencing) would indicate that the student knows the components that must be activated to find the cause of a particular malfunction, but does not know the order in which they should be activated to minimize time and cost. Hopefully, by replaying the problem, he will discover the optimum path.

DESIGN: AUTHOR MODE

This section describes the operation of DESIGN, a system for creating troubleshooting problems. Emphasis in development was on minimizing required instructor time and effort. Specifically, the instructor is relieved of tedious and lengthy graphic coordinate transcriptions and of learning a computer language.

In the DESIGN system, the instructor uses standard displays, modifying them on-line to create a new problem. Modification involves typing in text, entering codes, and tracing in schematics. Inputs specify the problem and its solution; therefore, the instructor must understand the troubleshooting process and be able to define his problem and its solution within this framework.

Function Keys

CATTS' DESIGN program makes extensive use of POGO function keys, which permit the user to designate the type of figure to be drawn on

* REMEDIES *

NOW THAT YOU HAVE FOUND THE DEFECTIVE COMPONENT, HOW WOULD YOU REPAIR IT? WHEN FINISHED TOUCH [DONE]

- [·] REPLENISH THE COMPONENT.
- [·] REPLACE THE COMPONENT.
- [·] CALL AN ELECTRICIAN.
- [·] REPAIR AND RETURN THE COMPONENT
- [·] REPOSITION THE COMPONENT.

QUIT	RETURN	BACK-UP	CONTINUE

Fig. 17 — Remedies: selection

* REMEDIES *

AFTER REPAIRING THE DEFECTIVE COMPONENT YOU MUST COMPLETE AFTO 781A BEFORE THE PLANE MAY FLY AGAIN. INDICATE ON THE FORM IN THE FOLLOWING ORDER WHERE YOU WOULD ENTER:

1) THE REMEDY.
2) YOUR NAME AND GRADE.
3) THE DATE.
4) YOUR INITIAL

SYM	DATE DISCD 10-09-9	CODE D	DISCREPANCY	REPORT NO	CORRECTIVE ACTION
ZERO PRESSURE INDICATION					
ON SYSTEM PRESSURE GAGE					
					DATE CORRECTED
	DISCOVERED BY J JONES,A1C			CORRECTED BY	INSPECTED BY

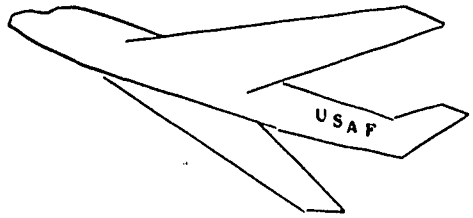
AFTO FORM SEP 68 781A — MAINTENANCE DISCREPANCY/WORK RECORD

WHEN FINISHED TOUCH CONTINUE

QUIT	RETURN	BACK-UP	CONTINUE

Fig. 18 — Remedies: completing the form

YOU HAVE NOW COMPLETED THE TROUBLESHOOTING PROCESS AND YOUR PLANE IS FLYING AGAIN.



YOUR SCORE IS ___________ %

DO YOU WISH

- [ ] AN ANALYSIS OF YOUR RESULTS.
- [ ] TO REPEAT THE SAME PROBLEM.
- [ ] TO TRY ANOTHER PROBLEM ON THE SAME SYSTEM.
- [ ] TO TRY A PROBLEM ON ANOTHER SYSTEM.
- [ ] TO QUIT.

Fig. 19 — Score

* SCORE ANALYSIS *

	POINTS
1. TIME	___________
2. VISUAL INSPECTION	___________
3. OPERATIONAL INSPECTION	___________
4. CONDITION VERIFICATION	___________
5. PROBABLE CAUSES	___________
6. CHECKING	___________
7. ANALYSIS (SEQUENCING)	___________
8. SOLUTION	___________
9. REMEDY	___________
10. FORM	___________
TOTAL	___________ 1000
PERCENT	___________

QUIT	RETURN	BACK-UP

Fig. 20 — Score analysis

the screen, or the type of action to be taken with the Tablet stylus; e.g., the user can draw boxes or circles and move or delete displays, as described in Sec. II.

File Control

A computer-graphics file may contain previously created displays. The CATTS file stores displays basic to the troubleshooting process. From these and instructor-supplied responses and schematics, new problems are created. Each troubleshooting problem has its own file of displays.

In the DESIGN phase the first display to appear on the screen is the first page of an index to the CATTS file. The file has three index pages, each providing space for ten displays (see the Appendix, Figs. A1, A2, and A3). The three boxes directly under the heading FILE CONTROL allow for branching to each page. The RESPONSE DISPLAYS PAGE (Fig. A3), is partially blank and numbered to allow storage of up to seven instructor-supplied-responses.

A display in the file is recalled (redisplayed) by touching the stylus to the box next to the display's description under the heading RECALL. A new display is stored by touching the stylus to a box under FILE CURRENT DISPLAY. A cursor will appear in the description for that item. The user then types in a name for the display and presses the typewriter key labeled "end." When the file operation has been completed, a large X appears in the FILE CURRENT DISPLAY box. To eliminate unused or incorrect displays, the user touches the respective boxes under DELETE ITEM NO. Return from FILE CONTROL to the current display is accomplished by touching RETURN. To return to FILE CONTROL from the current display, the user presses the function key labeled "file control."

For each display listed in the index, users perform the following:

1. Recall the display in sequence.
2. Take action required:

a. None
b. Enter text
c. Enter codes
d. Recall a schematic
e. Combination of (b)-(d).

3. Press "punch display."
   A name for the display is then requested. It is easiest and most logical to give the new display the standard display name appearing in the CATTS index.
4. Press "erase screen."
5. Press "file control."

If instructor responses are to be inserted, the following steps are performed before pressing "erase screen," step 4:

3.1. Press "reset pointers."
3.2. Enter response.
3.3. Press "file control," branching to Response Display Page.
3.4. File the response.
3.5. Return to current display.
3.6. Delete the response.
3.7. Repeat (3.2)-(3.6) for all instructor-supplied responses.

And then,

4. Press "erase screen."
5. Press "file control."

Schematics are entered by the instructor and stored at the end of the file. When a schematic is recalled in combination with a standard display, a new display pertaining to the current problem is created (see Fig. A18 in the appendix). Because the file can hold a maximum of seven instructor-supplied responses, the schematic may take up a needed spot. In this case, after its final use in creating the display Checking and Analysis--Search, the schematic may be deleted from the file, thus releasing its space.

Coding

To detect a student's action with the stylus during execution, areas on the screen must be made sensitive. This is accomplished by enclosing these areas in boxes or invisible boxes and assigning unique five-digit codes to each box. To assign codes, the user must press the function key "insert codes." A line of underscores will appear below the left corner of the box. Using the stylus, the instructor can then write in a five-digit code. CATTS uses two types of sensitive areas requiring instructor-supplied codes: multiple-selection areas and step-sequenced areas.

Coding for Multiple Selections. Multiple-selection alternatives appear in either a list or a schematic. The multiple-selection lists appearing in several standard displays pertain to a pneudraulics system. These lists may be totally or partially deleted and replaced with other alternatives. If the lists are changed, the codes for each selection must also be changed to fit the following format.

First digit:
1. Only the numbers 8 or 9 are acceptable.
2. Insert an 8 if the selection is correct.
3. Insert a 9 if the selection is incorrect.

Second digit:
1. The numbers 1-9 are acceptable.
2. Insert the number of correct selections in the current display; maximum number of correct selections per display is 9.

Third digit:
1. The numbers 1-9 are acceptable.
2. Insert a number that uniquely describes the selection; it is simplest to number right and wrong selections sequentially.

Fourth digit:
1. Insert a 0.

Fifth digit:
1. The numbers 0-7 are acceptable.
2. Inserting a 0 returns a standard response if this selection is chosen.
3. Inserting a positive number returns an instructor-created response; responses must be numbered sequentially as they appear at the end of the standard responses.

Figure 21 shows the codes for the standard display on visual inspection. The first three codes are interpreted as follows.

85100: Right selection, five right selections in list, first right selection, standard response returned.

95100: Wrong selection, five right selections in list, first wrong selection, standard response returned.

85200: Right selection, five right selections in list, second right selection, standard response returned.

In Fig. 22 the instructor has altered the codes to recall his responses when a wrong selection is made. The 8xxxx codes above remain the same, but the 9xxxx code has the following interpretation.

95103: Wrong selection, five right selections in list, first wrong selection, third instructor's response returned.

In this example, the rest of the display remains unchanged.

CATTS also requires the student to make component selections from a schematic, another form of multiple-selection testing. When creating a multiple-selection display with the schematic, the instructor must insert codes for each component. The codes follow the same format described above. For example, the display in Fig. 23 was created as follows:

1. Recalled "Probable Causes" display.
2. Recalled schematic.
3. Inserted codes.
4. Pressed "output display."

The codes for the reservoir and the power pump have the following interpretation.

96100: Wrong component, six right components on page, first wrong, standard response returned.

86200: Right component, six right components on page, second right component, standard response returned.

```
                    * SYSTEM INSPECTION *
BEFORE DIAGNOSING A MALFUNCTIONING SYSTEM, IT IS NECESSARY
TO PERFORM THE FOLLOWING IN ORDER:
              1) A VISUAL INSPECTION,
              2) AN OPERATIONAL INSPECTION,
              3) A CONDITION VERIFICATION
WHICH OF THE FOLLOWING WOULD YOU VISUALLY INSPECT?
                           WHEN FINISHED TOUCH [DONE]
                                               ----55
              [.] RESERVOIR FLUID LEVEL
              #5100
              [.] INTERNAL LEAKAGE
              #5100
              [.] MECHANICAL LINKAGES
              #5200
              [.] ACCUMULATOR GAGE
              #5300
              [.] ELECTRICAL CONNECTIONS
              #5400
              [.] SYSTEM RELIEF VALVE
              #5200
              [.] EXTERNAL LEAKAGE
              #5500
              [.] POWER PUMP
              #5300
              -----

                       [QUIT] [BACK-UP] [RETURN] [CONTINUE]
                       ----51  ----52    ----53   ----54
```

