UNIT DIAGNOSTICS HANDBOOK

General Diagnostic Information

Part No. 945400-9701 *M
July 1984

Volume 1

TEXAS INSTRUMENTS

Printed in U.S.A.

MANUAL REVISION HISTORY

Unit Diagnostics Handbook, Volume 1, General Diagnostic Information (945400-9701)

Original Issue	January 1972
Revision	August 1978
Revision	November 1979
Revision	June 1980
Revision	October 1981
Revision	March 1982
Revision	August 1982
Revision	November 1982
Revision	October 1983
Revision	March 1984
Revision	July 1984

The total number of pages in this publication is 272.

The computers, as well as the programs that TI has created to use with them, are tools that can help people better manage the information used in their business; but tools—including TI computers—cannot replace sound judgment nor make the manager's business decisions.

Consequently, TI cannot warrant that its systems are suitable for any specific customer application. The manager must rely on judgment of what is best for his or her business.

Preface

Unit diagnostics is a collection of tests that are used to evaluate Texas Instruments 990-based computers and their associated peripheral devices.

This handook is intended as a reference for systems analysts, customer representatives, factory service personnel, and other technical users.

The Unit Diagnostics Handbook consists of a set of seven volumes that describe the diagnostic tests for your 990-based system. Volume 1 defines the two categories of diagnostics: stand-alone tests (SATs) and the Diagnostic Operational Control System (DOCS). Both categories use tests that evaluate your computer equipment and devices. However, the manner in which these tests are implemented differs. You must be familiar with the information presented in Volume 1 to use the diagnostic tests.

Volume 1 is organized such that you need only read sections that apply to your particular software and hardware. This volume is divided into the following sections and appendixes:

Section

1 Diagnostics Overview -- Describes unit diagnostics and its hardware requirements, and lists the diagnostic tests.

2 Stand-Alone Tests (SATs) -- Describes the output device selections and other options available with SATs.

3 Standard Diagnostic Operational Control System (DOCS) -- Discusses Standard DOCS object modules, supported equipment and operating procedures.

4 Dynamic DOCS -- Discusses Dynamic DOCS object modules, supported equipment and operating procedures.

5 S200DOCS -- Discusses S200DOCS supported equipment and operating procedures.

6 DOCS Verbs -- Lists and explains the verbs used by Standard DOCS, Dynamic DOCS, and S200DOCS.

7 Diagnostic Loading Procedures -- Contains the procedures required to load Standard and Dynamic DOCS and the diagnostic tests.

8 DOCS Building Procedures -- Explains how to build DOCS on several types of media for Standard DOCS, Dynamic DOCS, and S200DOCS.

9 Batch Command Stream DOCS -- Describes the batch command stream feature.

10 Debug DOCS -- Contains information on selection, usage, and limitations of Debug DOCS.

Appendix

A Diagnostic Part Numbers -- Identifies all diagnostic information by part number.

B Operation of the 990 Maintenance Diagnostic Unit (MDU) -- Describes the operation of the 990 MDU.

C Operation of the 990 Programmer Panel -- Describes the operation of the programmer panel.

D DOCS Message Descriptions -- Lists and explains the messages that can occur when you are running DOCS or diagnostic tests.

E Assembly/Diagnostic Cross-Reference -- Cross-references the diagnostic tests with part numbers of target assemblies associated with the tests.

F Reference Documents -- Lists documents that contain operation, maintenance, and troubleshooting information.

G CONVER Diagnostic Utility -- Describes the CONVER diagnostic utility.

H ASCII AND HEXADECIMAL CHARACTERS -- Cross-references ASCII Characters with Hexadecimal Code.

I S200 SELF-TESTS -- Describes the self-test and extended-test diagnostic routines developed for the Business System 200.

J LOOPBACK MODES -- Explains which system components are tested in each mode, and prerequisites to the use of each mode.

K Operation of the Universal Systems Terminal (UST) -- Describes diagnostic testing using the UST.

