

Research Note 83-55

RESEARCH TO PROVIDE AUTOMATED TRAINING AND TESTING CAPABILITY
FOR AIR DEFENSE WEAPONS SYSTEMS

Ray E. Eberts
Honeywell, Inc.

John T. Larson
Contracting Officer's Technical Representative

Submitted by

Bruce W. Knerr, Acting Chief
Systems Manning Technical Area

and

Jerrold M. Levine, Director
Systems Research Laboratory



U. S. Army

Research Institute for the Behavioral and Social Sciences

December 1983

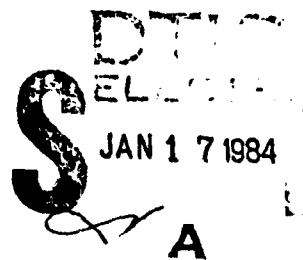
84 01 16 025

Approved for public release; distribution unlimited.

This report, as submitted by the contractor, has been cleared for release to Defense Technical Information Center (DTIC) to comply with regulatory requirements. It has been given no primary distribution other than to DTIC and will be available only through DTIC or other reference services such as the National Technical Information Service (NTIS). The views, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other official documentation.

DTIC FILE COPY

AD A136994



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER RN 83-55	2. GOVT ACCESSION NO. AD-A136994	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Research to Provide Automated Training and Testing Capability for Air Defense Weapon Systems		5. TYPE OF REPORT & PERIOD COVERED Final Report
		6. PERFORMING ORG. REPORT NUMBER 81SRC80
7. AUTHOR(s) Ray E. Eberts, Honeywell, Inc.		8. CONTRACT OR GRANT NUMBER(s) MDA 903-81-C-0503
9. PERFORMING ORGANIZATION NAME AND ADDRESS Perceptronics, Inc. Washington DC Operations 1911 N. Fort Myer Dr., Suite 308 Arlington, VA 22209		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 2Q263743A794
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE December 1983
		13. NUMBER OF PAGES 122
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) US Army Research Institute for the Behavioral and Social Sciences 5001 Eisenhower Ave. Alexandria, VA 22333		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Skill Qualification Test Embedded Testing Computer-Based Testing Missile Minder Air Defense System Programming Language for Interactive Teaching PLANIT		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The objective of the contract "Research to provide automated training and testing capability for Air Defense weapon systems" is to demonstrate the training and testing potential of a portable, microcomputer-based instructional delivery system utilizing videodisc laser image access, videographic overlay of computer generated imagery, task action responses via equipment console mock-up, and courseware authoring/control with enhanced Programming Language for Interactive Teaching (PLANIT). A 20-minute sample Skill Qualification Test (SQT) Hands-On Component will be developed for Military Occupational Specialty (MOS)		

DD FORM 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

20. (continued)

25L--the AN/TSQ-73 (Missile Minder) air defense artillery command and control system operator/repairer. This report provides PLANIT programming for a representative 3 of 12 tasks identified as critical AN/TSQ-73 console operator's tasks. The programming strategy allows errors to be tested in a realistic manner, by an algorithm which evaluates console mock-up switch sequences in terms of optional order and previously activated switches. The programming for these tasks can serve as a model for programming additional tasks.



Distribution	
13-1411	
140-145	
140-146	
140-147	
140-148	
140-149	
140-150	
140-151	
140-152	
140-153	
140-154	
140-155	
140-156	
140-157	
140-158	
140-159	
140-160	
140-161	
140-162	
140-163	
140-164	
140-165	
140-166	
140-167	
140-168	
140-169	
140-170	
140-171	
140-172	
140-173	
140-174	
140-175	
140-176	
140-177	
140-178	
140-179	
140-180	
140-181	
140-182	
140-183	
140-184	
140-185	
140-186	
140-187	
140-188	
140-189	
140-190	
140-191	
140-192	
140-193	
140-194	
140-195	
140-196	
140-197	
140-198	
140-199	
140-200	
140-201	
140-202	
140-203	
140-204	
140-205	
140-206	
140-207	
140-208	
140-209	
140-210	
140-211	
140-212	
140-213	
140-214	
140-215	
140-216	
140-217	
140-218	
140-219	
140-220	
140-221	
140-222	
140-223	
140-224	
140-225	
140-226	
140-227	
140-228	
140-229	
140-230	
140-231	
140-232	
140-233	
140-234	
140-235	
140-236	
140-237	
140-238	
140-239	
140-240	
140-241	
140-242	
140-243	
140-244	
140-245	
140-246	
140-247	
140-248	
140-249	
140-250	
140-251	
140-252	
140-253	
140-254	
140-255	
140-256	
140-257	
140-258	
140-259	
140-260	
140-261	
140-262	
140-263	
140-264	
140-265	
140-266	
140-267	
140-268	
140-269	
140-270	
140-271	
140-272	
140-273	
140-274	
140-275	
140-276	
140-277	
140-278	
140-279	
140-280	
140-281	
140-282	
140-283	
140-284	
140-285	
140-286	
140-287	
140-288	
140-289	
140-290	
140-291	
140-292	
140-293	
140-294	
140-295	
140-296	
140-297	
140-298	
140-299	
140-300	
140-301	
140-302	
140-303	
140-304	
140-305	
140-306	
140-307	
140-308	
140-309	
140-310	
140-311	
140-312	
140-313	
140-314	
140-315	
140-316	
140-317	
140-318	
140-319	
140-320	
140-321	
140-322	
140-323	
140-324	
140-325	
140-326	
140-327	
140-328	
140-329	
140-330	
140-331	
140-332	
140-333	
140-334	
140-335	
140-336	
140-337	
140-338	
140-339	
140-340	
140-341	
140-342	
140-343	
140-344	
140-345	
140-346	
140-347	
140-348	
140-349	
140-350	
140-351	
140-352	
140-353	
140-354	
140-355	
140-356	
140-357	
140-358	
140-359	
140-360	
140-361	
140-362	
140-363	
140-364	
140-365	
140-366	
140-367	
140-368	
140-369	
140-370	
140-371	
140-372	
140-373	
140-374	
140-375	
140-376	
140-377	
140-378	
140-379	
140-380	
140-381	
140-382	
140-383	
140-384	
140-385	
140-386	
140-387	
140-388	
140-389	
140-390	
140-391	
140-392	
140-393	
140-394	
140-395	
140-396	
140-397	
140-398	
140-399	
140-400	
140-401	
140-402	
140-403	
140-404	
140-405	
140-406	
140-407	
140-408	
140-409	
140-410	
140-411	
140-412	
140-413	
140-414	
140-415	
140-416	
140-417	
140-418	
140-419	
140-420	
140-421	
140-422	
140-423	
140-424	
140-425	
140-426	
140-427	
140-428	
140-429	
140-430	
140-431	
140-432	
140-433	
140-434	
140-435	
140-436	
140-437	
140-438	
140-439	
140-440	
140-441	
140-442	
140-443	
140-444	
140-445	
140-446	
140-447	
140-448	
140-449	
140-450	
140-451	
140-452	
140-453	
140-454	
140-455	
140-456	
140-457	
140-458	
140-459	
140-460	
140-461	
140-462	
140-463	
140-464	
140-465	
140-466	
140-467	
140-468	
140-469	
140-470	
140-471	
140-472	
140-473	
140-474	
140-475	
140-476	
140-477	
140-478	
140-479	
140-480	
140-481	
140-482	
140-483	
140-484	
140-485	
140-486	
140-487	
140-488	
140-489	
140-490	
140-491	
140-492	
140-493	
140-494	
140-495	
140-496	
140-497	
140-498	
140-499	
140-500	
140-501	
140-502	
140-503	
140-504	
140-505	
140-506	
140-507	
140-508	
140-509	
140-510	
140-511	
140-512	
140-513	
140-514	
140-515	
140-516	
140-517	
140-518	
140-519	
140-520	
140-521	
140-522	
140-523	
140-524	
140-525	
140-526	
140-527	
140-528	
140-529	
140-530	
140-531	
140-532	
140-533	
140-534	
140-535	
140-536	
140-537	
140-538	
140-539	
140-540	
140-541	
140-542	
140-543	
140-544	
140-545	
140-546	
140-547	
140-548	
140-549	
140-550	
140-551	
140-552	
140-553	
140-554	
140-555	
140-556	
140-557	
140-558	
140-559	
140-560	
140-561	
140-562	
140-563	
140-564	
140-565	
140-566	
140-567	
140-568	
140-569	
140-570	
140-571	
140-572	
140-573	
140-574	
140-575	
140-576	
140-577	
140-578	
140-579	
140-580	
140-581	
140-582	
140-583	
140-584	
140-585	
140-586	
140-587	
140-588	
140-589	
140-590	
140-591	
140-592	
140-593	
140-594	
140-595	
140-596	
140-597	
140-598	
140-599	
140-600	
140-601	
140-602	
140-603	
140-604	
140-605	
140-606	
140-607	
140-608	
140-609	
140-610	
140-611	
140-612	
140-613	
140-614	
140-615	
140-616	
140-617	
140-618	
140-619	
140-620	
140-621	
140-622	
140-623	
140-624	
140-625	
140-626	
140-627	
140-628	
140-629	
140-630	
140-631	
140-632	
140-633	
140-634	
140-635	
140-636	
140-637	
140-638	
140-639	
140-640	
140-641	
140-642	
140-643	
140-644	
140-645	
140-646	
140-647	
140-648	
140-649	
140-650	
140-651	
140-652	
140-653	
140-654	
140-655	
140-656	
140-657	
140-658	
140-659	
140-660	
140-661	
140-662	
140-663	
140-664	
140-665	
140-666	
140-667	
140-668	
140-669	
140-670	
140-671	
140-672	
140-673	
140-674	
140-675	
140-676	
140-677	
140-678	
140-679	
140-680	
140-681	
140-682	
140-683	
140-684	
140-685	
140-686	
140-687	
140-688	
140-689	
140-690	
140-691	
140-692	
140-693	
140-694	
140-695	
140-696	
140-697	
140-698	
140-699	
140-700	
140-701	
140-702	
140-703	
140-704	
140-705	
140-706	
140-707	
140-708	
140-709	
140-710	
140-711	
140-712	
140-713	
140-714	
140-715	
140-716	
140-717	
140-718	
140-719	
140-720	
140-721	
140-722	
140-723	
140-724	
140-725	
140-726	
140-727	
140-728	
140-729	
140-730	
140-731	
140-732	
140-733	
140-734	
140-735	
140-736	
140-737	
140-738	
140-739	
140-740	
140-741	
140-742	
140-743	
140-744	
140-745	
140-746	
140-747	
140-748	
140-749	
140-750	
140-751	
140-752	
140-753	
140-754	
140-755	
140-756	
140-757	
140-758	
140-759	
140-760	
140-761	
140-762	
140-763	
140-764	
140-765	
140-766	
140-767	
140-768	
140-769	
140-770	
140-771	
140-772	
140-773	
140-774	
140-775	
140-776	
140-777</	