Fig. 21 — Visual inspection display with standard responses coded

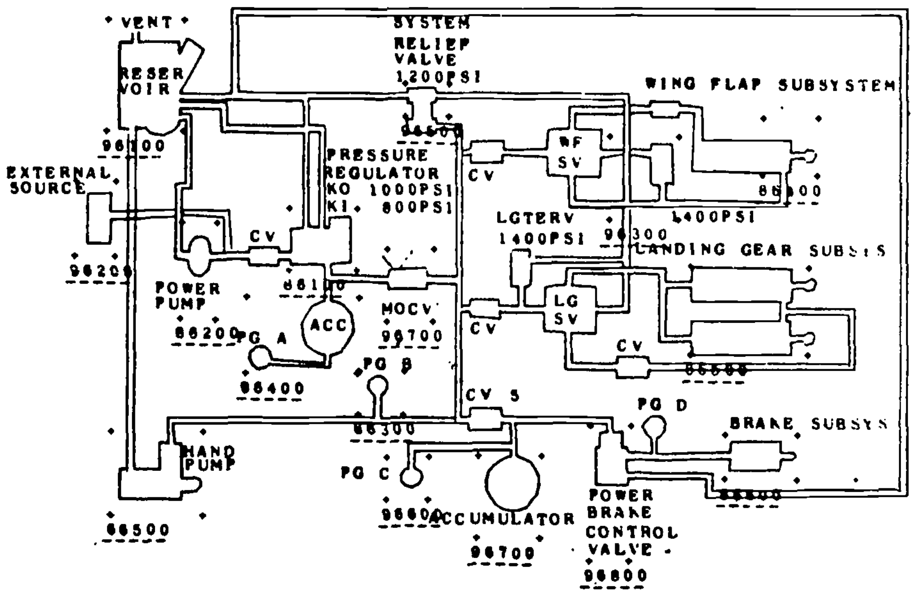
```
                    * SYSTEM INSPECTION *
BEFORE DIAGNOSING A MALFUNCTIONING SYSTEM, IT IS NECESSARY
TO PERFORM THE FOLLOWING IN ORDER:
              1) A VISUAL INSPECTION,
              2) AN OPERATIONAL INSPECTION,
              3) A CONDITION VERIFICATION
WHICH OF THE FOLLOWING WOULD YOU VISUALLY INSPECT?
                           WHEN FINISHED TOUCH [DONE]
                                               ----55
              [.] RESERVOIR FLUID LEVEL
              #5100
              [.] INTERNAL LEAKAGE
              #5103
              [.] MECHANICAL LINKAGES
              #5200
              [.] ACCUMULATOR GAGE
              #5300
              [.] ELECTRICAL CONNECTIONS
              #5400
              [.] SYSTEM RELIEF VALVE
              #5204
              [.] EXTERNAL LEAKAGE
              #5500
              [.] POWER PUMP
              #5305
              -----

                       [QUIT.] [RETURN] [BACK-UP] [CONTINUE]
                       ----51  ----52   ----53    ----54
```

Fig. 22 — Visual inspection display with instruction-stored responses coded

*** PROBABLE CAUSES ***

INDICATE ON THE SCHEMATIC WHICH COMPONENTS, IF DEFECTIVE COULD CAUSE THE TROUBLE. WHEN FINISHED TOUCH DONE

55



QUIT	RETURN	BACK-UP	CONTINUE
51	52	53	54

Fig. 23 — Creation of probable causes display

Coding for Step-Sequencing. A system should be checked so as to minimize cost and time. For any given malfunction there is a step and component sequence that accomplishes this. The instructor inputs this sequence by means of five-digit codes. A component is defined (i.e., coded) by its first use in the correct sequence of steps. The codes have the following format.

First digit:
1. Only the numbers 1-7, 9 are acceptable.
2. The numbers 1-7 indicate the step number.
3. The number 9 indicates that the component is not used in any step.

Second digit:
1. The numbers 1-9 are acceptable.
2. The numbers indicate the component's position within the step.

Third and fourth digits:
1. Numbers 00, 11-79 are acceptable.
2. The numbers specify the first two digits of a component in a previous step that is also necessary for completion of this step. A 00 means no previous component required.

Fifth digit:
1. Numbers 0-7 are acceptable.
2. A 0 should appear in all codes for components not completing a step.
3. The numbers 1-7 give the response number when a step has been completed correctly.

In the example problem for the student (Figs. 5-20), the malfunction is a zero reading on the system pressure gage. The step and component checkout sequence as described in CDC-42152-04-0269 is performed as follows:

Step 1: Check the power pump by connecting an external source of pressure to the aircraft and pressurize the system.
Component 1: External source
Previous component: None

Step 2: Determine whether the malfunction is in the power or the actuating system by setting subsystem selector valves with the external source of pressure on.
Component 1: Wing flap selector valve
Component 2: Landing gear selector valve
Component 3: Power brake control valve
Previous component: External source

Step 3: If the trouble is in the power system, check the pressure regulator by setting the MOCV and using the hand pump.
Component 1: MOCV
Component 2: Hand pump
Previous component: None

Figure 24 shows how the components in the above steps were coded. All components not used in the steps are given 9xxxx codes. The active components are coded as follows:

External source:	11001:	Step 1, component 1, no previous components necessary, step complete--Response 1.
Wing flat SV:	21110:	Step 2, component 1, external source necessary, step incomplete.
Landing gear SV:	22110:	Step 2, component 2, external source necessary, step incomplete.
Power brake CV:	23111:	Step 2, component 3, external source necessary, step complete--Response 1.
MOCV:	31000:	Step 3, component 1, no previous component necessary, step incomplete.
Hand pump:	32002:	Step 3, component 2, no previous component necessary, step complete--Response 2.

Problem Design and Example

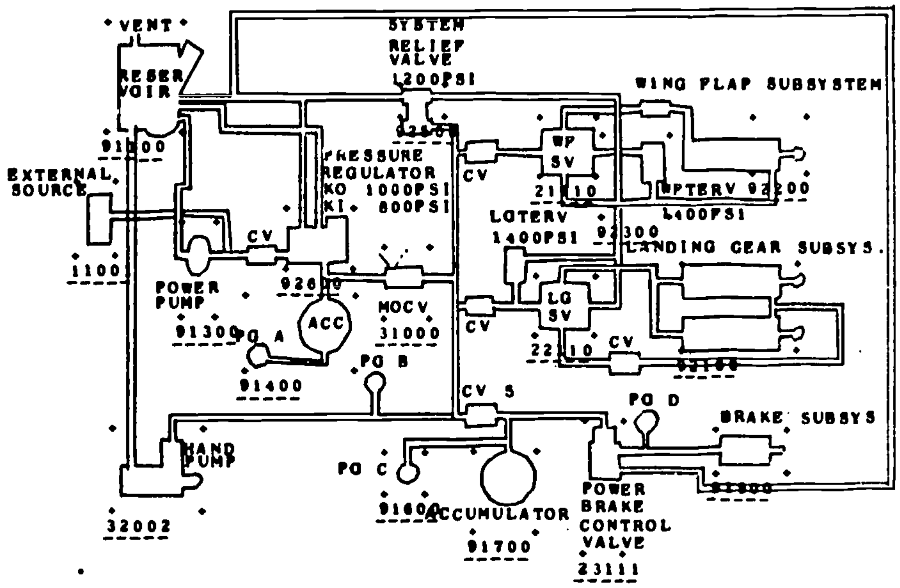
New problems are created from standard displays provided by CATTS and instructor-supplied displays. The appendix details the purpose of each standard display and the required instructor actions. To illustrate the use of each display, the pneudraulics system problem, presented earlier in this section, is created.

File Creation

After all displays for a problem are punched out on cards in the correct sequence, the cards are read in and stored on disk in a new file. It is then recommended that each display in the new file be checked for errors. This is accomplished by entering the DESIGN phase with the new file. A file control page will appear on the screen, listing the names of all the displays the instructor has just stored. The

* CHECKING AND ANALYSIS *

INDICATE ON THE SCHEMATIC THE COMPONENTS TO ACTIVATE FOR CHECKING OUT THE PROBABLE CAUSES. THE ORDER IN WHICH YOU CHECK OUT THE PROBABLE CAUSES SHOULD BE DETERMINED BY: 1) THE TIME AND EFFORT NEEDED TO CHECK OUT EACH CAUSE, THE EASIEST FIRST; 2) INFORMATION FROM THE 761 SERIES. WHEN YOU WANT A SYSTEM RESPONSE TO YOUR ACTIONS TOUCH [RESPONSE] 56.



WHEN YOU HAVE DETERMINED THE CAUSE TOUCH CONTINUE.

QUIT	RETURN	BACK-UP	CONTINUE
51	52	53	54

Fig. 24 — Checking and analysis: coding the components

file control index for the example problem created is given in Figs. 25, 26, and 27. Each display can then be recalled and examined. Corrections may be made with the keyboard and function keys after which the corrected display can be refiled.

Problem Changes and Variations

To change previously created problems or to create new problems from old ones, the instructor must use the problem file in the DESIGN phase rather than the CATTS standard file. Changing a problem erases the original problem from the file. This would be desirable if students had mastered the original problem, if they had copied solutions and passed them on to other students, or if the instructor had found the original problem too easy. A problem may be changed in several basic ways:

1. Change the "trouble."
2. Change the information on the flight manual forms to give pertinent rather than nonpertinent information or vice versa.
3. Change the cause of the trouble (for any given malfunction there are generally a number of possible causes).
4. Change the multiple-selection alternatives.
5. Any combination of (1)-(4).

Making a basic change generally requires making other changes in the original problem. The necessary changes for each of the above possibilities are listed below:

1. Trouble
    a. Change "trouble" display.
    b. Change codes on "condition verification" display.
    c. Change system response to condition verification.
    d. Change codes on "probable causes."
    e. Change codes for Checking and Analysis--Search to new correct sequence.
    f. Change system responses to checking steps.
    g. Change codes for Checking and Analysis--Solution to give new correct component.
    h. If necessary, change codes in Remedies to give correct remedy.

FILE CONTROL

STANDARD DISPLAYS 1-10	STANDARD DISPLAYS 11-20	RESPONSE DISPLAYS PAGE

RECALL	ITEM	DESCRIPTION	FILE CURRENT DISPLAY AS ITEM NO.	DELETE ITEM NO.
⊡	1	CONTROL PAGE	⊡	⊡
⊡	2	TROUBLE	⊡	⊡
⊡	3	SYSTEM INSPEC - VISUAL	⊡	⊡
⊡	4	SYSTEM INSPEC - OPERATION	⊡	⊡
⊡	5	CONDITION VERIFICATION	⊡	⊡
⊡	6	PROBABLE CAUSES	⊡	⊡
⊡	7	STUDY	⊡	⊡
⊡	8	CHK & ANAL	⊡	⊡
⊡	9	SOLUTION	⊡	⊡
⊡	10	REMEDIES	⊡	⊡

RETURN

Fig. 25 — File control, 1-10

FILE CONTROL

STANDARD DISPLAYS 1-10	STANDARD DISPLAYS 11-20	RESPONSE DISPLAYS PAGE

RECALL	ITEM	DESCRIPTION	FILE CURRENT DISPLAY AS ITEM NO	DELETE ITEM NO.
⊡	11	FORM	⊡	⊡
⊡	12	FINISH	⊡	⊡
⊡	13	SCORE ANALYSIS	⊡	⊡
⊡	14	781A	⊡	⊡
⊡	15	781D	⊡	⊡
⊡	16	781K	⊡	⊡
⊡	17	STD RESP 1	⊡	⊡
⊡	18	STD RESP 2	⊡	⊡
⊡	19	STD RESP 3	⊡	⊡
⊡	20	STD RESP 4	⊡	⊡

RETURN

Fig. 26 — File control, 11-20

FILE CONTROL

STANDARD DISPLAYS 1-10	STANDARD DISPLAYS 11-20	RESPONSE DISPLAYS PAGE

ECALL	ITEM	DESCRIPTION	FILE CURRENT DISPLAY AS ITEM NO.	DELETE ITEM NO
☐	21	STD RESP 5	☐	☐
☐	22	STD RESP 6	☐	☐
☐	23	PROBABLE CAUSES RESPONSE	☐	☐
☐	1	RESPONSE 1	☐	☐
☐	2	RESPONSE 2	☐	☐
☐	3	RESPONSE 3	☐	☐
☐	4	RESPONSE 4	☐	☐
☐	5	RESPONSE 5	☐	☐
☐	6	RESPONSE 6	☐	☐
☐	7	RESPONSE 7	☐	☐

RETURN

Fig. 27 — File control, response display page

   i. Enter new trouble on Form.
   j. If desired, change information on flight manual forms.