The remaining volumes contain operational and supplemental information for the specific diagnostics, which are grouped by volume according to equipment similarity. Table 1-1 lists the diagnostic tests alphabetically by their mnemonic names and indicates the volume(s) where each test is found.

Volume 2 Diagnostics for 990 Processors and Memories, part number 945400-9702, describes the diagnostic tests for the processors and memory devices associated with 990 computers. This volume includes the two TILINE(TM) diagnostic tests.

Volume 3 Diagnostics for 990 Mass Storage Devices, part number 945400-9703, explains the diagnostic tests for the storage media associated with 990 computers.

Volume 4 Diagnostics for 990 Printers, Terminals, and Interface Modules, part number 945400-9704, explains the diagnostic tests for the printers, terminals, and interface modules associated with 990 computers.

Volume 5 Diagnostics for 990 Industrial Systems, part number 945400-9705, describes the diagnostic tests for the industrial systems associated with 990 computers.

Volume 6 Diagnostics for 990 Communications Interfaces, part number 945400-9706, explains the diagnostics for the communications interfaces associated with 990 computers. Volume 6 includes the Ethernet(R) test.

Volume 7 Diagnostics for 990 AMPL Systems, part number 945400-9707, describes the diagnostic tests for the AMPL systems associated with 990 computers.

TILINE is a trademark of Texas Instruments Incorporated.
Ethernet is a registered trademark of Texas Instruments Incorporated.

CONTENTS

4 -- Dynamic Diagnostics Operational Control System (DOCS)

5 -- S200DOCS

6 -- DOCS Verbs

7 -- Diagnostic Loading Procedures

8 -- DOCS Building Procedures

9 -- Batch Command Stream DOCS (BCSDOCS)

10 -- Debug DOCS

APPENDIXES

ILLUSTRATIONS

TABLES

Section 1

Diagnostics Overview

1.1 INTRODUCTION

A diagnostic test is a program that generates data patterns to check data flow and control logic in the device the test evaluates. Unit diagnostics is comprised of two different types of tests:

* Stand-alone tests (SATs) -- SATs perform extensive repetitive testing and have minimal input/output (I/O) requirements.

* Diagnostics Operational Control System (DOCS) tests -- DOCS is a small operating system that supports tests requiring user interaction.

The test category depends on the type of hardware being tested. Table 1-1 lists the SATs and DOCS tests alphabetically by mnemonic name and indicates the volume(s) where each test is found.

Refer to Section 2 of this volume and Volume 2 for more information about SATs.

DOCS is divided into three categories:

* Standard DOCS -- Used if you are using a unit record (cassette/UST) loader for a 990 computer. (Standard DOCS is also used for testing 774 computers.) Standard DOCS is discussed in Section 3.

* Dynamic DOCS -- Used if you are using a disk or magnetic tape loader for a 990 computer. Dynamic DOCS is described in Section 4.

* S200DOCS -- Used if you are using an S200 system. S200DOCS is described in Section 5.

Standard DOCS, Dynamic DOCS, and S200DOCS, each perform the following functions:

* Identifies the type of computer on which it is loaded
* Dynamically reduces its size after determining which interfaces you are using for the interactive terminal and for an error message device. (This is not true for S200DOCS.)
* Initializes all interrupt and extended operation (XOP) trap locations
* Determines the interrupt level of the real-time clock
* Determines the type of terminal at which you are responding
* Initializes all system parameters, enabling you to send error messages to a printer, and lets the user define error message device parameters.
* Loads diagnostic test modules into memory
* Optionally passes control to the test module initialization routine through the Initialize Test (IT) verb
* Optionally executes all standard parts of the test module through the Execute All (EA) verb
* Controls the verb execution mechanism
* Processes all test module service calls
* Processes all DOCS-supported verbs
* Controls all I/O functions between you and the test module

A DOCS verb is a command that is associated with an action. These verbs enable you to load, initialize, and execute diagnostic tests. Each diagnostic test supports a set of verbs related to the device that the test evaluates. In addition, DOCS supports a set of general verbs that help you diagnose problems. Section 6 describes the DOCS verbs in detail.