CONTENTS

<u>Section</u>		<u>Page</u>
1	INTRODUCTION AND OVERVIEW	1
	1.1 Contract Overview	1
	1.2 Scope	1
2	PLANIT AND THE SQT	3
	2.1 Introduction to PLANIT	3
	2.2 SQT Overview	5
3	COURSEWARE MATERIALS	11
	3.1 Introduction	11
	3.2 Features of the Courseware	12
4	COURSEWARE	17
	4.1 Introduction	17
	4.2 Task 1 - Enter Fixed Point Site	17
	4.3 Task 2 - Perform Hooking Procedures	19
	4.4 Task 3 - Enter FU Data on DDG and ARO	22
	4.5 Other Features	24
	4.6 Summary	24
APPENDIX A	ENTER FIXED POINT SITE	25
APPENDIX B	PERFORM HOOKING PROCEDURES	57
APPENDIX C	ENTER FU DATA ON DDG AND ARO	83
APPENDIX D	SETUP AND INSTRUCTIONS	113

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1	An Example of a Q Frame	4
2	An Example of a D Frame	6
3	An Example of a P Frame	7
4	Hands-On Test Worksheet - Enter Fixed Point Site	18
5	Example Responses	20
6	Hands-On Test Worksheet - Perform Hooking Procedures	21
7	Hands-On Test Worksheet - Enter FU Data on DDG and ARO	23

SECTION 1

INTRODUCTION AND OVERVIEW

1.1 CONTRACT OVERVIEW

This report contains courseware materials for the contract "Research to Provide Automated Training and Testing Capability for Air Defense Weapon Systems" (Contract No. MDA 903-81-C-0218). This contract is being performed by Perceptronics and Honeywell for the U.S. Army Research Institute for the Behavioral and Social Sciences. The objective of this contract is to demonstrate the training and testing potential of a microcomputer-based videodisc instructional delivery system for U.S. Army Air Defense weapon systems. Sample Skill Qualification Test (SQT) Hands-On Component (HOC) are to be developed for the MOS 25L -- the AN/TSQ-73 (Missile Minder) air defense artillery command and control system operator/repairer.

The SQT is to be administered on a portable, relatively inexpensive trainer/tester delivery system, designed and built by Perceptronics. The functional capabilities of the system include an LSI 11/23 microprocessor with memory management, videodisc laser image access, videographic overlay of computer-generated imagery, and interactive real-time task action responses via equipment console mock-up. The courseware is to be done by using the PLANIT (Programming LANguage for Interactive Teaching) authoring system. PLANIT will handle the instruction capability for the simulator.

SECTION 2

PLANIT AND THE SQT

2.1 INTRODUCTION TO PLANIT

PLANIT is a computer assisted instruction (CAI) and automatic testing system. It can be installed on almost any computer-based system to be used as a framework for instruction or testing. PLANIT operates in two modes: the authoring mode which the programmer uses to program the lesson or tasks; and the student mode which the student uses to take the tests or sequence through the instructions.

The main programming unit for PLANIT is the frame. A primary purpose of the frame is to provide a basis to query the student, receive the answer, and interpret and evaluate the student answers. Within the program, frames can be entered sequentially (from the previous frame) or control can branch to various frames depending on the particular student answer that was given. Frames can also be used as basic programming units; IF-THEN statements or FOR loops are examples of programming statements that can be used. When a student is sequencing through a task, PLANIT updates and stores information about the correctness of responses for each student.

A frame itself is further divided into groups and lines. The lines are used for recording the actual programming code. The number of groups within a frame depends on the particular kind of frame. Three different kinds of frames are used in the Missile Minder courseware: Q (Question), D (Decision), and P (Programming) frames. Each of these frames will be described in more detail.

The Q Frame. An example of a Q (or question) frame is provided in Figure 1. More lines for each of the groups can be added as needed. Group 1 is the label.

A label can be a number (a whole number or one with a decimal point) or a name label. Group 2 is the text. This is what is printed out to the student. It can include instructions or test questions. Group 3 contains the answers to the text of the previous group. Each possible answer is labeled and referred for action to Group 4. Group 4 is the action that should be taken depending on each answer. Possible actions include selective feedback, branching to other frames, and changing the values of variables.

The D Frame. An example of a D (or decision) frame is provided in Figure 2. A D frame is composed of two groups: the label (same as for the Q frame) and the criteria group. The criteria group is used to prescribe actions based on student performance. The actions are usually prescribed through the use of IF-THEN statements or FOR loops.

The P Frame. An example of a P (or programming) frame is provided in Figure 3. A P frame is also composed of two groups: the label and the statements group. The P frame functions as a subroutine. Statements that always occur together in the courseware can be combined in a P frame. Another frame can then call this P frame. After execution of the commands, the P frame will return control to the succeeding statement in the calling frame. The use of a P frame can make the program more efficient.

2.2 SQT OVERVIEW

The SQT Requirements Analysis Report (Interim Report) provides a summary of the 12 Missile Minder tasks that were to be tested. Each of the tasks is broken down into discrete performance measures that can be used to evaluate student performance. The number of performance measures for the 12 tasks range from four to 13 and include the following kinds of actions:

- o Switch action

FRAME (P) LABEL=
G2. STATEMENTS
1)

Figure 3

- o Keyboard entries
- o Hooking (a procedure where a symbol is specified on the radar screen in real-time and is flagged for further processing)

This report will describe how the courseware can be programmed to evaluate these kinds of performance measures.

For several reasons, evaluation of these kinds of performance is different from what PLANIT was originally designed to do. The Missile Minder system uses very little text processing and, rather, operates in real-time. The operator is required to manipulate graphics symbols on a radar screen. Furthermore, a CAI and testing system for Missile Minder will be required to analyze sequences of actions instead of discrete keyboard entries as is typical for CAI programs. The challenge was to modify PLANIT so that it could be used for this real-time, graphics manipulation system. Many of the modifications to PLANIT were done directly to the courseware by Perceptronics and PLANIT's developer, Dr. Charles Frye. The courseware described in this report will refer to these modifications at various points but will mostly rely on changing PLANIT within its existing structure to enable the needed capabilities for Missile Minder.

PLANIT is designed to handle only three kinds of errors: deletions, wrong entries, and exceeding the time limit. For Missile Minder, six different types of errors are possible within a task:

- o Extra switch actions were included, but not needed
- o Action deleted
- o Substituted wrong action for the correct one
- o Wrong keyboard entry
- o Action included, but done out of order
- o Task exceeded the allotted time limit

When taking a test, it is important that the student be given the most detailed performance feedback possible. Analyzing a sequence of actions is difficult for an instructor and, thus, is difficult for PLANIT. The same response can be interpreted in several ways. The challenge for this task was to write the courseware so that the student answers could be evaluated as intelligently as possible and the students could be given useful feedback for evaluating their own performance.

SECTION 3

COURSEWARE MATERIALS

3.1 INTRODUCTION

To present the courseware, it has been written in frames such as those presented in Figures 1-3. Each page will represent a separate frame. The courseware material for three of the tasks is presented in Appendixes A, B, and C. The courseware presented here has not been implemented and tested on the simulator; it has only been desk-checked. It is expected that some bugs in the programming will turn up when this courseware is implemented.

The following three tasks were programmed:

- o Task S079 - Enter fixed point site
- o Task S071 - Perform hooking procedures
- o Task S078 - Enter FU data on DDG and ARO

These three tasks were programmed because they were judged to be representative of the other tasks and could be used to demonstrate the full power of the courseware.

The goal of the courseware programming was to be able to "intelligently" interpret errors. Error interpretation presented several problems. First, there are many possible sequences of responses. As an example for a simple three switch-action sequence, if there are three operational switches, all other responses are lumped into one category, and deletions are counted, there would be 5^3 or 125 possible response sequences. This doesn't include the possibility that more than three switch actions could erroneously be made during the task.

Having a discrete path for each sequence is not possible; a method must be found where novel responses can be interpreted. A second problem is that there can be several different ways to interpret errors. As an example, if all three switch actions were included but the first one was done last, then it could be interpreted either as: 1) the first switch action was performed out of order; or 2) the first switch action was deleted and there was an extra, unneeded response at the end of the sequence. The program should be able to determine which is more plausible if it is to act "intelligently."

The method that is used to analyze the errors is a hypothesis-test method. The program makes an hypothesis about the type of error; if later information disconfirms the hypothesis, then another hypothesis is made on the basis of the new information. As an example of how this works, consider a sequence-- A, B, C--in which the student erroneously responds with X, B, C. The program evaluates the responses as they occur. When the X is entered, the program hypothesizes that it is an extra, unneeded response at the beginning. The next response, B, provides information that disconfirms that hypothesis: X is, rather, a substitution for the correct response, A. The program then makes the new hypothesis that a substitution error has occurred in the first position. The last response, C, offers further evidence for that hypothesis. Using this hypothesis-test method, the program can intelligently interpret errors and respond to novel responses.

3.2 FEATURES OF THE COURSEWARE

This section describes the features that are common to the courseware programs presented in the appendix.