2. Flight manual forms
   a. Enter new information on appropriate form.
   b. Change codes for Checking and Analysis--Search (#8) to give new correct sequence of steps based on the new information.

3. Cause
   a. Change codes in Checking and Analysis--Solution (#9) to give new desired defective component as cause.
   b. Change system responses to Checking and Analysis--Search (#8) to reflect new cause.
   c. If necessary, change codes in Remedies--Repair (#10) to give new correct remedy.

4. Multiple-selection alternatives
   a. Insert new selections, deleting old if desired.
   b. Update codes to give new correct format.

Each display that is changed must be refiled in the same position as the original. The new display may be given the same name as the old one. Frequent minor changes can keep students actively participating. When confronted with the same question time and again, some students may memorize the position of the correct answer rather than the answer itself. For this reason, it is suggested that the multiple-selection lists be changed or at least rearranged frequently.

It is desirable to create a new problem from an old one when the same system (schematic) is to be used. This eliminates the need to trace in the schematic again and overlay it with the standard displays. The same changes as described above can be made to create a new problem. The difference is that the displays are punched out rather than refiled in the same file. The steps are

1. Recall display.
2. Make changes.
3. Press "punch display."
4. Press "erase screen."
5. Press "file control."
6. Repeat (1)-(5) for each display in sequence.

Finally, a new file must be set up for the new problem.

New standard files may also be created from the CATTS file. If several troubleshooting problems are to be created for a system that defies visual inspection, for example, the visual inspection display (#3) may be recalled and altered except for control boxes. In fact,

1. Any multiple-selection display may be replaced by another multiple-selection display.
2. Any step-sequence display may be replaced by another step-sequence display.
3. Any form display may be replaced by another form display (except #11).

Creating a new standard file involves 23 standard displays. Starting with Item 1 and continuing through Item 23, *in sequence*, several steps are required:

1. Recall display:
   go to (2) if the standard display is to be altered; otherwise go to (4).
2. Delete any or all of display except control boxes.
3. Create new display of correct type for position in file.
4. Press "punch display."
5. Press "erase screen."
6. Press "file control."
   Go to (1).

After all 23 standard displays have been punched out, a new file must be set up.

Table 3 describes the type of display for each position in the file. Displays with type "xxx" may not be changed.

Table 3

DISPLAY TYPES

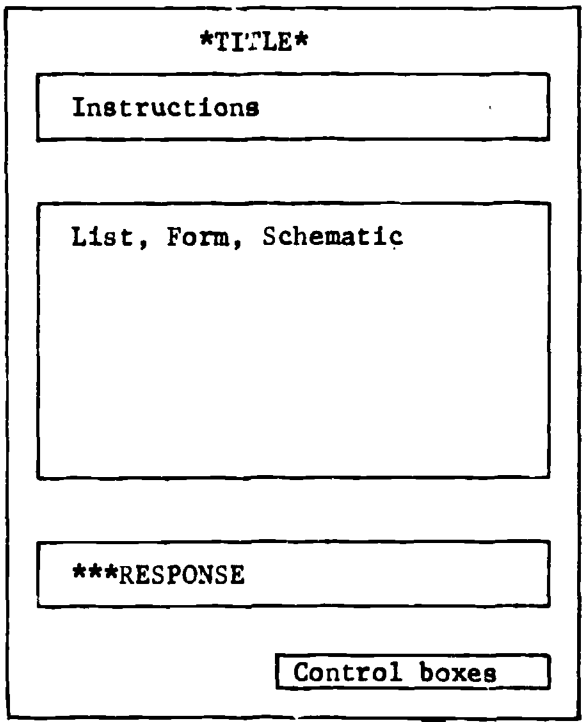
Position	Type	Current Display Available
1	xxx	Control
2	Form	Trouble
3	MS[a]	System Inspection--Visual
4	MS	System Inspection--Operational
5	SS[b]	System Inspection--Verification
6	MS	Probable Causes
7	Form Index	Study
8	SS	Checking and Analysis--Search
9	MS	Checking and Analysis--Solution
10	MS	Remedies--Repair
11	xxx	Remedies--Form
12	xxx	Finish
13	xxx	Score Analysis
14	Form	781A
15	Form	781D
16	Form	781K
17-23	Response[c]	Standard response

[a]Multiple-selection.

[b]Step-sequence.

[c]Wording only, not meaning, may be changed.

When creating a new standard display, the layout format used in CATTS displays should be kept in mind. An example of such a format (which can be altered) follows.



IV. SYSTEM TEST

After implementation, the author mode of CATTS was tested by a pneudraulics instructor from Chanute AFB. The instructor had no previous experience with computers or computer-aided instruction. We wished to observe his initial reaction to the system and the time it took for him to learn how to use CATTS. The latter step included the development of a new problem. Of course, we were also interested in his criticisms and suggestions.

OBSERVATIONS

The instructor spent half an hour twice a day for four days at the terminal. His immediate reaction to CATTS was positive and it remained so throughout the experiment. During the first session, he learned to use the function keys, Rand Tablet, light pen, and CRT. He learned the physical manipulation of these devices in less than five minutes.

To familiarize the instructor with the system, the next several sessions we - devoted to executing the system's author and student modes. The instructor found no difficulty in creating and using all displays, except for the step-sequencing ones. The sequencing concepts within the framework of a coding system were foreign to him. Once explained, however, he found the idea of coded sequencing quite appealing, not only for CATTS, but also as a teaching aide. Although the objective of improving the instructor's ability to discuss and teach the subject matter was not included in the system design, it is a possible CAI benefit that should be explored.

The instructor made several recommendations on improvements in the wording of directions. In addition, certain situations in sequencing occurred to him that were not explained in the system description. For example, what if the components in a step do not need to be specifically ordered? It was determined that the situation may be handled by making each component a separate step referencing each other.

The instructor also noted the value of preparing the problem in advance of the terminal session, either with notes or by using a set of standard display pictures. This approach minimizes design time at the terminal. Of course, with the problem definition clear from the filled-in standard display pictures and the problem insertion process so easy, an assistant or clerk could do the work at the terminal. In other words, the functions of problem creation and problem insertion may or may not be separated, depending only on (1) the desire to minimize computer costs, and (2) the instructor's preference.

In conclusion, the instructor felt that CATTS provides a potentially useful teaching medium for conveying troubleshooting concepts. He was able to use the system in a very short time with minimum effort. It also seems evident that involving the eventual system user, the instructor, throughout the design of the CAI course would produce better and more acceptable CAI packages.

SYSTEM EVALUATION

In summary, to create a troubleshooting problem with CATTS, an instructor must know how to use the POGO function keys and he must know how to define his problem with simple codes. It is not necessary to learn any computer language at this level, to use special cameras, or to resort to off-line coordinate computation and card punching (the only card punching CATTS requires currently is done automatically).

Although the instructor-user will rarely involve himself at the programming level, the programming simplicity of POGO should interest course developers. The program behind the student mode (EXECUTE), written in FORTRAN IV, uses only six different types of FORTRAN statements in combination with calls to POGO routines, to total less than 200 statements. All possible contingencies in the problem-solving tree are taken into account. In comparison, the Navy's CAI course in troubleshooting required some 200 to 300 lines in COURSEWRITER for *each* path through the problem tree.

After examining the description of DESIGN and EXECUTE, the reader may perceive several limitations of CATTS and the graphics approach.

As for CATTS, the reader may consider the maximum of seven instructor-supplied responses and 30 total displays as a rather unrealistic limitation. The limit does not really exist, however, as the file for CATTS and the file control pages may be expanded to include any reasonable number of instructor-supplied responses. The limit of 30 was imposed on CATTS simply because it was sufficient at the time.

The use of multiple-selection questions in several displays may be criticized on the basis that it primes the student for right and wrong answers. An alternative approach allows the student to type in his own response to questions. This capability is available in most course author languages, but unfortunately is not incorporated into CATTS (or POGO); however, Rand is implementing a new graphics system that will allow modification of CATTS and POGO to accept typed input during execution. Means for analyzing such input exist in programs and routines developed elsewhere, and a way to incorporate all these capabilities into a "CAI System" is discussed in the concluding section.

The new Video Graphics System at Rand can also provide another solution to the multiple selection problem. In one instance, a multiple selection question is used in CATTS because a schematic could not describe the aircraft's visual appearance and sounds. Under the new system, it will be possible to recall photographs from disk and sound from tapes as required. These developments should lend even more realism to the learning environment.

The high cost of hardware necessary to support a graphics system is, of course, an obvious drawback to the graphics approach. Graphics hardware is 10 to 20 times more expensive than text-displaying equipment. Furthermore, graphics software has not been available in the past to make simulations and graphical displays easy to describe and input. For example, the creation of the least complex CATTS display using a standard graphics software package would take approximately 30 minutes of sketching, computing coordinates, coding, executing, and debugging. The more complex displays would take two hours or longer of the same process. In addition to the excessive time involved, most instructors would find the process tedious and unnatural. With POGO, the time range for creating displays is 2 to 15 minutes with no off-line display sketching, coordinate computing, or programming required. The process is natural in that the user draws and sees his drawing simultaneously. The purpose of this Report is to show how POGO could be used in a CAI application to eliminate the graphic input problem. To the extent that it does, CAI becomes a more feasible teaching medium. And, to the extent that savings are realized due to more effective and efficient teaching through CAI, the higher costs of the graphics approach may be offset.

V. CONCLUSIONS AND RECOMMENDATIONS

In describing the functional approach to CAI, we indicated that an author requires an easy means to create CAI courses, whereas a student requires facilities for interacting with the information displayed by the CAI course program. The CATTS system demonstrates POGO's role in providing these functions. In addition to these functions, however, both author and student may need access to computational facilities that may not be a part of the CAI package. Also, an instructor should be able to switch from author-mode to student-mode on-line to check out his CAI course. Unfortunately, the typical teaching system has been designed as special-purpose software. This means that the system is limited to performing functions specifically programmed into the teaching system. The addition of other features requires extensive additional programming.

Consequently, the development of a CAI system permitting the interleaving of programs has been recommended [4]. The "coherently programmed" system should be a set of conventions and techniques designed to shape the growth of a library of programs that would enable the user to draw upon them freely with minimal concern about the details of their compatibility. A CAI system user is likely to make use of such system facilities as conventional compilers, text editors, and text analyzers. He should be able to use them without recourse to either a complex job control language or the more difficult alternative of having to terminate the process. It should be a primary goal of the system to allow users to work on-line in several languages or modes.

In 1968, K. J. Engvold and J. L. Hughes of the IBM Education Center suggested that a system they designed called ADEPT (A Display-Expedited Processing and Tutorial) could solve the difficulty of interleaving programs [11]. ADEPT is a program that controls the standard operating system by terminating and rescheduling itself automatically, relinquishing computer resources allocated to it, and surrendering control to the operating system to perform other jobs. By means of a display unit and light pen, the user may access any of the cataloged programs that function as subroutines under the control of ADEPT once it is in execution. Thus all features added to the standard operating system in the future would automatically come under control of ADEPT without the need for additional programming.

The coherent programming approach to CAI, exemplified by ADEPT, expands the power and flexibility of CAI by making immediately available to students, teachers, and other users the full resources of the operating system. Graphic facilities such as POGO could be used in combination with previously programmed text analyzers and computational facilities, for example, thus satisfying all the functional requirements of both course author and student.