Table 1-1 Diagnostic Tests

Mnemonic Name	Volume	Type	Description
ACUTST	6	DOCS	Automatic Call Unit Test
ADCHK	5	DOCS	Analog-to-Digital Converter Test
AU04	2	SAT	990/4 Arithmetic Unit Test (8K-Byte Version)
AU04TST	2	SAT	990/4 Arithmetic Unit Test (Extended Test Version)
AU05	2	SAT	990/5 Arithmetic Unit Test
CPU10	2	SAT	990/10 Arithmetic Unit Test
CPU12	2	SAT	990/12 Arithmetic Unit Test
AU200	8	DOCS	S200 Processor Board Test
BCAIMT	6	DOCS	BCAIM and BCAIM X.21 Communications Controller Test
COM90X	6	DOCS	TMS 9902/9903 Communications IC Test
CONVER	1	DOCS	Configuration Verification Diagnostic Utility
CPTEST	4	DOCS	Character Printer Test
CRCOMM	6	DOCS	Communications Interface Module Test
CRDRDR	4	DOCS	Model 804 Card Reader Test
CRT911	4	DOCS	911 Video Display Terminal Test
CRT913	4	DOCS	913 Video Display Terminal Test
CRUEXP	4	DOCS	CRU Expansion Chassis Interface Test
CRUTIC	2	DOCS	990/10A and S300/S300A CRU Chip Test
CTLTST	2	DOCS	Random Access Memory Diagnostic for the 990/10, 990/10A, 990/12, S300 and S300A
DACHK	5	DOCS	Digital-to-Analog Converter Test
DDFLOP	3	DOCS	TILINE DSDD Diskette Test (FD1000)
DS10PD	3	DOCS	DS10 Disk Test
DS990R	3	DOCS	Universal TILINE Disk Utility
DSKCD1	3	DOCS	CD1400 and DS80/300 Disk Systems Test - Part 1
DSKCD2	3	DOCS	CD1400 and DS80/300 Disk Systems Test - Part 2
DSKCOM	3	DOCS	Common Disk Drive Test
DSKM3X	3	DOCS	DS31/32 Disk Test
DSKSA	3	DOCS	Disk Surface Analysis
DSKTRI	3	DOCS	990 Trident Disk Test (T-25, T-50, T-200)
DSKWD5	3	DOCS	5 1/4" Winchester Disk Test
DSKWD8	3	DOCS	8" Winchester Disk Test
DS990R	3	DOCS	Universal TILINE Disk Utility
EMU900	7	DOCS	TMS 9900/9980 Emulator, Buffer, and Expansion Memory Test Second Generation AMPL
EMU940	7	DOCS	TMS 9940 Emulator and Buffer Test
EMUTST	7	DOCS	TMS 9900/9980 Emulator and Buffer Test First Generation AMPL
ENETST	6	DOCS	EI990 and EI300 Interface Boards

Table 1-1 Diagnostic Tests (Continued)