Error Storage

Errors are stored in arrays with a length equal to the number of possible switch actions or keyboard entires, as appropriate for the error type. The following six error types are stored:

- o ARRAY1 - an unnecessary response was recorded
- o ARRAY2 - a switch action was deleted
- o ARRAY3 - substituted a wrong switch action for a correct one
- o ARRAY4 - a wrong keyboard entry was made
- o ARRAY5 - a switch action was performed out of order
- o ARRAY6 - the task took too long to complete

A 0 in the array means that no error occurred; 1 or higher means that an error occurred. The value can be greater than 1 for ARRAY1 when several unneeded responses can and might occur. The error arrays are changed to reflect the current hypothesis.

PM Array

The PM array is used in the feedback stage to store the correctness of each performance measure for a task. The size of the array is the same as the number of performance measures for the task; it can range from four to 13. Again, a 0 means that the performance measure was performed correctly and a 1 means that an error occurred for the particular performance measure indexed.

Task Termination

Task termination is a difficult problem. If the student does the task incorrectly, the last response listed for the task may or may not be the last response that the student makes. Several possibilities exist for how to terminate the task:

- o Terminate on the last response listed
- o Terminate after the required number of switch actions
- o Terminate after a certain time limit between responses
- o Have the student push a special switch when the task is done

The first three methods were rejected because they couldn't allow for certain kinds of errors or they would place long, unnecessary pauses in the task. The last method, a special switch, was chosen. This method gives more control to the student. In the courseware, this switch is listed as the "ACT" action switch.

Switch Actions

In the courseware, each switch action will be interpreted as a discrete event. A switch action will prompt the Q frame to perform the appropriate action contingent upon the response.

Keyboard Responses

The purpose of certain switch actions is to read the keyboard entries. Thus, unlike switch actions, a keyboard entry will not prompt the Q frame to proceed to the appropriate action. Rather, the program will determine, based upon a switch action, when the keyboard entries are to be read.

Hooking

A hooking procedure is used to specify that a target, fire unit, or other radar symbol is to be processed further. Once a symbol is hooked, a graphics feature appears around the hooked symbol and follows it across the radar screen. Thus, each symbol must be potentially "hookable." To accomplish this, each radar symbol must have a hook symbol around it which can either be in the off or on state. When hooked, the symbol would change from off to on. This capability is a modification to the courseware that is not handled by Honeywell. During the courseware, the program will refer to, but not specify, this capability.

Cursor Control

One of the methods used to hook a symbol on the radar screen is to use the position tab, which controls a cursor, to position the cursor over the symbol. This feature should be hardwired and so is not considered in this courseware.

SECTION 4

COURSEWARE

4.1 INTRODUCTION

The courseware for each of the three tasks appears in the appendix. In this section of the report, the courseware will be described and summarized.

4.2 TASK 1 - ENTER FIXED POINT SITE

The hands-on test worksheet for this task, taken from the Interim Report, is reproduced as Figure 4. The courseware for the task is in Appendix A. This task was chosen as representative of the others because it must be performed exactly in the order listed in Figure 4. It provides a good example of how the hypothesis-test method can be used to interpret the results.

To explain how the program works, refer to Appendix A. The first three frames are the instructions. They specify the specific conditions and what the students are required to do. The instructions are taken from the appropriate boxes in Figure 4. To advance from one frame to the next in the instructions, the student is required to push the action (ACT) switch.

The task starts at Frame 1 and waits on this frame until any switch action is made. The program spells out the actions to be taken by listing the allowable switch actions (TS1, TF4, or TF5). The action group in the frame indicates what should be done if other switches are pushed ("- " in G4). Also, the action switch will terminate the task. If the student takes too long, the program will branch to the TIMOUT frame and also terminate the task.

HANDS-ON TEST WORKSHEET

TASK NUMBER: 441-074-1035	LOCAL NUMBER: S079	SKILL LEVEL: 1
TASK STATEMENT: Enter fixed point site.		
CONDITIONS: Task is performed on the AN/TSQ-73 console. System is energized and operational. TM 9-1430-652-10-3 is available. System is configured to either Battalion or Group. GEOREF coordinates for a fixed point site will be provided.		
STANDARDS: A fixed point site must be entered IAW TM 9-1430-652-10-3, Figure 4-28, without error within 2 minutes.		
PREREQUISITE DATA: Soldier is required to enter a radar site at GEOREF coordinates <u>NKHH4715</u> .		
PERFORMANCE MEASURES: 1. Press TASK SELECTIONS-TRACK DATA. 2. Enter GEOREF coordinates on AN keyboard. 3. Press TASK FUNCTIONS-POSN ENTRY. 4. Enter symbol code on AN keyboard. 5. Press TASK FUNCTIONS-SITE INIT-IDENT. 6. Complete performance measures 1-5 within 2 minutes.		INPUTS: TS1 "NKHH4715" TF4 "R" TF5
REFERENCES: TM 9-1430-652-10-3, Figure 4-28, p. 4-47; SM FM 44-25L, p. 2-199		
CONSOLE SET-UP: Map shown on PPI.		
CONSOLE FUNCTIONALITY: TS1, TF4, TF5. AN keyboard		

Depending on the switch action, the program will branch to other frames within the program to continue with the hypothesis and test procedure. It is constantly evaluating responses and filling in the error arrays when appropriate.

The last frames in the program are the feedback frames. FEED provides detailed feedback on the type of error and the point that the error occurred. SUMMARY provides a summary of the results and indicates, for each performance measure, whether it was done correctly or not.

Figure 5 contains a list of some of the possible sequences of responses for this task and how the system would interpret that sequence. In this figure, "other" stands for a response other than those activated (TS1, TF4, or TF5) and PM stands for performance measure (refer back to Figure 4). For this example the keyboard entries are evaluated separately, a fairly straightforward process, and are not presented in the examples. This is just a small subset of the possible responses but provides an example of how the program intelligently interprets the response sequences.

4.3 TASK 2 - PERFORM HOOKING PROCEDURES

The hands-on test worksheet is reproduced as Figure 6 and the courseware for this task is presented in Appendix B. This task was chosen as representative of the others because it is embedded in most of the tasks. Also, it provides an example of a real-time control problem. The task is divided into four subtasks; the four can be performed in any order desired. Once within a subtask, however, the steps must be performed in order.

The first couple instruction frames and the feedback frames are much the same as those in the first task. The fourth frame, labelled HOOK, is used to branch the program to the appropriate subtask. The branch occurs according to the switch action taken. If the switch action is the first action in a subtask it will branch to that particular subtask. If the program is at HOOK and

<u>Response</u>					<u>Interpretation</u>
TS1		TF4		TF5	Correct response
TS1				TF5	PM3 deleted
TS1		X		TF5	PM3 substituted
TS1		TF4		X	PM5 substituted
X	X	TS1	TF4	TF5	two unneeded responses before PM1
TS1	TF4	X		TF5	one unneeded response before PM5
TF5		TF4		TS1	PM1, PM3, PM5 performed out of order
TF4		TS1		TF5	PM1, PM2 performed out of order
X		TF4		TF5 TS1	PM1 substituted, unneeded response after PM5

PM = performance measure (see Figure 4)

X = response other than TS1, TF4, or TF5

Figure 5. Example Responses

HANDS-ON TEST WORKSHEET

TASK NUMBER: 441-074-1029	LOCAL NUMBER: S071	SKILL LEVEL: 1
TASK STATEMENT: Perform hooking procedures.		
CONDITIONS: Task is performed on the AN/TSQ-73 console. System is energized and operational. TM 9-1430-652-10-3 is available. System is configured to either Battalion or Group. Requirements for hooking tracks by four methods will be provided.		
STANDARDS: Number, GEOREF, position, and sequence hooking procedures must be performed IAW TM 9-1430-652-10-3, Figure 4-14 and Figure 4-13, without error within 5 minutes.		
PREREQUISITE DATA: Soldier is required to perform hooking procedures as follows: 1) number hook on ATDL-1 track number AB 120; b) GEOREF position hook for a track at coordinates PDEC 3040; c) position hook for the track designated as AB121; and d) sequence hook for all high-threat tracks.		
PERFORMANCE MEASURES: A. Number hook 1. Enter track number on AN keyboard. 2. Press TASK FUNCTIONS-NUMBER HOOK. B. GEOREF position hook 3. Press TASK SELECTIONS-TRACK DATA. 4. Enter GEOREF coordinates on AN keyboard. 5. Press TASK FUNCTIONS-POSN ENTRY.. 6. Press TASK FUNCTIONS-POSN HOOK. C. Position hook 7. Position tab on symbol to be hooked. 8. Press TASK FUNCTIONS-POSN HOOK. D. Sequence hook 9. Enter sequence hook category on AN keyboard. 10. Press TASK FUNCTIONS-SEQ HOOK. 11. Complete performance measures 1-10 within 5 minutes.		INPUTS: "AB120" MI4 TS1 "PDEC3040" TF4 MI6 TA1 (opt.) TA2 MI6 "H" MI3 MI3 (opt.)
REFERENCES: TM 9-1430-652-10-3, Figure 4-14, Figure 4-13, p. 4-29, p. 4-27; SM FM 44-25L, p. 2-186, p. 2-185		
CONSOLE SET-UP: At least four tracks shown on PPI. No symbols are hooked. One track is located at coordinates <u>PDEC 3040</u> . Two tracks are designated as <u>AB120</u> and <u>AB121</u> . Two tracks are designated as high-threat tracks.		
CONSOLE FUNCTIONALITY: TS1, MI4, MI3, MI6, TF4, TA1, TA2, AN keyboard		

Figure 6

receives an input other than the four subtask switch actions, it will interpret this response as unneeded and wait until it receives one that it can interpret.

Another different feature of this task is that it assumes that an action is deleted until it encounters information to indicate otherwise. Also, the subtask for the position hook largely depends on cursor control, a function that cannot be done by PLANIT courseware. This capability takes special commands that were not developed by Honeywell.

4.4 TASK 3 - ENTER FU DATA ON DDG AND ARO

The hands-on test worksheet for this task is reproduced as Figure 7 and the courseware for the task is presented in Appendix C. This task was chosen because, of all the tasks in the interim report, it contains the most performance measures (13). The switch actions can be done in any order and the task contains a large number of keyboard entries.