Appendix

PROBLEM DESIGN AND EXAMPLE

The following pages detail the purpose of each display. Standard displays must be handled in the sequence in which they appear in the CATTS files. Optional instructor actions are designated by a single asterisk. All other actions described are required. The pneudraulics system problem from Sec. III is developed to illustrate the use of each display.

In the descriptions, display numbers in parentheses refer to displays listed on the file control pages presented in Figs. A1, A2, and A3.

Figures A4 through A23 are described by purpose, instructor actions, and example. Standard displays appear on the top half of the page; if nothing appears on the bottom half, the standard display is used without change. Displays on the bottom half are either (1) standard displays that have been altered or (2) instructor-supplied displays. In the latter case, of course, no display appears on the top half of the page.

FILE CONTROL

STANDARD DISPLAYS 1-10 | STANDARD DISPLAYS 11-20 | RESPONSE DISPLAYS PAGE

RECALL	ITEM	DESCRIPTION	FILE CURRENT DISPLAY AS ITEM NO.	DELETE ITEM NO.
⊡	1	CONTROL	⊡	⊡
⊡	2	TROUBLE	⊡	⊡
⊡	3	SYSTEM INSPECTION - VISUAL	⊡	⊡
⊡	4	SYSTEM INSPECTION - OPERATIONAL	⊡	⊡
⊡	5	SYSTEM INSPECTION - VERIFICATION	⊡	⊡
⊡	6	PROBABLE CAUSES	⊡	⊡
⊡	7	STUDY	⊡	⊡
⊡	8	CHECKING & ANALYSIS - SEARCH	⊡	⊡
⊡	9	CHECKING & ANALYSIS - SOLUTION	⊡	⊡
⊡	10	REMEDIES - REPAIR	⊡	⊡

RETURN

Fig. A1 — File control, 1-10

FILE CONTROL

STANDARD DISPLAYS 1-10 | STANDARD DISPLAYS 11-20 | RESPONSE DISPLAYS PAGE

RECALL	ITEM	DESCRIPTION	FILE CURRENT DISPLAY AS ITEM NO.	DELETE ITEM NO.
⊡	11	REMEDIES - FORM	⊡	⊡
⊡	12	FINISH	⊡	⊡
⊡	13	SCORE ANALYSIS	⊡	⊡
⊡	14	781A	⊡	⊡
⊡	15	781D	⊡	⊡
⊡	16	781K	⊡	⊡
⊡	17	STD RESP 1	⊡	⊡
⊡	18	STD RESP 2	⊡	⊡
⊡	19	STD RESP 3	⊡	⊡
⊡	20	STD RESP 4	⊡	⊡

RETURN

Fig. A2 — File control, 11-20

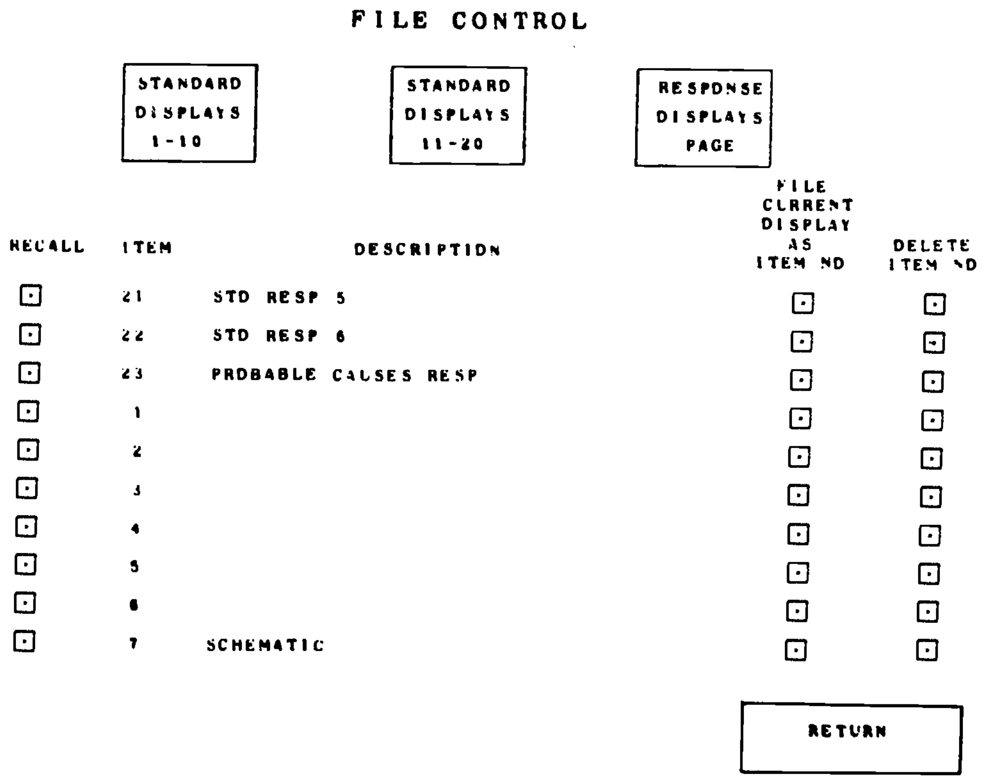


Fig. A3 — Response displays page with schematic filed

1. Schematic

Purpose

To be used to create other displays that test the student on his knowledge of and ability to use aircraft schematics.

Instructor Actions

1. Recall the Checking and Analysis - Search display (#8) for alignment purposes;
2. Press "reset pointers";
3. Position the schematic on the Rand Tablet so that it fits into the display, leaving room at the bottom of the display for responses;
4. Trace the schematic using the stylus and function keys;
5. Place an invisible box around each component without overlapping;
6. Insert 9xxxx codes for all components;
7. Press "file control", branching to Response Displays Page;
8. File the schematic at the end of the file;
9. Press "erase screen";
10. Press "file control."

Example

1. Recalled standard display #8, Checking and Analysis - Search;
2. Pressed "reset pointers";
3. Traced schematic, labeled components;
4. Placed invisible boxes around each component without overlapping;
5. Inserted 9xxxx codes for all components;
6. Pressed "file control," branched to Response Displays Page;
7. Filed display temporarily as Item 7 (see Fig. 27);
8. Pressed "erase screen";
9. Pressed "file control."

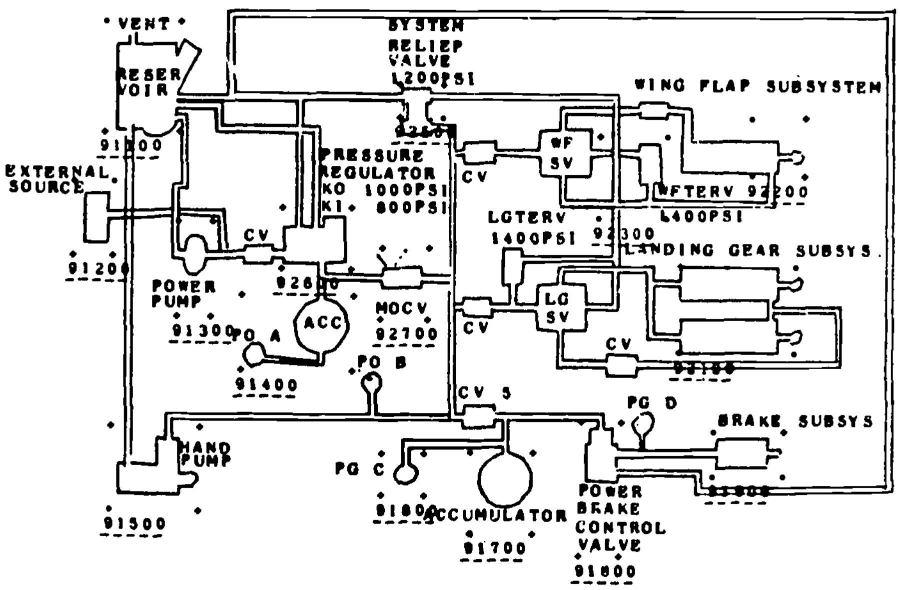


Fig. A4 — Schematic

2. Control

Purpose

To allow the student to branch among the various steps of troubleshooting.

Instructor Actions

1. Recall display;
2. Inputs: None;
3. Press "punch display";
4. Press "erase screen";
5. Press "file control."

Example

1. Recalled display;
2. Pressed "punch display";
3. Pressed "erase screen";
4. Pressed "file control."

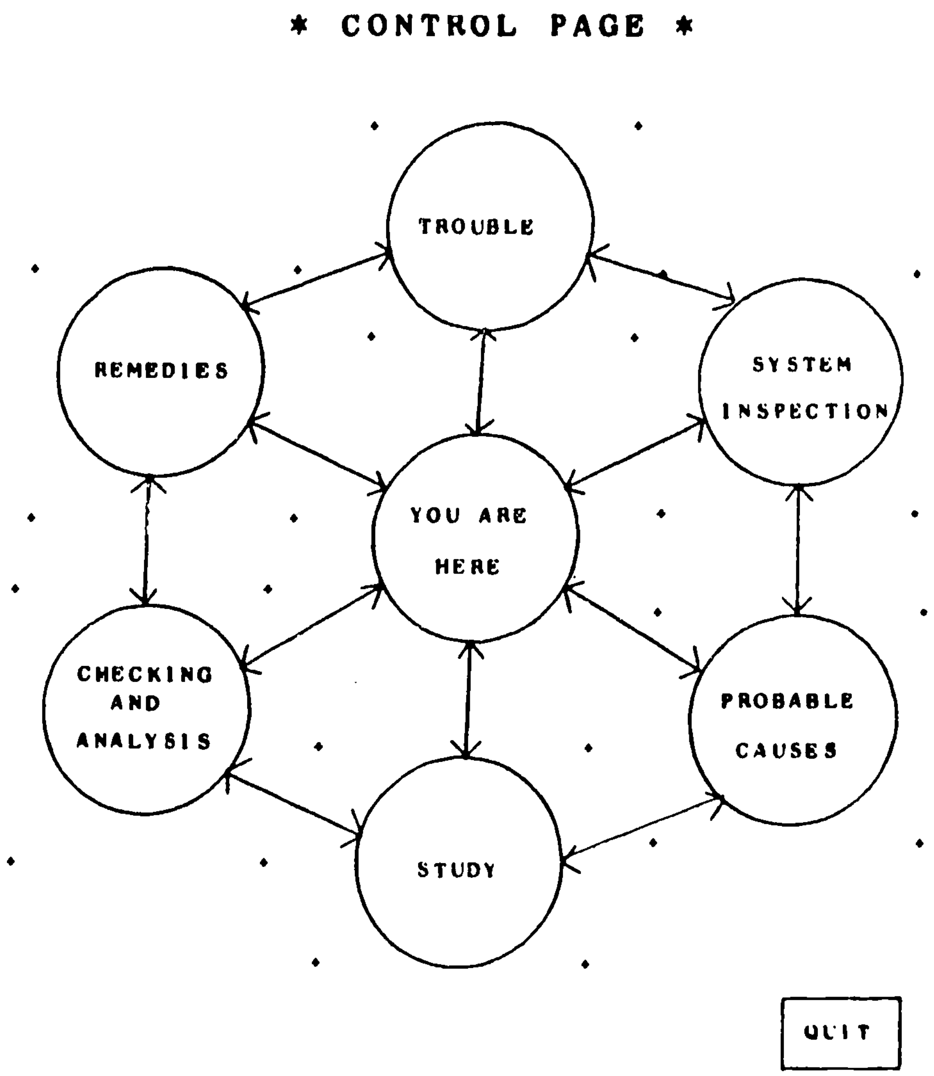


Fig. A5 — Control page

3. Trouble

Purpose

To present the malfunction to the student on the form on which he would receive it on the job.