Mnemonic Name	Volume	Type	Description
EROMBT	5	DOCS	EPROM Memory Board Test
EXTACU	6	DOCS	External Autocall Unit Interface Test
FC3TST	6	DOCS	Four-Channel Communications Controller Test
FIVMOD	5	DOCS	5MT/6MT Serial Interface Module Test
FLPDSK	3	DOCS	Single-Sided, Single-Density Diskette Test
FLPTST	4	DOCS	Fast Line Printer Test
HSTSLV	2	DOCS	990/5 Host-Slave Test
INPMOD	5	DOCS	32-Bit Input/Transition Detection Module Test
IO16	4	DOCS	16 I/O TTL Module Test
LLMTST	6	DOCS	Local Line Module Test
LP810	4	DOCS	Model 810 Line Printer Test
LPTEST	4	DOCS	Line Printer Test
LQPTST	4	DOCS	LQ45 Letter Quality Line Printer Test
MAP12	2	DOCS	Map Logic Test - 990/12 only
MAPTIC	2	DOCS	990/10A, S300 and S300A Mapping Logic Test
MAPTST	2	SAT	Map Logic Test - 990/10 only
MEMPRT	2	DOCS	990/4 Memory Protect Logic Test
MEMTST	2	DOCS	General Random Access Memory Test and /10A and Cache Controller Memory Test
MTCTST1	3	DOCS	Magnetic Tape Cartridge Test 1
MTCTST2	3	DOCS	Magnetic Tape Cartridge Test 2
MUXTST	6	DOCS	CI403 and CI404 Asynchronous Communications Interface Test
OUTMOD	5	DOCS	32-Bit Output Data Module Test
PROMPG	5	DOCS	PROM Programmer Interface/Personality Card Test
PRTIF	4	DOCS	Printer Interface Module Test
RAM04	2	DOCS	990/4 Random Access Memory Test
RMTEIA	4	DOCS	TTY/EIA Remote Terminal Test
RMTFLP	3	DOCS	Remote Diskette System Loopback Test
TAPTST	3	DOCS	Model 979/979A and MT1600 Tape Test
TFMTST	5	DOCS	Time Function Module Test
TILCOU	2	DOCS	TILINE Coupler Logic Test
TLCPLR	2	DOCS	TILINE Coupler Test
TNXTST	2	DOCS	Ten X99 COBOL(TM) Accelerator Test
TPBITS	3	DOCS	TILINE Peripheral Bus Interface Test
TRACE	7	DOCS	Trace Module Test
TST733	4	DOCS	733/743 ASR/KSR Data Terminal Test
TST8X0	4	DOCS	820/840 Data Terminal Test

Ten X99 COBOL Accelerator is a trademark of Ten X Technology, Inc.

Table 1-1 Diagnostic Tests (Continued)

Mnemonic Name	Volume	Type	Description
TST931	4	DOCS	931 Video Display Terminal Test
TST940	4	DOCS	940 Electronic Video Terminal Test
TTYEIA	4	DOCS	Full Duplex TTY/EIA Interface Module Test
WATDOG	5	DOCS	System ID and Watchdog Timer Interface Test
WDFLPY	3	DOCS	WD500 Floppy Disk Drive Diagnostic

1.2 EQUIPMENT REQUIREMENTS

Each diagnostic test has certain minimum hardware requirements. Your system must contain the following hardware in order to use diagnostic tests:

* A 990-based computer with sufficient memory. The TILINE peripheral tests require 64K bytes of memory. Tests for less complex devices usually require 32K bytes of memory.

* Figure 1-1 illustrates typical memory configurations for SATs and DOCS tests. (Refer to Appendix A for information on DOCS sizes. Add the DOCS and verb module sizes to the test size to determine the total memory size needed.)

* A device to load the test into the computer. For example, a 733 ASR, a 990 diskette drive (double-density), any 990 disk drive, a 990 maintenance diagnostic unit (MDU), a Universal Systems Terminal (UST), or a magnetic tape drive (open reel or cartridge).

* An interactive device such as a 733 ASR/KSR, 743 KSR, 911 VDT, 913 VDT, 931 VDT, 940 EVT, a programmer panel (MDU), 820/840 KSR, or UST can be used.