The instruction and feedback frames are similar to the previous two tasks and will not be discussed further. Because the switch actions can be performed in any order, this presents a problem for how to interpret "other" responses; those not listed the SETUP frame. The "other" responses can be interpreted as either a substitution or an unneeded response. The program interprets the responses as follows. When an "other" response is received, it queries the keyboard input to see if the "other" response can be interpreted, as a substitution, from the keyboard entry. If the keyboard entry matches one of the possible entries and the switch action that triggers the matched keyboard entry has not been performed yet, then the program will hypothesize that the "other" response is a substitution for the switch action that was supposed to trigger the keyboard entry. Otherwise, it will interpret it as an unneeded response. If the switch action, that was hypothesized to be substituted for is performed later, the hypothesis will have to be changed. In this case the "other" response is reinterpreted to be an unneeded response. Besides this feature, the programming for this task is fairly simple.

HANDS-ON TEST WORKSHEET

TASK NUMBER: 441-074-1034	LOCAL NUMBER: S078	SKILL LEVEL: 1												
TASK STATEMENT: Enter FU data on DDG and ARO.														
CONDITIONS: Task is performed on the AN/TSQ-73 console. System is energized and operational. TM 9-1430-652-10-3 is available. System is configured to Battalion. Required fire unit data will be provided.														
STANDARDS: Fire unit data must be entered IAW TM 9-1430-652-10-3, Figure 4-27, without error within 6 minutes.														
PREREQUISITE DATA: Soldier is required to enter the following fire unit data on ARO and DDG: <table border="0"> <tr> <td>FU ADDRESS</td> <td>A12</td> </tr> <tr> <td>ROW NUMBER</td> <td>05</td> </tr> <tr> <td>FU ALERT STATUS</td> <td>15</td> </tr> <tr> <td>HOT MISSILE COUNT</td> <td>20 (WARHEAD TYPE C)</td> </tr> <tr> <td>COLD MISSILE COUNT</td> <td>15 (WARHEAD TYPE C)</td> </tr> <tr> <td>FU STATUS</td> <td>READY</td> </tr> </table>			FU ADDRESS	A12	ROW NUMBER	05	FU ALERT STATUS	15	HOT MISSILE COUNT	20 (WARHEAD TYPE C)	COLD MISSILE COUNT	15 (WARHEAD TYPE C)	FU STATUS	READY
FU ADDRESS	A12													
ROW NUMBER	05													
FU ALERT STATUS	15													
HOT MISSILE COUNT	20 (WARHEAD TYPE C)													
COLD MISSILE COUNT	15 (WARHEAD TYPE C)													
FU STATUS	READY													
PERFORMANCE MEASURES: - Press TASK SELECTIONS-STATUS BOARD DATA. - Hook fire unit. - Row number - Enter DDG row number on AN keyboard - Press TASK FUNCTIONS-ASSIGN ROW NUMBER - Alert status - Enter FU alert status on AN keyboard. - Press TASK FUNCTIONS-FU ALERT STATUS - Missile count - Enter missile type and count on AN keyboard. - Press TASK FUNCTIONS-MSL COUNT HOT. - Enter missile type and count on AN keyboard. - Press TASK FUNCTIONS-MSL COUNT COLD. - FU status - Enter fire unit status on AN keyboard. - Press TASK FUNCTIONS-FU STATUS. - Complete performance measures 1-12 within 6 minutes.		INPUTS: TS9 "A1205" TF1 "15" TF3 "C020" TF5 "C15" TF6 "R" TF4												
REFERENCES: TM 9-1430-652-10-3, Figure 4-27, p. 4-45; SM FM 44-25L, p. 2-198														
CONSOLE SET-UP: Fire unit designated as A12 is shown on PPI. FU is not hooked.														
CONSOLE FUNCTIONALITY: TS9, TF1, TF3, TF4, TF5, TF6, AN keyboard, hooking procedure														

4.5 OTHER FEATURES

The courseware can also be designed to enable the student to enter information such as his name, social security number, rank, and the date. This can be done similar to how it was done for the TACFIRE PLANIT system that Honeywell authored. The relevant portion of that courseware has been reproduced in Appendix D. Also, at the beginning of the testing session, the student is required to read and page through a set of general instructions.

Presenting the instructions is very simple on PLANIT. The text is presented on a Q frame and the system waits for the appropriate response (such as an action switch input) to page to the next set of instructions. The instructions for the testing session are presented in text form in Appendix D.

Another consideration in programming this Missile Minder simulator is that it can be designed to make the user-computer interaction very simple through the use of the touch panel. The touch panel could be used in the instruction and interaction segments. A possibility is a menu selection system that could be used to enter the date or select tests by touching the panel. Students (and instructors) would find this system very easy to use.

4.6 SUMMARY

These three tasks show how PLANIT can be used to intelligently interpret and evaluate a response string in real-time. Modifying PLANIT to accomplish this was a difficult process. At many points I was frustrated because the PLANIT structure seemed to hinder the programming rather than facilitate it. I felt that a language such as FORTRAN could have handled some of the problems much easier. Nevertheless, PLANIT has been demonstrated, through these examples, to handle evaluation of these real-time tasks.

APPENDIX A

ENTER FIXED POINT SITE

26
✓ 19

FRAME (Q) LABEL= START 1

G2. TEXT

- 1) TEST 1
- 2) ENTER FIXED POINT SITE
- 3) (TASK 441-074-1035)
- 4)
- 5)

(MORE)

G3. ANSWERS

- 1) O WAIT 60
- 2) A+ACT

G4. ACTIONS

- 1) -F: PRESS THE "ACTION" SWITCH TO PROCEED

FRAME (Q) LABEL= COND 1

G2. TEXT

- 1) THE FOLLOWING CONDITIONS PERTAIN TO THIS TEST:
- 2) 1. YOU ARE OPERATING A GROUP OR BATTALION MISSILE
- 3) MINDER COMPUTER SYSTEM WHICH IS ENERGIZED
- 4) AND OPERATIONAL
- 5) 2. YOU MAY REFER TO THE APPROPRIATE TECHNICAL
- 6) MANUAL(S)

G3. ANSWERS

- 1) 0 WAIT 60
- 2) A+ACT

G4. ACTIONS

- 1) '-F:PRESS THE "ACTION" SWITCH TO PROCEED

FRAME (Q) LABEL=INSTRUCT 1

G2. TEXT

- 1) YOU ARE REQUIRED TO DO THE FOLLOWING.
- 2)
- 3) ENTER A RADAR SITE AT GEOREF COORDINATES NKHH4715
- 4) COMPLETE THE TASK WITHIN 2 MINUTES
- 5) (MORE)

G3. ANSWERS

- 1) O WAIT 60
- 2) A+ACT

G4. ACTIONS

- 1) '-F:PRESS THE "ACTION" SWITCH TO PROCEED

FRAME (Q) LABEL= 1

G2. TEXT

1) \$

G3. ANSWERS

- 1) A+TS1
- 2) B+TF4
- 3) C+TF5
- 4) D+ACT

G4. ACTIONS

- 1) A B:2
- 2) B C:FLAG=FLAG-1 B:1.1
- 3) C C:FLAG=FLAG-1 B:1.2
- 4) - B:1.3
- 5) D B:1.4
- 6) ' B:TIMOUT

FRAME (D) LABEL= 1.1

G2. CRITERIA

- (1) C:FRMNUM=1.1
- 2) IF FLAG EQ 0 B:READAN 1
- 3) C:ARRAY5(2)=ARRAY5(2)+1
- 4) IF ARRAY1(1) GT 0 C:ARRAY1(1)=ARRAY1(1)-1 C:ARRAY3(1)=1
C:ARRAY6(2)=ARRAY6(2)-1 B:3
- 5) C:ARRAY2(1)=1 B:3

FRAME (D) LABEL=1.2

G2. CRITERIA

- (1) C:FRMNUM=1.2
- 2) IF FLAG EQ 0 B:READAN2
- 3) IF ARRAY1(1) GT 1 C:ARRAY1(1)=ARRAY1(1)=2
C:ARRAY2(1)=ARRAY2(1)+1 C:ARRAY2(2)=ARRAY2(2)+1
- 4) C:ARRAY5(3)=ARRAY5(3)+1
- 5) B:2

FRAME (D) LABEL= 1.3

G2. CRITERIA

- (1) C:ARRAY1(1)=ARRAY1(1)+1
- (2) B:1

FRAME (P) LABEL=1.4

G2. STATEMENTS

- 1) C:ARRAY2(1)=1 C:ARRAY2(2)=1 C:ARRAY2(3)=1
- 2) B:FINISH

FRAME (Q) LABEL=2

G2. TEXT

1) \$

G3. ANSWERS

1) A+TF4
2) B+TS1
3) C+TF5
4) D+ACT

G4. ACTIONS

1) A C:FLAG=FLAG-1 B:2.1
2) B C:ARRAY6(1)=ARRAY6(1)+1 B:3
3) C C:FLAG=FLAG-1 B:2.2
4) D B:2.4
5) - B:2.3
6) ' B:TIMEOUT

FRAME (0) LABEL=2.1

G2. CRITERIA

- 1) C:FRMNUM=2.1
- 2) IF FLAG=0 B:READAN1
- 3) B:3

FRAME (D) LABEL= 2.2

G2. CRITERIA

- (1) C:FRMNUM=2.1
- (2) IF FLAG EQ 0 B:READAN2
- (3) C:ARRAY5(3)=ARRAY5(3)+1
- (4) IF ARRAY1(2) GT 0 C:ARRAY3(2)=1 C:ARRAY5(3)=ARRAY5(3)-1
C:ARRAY1(2)=ARRAY1(2)-1 B:4
- (5) C:ARRAY2(2)=ARRAY2(2)-1 B:3

FRAME (D) LABEL=2.3

G2. CRITERIA

- 1) C:ARRAY1(2)=ARRAY1(2)+1
- 2) B:2

FRAME (D) LABEL=2.4

G2. CRITERIA

- (1) IF ARRAY1(1) GT 0 C:ARRAY3(1)=ARRAY3(1)+1
C:ARRAY1(1)=ARRAY(1)-1
- (2) IF ARRAY1(1) GT 0 C:ARRAY3(2)=ARRAY3(2)+1
C:ARRAY1(1)=ARRAY1(1)-1
- (3) C:ARRAY2(2)=1 C:ARRAY2(3)=1
- (4) B:TIMEOUT