Instructor Actions

1. Recall display;
2. Inputs: Using "small characters" enter
   a. Date disc'd
   b. Code
   c. Problem
   d. Disc'd by
3. Press "punch display";
4. Press "erase screen";
5. Press "file control."

Example

1. Recalled display;
2. Inputs: Using "small characters" entered
   a. Date disc'd: 10-09-9
   b. Code D
   c. Problem: Zero pressure indication on system pressure gage
   d. Disc'd by: J Jones, AIC
3. Pressed "punch display";
4. Pressed "erase screen";
5. Pressed "file control."

* TROUBLE *

SYM	DATE DISD	CODE	DISCREPANCY	REPORT NO	CORRECTIVE ACTION
					DATE CORRECTED
			DISCOVERED BY	CORRECTED BY	INSPECTED BY

AFTO FORM DEC 61 781A MAINTENANCE DISCREPANCY/WORK RECORD

WHEN YOU ARE READY TO GO ON TOUCH CONTINUE

QUIT	RETURN	BACK-UP	CONTINUE

* TROUBLE *

SYM	DATE DISD	CODE	DISCREPANCY	REPORT NO	CORRECTIVE ACTION
	10-08-8	D			
			ZERO PRESSURE INDICATION		
			ON SYSTEM PRESSURE GAGE.		
					DATE CORRECTED
			DISCOVERED BY J JONES, A1C	CORRECTED BY	INSPECTED BY

AFTO FORM DEC 61 781A MAINTENANCE DISCREPANCY/WORK RECORD

WHEN YOU ARE READY TO GO ON TOUCH CONTINUE.

QUIT	RETURN	BACK-UP	CONTINUE

Fig. A6 — Trouble

4. *System Inspection--Visual*

Purpose

To test the student on his knowledge of visual inspection requirements. (Optional instructor actions are designated by a single asterisk.)

Instructor Actions

1. Recall display;
2. Inputs: None if standard display is acceptable; if other selections or responses are desired, they must be entered and the codes changed according to the format described under Coding for Multiple Selections, Sec. III;
3. Press "punch display";
*4. Press "reset pointers";
*5. Response input: If an instructor-response is to be inserted, enter the response below the list using large characters and preceding it with three asterisks;
*6. Press "file control," branching to Response Displays Page;
*7. Store the response at the end of standard displays in the position indicated by the code;
*8. Return to the current display;
*9. Delete response;
*10. Repeat steps (5)-(9) for all instructor-responses indicated by codes with last two digits nonzero;
11. Press "erase screen";
12. Press "file control."

Example

1. Recalled display;
2. Inputs: The instructor has decided to insert his own responses for wrong answers. Therefore each 9xxxx code has been changed to indicate the position of the response in the file. For example,

   95103: Wrong answer, five right on page, first wrong, return third instructor-stored response in file.

3. Pressed "punch display";
4. Pressed "reset pointers";
5. Response input: Response to the selection "internal leakage";
   *** No. Internal leakage is not visible.
6. Pressed "file control," branched to Response Displays Page;
7. Filed response in position 3 because response digit (i.e., last digit) of code for the selection is a 3;
8. Returned to current display;
9. Deleted response;
10. Response input: Response to the selection "system relief valve";
    *** No. System relief valve defects cannot be directly determined by visual inspection.
11. Pressed "file control," branched to Response Displays Page;
12. Filed response in position 4;
13. Returned to current display;
14. Deleted response;
15. Response input: Response to the selection "power pump";
    *** No. Defects in the power pump are not visually apparent.
16. Pressed "file control," branched to Response Displays Page;
17. Filed response in position 5;
18. Pressed "erase screen";
19. Pressed "file control."

* SYSTEM INSPECTION *

BEFORE DIAGNOSING A MALFUNCTIONING SYSTEM, IT IS NECESSARY
TO PERFORM THE FOLLOWING IN ORDER:

1) A VISUAL INSPECTION,
2) AN OPERATIONAL INSPECTION,
3) A CONDITION VERIFICATION.

WHICH OF THE FOLLOWING WOULD YOU VISUALLY INSPECT?

WHEN FINISHED TOUCH DONE
55

RESERVOIR FLUID LEVEL
95100
INTERNAL LEAKAGE
95100
MECHANICAL LINKAGES
95200
ACCUMULATOR GAGE
95300
ELECTRICAL CONNECTIONS
95400
SYSTEM RELIEF VALVE
95200
EXTERNAL LEAKAGE
95500
POWER PUMP
95300

QUIT.	BACK-UP	RETURN	CONTINUE
51	52	53	54

* SYSTEM INSPECTION *

BEFORE DIAGNOSING A MALFUNCTIONING SYSTEM, IT IS NECESSARY
TO PERFORM THE FOLLOWING IN ORDER:

1) A VISUAL INSPECTION,
2) AN OPERATIONAL INSPECTION,
3) A CONDITION VERIFICATION.

WHICH OF THE FOLLOWING WOULD YOU VISUALLY INSPECT?

WHEN FINISHED TOUCH DONE
55

RESERVOIR FLUID LEVEL
95100
INTERNAL LEAKAGE
95100
MECHANICAL LINKAGES
95200
ACCUMULATOR GAGE
95300
ELECTRICAL CONNECTIONS
95400
SYSTEM RELIEF VALVE
95200
EXTERNAL LEAKAGE
95500
POWER PUMP
95300

QUIT.	RETURN	BACK-UP	CONTINUE
51	52	53	54

Fig. A7 — System inspection: visual

5. *System Inspection--Operational*

Purpose

To test the student on his knowledge of operational inspection requirements.

Instructor Actions

1. Recall display;
2. Inputs: None if standard display is acceptable; if other selections are desired, they must be entered and the codes changed according to the format described under Coding for Multiple Selections, Sec. III;
3. Press "punch display";

*4. Press "reset pointers";

*5. Response inputs: If an instructor response is to be inserted, enter the response below the list using "large characters" and preceding it with three asterisks;

*6. Press "file control," branching to Response Displays Page;

*7. File the response at the end of the standard displays in position indicated by codes;

*8. Return to current display;

*9. Delete response;

*10. Repeat steps (5)-(7) for all instructor responses indicated by codes with last two digits nonzero;

11. Press "erase screen";
12. Press "file control."

Example

1. Recalled display;
2. Inputs: The user has decided to insert his own responses for wrong answers; therefore each 9xxxx code has been changed to indicate the position of the response in the file.

   93107: Wrong answer, three right on page, first wrong, return seventh instructor-stored response in file.

   93206: Wrong answer, three right on page, second wrong, return sixth instructor-stored response in file.

3. Pressed "punch display";
4. Pressed "reset pointers";
5. Response input: Response to the selection, "the hand pump is inoperative":

   *** No. The hand pump cannot be tested when the power pump is on.
6. Pressed "file control," branched to Response Displays Page;
7. Filed response as Item 7;
8. Returned to current display;
9. Deleted response;
10. Response input: Response to the selection, "the pressure regulator is stuck":

    *** No. With the power pump on the pressure regulator cannot be isolated for checking;
11. Pressed "file control," branched to Response Displays Page;
12. Filed response as Item 6;
13. Pressed "erase screen";
14. Pressed "file control."

* SYSTEM INSPECTION *

AN OPERATIONAL CHECK MUST BE MADE NEXT. WHICH OF THE FOLLOWING WOULD THIS CHECK DETERMINE? WHEN FINISHED TOUCH [DONE]
55

- [•] THE SYSTEM IS OPERATING
  03100
- [•] THE HAND PUMP IS INOPERATIVE
  03100
- [•] THE PRESSURE REGULATOR IS STUCK
  03200
- [•] THE SYSTEM IS OPERATING BUT TOO SLOW OR TOO FAST.
  03200
- [•] ABNORMAL NOISES INDICATING RUBBING OR BINDING
  03300

QUIT.	RETURN	BACK-UP	CONTINUE
51	52	53	54

* SYSTEM INSPECTION *

AN OPERATIONAL CHECK MUST BE MADE NEXT. WHICH OF THE FOLLOWING WOULD THIS CHECK DETERMINE? WHEN FINISHED TOUCH [DONE]
55

- [•] THE SYSTEM IS OPERATING.
  03100
- [•] THE HAND PUMP IS INOPERATIVE.
  03107
- [•] THE PRESSURE REGULATOR IS STUCK
  03208
- [•] THE SYSTEM IS OPERATING BUT TOO SLOW OR TOO FAST
  03200
- [•] ABNORMAL NOISES INDICATING RUBBING OR BINDING
  03300

QUIT	RETURN	BACK-UP	CONTINUE
51	52	53	54

Fig. A8 — System inspection: operational

6. *System Inspection--Verification*

Purpose

To have the student verify that the malfunction exists as reported.

Instructor Actions

1. Recall display;
2. Recall schematic;
3. Inputs: Insert codes for components according to the step-sequence format described under Coding for Step-Sequencing, Sec. III;
4. Press "punch display";
5. Press "reset pointers";
6. Response inputs: Draw and/or enter system response to correct student action;
7. Press "file control," branching to Response Displays Page;
8. File response at end of standard displays in position indicated by the codes;
9. Press "erase screen";
10. Press "file control."

Example

1. Recalled display;
2. Recalled schematic;
3. Inputs: To verify that the system pressure gage is reading zero, the power pump must be turned on (activated). The power pump is therefore coded:

   11001: Step 1, component 1, no previous component required, return response 1.

   All other components retain 9xxxx codes.
4. Pressed "punch display";
5. Pressed reset pointers;
6. Response inputs: The gages indicate the system's response to activation of the power pump. Therefore gage readings are entered which show a zero pressure indication on PG B, the system gage, as

reported on the 781A. Normal readings are entered for the other gages.

300 PSI 0 PSI 1000 PSI 0 PSI

7. Pressed "file control," branched to Response Displays Page;
8. Filed response as Item 1;
9. Pressed "erase screen";
10. Pressed "file control."