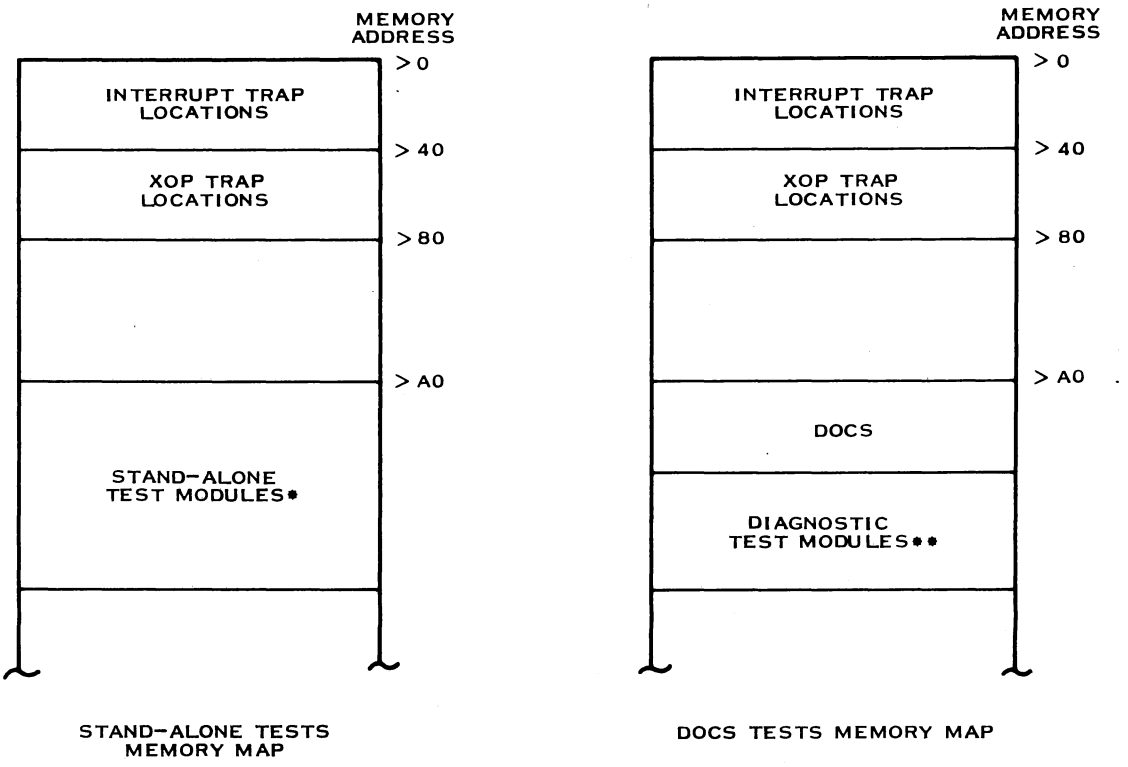
* An optional error message device to receive header and error messages, such as a 733 ASR/KSR, 743 KSR, 820/840 KSR, 810, 850, 855 or 880 line printer, MDU, or UST.

* The unit that is being tested.

NOTE

As may be seen in the lists above, the UST can function as the loading, interactive, and error message device simultaneously.

Figure 1-1 shows the memory configurations for SATs and DOCS tests.



* STAND-ALONE TESTS CONTAIN ALL CODE NEEDED TO PERFORM TESTS AND OUTPUT RESULTS TO THE USER.

** DOCS TESTS REQUIRE DOCS TO HANDLE USER I/O, DEFAULT INTERRUPT PROCESSING, VERB DECODING, AND OTHER FUNCTIONS. DOCS MUST BE LOADED FIRST IN ORDER TO LOAD THE TEST MODULE.

2283756

Figure 1-1 Stand-Alone and DOCS Memory Maps

Section 2

Stand-Alone Tests (SATs)

2.1 INTRODUCTION

A stand-alone test (SAT) makes minimal hardware demands on system capability during execution. If you are uncertain whether the entire central processing unit (CPU) is operational, then you should use a SAT.