FRAME (Q) LABEL= 3

G2. TEXT

1) S

G3. ANSWERS

- 1) A+TF5
- 2) B+TS1
- 3) C+TF4
- 4) D+ACT

G4. ACTIONS

- 1) A C:FLAG=FLAG-1 B:3.1
- 2) B C:ARRAY5(1)=ARRAY5(1)+1 B:4
- 3) C C:FLAG=FLAG-1 B:3.2
- 4) D B:3.3
- 5) - B:3.4
- 6) ' B:TIMEOUT

FRAME (D) LABEL=3.1

G2. CRITERIA

- 1) C:FRMNUM=3.1
- 2) IF FLAG EQ 0 B:READAN1
- 3) B:4

FRAME (D) LABEL=3.2

G2. CRITERIA

- (1) C:FRMNUM=3.2
- (2) IF FLAG EQ 0 B:READAN2
- (3) C:ARRAY5(2)=ARRAY5(2)+1
- (4) IF ARRAY5(2) AND ARRAY2(1) GT 0 C:ARRAY2(1)=ARRAY2(1)-1
- (5) B:4

FRAME (D) LABEL= 3.3

G2. CRITERIA

- (1) IF ARRAY1(3) GT 0 C:ARRAY3(3)=ARRAY3(3)+1
C:ARRAY1(3)=ARRAY1(3)-1 B:FINISH
- (2) C:ARRAY2(3)=ARRAY2(3)+1
- (3) B:FINISH

FRAME (D) LABEL= 3.4

G2. CRITERIA

- 1) B:ARRAY1(3)=ARRAY1(3)+1
- 2) B:3

FRAME (Q) LABEL= 4

G2. TEXT

1) \$

G3. ANSWERS

1) A+TS1
2) B+TF4
3) C+TF5
4) D+ACT

G4. ACTIONS

1) A B:4.1
2) B C:FLAG=FLAG-1 B:4.2
3) C C:FLAG=FLAG-1 B:4.3
4) D B:FINISH
5) - B:4.4
6) ' B:TIMEOUT

FRAME (D) LABEL= 4.1

G2. CRITERIA

- 1) C:ARRAY5(1)=ARRAY5(1)+1
- 2) B:4

FRAME (D) LABEL = 4.2

G2. CRITERIA

- (1) C:FRMNUM=4.2
- (2) IF FLAG EQ 0 B:READAN1
- (3) C:ARRAY5(2)=ARRAY5(2)+1
- (4) B:4

FRAME (D) LABEL= 4.3

G2. CRITERIA

- 1) C:FRMNUM=4.3
- 2) IF FLAG EQ 0 B:READAN2
- 3) C:ARRAY5(3)=ARRAY5(3)+1
- 4) B:4

FRAME (D) LABEL=4.4

G2. CRITERIA

- 1) C:ARRAY1(3)=ARRAY1(3)+1
- 2) B:4

FRAME (Q) LABEL=READAN 1

G2. TEXT

1) \$

G3. ANSWERS

1) A+NKHH4715

G4. ACTIONS

1) A C:FLAG=FLAG+1 B:FRMNUN
2) - C:FLAG=FLAG+1 C:ARRAY4(1)=ARRAY4(1)+1 B:FRMNUM

FRAME (Q) LABEL= READAN2

G2. TEXT

1) \$

G3. ANSWERS

1) A+R

G4. ACTIONS

1) A C: FLAG=FLAG+1

2) - C: FLAG=FLAG+1 C: ARRAY4(1)=ARRAY4(1)+1 B: FRMNUM

FRAME (D) LABEL=FINISH

G2. CRITERIA

- (1) FOR I=1,2
- (2) IF ARRAY2(I) GT 0 AND ARRAY6(I+1) GT 0
THEN ARRAY5(I+1)=ARRAY5(I+1)-1 B:L9
- (3) FOR (I=1,3) IF ARRAY5(I) GT 0 AND ARRAY5(I) GT 0
THEN ARRAY2(I)=ARRAY2(I)-1
- (4) B:FEED

FRAME (D) LABEL=FEED

G2. CRITERIA

- 1) C:I=0
- 2) IF ARRAY 1(I) EQ 0 B:L4
- 3) C:ARRAY 1(I) UNNEEDED SWITCH ACTIONS AT STEP' I*2 C:PM(I)=1
- 4) C:I=I+1
- 5) IF I LT 3 B:L2
- 6) C:I=1
- 7) IF ARRAY 2(3) EQ 0 B:L9
- 8) C:'THE SWITCH ACTION AT STEP' I*2 'was deleted' C:PM(I*2)=1
- 9) C:I=I+1
- 10) IF I LT 3 B:L7
- 11) C:I=1
- 12) IF ARRAY 3(I) EQ 0 B:L14
- 13) C:'THE SWITCH ACTION AT STEP' 2*I 'WAS SUBSTITUTED WITH A WRONG RESPONSE' C:PM(I*2)=1
- 14) C:I=I+1
- 15) IF I LT 3 B:L12
- 16) C:I=1
- 17) IF ARRAY 4(I) EQ 0 B:L21
- 18) C:'THE WRONG INFORMATION WAS ENTERED ON THE KEYBOARD AT STEP' C:I+2 C:PM(I+2)=1
- 19) C:I=I+1
- 20) IF I LT 2 B:L17
- 21) C:I=1
- 22) IF ARRAY 5(I) EQ 0 B:L26
- 23) C:'SWITCH ACTION' 2*I 'WAS PERFORMED OUT OF ORDER' C:PM(I)=1
- 24) C:I=I+1
- 25) IF I LT 3 B:L22
- 26) IF PM(I) EQ 0 FOR (I=1,6) F:TASK WAS PERFORMED CORRECTLY
- 27) B:SUMMARY

FRAME (D) LABEL=TIMOUT

G2. CRITERIA

- 1) F:THE TASK EXCEEDED THE TIME LIMIT
- 2) C:PM(6)=1

FRAME (D) LABEL= SUMMARY

G2. CRITERIA

- 1) F:*****
- 2) F: SUMMARY
- 3) F: PERFORMANCE MEASURE PASS FAIL
- 4) C: I=1 C: COLUMN=26+PM(I)*8
- 5) ALIGN X, COLUMN

4,6

58

100

APPENDIX B

PERFORM HOOKING PROCEDURES

38

65

FRAME (Q) LABEL= START2

G2. TEXT

- 1) TEST 2
- 2) PERFORM HOOKING PROCEDURES
- 3) (TASK 441-074-1029)
- 4)
- 5) (MORE)

G3. ANSWERS

- 1) O WAIT 60
- 2) A+ACT

G4. ACTIONS

- 1) '-PRESS THE "ACTION" SWITCH TO PROCEED

FRAME (Q) LABEL=COND2

G2. TEXT

- 1) THE FOLLOWING CONDITIONS PERTAIN TO THIS TEST
- 2) 1. YOU ARE OPERATING A GROUP OR BATTALION MISSILE
- 3) MINDER COMPUTER SYSTEM WHICH IS ENERGIZED
- 4) AND OPERATIONAL
- 5) 2. YOU MAY REFER TO THE APPROPRIATE TECHNICAL
- 6) MANUAL(S)
- 7) (MORE)

G3. ANSWERS

- 1) O WAIT 60
- 2) A+ACT

G4. ACTIONS

- 1) '-F: PRESS THE "ACTION" SWITCH TO PROCEED

FRAME (Q) LABEL= INSTRUCT2

G2. TEXT

- 1) YOU ARE REQUIRED TO DO THE FOLLOWING:
- 2)
- 3) 1. PERFORM A NUMBER HOOK ON ATDL-1 TRACK NUMBER
AB 120
- 4) 2. PERFORM A GEOREF POSITION HOOK FOR A TRACK
AT COORDINATES PDEC 3040
- 5) 3. PERFORM A POSITION HOOK FOR THE TRACK DESIGNATED
AS AB121
- 6) 4. PERFORM A SEQUENCE HOOK FOR ALL HIGH-THREAT
TRACKS
- 7) 5. COMPLETE THE TASK WITHIN 5 MINUTES
- 8) (MORE)

G3. ANSWERS

- 1) O WAIT 60
- 2) A+ACT

G4. ACTIONS

- 1) '-F: PRESS THE "ACTION" SWITCH TO PROCEED

FRAME (D) LABEL= SEQ

G2. CRITERIA

- 1) IF ARRAY2(1) AND ARRAY2(2) AND ARRAY2(5)
AND ARRAY2(6) EQ 0 B: FINISH
- 2) ELSE B: HOOK

FRAME (Q) LABEL= HOOK

G2. TEXT

1) \$

G3. ANSWERS

- 1) A+MI4
- 2) B+TS1
- 3) C+MI6
- 4) D+MI3
- 5) E+ACT

G4. ACTIONS

- 1) A B:9
- 2) B B:19
- 3) C B:29
- 4) D B:39
- 5) E B:FINISH
- 6) - B:1
- 7) ' B:TIMOUT

FRAME (D) LABEL=9

G2. CRITERIA

- 1) IF ARRAY2(1) EQ 0 B:SEQ
- 2) B:10

FRAME (Q) LABEL=10

G2. TEXT

1) S

G3. ANSWERS

1) A+ AB120

G4. ACTIONS

- 1) A C: Hook track* B: SEQ
- 2) - C:ARRAY4(1) = 1 B:SEQ

*Done through hardware

FRAME (D) LABEL=19

G2. CRITERIA

- 1) IF ARRAY2(2) EQ 0 B:SEQ
- 2) B:20

FRAME (Q) LABEL=20

G2. TEXT

1) \$

G3. ANSWERS

- 1) A+TF4
- 2) B+MI6
- 3) C+ACT

G4. ACTIONS

- 1) A C:ARRAY2(3)=0 C:FRMNUM=21 B:READAN2
- 2) B C:ARRAY2(4)=0 B:20.1
- 3) C B:FINISH
- 4) - B:20.2
- 5) ' B:TIMOUT

FRAME (Q) LABEL = READAN2

G2. TEXT

1) \$

G3. ANSWERS

1) A+ PDEC3040

G4. ACTIONS

1) A B:FRMNUM

2) - ARRAY4(1)=1 B:FRMNUM

FRAME (D) LABEL=20.1

G2. CRITERIA

- 1) C:ARRAY5(4)=ARRAY5(4)+1
- 2) IF ARRAY1(4) GT 0 C:ARRAY1(4)=ARRAY1(4)-1
C:ARRAY3(4)=1 C:ARRAY5(4)=ARRAY5(4)-1 B:21
- 3) ELSE C:ARRAY2(4)-1 B:21