*** SYSTEM INSPECTION ***

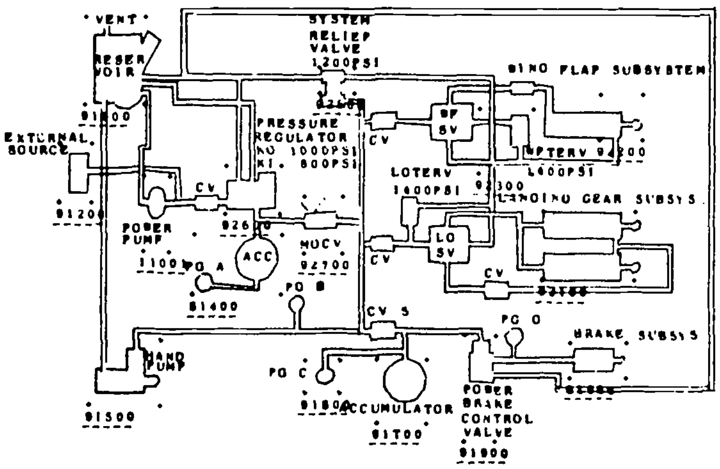
FINALLY YOU MUST VERIFY THAT THE SYSTEM IS MALFUNCTIONING AS REPORTED. ON THE SCHEMATIC ACTIVATE THE COMPONENT(S) THAT WILL DEMONSTRATE THE CONDITION AND DETERMINE IF IT EXISTS. WHEN YOU WANT A SYSTEM RESPONSE TO YOUR ACTIONS TOUCH [RESPONSE]

WHEN FINISHED TOUCH CONTINUE

QUIT	RETURN	BACK-UP	CONTINUE

*** SYSTEM INSPECTION ***

FINALLY YOU MUST VERIFY THAT THE SYSTEM IS MALFUNCTIONING AS REPORTED. ON THE SCHEMATIC ACTIVATE THE COMPONENT(S) THAT WILL DEMONSTRATE THE CONDITION AND DETERMINE IF IT EXISTS. WHEN YOU WANT A SYSTEM RESPONSE TO YOUR ACTIONS TOUCH [RESPONSE]
56



WHEN FINISHED TOUCH CONTINUE

QUIT	RETURN	BACK-UP	CONTINUE
51	52	53	54

Fig. A9 — System inspection: condition verification

7. *Probable Causes*

Purpose

To test the student's knowledge of common causes of the malfunction given in the "trouble" display.

Instructor Actions

1. Recall display;
2. Recall schematic;
3. Inputs: Insert codes for components according to the multiple-selection coding format described under Coding for Multiple Selection, Sec. III;
4. Press "punch display";
*5. Press "reset pointers";
*6. Response inputs: Enter response below schematic using "large characters" and preceding it with three asterisks;
*7. Press "file control," branching to Response Displays Page;
*8. File response at the end of standard displays in the position indicated by code;
*9. Delete response;
*10. Repeat steps (6)-(8) for all instructor-responses indicated by codes with last two digits nonzero;
11. Press "erase screen";
12. Press "file control."

Example

1. Recalled display;
2. Recalled schematic;
3. Inputs: The probable causes of a zero reading on the system pressure gage are a defective pressure regulator, power pump, system pressure gage and actuating cylinder(s). These components are given "right" codes:

Pressure regulator:	86100:	Right, six right, first right, standard response returned.
Power pump:	86200:	Right, six right, second right, standard response returned.

PG B:	86300:	Right, six right, third right, standard response returned.
Wing flap actuating cylinder:	86400:	Right, six right, fourth right, standard response returned.
Landing gear actuating cylinder:	86500:	Right, six right, fifth right, standard response returned.
Brake actuating cylinder:	86600:	Right, six right, sixth right, standard response returned.

All other components retain the "wrong" 9xxxx codes.

4. Pressed "punch display";
5. Pressed "erase screen";
6. Pressed "file control."

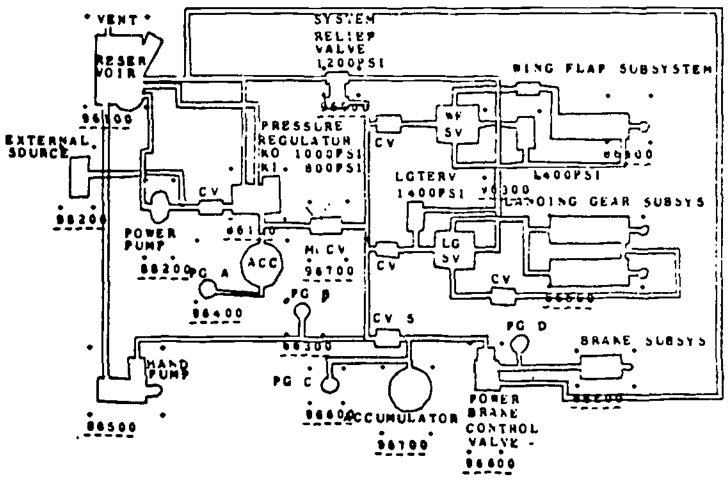
* PROBABLE CAUSES *

INDICATE ON THE SCHEMATIC WHICH COMPONENTS, IF DEFECTIVE, COULD CAUSE THE TROUBLE WHEN FINISHED TOUCH DONE

QUIT | RETURN | BACK-UP | CONTINUE

* PROBABLE CAUSES *

INDICATE ON THE SCHEMATIC WHICH COMPONENTS, IF DEFECTIVE COULD CAUSE THE TROUBLE. WHEN FINISHED TOUCH DONE
55



QUIT | RETURN | BACK-UP | CONTINUE
51 52 53 54

Fig. A10 — Probable causes

8. *Study*

Purpose

To provide the student with the aircraft's history as recorded in the flight manual.

Instructor Actions

1. Recall display;
2. Inputs: None;
3. Press "punch display";
4. Press "erase screen";
5. Press "file control."

Example

1. Recalled display;
2. Pressed "punch display";
3. Pressed "erase screen";
4. Pressed "file control."

* STUDY *

SELECT ONE OF THE FOLLOWING FORMS TO STUDY.

YOU MAY RETURN AND SELECT ANOTHER.

- [·] 781A (PREVIOUS ENTRIES
- [·] 781K
- [·] 781D

QUIT	RETURN	BACK-UP	CONTINUE

Fig. A11 — Study

9. *Checking and Analysis--Search*

Purpose

To allow the student to check out the malfunctioning system using a schematic.

Instructor Actions

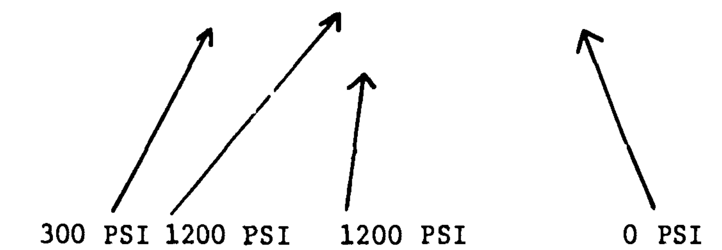
1. Recall display;
2. Recall schematic;
3. Inputs: Insert codes for components according to the step-sequence coding format described in Sec. III;
4. Press "punch display";
5. Press "reset pointers";
6. Response inputs: Draw and/or enter system responses to a student's correct completion of a step-sequence;
7. Press "file control," branching to Response Displays Page;
8. File response at end of standard displays in the position indicated by code;
9. Return to current display;
10. Delete response;
11. Repeat (6)-(8) for each checkout step; the number of instructor-responses required is less than or equal to the number of steps necessary to troubleshoot the system for the given malfunction (i.e., some step may cause the same system response);
12. Press "erase screen";
13. Press "file control."

Example

1. Recalled display;
2. Recalled schematic;
3. Inputs: The step and component sequence that minimizes cost and time for this malfunction requires coding as follows (described in greater detail in Sec. III);

External source:	11001
Wing Flap SV:	21110
Landing Gear SV:	22110
Power Brake CV:	23111
MOCV:	31000
Hand Pump:	32002

4. Pressed "punch display";
5. Pressed "reset pointers";
6. Response input: The gages indicate the system's response to correct activation of components in a step. In this problem, the power pump is not defective. Therefore, when the external source is activated the same response is returned as when the power pump was activated (Response 1). The actuating cylinders are also not defective so Response 1 is returned again. Since Response 1 is already stored in the response file it does not have to be entered again. The defective component is the pressure regulator and therefore a new response (Response 2) was created for return when Step 3 is completed.



7. Pressed "file control," branched to Response Displays Page;
8. Filed new response as Item 2;
9. Pressed "erase screen";
10. Pressed "file control."

* CHECKING AND ANALYSIS *

INDICATE ON THE SCHEMATIC THE COMPONENTS TO ACTIVATE FOR CHECKING OUT THE PROBABLE CAUSES. THE ORDER IN WHICH YOU CHECK OUT THE PROBABLE CAUSES SHOULD BE DETERMINED BY: 1) THE TIME AND EFFORT NEEDED TO CHECK OUT EACH CAUSE, THE EASIEST FIRST, 2) INFORMATION FROM THE 78 SERIES. WHEN YOU WANT A SYSTEM RESPONSE TO YOUR ACTIONS TOUCH [RESPONSE]

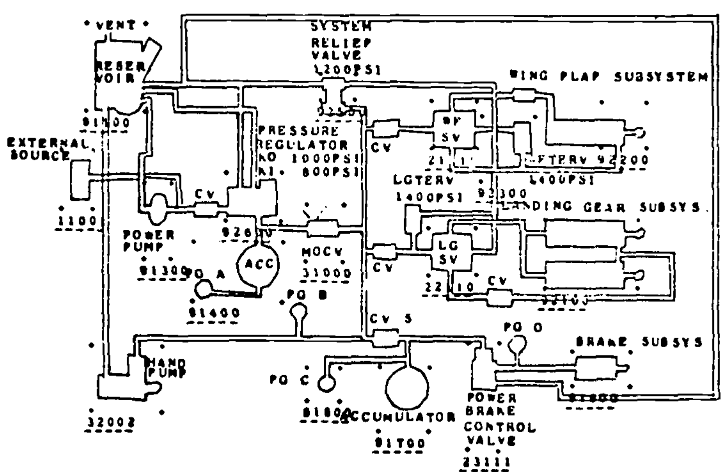
WHEN YOU HAVE DETERMINED THE CAUSE TOUCH CONTINUE

QUIT	RETURN	BACK-UP	CONTINUE

* CHECKING AND ANALYSIS *

INDICATE ON THE SCHEMATIC THE COMPONENTS TO ACTIVATE FOR CHECKING OUT THE PROBABLE CAUSES. THE ORDER IN WHICH YOU CHECK OUT THE PROBABLE CAUSES SHOULD BE DETERMINED BY: 1) THE TIME AND EFFORT NEEDED TO CHECK OUT EACH CAUSE, THE EASIEST FIRST, 2) INFORMATION FROM THE 731 SERIES. WHEN YOU WANT A SYSTEM RESPONSE TO YOUR ACTIONS TOUCH [RESPONSE]

58



WHEN YOU HAVE DETERMINED THE CAUSE TOUCH CONTINUE

QUIT	RETURN	BACK-UP	CONTINUE
51	52	53	54

Fig. A12 — Checking and analysis: search

10. Checking and Analysis--Solution

Purpose

To have the student pinpoint the cause of the malfunction.

Instructor Actions

1. Recall display;
2. Recall schematic;
3. Inputs: Insert an 8xxxx code for the component that is defective according to the format for multiple-selection codes;
4. Press "punch display";

*5. Press "reset pointers";

*6. Response input: Enter response below schematic using "large characters" and preceding it with three asterisks;

*7. Press "file control," branching to the Response Displays Page;

*8. File response at the end of standard displays in the position indicated by last digit of code;

*9. Return to current display;

*10. Delete response;

*11. Repeat steps (6)-(8) for all instructor-responses indicated by codes with last digit nonzero;

12. Press "erase screen";
13. Press "file control."

Example

1. Recalled display;
2. Recalled schematic;
3. Inputs: The defective component, the pressure regulator, was given the code 81100 (right answer, one right, first right, standard response) while all other components retained 9xxxx codes;
4. Pressed "punch display";
5. Pressed "file control."