You do not need to initialize a SAT since the test begins executing as soon as you load it into memory. SATs can issue header and error messages to one or more output devices, depending on the selections you specify. (Refer to paragraph 2.2 for information on selecting output options.) If you use a hardware configuration that includes an output device, a SAT generates messages in the following manner:

Test name and version number
Header messages
Error messages (if any errors occur)
Completion message
Loop count

You can load a SAT in one of the following manners:

* Load the SAT directly by using the 990 ROM loader. Consult the Model 990 Computer Universal ROM Loader Users Guide part number 2270534-9701 for complete instructions on this process. The test executes immediately after loading. Upon completion, the SAT displays the loop count on the output terminal or programmer panel and repeats in a continuous loop until the CPU is halted. This is not true, of course, in the case of either CPU-10 or CPU-12. These tests utilize the following type of loading.

* Load the SAT using the DOCS .LD verb. The test executes in stand-alone mode. Upon completion, control is returned to DOCS (providing that the test does not overlay any part of DOCS memory space) and you can reexecute the SAT by issuing the Execute Stand-Alone (.ES) DOCS verb. If you want to run a group of diagnostics that includes both DOCS tests and SATs, use DOCS to load SATs. Then, the SATs will send all of the output to input/output (I/O) and error devices.

Refer to Volume 2 of the Unit Diagnostics Handbook for detailed information on each of the SATs.

The following paragraphs explain how to select output devices and test options for SATs.

2.2 OUTPUT DEVICE SELECTION

The SATs can send test results to one or more devices. The methods of output device selection vary among the AU04, AU4TST, AU05, CPU10, CPU12, and MAPTST diagnostics as explained in the following paragraphs. The UST is itself both the loading and output device, and runs SATs only under the control of DOCS.

2.2.1 AU Diagnostic Device Selection

The default device selection for an AU and CPU diagnostic depends on how the test is loaded. The test chooses a default device according to whether DOCS exists wholly in memory.

The AU diagnostic can determine that DOCS is not completely resident in memory if:

* You load the test using the ROM loader

* You load the test and overlay all or part of DOCS

If the SAT is of CPU10 or one of the CPU12 diagnostics, and one of the above situations is true, the SAT displays the test number and loop count on the front panel.

If the SAT is of AU04, AU4TST, or AU05, and one of these situations is true, the test performs the default device selection algorithm. The algorithm works in this manner:

1. The test uses the default device selection parameters defined as follows:

```
    0 = ASR/KSR
    1 = 913 VDT
    2 = 911 VDT
    3 = 810 Line printer
    4 = 940 EVT
>5555 = Search for one video display terminal
        (VDT) and one hardcopy device
>FFFF = Attempt output to four devices.
```

   The default for the output device selection is >5555.

NOTE

A value preceded by a right angle bracket (>) indicates a hexadecimal value.

2. If you accept the default, the test searches the predefined CRU base addresses for one VDT and one hardcopy device. The predefined addresses are:

Address	Device
>0000	ASR/KSR
>0060	Line printer
>0040	940 EVT (on communications interface board)
>00C0	911 or 913 VDT
>0100	911 or 913 VDT
>1700	ASR/KSR (AU05 only)
>1700	940 EVT (on /10A communications port, /10 with CI402, and S300 computer)
>1740	Line printer (AU05 only)

If you do not accept the default, choose an acceptable parameter listed in item 1. If you use a CRU base address that is not predefined, you will need to modify the CRU base address location of the output terminal according to the specifications listed in Table 2-3.

NOTE

The test displays error numbers and loop counts on the programmer panel, regardless of whether another output device is found.

If the AU diagnostic determines that DOCS resides completely in memory and is loaded below the test code, then it checks whether the loaded version of DOCS is a revision prior to 8/80 *K. If the revision is prior to 8/80 *K, then the test uses the default device selection algorithm described previously. If the revision is 8/80 *K or later, the AU diagnostic makes inquiries of DOCS and uses the I/O and error message devices you selected during DOCS initialization.

If you want to send test results to a device located at a CRU address that the algorithm does not search, you can modify certain flag locations to redirect the output. Refer to paragraph 2.3 and Table 2-1 for information on output options.