FRAME (D) LABEL= 20.2

G2. CRITERIA

- 1) C:ARRAY1 (4)=ARRAY4(1)+1
- 2) B:20

FRAME (Q) LABEL=22

G2. TEXT

1) \$

G3. ANSWERS

- 1) A+MI6
- 2) B+TF4
- 3) C+ACT

G4. ACTIONS

- 1) A C:ARRAY2(4)=0 B:22.1
- 2) B C:ARRAY2(3)=0 C:FRMNUM=22.2 B:READAN2
- 3) C B:FINISH
- 4) - B:22.3
- 5) ' TIMEOUT

FRAME (P) LABEL= 22.1

G2. STATEMENTS

1) B:SEQ

FRAME (D) LABEL= 22.2

G2. CRITERIA

- 1) $ARRAY5(3)=ARRAY5(3)+1$
- 2) B:HOOK1

FRAME (D) LABEL=22.3

G2. CRITERIA

- 1) $ARRAY1(4) = ARRAY1(4) + 1$
- 2) 8:22

·FRAME (D) LABEL=29

G2. CRITERIA

- 1) IF ARRAY2(5) EQ 0 B:SEQ
- 2) B:30

FRAME (Q) LABEL=30

G2. TEXT

1) S

G3. ANSWERS

1) A+ (Within acceptable error)*

G4. ACTIONS

1) A C:Hook track* B:SEQ
2) - C:ARRAY7(1)=1 C:Hook track* B:SEQ

*Done through hardware

FRAME (D) LABEL= 39

G2. CRITERIA

- 1) IF ARRAY2(6) EQ 0 B:SEQ
- 2) B:40

FRAME (Q) LABEL=40

G2. TEXT

1)

G3. ANSWERS

- 1) A+ L
- 2) B+ T
- 3) C+ E
- 4) D+ H
- 5) E+ P
- 6) F+ A
- 7) G+ F

G4. ACTIONS

- 1) A C:ARRAY4(3)=1 C:Hook local tracks* B:SEQ
- 2) B C:ARRAY4(3)=1 C:Hook all central file tracks* B:SEQ
- 3) C C:ARRAY4(3)=1 C:Hook jam strobes* B:SEQ
- 4) D C:Hook high priority track* B:SEQ
- 5) E C:ARRAY4(3)=1 C:Hook poor quality tracks* B:SEQ
- 6) F C:ARRAY4(3)=1 C:Hook alerts* B:SEQ
- 7) G C:ARRAY4(3)=1 C:Hook fire units* B:SEQ
- 8) - C:ARRAY4(3)=1

*Done through hardware

FRAME (P) LABEL=FEED
G2. STATEMENTS

- 1) C:J(1)=2 C:K(1)=1
- 2) C:J(2)=3 C:K(2)=4
- 3) C:J(3)=5 C:K(3)=9
- 4) C:J(4)=6 C:L(1)=7
- 5) C:J(5)=8
- 6) C:J(6)=10

FRAME (P) LABEL+FEED1
G2. STATEMENTS

```
1) C:I=1
2) IF ARRAY1(I) EQ 0 B:L4
3) C:ARRAY1(I) 'UNNEEDED SWITCH ACTIONS AT STEP' J(I)
   C:PM(J(I))=1
4) C:I=I+1
5) IF I LT 6 B:L2
6) C:I=1
7) IF ARRAY2(J) EQ 0 B:L9
8) C:'THE SWITCH ACTION AT STEP' J(I) 'WAS DELETED'
   C:PM(J(I))=1
9) C:I=I+1
10) IF I LT 6 B:L7
11) C:I=1
12) IF ARRAY3(I) EQ 0 B:L14
13) C:'THE SWITCH ACTION AT STEP' J(I) 'WAS SUBSTITUTED
   WITH A WRONG RESPONSE' C:PM(J(I))=1
14) C:I=I+1
15) IF I LT 6 B:L12
16) C:I=1
17) IF ARRAY4(I) EQ 0 B:L21
18) C:'THE WRONG INFORMATION WAS ENTERED ON THE
   KEYBOARD AT STEP' K(I) C:PM(K(I))=1
19) C:I=I+1
20) IF I LT 3 B:L17
21) C:I=1
22) IF ARRAY5(I) EQ 0 B:L26
23) C:'SWITCH ACTION' 2+I 'WAS PERFORMED OUT OF ORDER'
   C:PM(J(I))+1
24) C:I=I+1
25) IF I LT 6 B:L22
26) C:I=1
27) IF ARRAY6(I) EQ 0 B:L29
28) C:'WRONG SYMBOL HOOKED AT STEP 7' C:PM(7)=1
29) IF PM(I) EQ 0 FOR (I=1,11) F:TASK PERFORMED CORRECTLY
30) B:SUMMARY
```

FRAME (D) LABEL= SUMMARY

G2. CRITERIA

- 1) F:*****
- 2) F: SUMMARY
- 3) F: PERFORMANCE MEASURE PASS FAIL
- 4) C: I=1 C: COLUMN=26+PM(I)*8 FOR(I=1,6)
- 5) ALIGN X, COLUMN

82

2

APPENDIX C

ENTER FU DATA ON DDG AND ARO

84

FRAME (Q) LABEL= START3

G2. TEXT

- 1) TEST 3
- 2) ENTER FU DATA ON ARO AND DDG
- 3) (TASK 441-074-1034)
- 4)
- 5) (MORE)

G3. ANSWERS

- 1) O WAIT 60
- 2) A+ACT

G4. ACTIONS

- 1) '-PRESS THE "ACTION" SWITCH TO PROCEED

FRAME (Q) LABEL= COND3

G2. TEXT

- 1) THE FOLLOWING CONDITIONS PERTAIN TO THIS TEST
- 2) 1. YOU ARE OPERATING A GROUP OR BATTALION MISSILE
- 3) MINDER COMPUTER SYSTEM WHICH IS ENERGIZED
- 4) AND OPERATIONAL
- 5) 2. YOU MAY REFER TO THE APPROPRIATE TECHNICAL
- 6) MANUAL(S)
- 7) (MORE)

G3. ANSWERS

- 1) O WAIT 60
- 2) A+ACT

G4. ACTIONS

- 1) '-F:PRESS THE "ACTION" SWITCH TO PROCEED

FRAME (Q) LABEL=INSTRUCT3

G2. TEXT

- 1) YOU ARE REQUIRED TO ENTER THE FOLLOWING FIRE UNIT
- 2) DATA ON THE ARO AND DDG
- 3) FU ADDRESS A12
- 4) ROW NUMBER 05
- 5) FU ALERT STATUS 15
- 6) HOT MISSILE COUNT 20 (WARHEAD TYPE C)
- 7) COLD MISSILE COUNT 15 (WARHEAD TYPE C)
- 8) FU STATUS READY
- 9) THIS INFORMATION SHOULD BE ENTERED WITHIN 6 MINUTES
- 10) (MORE)

G3. ANSWERS

- 1) O WAIT 60
- 2) A+ACT

G4. ACTIONS

- 1) '-F:PRESS THE "ACTION" SWITCH TO PROCEED

FRAME (Q) LABEL=SETUP

G2. TEXT

1) \$

G3. ANSWERS

- 1) A+ TS9
- 2) B+ HOOK
- 3) C+ TF1
- 4) D+ TF3
- 5) E+ TF5
- 6) F+ TF6
- 7) G+ TF4
- 8) H+ ACT

G4. ACTIONS

- 1) A B:1
- 2) B C:FLAG=FLAG-1 B:2
- 3) C C:FLAG=FLAG-1 B:3
- 4) D C:FLAG=FLAG-1 B:4
- 5) E C:FLAG=FLAG-1 B:5
- 6) F C:FLAG=FLAG-1 B:6
- 7) G C:FLAG=FLAG-1 B:7
- 8) H B:FEED
- 9) - B:8
- 10) ' B:TIMEOUT

FRAME (D) LABEL=1

G2. CRITERIA

- 1) C:ARRAY2(1)=0
- 2) B:SETUP

FRAME (D) LABEL=2

G2. CRITERIA

- 1) C:FRMNUM=2
- 2) IF FLAG EQ 0 B:READAN1
- 3) C:ARRAY2(2)=0

FRAME (Q) LABEL= READAN1

G2. TEXT

1) \$

G3. ANSWERS

1) A+A1205

G4. ACTIONS

1) A C:FLAG=FLAG+1 B:FRMNUM

2) - C:FLAG=FLAG+1 C:ARRAY4(1)=ARRAY4(1)+1 B:FRMNUM

FRAME (D) LABEL=3

G2. CRITERIA

- 1) C:FRMNUM=3
- 2) IF FLAG EQ 0 B:READAN2
- 3) C:ARRAY2(3)=0
- 4) IF ARRAY3(3) EQ 1 C:ARRAY3(3)=0 C:ARRAY1(1)=ARRAY1(1)+1
- 5) B:SETUP

FRAME (Q) LABEL=READAN2

G2. TEXT

1) \$

G3. ANSWERS

1) A+15

G4. ACTIONS

1) A C:FLAG=FLAG+1 B:FRMNUM

2) - C:FLAG=FLAG+1 C:ARRAY4(2)=ARRAY4(2)+1 B:FRMNUM

FRAME (D) LABEL=4

G2. CRITERIA

- 1) C:FRMNUM=4
- 2) IF FLAG EQ 0 B:READAN3
- 3) C:ARRAY2(4)=0
- 4) IF ARRAY3(4) EQ 1 C:ARRAY3(4)=0 C:ARRAY1(1)=ARRAY1(1)+1
- 5) B:SETUP