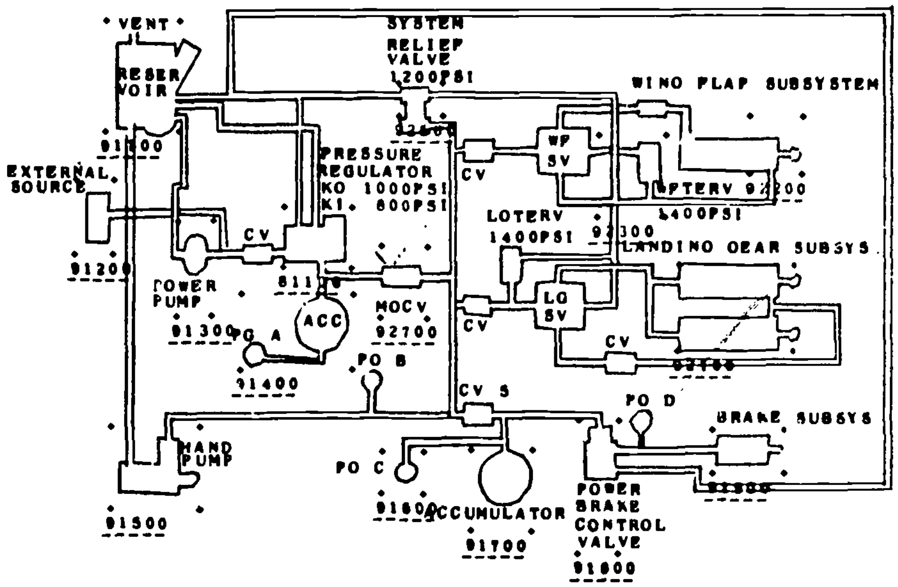
* CHECKING AND ANALYSIS *

INDICATE ON THE SCHEMATIC THE COMPONENT THAT YOU HAVE DETERMINED IS DEFECTIVE.

QUIT	RETURN	BACK-UP	CONTINUE

* CHECKING AND ANALYSIS *

INDICATE ON THE SCHEMATIC THE COMPONENT THAT YOU HAVE DETERMINED IS DEFECTIVE.



QUIT	RETURN	BACK-UP	CONTINUE
51	52	53	54

Fig. A13 — Checking and analysis: solution

11. Remedies--Repair

Purpose

To test the student on his knowledge of the means to repair the given defective component.

Instructor Actions

1. Recall display;
2. Inputs: None if standard display is acceptable; if other selections are desired, they must be entered and the codes changed according to the format described in Sec. III;
3. Press "punch display";

*4. Press "reset pointers";

*5. Response input: If an instructor response is to be inserted, enter the response below the list using "large characters" and preceding the response with three asterisks;

*6. Press "file control," branching to Response Displays Page;

*7. File response at the end of the standard displays in the position indicated by the code;

*8. Return to current display;

*9. Delete response;

*10. Repeat steps (5)-(8) for all instructor responses indicated by codes with last digit nonzero;

11. Press "erase screen";
12. Press "file control."

Example

1. Recalled display;
2. Pressed "punch display";
3. Pressed "erase screen";
4. Pressed "file control."

* REMEDIES *

NOW THAT YOU HAVE FOUND THE DEFECTIVE COMPONENT, HOW WOULD YOU REPAIR IT? WHEN FINISHED TOUCH DONE

- REPLENISH THE COMPONENT.
- REPLACE THE COMPONENT.
- CALL AN ELECTRICIAN.
- REPAIR AND RETURN THE COMPONENT.
- REPOSITION THE COMPONENT.

QUIT	RETURN	BACK-UP	CONTINUE

Fig. A14 — Remedies: repair

12. *Remedies--Form*

Purpose

To test the student on the completion of AF Form 781A.

Instructor Actions

1. Recall display;
2. Inputs: Using "small characters" enter the same information appearing in the Trouble display,
   a. Date Disc'd
   b. Code
   c. Problem
   d. Disc'd by
3. Press "punch display";
4. Press "erase screen";
5. Press "file control."

Example

1. Recalled display;
2. Inputs: Using "small characters" entered,
   a. Date disc'd: 10-09-9
   b. Code: D
   c. Problem: Zero pressure indication on system pressure gage.
   d. Disc'd by: J Jones, AIC
3. Pressed "punch display";
4. Pressed "erase screen";
5. Pressed "file control."

* REMEDIES *

AFTER REPAIRING THE DEFECTIVE COMPONENT YOU MUST COMPLETE AFTO 781A BEFORE THE PLANE MAY FLY AGAIN. INDICATE ON THE FORM IN THE FOLLOWING ORDER WHERE YOU WOULD ENTER:

1) THE REMEDY,
2) YOUR NAME AND GRADE,
3) THE DATE,
4) YOUR INITIAL.

SYM	DATE DISD	CODE	DISCREPANCY	REPORT NO.	CORRECTIVE ACTION
					DATE CORRECTED
		DISCOVERED BY		CORRECTED BY	INSPECTED BY

AFTO FORM 781A DEC 61 — MAINTENANCE DISCREPANCY/WORK RECORD

QUIT	RETURN	BACK-UP	CONTINUE

* REMEDIES *

AFTER REPAIRING THE DEFECTIVE COMPONENT YOU MUST COMPLETE AFTO 781A BEFORE THE PLANE MAY FLY AGAIN. INDICATE ON THE FORM IN THE FOLLOWING ORDER WHERE YOU WOULD ENTER:

1) THE REMEDY,
2) YOUR NAME AND GRADE,
3) THE DATE,
4) YOUR INITIAL.

SYM	DATE DISCD	CODE	DISCREPANCY	REPORT NO	CORRECTIVE ACTION
	10-09-9	0			
	ZERO PRESSURE INDICATION				
	ON SYSTEM PRESSURE GAGE				
					DATE CORRECTED
		DISCOVERED BY J. JONES, A1C		CORRECTED BY	INSPECTED BY

AFTO FORM 781A SEP 68 — MAINTENANCE DISCREPANCY/WORK RECORD

WHEN FINISHED TOUCH CONTINUE

QUIT	RETURN	BACK-UP	CONTINUE

Fig. A15 — Remedies: form

13. Score

Purpose

To display the student's score and allow him to take various actions.

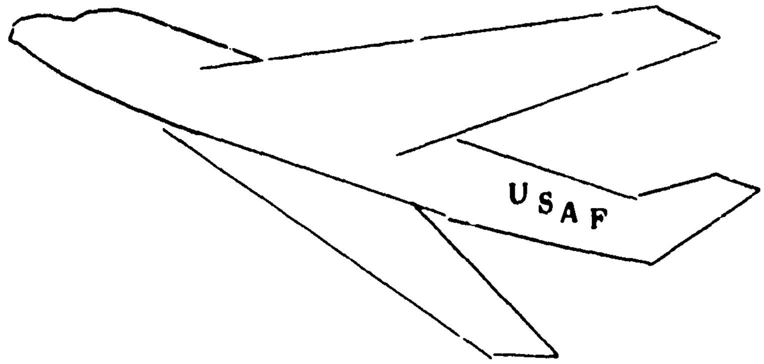
Instructor Actions

1. Recall display;
2. Inputs: None;
3. Press "punch display";
4. Press "erase screen";
5. Press "file control."

Example

1. Recalled display;
2. Pressed "punch display";
3. Pressed "erase screen";
4. Pressed "file control."

YOU HAVE NOW COMPLETED THE TROUBLESHOOTING PROCESS AND YOUR PLANE IS FLYING AGAIN.



YOUR SCORE IS *__________ %

DO YOU WISH

- [·] AN ANALYSIS OF YOUR RESULTS.
- [·] TO REPEAT THE SAME PROBLEM.
- [·] TO TRY ANOTHER PROBLEM ON THE SAME SYSTEM.
- [·] TO TRY A PROBLEM ON ANOTHER SYSTEM.
- [·] TO QUIT.

Fig. A16 — Score

14. *Score Analysis*

Purpose

To present a breakdown of the student's score according to the various steps in troubleshooting.

Instructor Actions

1. Recall display;
2. Inputs: None;
3. Press "punch display";
4. Press "erase screen";
5. Press "file control."

Example

1. Recalled display;
2. Pressed "punch display";
3. Pressed "erase screen";
4. Pressed "file control."

* SCORE ANALYSIS *

		POINTS
1.	TIME	*-----------
2.	VISUAL INSPECTION	*-----------
3.	OPERATIONAL INSPECTION	*-----------
4.	CONDITION VERIFICATION	*-----------
5.	PROBABLE CAUSES	*-----------
6	CHECKING	*-----------
7.	ANALYSIS (SEQUENCING)	*-----------
8.	SOLUTION	*-----------
9.	REMEDY	*-----------
10.	FORM	*-----------

TOTAL *----------- / 1000

PERCENT *-----------

QUIT	RETURN	BACK-UP

Fig. A17 — Score analysis

15. 781A

Purpose

To present the aircraft's Maintenance Discrepancy/Work Record, which is available in the flight manual.

Instructor Actions

1. Recall display;
2. Inputs: Using "small characters" enter past history about the aircraft; this history may or may not reveal information about the cause of the current malfunction;
3. Press "punch display";
4. Press "erase screen";
5. Press "file control."

Example

1. Recalled display;
2. Inputs: The user completed past 781A's with nonpertinent information for this problem:

Date From:	9-09-9
To:	10-09-9
Crew Chief:	A Baker, AIC
Orgn:	62TCW
Loc:	Norton AFB
MDS:	C-124C
Serial No:	50-83
Sym:	/
Date Disc'd:	9-09-9
Code:	K
Discrepancy:	Oil leak at oil cooler inlet tube
Disc'd by:	J Jones, AIC
Report No:	A 1456
Corrective Action:	"O" ring seal replace
Date corrected:	9-09-9
Corrected by:	J Jones, AIC
Sym:	/
Date Disc'd:	9-09-9
Code:	B

Discrepancy:	EQT indication fluctuates intermittently
Disc'd by:	A Baker, AIC
Report No:	A 1474
Corrective Action:	Thermocouple harness replaced, operational
Date corrected:	9-09-9
Corrected by:	D Evans, AIC

3. Pressed "punch display";
4. Pressed "erase screen";
5. Pressed "file control."

DATE FROM	TO	CREW CHIEF	ORGN	LOCATION	MDS	SERIAL NO

SYM	DATE DISCD	CODE	DISCREPANCY	REPORT NO	CORRECTIVE ACTION
					DATE CORRECTED
			DISCOVERED BY	CORRECTED BY	INSPECTED BY

SYM	DATE DISCD	CODE	DISCREPANCY	REPORT NO	CORRECTIVE ACTION
					DATE CORRECTED
			DISCOVERED BY	CORRECTED BY	INSPECTED BY

AFTO FORM SEP 66 781A MAINTENANCE DISCREPANCY/WORK RECORD

TO RETURN TO STUDY TOUCH [STUDY]

[QUIT] [RETURN] [BACK-UP] [CONTINUE]

DATE FROM	TO	CREW CHIEF	ORGN	LOCATION	MDS	SERIAL NO
9-09-9	10-09-9	A BAKER, AIC	82TCW	NORTON AFB	C-124C	50-83

SYM	DATE DISCD	CODE	DISCREPANCY	REPORT NO	CORRECTIVE ACTION
J	9-09-9	K		A 1450	
			OIL LEAK AT OIL COOLER INLET TUBE		"O" RING SEAL REPLACED.
					DATE CORRECTED 9-09-9
			DISCOVERED BY J JONES, AIC	CORRECTED BY J JONES, AIC	INSPECTED BY

SYM	DATE DISCD	CODE	DISCREPANCY	REPORT NO	CORRECTIVE ACTION
E	9-09-9	B		A 1474	
			EGT INDICATION FLUCTUATES INTERMITTENTLY		THERMOCOUPLE HARNESS REPLACE, OPERATIONAL.
					DATE CORRECTED 9-09-9
			DISCOVERED BY A BAKER, AIC	CORRECTED BY D EVANS, AIC	INSPECTED BY

AFTO FORM SEP 66 781A MAINTENANCE DISCREPANCY/WORK RECORD

TO RETURN TO STUDY TOUCH [STUDY]

[QUIT] [RETURN] [BACK-UP] [CONTINUE]

Fig. A18 — AFTO form 781A

16. 781D

Purpose

To present the aircraft's Calendar and Hourly Item Inspection Record, which is available in the flight manual.