2.2.2 MAPTST Diagnostic Device Selection

Since MAPTST must be loaded at a beginning address no higher than >100, this diagnostic is unable to communicate with DOCS about I/O devices. Therefore, MAPTST assumes that certain devices are located at specific CRU addresses. The test writes to each of the CRU addresses without attempting to verify that the correct device is present. False errors may be generated if the default devices are not located at the assumed CRU addresses. The CRU addresses and default devices are:

Address	Device
>0000	733 ASR
>0060	Line printer
>00C0	913 VDT
>0100	911 VDT

The MAPTST diagnostic attempts to write to all of these devices simultaneously. If any other device resides at one of these addresses, errors can result. When you check your test results, verify that no such residence exists.

If your system is not configured with the default devices at the specified CRU addresses, you can modify certain flag locations to redirect the output. Refer to paragraph 2.3 and Table 2-1 for information on output options.

2.3 STAND-ALONE OPTIONS

Certain operating options are available with SATs. These options are listed and explained in Table 2-1.

When you load a SAT using DOCS, you can select the options through the interactive terminal by executing the Modify Memory (.MM) verb. If you run a SAT without DOCS, you must select the options by performing a memory patch procedure on the programmer panel. To do this, perform the following steps immediately after the SAT begins execution:

1. Press HALT/SIE.

2. Enter the address of the option that you want to modify using the data switches. To determine this address, add the hexadecimal load bias (usually >A0) to the hexadecimal relative flag address. Refer to Table 2-1, Table 2-2, and Table 2-3 for relative flag addresses.

3. Press MA to enter the memory address.

4. Press MDD to display the contents of the location.
5. Enter the new memory data value using the data switches.
6. Press MDE to enter the data into memory.
7. Press CLR.
8. Press ENTER ST to clear the status register.
9. Enter the load point address using the data switches. The load point is usually >A0 if:
    * You are not using DOCS.
    * You are using DOCS and loaded the test over DOCS by selecting >A0 as the load bias.
10. Press MA to enter the load point.
11. Press MDD to view the contents. If DOCS is not present, the restart address of the SAT is displayed. If DOCS is present, the return address to DOCS is displayed.
12. Press ENTER PC to enter the restart address in the program counter.
13. Press RUN.

You can also modify SAT options by changing the object code on the media before loading the test into the computer. Refer to the Link Editor Reference Manual, part number 949617-9701, for instructions on modifying the object code.

Table 2-1 lists the SAT options and their functions, and explains how to modify the control flags to meet your requirements. The relative address and >A0 load bias address for each control flag are also listed. All of the options for SATs are at the locations listed unless specified otherwise.

Table 2-1 Stand-Alone Test Options

Options (Control Flag)	Relative Address	Address For >A0 Load Bias	Function
Idle-on-error	>32	>D2	Set this flag to >FFFF to cause the computer to idle when errors are detected. The computer will enter the idle mode after an error is written to an output device or displayed on the programmer panel. The default does not permit the computer to idle after detecting an error.
Error Print/Display	>1E	>BE	Set this flag to zero to prevent error messages from being printed. The error number is displayed on the programmer panel. However, unless you also select the idle-on-error option, the error number is not displayed long enough for you to see. The default is nonzero to print the error messages.
Header Print/Display	>20	>C0	Set this flag to zero to prevent header messages from being written or displayed. The default is nonzero to print header messages.

Table 2-1 Stand-Alone Test Options (Continued)