FRAME (Q) LABEL= READAN3

G2. TEXT

1) \$

G3. ANSWERS

1) A+15

G4. ACTIONS

1) A C:FLAG=FLAG+1 B:FRMNUM

2) - C:FLAG=FLAG+1 C:ARRAY4(3)=ARRAY4(3)+1 B:FRMNUM

FRAME (D) LABEL= 5

G2. CRITERIA

- 1) C:FRMNUM=5
- 2) IF FLAG EQ 0 B:READAN4
- 3) C:ARRAY2(5)=0
- 4) IF ARRAY3(5) EQ 1 C:ARRAY3(5)=0 C:ARRAY1(1)+ARRAY1(1)+1
- 5) B:SETUP

FRAME (Q) LABEL= READAN4

G2. TEXT

1) \$

G3. ANSWERS

1) A+ C020

G4. ACTIONS

1) A C:FLAG=FLAG+1 B:FRMNUM
2) - C:FLAG=FLAG+1 C:ARRAY4(4)=ARRAY4(4)+1 B:FRMNUM

FRAME (D) LABEL= 6

G2. CRITERIA

- 1) C:FRMNUM=6
- 2) IF FLAG EQ 0 B:READAN5
- 3) C:ARRAY2(6)=0
- 4) IF ARRAY3(6) EQ 1 C:ARRAY3(6)=0 C:ARRAY1(1)=ARRAY1(1)+1
- 5) B:SETUP

FRAME (Q) LABEL= READAN5

G2. TEXT

1) \$

G3. ANSWERS

1) A+ C15

G4. ACTIONS

1) A C:FLAG=FLAG+1 B:FRMNUM
2) = C:FLAG=FLAG+1 C:ARRAY4(5)=ARRAY4(5)+1 B:FRMNUM

FRAME (D) LABEL= 7

G2. CRITERIA

- 1) C:FRMNUM=7
- 2) IF FLAG EQ 0 B:READAN6
- 3) C:ARRAY2(7)=0
- 4) IF ARRAY3(7) EQ 1 C:ARRAY3(7)=0 C:ARRAY1(1)=ARRAY1(1)+1
- 5) B:SETUP

FRAME (Q) LABEL= READAN6

G2. TEXT

1) \$

G3. ANSWERS

1) A+ R

G4. ACTIONS

1) A C:FLAG=FLAG+1 B:FRMNUM

2) - C:FLAG=FLAG+1 C:ARRAY4(6)=ARRAY4(6)+1 B:FRMNUM

FRAME (Q) LABEL=8

G2. TEXT

1) \$ READ A/N KEYS

G3. ANSWERS

- 1) A+ A1205
- 2) B+ 15
- 3) C+ C020
- 4) D+ C15
- 5) E+ \$

G4. ACTIONS

- 1) A B:10
- 2) B B:11
- 3) C B:12
- 4) D B:13
- 5) E B:14
- 6) - B:15

FRAME (D) LABEL=10

G2. CRITERIA

- 1) IF ARRAY2(3) EQ 0 C:ARRAY1(1)=ARRAY1(1)+1 B:SETUP
- 2) C:ARRAY3(3)=ARRAY3(3)+1 B:SETUP

FRAME (D) LABEL=11

G2. CRITERIA

- 1) IF ARRAY2(4) EQ 0 C:ARRAY1(1)=ARRAY1(1)+1 B:SETUP
- 2) C:ARRAY3(4)=ARRAY3(4)+1 B:SETUP

FRAME (D) LABEL=12

G2. CRITERIA

- 1) IF ARRAY2(5) EQ 0 C:ARRAY1(1)=ARRAY1(1)+1 B:SETUP
- 2) C:ARRAY3(5)=ARRAY3(5)+1 B:SETUP

FRAME (D) LABEL= 13

G2. CRITERIA

- 1) IF ARRAY2(6) EQ 0 C:ARRAY1(1)=ARRAY1(1)+1 B:SETUP
- 2) C:ARRAY3(6)+ARRAY3(6)+1 B:SETUP

FRAME (D) LABEL=14

G2. CRITERIA

- 1) IF ARRAY2(7) EQ 0 C:ARRAY1(1)=ARRAY1(1)+1 B:SETUP
- 2) C:ARRAY3(7)=ARRAY3(7)+1 B:SETUP

FRAME (D) LABEL= 15

G2. CRITERIA

- 1) C:ARRAY1(1)=ARRAY1(1)+1
- 2) B:SETUP

FRAME (D) LABEL=FEED

G2. CRITERIA

- 1) C:J(1)=1
- 2) FOR I=2,6 (C:J(I)=I*2)
- 3) FOR I=1,5 (C:K(I)+I*2+1)

FRAME (D) LABEL= FEED1

G2. CRITERIA

- 1) IF ARRAY1(1) EQ 0 B:L3
- 2) C:ARRAY1(1) 'UNNEEDED SWITCH ACTIONS OCCURRED
- 3) C:I=1
- 4) IF ARRAY2(I) EQ 0 B:L7
- 5) C:'THE SWITCH ACTION AT STEP' J(I) 'WAS DELETED'
C:PM(J(I))=1
- 6) C:I=I+1
- 7) IF I LT 6 B:L4
- 3) C:I=1
- 9) IF ARRAY3(I) EQ 0 B:L11
- 10) C:'THE SWITCH ACTION AT STEP' J(I) 'WAS SUBSTITUTED
WITH A WRONG RESPONSE' C:PM(J(I))=1
- 11) C:I=I+1
- 12) IF I LT 6 B:L9
- 13) C:I=1
- 14) IF ARRAY4(I) EQ 0 B:L19
- 15) C:'THE WRONG INFORMATION WAS ENTERED ON
THE KEYBOARD AT STEP' K(I) C:PM(K(I))=1
- 16) C:I=I+1
- 17) IF I LT 5 B:L14
- 18) IF PM(I) EQ 0 FOR (I=1,12) F:TASK WAS
PERFORMED CORRECTLY
- 19) B:SUMMARY

FRAME (D) LABEL=TIMEOUT

G2. CRITERIA

- 1) F:THE TASK EXCEEDED THE TIME LIMIT
- 2) C:PM(6)=1

FRAME (D) LABEL= SUMMARY

G2. CRITERIA

- 1) F:*****
- 2) F: SUMMARY
- 3) F: PERFORMANCE MEASURE PASS FAIL
- 4) C: I=1 C: COLUMN=26+PM(I)*8 FOR(I=1,6)
- 5) ALIGN X, COLUMN

APPENDIX D

SETUP AND INSTRUCTIONS

114

125

LESSON NAME SQT3

DATE 2/3/80

FRAME 7.00 (Q)

G2. TEXT

1) PLEASE TYPE IN THE NAME OF THE TSM ON THE C/ED SCREEN.

2) THEN PRESS "C/ED CMPTR ACTION".

3) \$\$\$ CE-WRITE

4) TSM IDENTIFICATION --

5) LAST NAME:

6) RANK:

7) \$\$\$ SA-CLEAR #

G3. ANSWERS

1) A+CA

FRAME 8.00 (Q)

G2. TEXT

1) \$

G3. ANSWERS

1) 0 SP(3,11,15)

2) A XYZ

G4. ACTIONS

1) A-* C:SET REPLY(7) C:SP(4,6,5) R:\$

2) A-* C:SET REPLY(9)

FRAME 9.00 (Q) LABEL=START

G2. TEXT

1) *****

2) **

3) ** SKILL QUALIFICATION TEST 1 -- HANDS-ON COMPONENT **

4) ** MOS 100 **

5) **

6) *****

7) PRESS THE "PRIORITY MESSAGE" SWITCH TO PROCEED.

8) \$\$\$ SA-CLEAR CE-CLEAR PM-ON #

G3. ANSWERS

1) A+PM

2) S SP

G4. ACTIONS

1) F:\$\$\$ SA-CLEAR

2) S B:OUT

FRAME 10.00 (Q)

G2. TEXT

1) PLEASE TYPE THE FOLLOWING INFORMATION ON THE C/ED SCREEN IN THE

2) DESIGNATED LINES AND FIELDS.

3) LINE 1: LAST NAME, FIRST NAME, MIDDLE INITIAL

4) LINE 2: SOCIAL SECURITY NUMBER AS IT APPEARS ON YOUR MILITARY ID CARD

5) LINE 3: YOUR RANK

6) LINE 4: TODAY'S DATE-- DAY MONTH YEAR

7) THEN PRESS THE "C/ED CMPTR ACTION" SWITCH.

8) \$\$\$ CE-WRITE COPY-OFF

9) LAST NAME:

FIRST NAME:

M. I. :

10) S-S-NO.

LESSON NAME SQT3 DATE 2/3/60

11) RANK:

12) DAY: MONTH: YEAR:

13) \$\$\$ SA-CLEAR RD-WRITE COPY-ON \$

G3. ANSWERS

1) A+CA

G4. ACTIONS

1) A B:11

2) - F:NO. YOU PRESSED THE WRONG SWITCH. PRESS THE "C/ED

3) F:CMPTTR ACTION" SWITCH. R:\$\$\$ SA-CLEAR \$

FRAME 10.50 (Q)

G2. TEXT

1) PLEASE TYPE THE FOLLOWING INFORMATION ON THE C/ED SCREEN IN THE

2) DESIGNATED LINES AND FIELDS. THEN PRESS "C/ED CMPTTR ACTION."