Instructor Actions

1. Recall display;
2. Inputs: Using "small characters" enter past history about the aircraft; this history may or may not reveal information about the causes of the current malfunction;
3. Press "punch display";
4. Press "erase screen";
5. Press "file control."

Example

1. Recalled display;
2. Inputs:

Aircraft MDS:	C-124C
Serial No:	50-83
Page No:	3
Inspection Item:	Missile ammonia bottle
Frequency:	24 mos.
Next due:	15-10-9

3. Pressed "punch display";
4. Pressed "erase screen";
5. Pressed "file control."

CALENDAR AND HOURLY ITEM	AIRCRAFT MDS		SERIAL NO		PAGE NO
INSPECTION ITEM	FREQUENCY	NEXT DUE	NEXT DUE	NEXT DUE	NEXT DUE

AFTO FORM MAY 65 781D

TO RETURN TO STUDY TOUCH STUDY

QUIT | RETURN | BACK-UP | CONTINUE

CALENDAR AND HOURLY ITEM	AIRCRAFT MDS C-124C		SERIAL NO 50-83		PAGE NO 3
INSPECTION ITEM	FREQUENCY	NEXT DUE	NEXT DUE	NEXT DUE	NEXT DUE
MISSILE AMMONIA BOTTLE	24 MO	15-10-9			

AFTO FORM MAY 65 781D

TO RETURN TO STUDY TOUCH STUDY

QUIT | RETURN | BACK-UP | CONTINUE

Fig. A19 — AFTO form 781D

17. 781K

Purpose

To present the aircraft's Delayed Discrepancy and Calendar Item Inspection Record, which is available in the flight manual.

Instructor Actions

1. Recall display;
2. Inputs: Using "small characters" enter past history about the aircraft; this history may or may not reveal useful information about the cause of the current malfunction;
3. Press "punch display";
4. Press "erase screen";
5. Press "file control."

Example

1. Recalled display;
2. Inputs:

Date From:	1-09-9
Crew Chief:	A Baker
Orgn:	62TCW
Loc:	Norton
AFCT MDS:	C-124C
AFCT Serial No:	50-83
Symbol:	/
Description:	Top both wings need repainting
Sys:	11
Date:	1-09-9
AFCT Time:	4224
Code:	M
Date CW:	6-09-9
Verified by:	Jackson, SG

3. Pressed "punch display";
4. Pressed "erase screen";
5. Pressed "file control."

DATE FROM	TO	CREW CHIEF	ORGN	LOCATION	ACFT MDS	ACFT SERIAL NO

OUTSTANDING IMMEDIATE ACTION AND URGENT ACTION
T O S AND SUMMARY OF DELAYED DISCREPANCIES

SYM	T O NUMBER AND PUBLICATION DATE OR DISCREPANCY	SYS	WHEN DISCD/ENTERED DATE	ACFT TIME		DATE TO AFTO 349 OR 781A OR DATE CW	TRANSFERRED BY OR COMP VERIFIED BY (SIGNATURE)

AFTO FORM NOV 67 781K

DELAYED DESCREPANCY AND CALENDAR ITEM INSPECTION RECORD

TO RETURN TO STUDY TOUCH | STUDY

QUIT	RETURN	BACK-UP	CONTINUE

DATE FROM	TO	CREW CHIEF	ORGN	LOCATION	ACFT MDS	ACFT SERIAL NO
1-09-9		A BAKER	82TCW	NORTON	C-124C	50-83

OUTSTANDING IMMEDIATE ACTION AND URGENT ACTION
T.O S AND SUMMARY OF DELAYED DISCREPANCIES

SYM	T.O. NUMBER AND PUBLICATION DATE OR DISCREPANCY	SYS	WHEN DISCD/ENTERED DATE	ACFT TIME	CODE	DATE TO AFTO 349 OR 781A OR DATE CW	TRANSFERRED BY OR COMP VERIFIED BY (SIGNATURE)
/	TOP BOTH WINGS						
	NEED REPAINTING	11	1-09-9	4224	M	8-09-9	JACKSON,SG

AFTO FORM NOV 67 781K

DELAYED DESCREPANCY AND CALENDAR ITEM INSPECTION RECORD

TO RETURN TO STUDY TOUCH | STUDY

QUIT	RETURN	BACK-UP	CONTINUE

Fig. A20 — AFTO form 781K

18. Standard Responses

Purpose

To provide default responses to student actions.

Instructor Actions

1. Recall display;
2. Press "punch display";
3. Press "erase screen";
4. Press "file control";
5. Repeat (1)-(4) for all standard responses.

Example

1. Recalled display;
2. Pressed "punch display";
3. Pressed "erase screen";
4. Pressed "file control";
5. Repeated these four steps for standard responses (1)-(6).

*** YOUR PREVIOUS ANALYSIS DID NOT INDICATE THAT THIS COMPONENT IS DEFECTIVE. ARE YOU GUESSING? BACK UP AND CHECK OUT THE SYSTEM AGAIN.

*** NO. THERE ARE MORE ITEMS.

*** THAT'S RIGHT.

*** NO. TRY AGAIN.

*** NO. YOU HAVE ACTIVATED THE COMPONENTS OUT OF SEQUENCE.

*** NO. YOU HAVE ACTIVATED AN EXTRANEOUS COMPONENT. TRY AGAIN.

Fig. A21 — Standard responses, 1-6

19. Probable Causes Response

Purpose

To provide the student with the probable causes if he misses too many on that display.

Instructor Actions

1. Recall display;
2. Inputs: Complete response giving malfunction and its probable causes;
3. Press "punch display";
4. Press "erase screen";
5. Press "file control."

Example

1. Recalled display;
2. Inputs: Completed response,

   .. a zero pressure indication on the system pressure gage are the power pump, the pressure regulator, the system pressure gage and any of the actuating cylinders.

   Let's go on.
3. Pressed "punch display";
4. Pressed "erase screen";
5. Pressed "file control."

*** NO, YOU'RE GUESSING. THE COMPONENTS, IF DEFECTIVE, THAT CAN CAUSE

*** NO, YOU'RE GUESSING. THE COMPONENTS, IF DEFECTIVE, THAT CAN CAUSE A ZERO PRESSURE INDICATION ON THE SYSTEM PRESSURE GAGE ARE THE POWER PUMP, THE PRESSURE REGULATOR, AND THE SYSTEM PRESSURE GAGE. LET'S GO ON

Fig. A22 — Probable causes response

20. *Instructor-Stored Responses*

Purpose

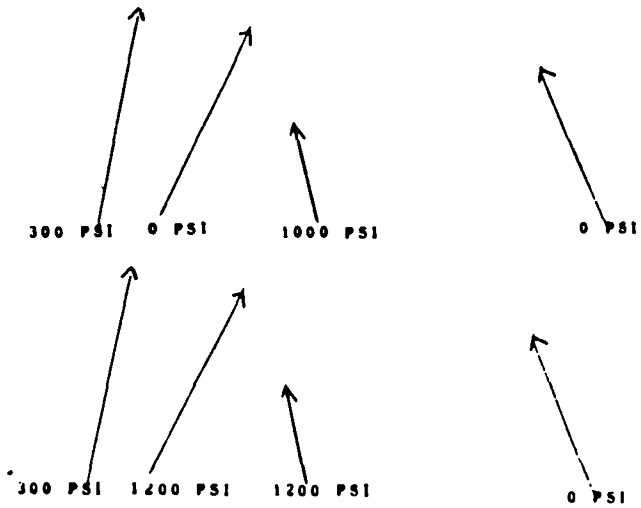
To provide specific instructor-supplied responses to student actions.

Instructor Actions

1. Recall display;
2. Press "punch display";
3. Press "erase screen";
4. Press "file control";
5. Delete response from file;
6. Repeat (1)-(5) for all instructor-stored responses.

Example

1. Recalled display;
2. Pressed "punch display";
3. Pressed "erase screen";
4. Pressed "file control";
5. Deleted response from file;
6. Repeated (1)-(5) for all instructor-stored responses (1)-(7).



*** NO. INTERNAL LEAKAGE IS NOT VISIBLE

*** NO. SYSTEM RELIEF VALVE DEFECTS CANNOT BE DIRECTLY DETERMINED BY VISUAL INSPECTION.

*** NO. DEFECTS IN THE POWER PUMP ARE NOT VISUALLY APPARENT.

*** NO. WITH THE POWER PUMP ON THE PRESSURE REGULATOR CANNOT BE ISOLATED FOR CHECKING.

*** NO. THE HAND PUMP CANNOT BE TESTED WHEN THE POWER PUMP IS ON.

Fig. A23 — Instructor-stored responses

REFERENCES

1. *Computer-Aided Instruction Plan 1968*, prepared by Headquarters ATC, DCS/Plans (ATXTD), August 10, 1968.

2. Rogers, J. L., "Current Problems in CAI," *Datamation*, September 1968, pp. 28-33.

3. *In-House Seminar on Computer Assisted Instruction*, report of the Department of Defense, December 8, 1967.

4. Seidel, R. J., *Project IMPACT: Computer-Administered Instruction Concepts and Initial Development*, Human Resources Research Office, George Washington University, Technical Report 69-3, March 1969.

5. Frye, C. H., "CAI Languages: Capabilities and Applications," *Datamation*, September 1968, pp. 34-37.

6. Zinn, K. L., "Programming Conversational Use of Computers for Instruction," *Proceedings: 1968 ACM National Conference*, 1968, pp. 85-92.

7. Boehm, B. W., V. R. Lamb, R. L. Mobley, and J. E. Rieber, *POGO: Programmer-Oriented Graphics Operation*, The Rand Corporation, RM-5825-PR, June 1969.

8. Rigney, J. W., *The Application of Computer Time-Sharing Technology to Training Requirements at the Electronics Schools Command, Mare Island*, Department of Psychology, University of Southern California, Technical Report No. 56, December 1967.

9. -----, *Training Corrective Maintenance and Performance on Electronics Equipment with CAI Terminals: I. A Feasibility Study*, Department of Psychology, University of Southern California, December 1966.

10. *Aircraft Pneudraulic Repairman/Technician*, CDC-42152-04-0269, Vol. 4, prepared by the Technical Training Center, Chanute AFB, Illinois, for the Extension Course Institute, Air University.

11. Engvold, K. J., and J. L. Hughes, "A General-Purpose Display Processing and Tutorial System," *Communications of the ACM*, Vol. 11, October 1968, pp. 697-702.