3) LINE 1: LAST NAME, FIRST NAME, MIDDLE INITIAL

4) LINE 2: SOCIAL SECURITY NUMBER AS IT APPEARS ON YOUR MILITARY ID CARD

5) LINE 3: YOUR RANK

6) LINE 4: TODAY'S DATE-- DAY MONTH YEAR

7) \$\$\$ CE-WRITE COPY-OFF

8) LAST NAME:

FIRST NAME:

M. I.:

9) S. S. NO.:

10) RANK:

11) DAY: MONTH: YEAR:

12) \$\$\$ SA-CLEAR RD-WRITE COPY-ON \$

G3. ANSWERS

1) 0 SET KW ON

2) A+CA

G4. ACTIONS

1) - F:NO. YOU PRESSED THE WRONG SWITCH. PRESS THE "C/ED

2) F:CMPTTR ACTION" SWITCH. R:\$\$\$ SA-CLEAR \$

FRAME 11.00 (Q)

G2. TEXT

1) \$\$\$ GET(1,11,17) \$

G3. ANSWERS

1) A XYZ

G4. ACTIONS

1) C:SET REPLY(1)

2) * F:NO. YOU DID NOT TYPE IN YOUR NAME. TRY AGAIN. B:10.5

FRAME 12.00 (Q)

G2. TEXT

1) \$\$\$ GET(1,40,17) \$

G3. ANSWERS

1) A XYZ

G4. ACTIONS

1) C:SET REPLY(2)

2) * F:NO. YOU DID NOT TYPE IN YOUR NAME. TRY AGAIN. B:10.5

LESSON NAME SQT3 DATE 2/3/80

FRAME 13.00 (Q)

G2. TEXT

1) \$\$\$ GET(1, C3, 2) \$

G3. ANSWERS

1) 0 KW OFF

2) A

G4. ACTIONS

1) A*

2) = C:SET REPLY(3)

FRAME 14.00 (Q)

G2. TEXT

1) \$\$\$ GET(2, 10, 11) \$

G3. ANSWERS

1) 0 KW OFF

2) A

G4. ACTIONS

1) C:SET REPLY(4)

2) A* F:NO YOU DID NOT TYPE IN YOUR S. S. NO. TRY AGAIN. B:10.5

FRAME 15.00 (Q)

G2. TEXT

1) \$\$\$ GET(3, 6, 4) \$

G3. ANSWERS

1) 0 SET KW ON

2) A XYZ

G4. ACTIONS

1) C:SET REPLY(5)

2) * F:NO YOU DID NOT TYPE IN YOUR RANK. TRY AGAIN. B:10.5

FRAME 16.00 (Q)

G2. TEXT

1) \$\$\$ GET(4, 5, 2) \$

G3. ANSWERS

1) A XY

2) 1 35

G4. ACTIONS

1) C:SET LINK(3)=RESPONSE

2) * F:NO YOU DID NOT TYPE IN THE DATE. B:10.5

FRAME 17.00 (Q)

G2. TEXT

1) \$\$\$ GET(4, 14, 3) \$

G3. ANSWERS

1) A+JAN

LESSON NAME SQT3 DATE 2/3/80

- 2) B+FEB
- 3) C+MAR
- 4) D+APR
- 5) E+MAY
- 6) F+JUN
- 7) G+JUL
- 8) H+AUG
- 9) I+SEP
- 10) J+OCT
- 11) K+NOV
- 12) L+DEC

G4. ACTIONS

- 1) C:SET REPLY(6)
- 2) * F:NO. YOU DID NOT TYPE IN THE DATE. B:10.5
- 3) A C:LINK(4)=1
- 4) B C:LINK(4)=2
- 5) C C:LINK(4)=3
- 6) D C:LINK(4)=4
- 7) E C:LINK(4)=5
- 8) F C:LINK(4)=6
- 9) G C:LINK(4)=7
- 10) H C:LINK(4)=8
- 11) I C:LINK(4)=9
- 12) J C:LINK(4)=10
- 13) K C:LINK(4)=11
- 14) L C:LINK(4)=12

FRAME 18.00 (Q)

G2. TEXT

- 1) \$\$\$ GET(4,23,2) \$

G3. ANSWERS

- 1) A XY
- 2) 1 90

G4. ACTIONS

- 1) C:SET LINK(5)=RESPONSE
- 2) A* F:NO. YOU DID NOT TYPE IN THE DATE. TRY AGAIN. B:10.5

FRAME 19.00 (D)

G2. CRITERIA

- 1) F:\$\$\$ RD-WRITE COPY-ON
- 2) F:WELCOME TO SKILL QUALIFICATION TEST 3 FOR MOS 13C. \$
- 3) C:PRINT REPLY(5) \$ C:PRINT / \$
- 4) C:PRINT REPLY(1) \$
- 5) C:PRINT /
- 6) IF SCORE(1,43) EQ 1 B:20

FRAME 19.50 (Q)

G2. TEXT

- 1) IF YOU HAVE AN ELP PRINTOUT OF THE INSTRUCTIONS FOR SQT3 HOC
- 2) AND DO NOT NEED TO REVIEW THE INSTRUCTIONS AT THIS TIME, PRESS
- 3) "PRIORITY MESSAGE" TO BEGIN THE FIRST HANDS-ON TEST.
- 4) IF YOU ARE NOT FAMILIAR WITH THE INSTRUCTIONS OR NEED TO REVIEW
- 5) THEM AT THIS TIME, PRESS "CYCLE MESSAGES" TO PROCEED WITH THE
- 6) INSTRUCTIONS.

LESSON-NAME SQT3 DATE 2/3/80

7) \$\$\$ SA-CLEAR CE-CLEAR \$

G3. ANSWERS

1) 0 SET KW ON

2) 0 WAIT 90

3) A+PM

4) B+CM

G4. ACTIONS

1) F:\$\$\$ SA-CLEAR

2) A-B-28

FRAME 20.00 (Q) LABEL=INTRO

G2. TEXT

1) THIS IS THE HANDS-ON COMPONENT OF SQT3. IT WILL TEST YOUR ABILITY TO

2) OPERATE THE ACC IN A MANNER THE SAME AS THE TACFIRE OPERATING SYSTEM.

3) IN THIS HOC, YOU WILL NORMALLY RECEIVE DIRECTIONS AND DESCRIPTION OF A

4) TACTICAL SITUATION ON THE RD SCREEN. WHENEVER "(MORE)" APPEARS AT THE

5) END OF A LINE AND "PRIORITY MESSAGE" IS LIT, PRESS "PRIORITY MESSAGE"

6) TO DISPLAY THE REMAINDER OF THE TEXT ON THE RD. (MORE)

7) \$\$\$ SA-CLEAR CE-CLEAR PM-ON \$

G3. ANSWERS

1) 0 WAIT 90

2) A+PM

G4. ACTIONS

1) F:\$\$\$ SA-CLEAR

2) A F:GOOD. PLEASE READ THE FOLLOWING INSTRUCTIONS FOR THIS HOC.

3) - F:YOU PRESSED A SWITCH OTHER THAN "PRIORITY MESSAGE."

4) R:PRESS THE "PRIORITY MESSAGE" SWITCH. \$

5) / R:PRESS THE "PRIORITY MESSAGE" SWITCH. \$

FRAME 21.00 (Q)

G2. TEXT

1) YOU WILL BE TESTED ON TASKS FROM THE MOS 130 SKILL LEVEL 3 SOLDIER'S

2) MANUAL. EACH TASK IS ORGANIZED INTO ONE OF MORE HANDS-ON TESTS.

3) COMPLEX TASKS WILL BE TESTED IN MORE THAN ONE TEST. YOU WILL BE GIVEN

4) THE CONDITIONS, STANDARDS, AND REQUIREMENTS FOR EACH TEST ON THE RD

5) SCREEN AND ON THE ELP FOR YOUR REFERENCE. THEN, YOU WILL BE ASKED TO

6) CARRY OUT THE PERFORMANCE REQUIREMENTS FOR THAT TEST. (MORE)

7) \$\$\$ SA-CLEAR CE-CLEAR PM-ON \$

G3. ANSWERS

1) 0 WAIT 90

2) A+PM

G4. ACTIONS

1) F:\$\$\$ SA-CLEAR

2) - R:NO. PRESS "PRIORITY MESSAGE" TO DISPLAY THE REMAINDER OF THE TEXT. \$

3) / R:PRESS THE "PRIORITY MESSAGE" SWITCH. \$

FRAME 21.50 (Q)

G2. TEXT

1) FOR EACH TEST, OPERATE THE PROPER SWITCH PANEL ASSEMBLY SWITCHES

2) AND/OR USE THE KEYBOARD TO MAKE ENTRIES IN THE APPROPRIATE SUBFIELDS

3) ON THE C/ED SCREEN. USE THE SAME PROCEDURES AS YOU WOULD IN THE

126

27

INSTRUCTIONS

This is the hands-on component of the SQT. It will test your ability to operate the simulator in the same manner as the Missile Minder operating system. In this HOC, you will normally receive directions and descriptions of a tactical situation on the screen. Whenever "(More)" appears at the end of the line, press the "action" switch to display the remainder of the text.

You will be tested on the MOS 25L Soldier's Manual. Each task is organized into one or more hands-on tests. Complex tasks will be tested in more than one test. You will be given the conditions, standards, and requirements for each test on the screen. Then, you will be asked to carry out the performance requirements for that test.

For each test, operate the proper switch panel action switches and/or use the alphanumeric keyboard to make the proper keyboard entries. Use the same procedures as you would in the field operating system for a battalion Missile Minder system. If you make an error, the system will respond as if your action was correct. Do not try to correct the error. The testing system will not be able to know when you have completed the test. At the end of all switch actions and keyboard entries, you must push the "action" switch to tell the system that you have finished the test. If you don't push this switch, the system will wait until the allotted test time is over and give you an error for taking too much time.

You will be supplied with the proper Missile Minder technical manual, TH9-1430-652-10-3. You can use this manual during the test.

When you finish each test, the computer will provide you will a list of the errors committed during the test, if any. There are several types of errors. It will tell you if 1) you made switch actions that weren't needed; 2) you

deleted a switch action that was needed; 3) you substituted a wrong switch action for the correct one; 4) you made a wrong keyboard entry; 5) a switch action was performed out of order; 6) you took too long to complete the task.

Each of the tasks is broken down into discrete steps which must be passed to correctly perform the task. At the end of the task, a summary of performance on each step will be provided with either a "go" or "no-go" score. A "go" means that the step has been successfully completed. A "no-go" means that the step was not done correctly.

If you understand the instructions and have technical manual TM-9-1430-652-10-3 and you are ready to proceed to the start of the first test, type "go" on the keyboard and press the "action" switch. If you wish to review the instructions, type "review" on the keyboard and press the "action" switch.

You have now begun the SQT for MOS 25L.