Options (Control Flag)	Relative Address	Address For >A0 Load Bias	Function
Output Terminal Selector	>3A	>DA	Set this flag to one of the following values to select an output device for test results. 0 = ASR/KSR 1 = 913 VDT 2 = 911 VDT 3 = 810 Line printer 4 = 940 EVT >5555 = Search for one VDT and one hardcopy device >FFFF = SAT will attempt output to four devices The default for the AU diagnostics is >5555. The default for the MAPTST diagnostic is >FFFF.
Baud Rate Selector			This option is not available for MAPTST. Set
AU04, AU4TST,	>40	>E0	this flag to nonzero to select a baud rate for a
	>5A	>FA	733, 810, 820, 840, 850, 855, or 880 terminal. The default is 0 for a baud rate of 1200.
AU05 - port 1 >1700	>3E	>DE	By entering the necessary
- port 2 >1740	>40	>E0	baud rate, you can modify the flag for AU05 ports 1 or 2 located at CRU address >1700 or >1740, respectively. The default for both ports is 4800.
Restart Location			Find the restart/return address in the first location of the diagnostic. If you load without DOCS, the address is usually >A0. If you load with DOCS, this address is at the load bias.

NOTE

If you are using the programmer panel to run SATs, set the error print/display and header print/display flags to zero. Then, set the idle-on-error flag to >FFFF to cause the computer to idle when an error is detected. The loop count and error message numbers will be displayed on the programmer panel.

If you want messages to be sent to an output device, you can update flags in the SAT. However, these addresses need not be modified if the diagnostic is loaded under the control of DOCS. The devices that you selected during DOCS initialization are used as output and error message devices for the SAT.

To update flags in the SAT, look at the output terminal selector flag description in Table 2-1. As an example, suppose your hardware configuration has only one 911 VDT located at CRU address >140 and you are executing the diagnostic test without the control of DOCS. You must modify relative address >3A to the value 2. You have just indicated that your output is only to be sent to the 911 VDT. If your release is dated the fourth quarter of 1979 (Release 4.1.0) or later look at Table 2-3. (If your release is dated prior to this time, use the information in Table 2-2 for this example.) Next, you must modify relative address >46 to the address where your 911 VDT is located (>140). Your output will now be sent to the 911 VDT.

Table 2-2 SAT Output Device CRU Base Addresses -- Prior to Release 4.1.0

Output Terminal	Default Value	AU04	AU04TST	AU05	MAPTST
733 ASR/KSR	>000	>40 (>E0)	>40 (>E0)	>40 (>E0)	>40 (>E0)
913 VDT	>0C0	>46 (>E6)	>48 (>E8)	NA NA	>18EA (>198A)
911 VDT	>100	>48 (>E8)	>4A (>EA)	NA NA	>1986 (>1A26)
Line printer (TTYEIA IF)	>060	>44 (>E4)	>46 (>E6)	NA NA	>1880 (>1920)

Notes:

Relative address locations are shown without parentheses.

Load bias address >A0 locations are shown with parentheses.

NA indicates not available for modification.

Table 2-3 SAT Output Device CRU Base Addresses -- Release 4.1.0 and Later Releases

Output Terminal	Default Value	AU04/AU05/AU04TST Diagnostics	MAPTST
733 ASR/KSR	>000	>42 (>E2)	>40 (>E0)
913 VDT	>0C0	>44 (>E4)	>18EA (>198A)
911 VDT	>100	>46 (>E6)	>1986 (>1A26)
940 EVT:			
S300 /10A communications port	>1700	>4A (>EA)	NA NA
CI-402 COMIF	>1700 >040	>4A (>EA)	NA NA
Line printer (TTYEIA IF)	>060	>48 (>E8)	>1880 (>1920)

Notes:

Relative address locations are shown without parentheses.

Load bias address >A0 locations are shown with parentheses.

NA means not applicable.

A SAT only supports one 940 EVT as an output device. If more than one 940 EVT is configured on a system where a SAT is loaded, the SAT first checks address >040 to see if a device is present. If no device is present at that address, it uses the device at address >1700.

Section 3

Standard Diagnostic Operational Control System (DOCS)

3.1 INTRODUCTION

The Diagnostics Operational Control System (DOCS) is a small operating system that supports and controls all diagnostic tests, except stand-alone tests (SATs). Standard DOCS contains verb modules that are already linked with DOCS.

If the user has never before attempted to use DOCS, he should first completely familiarize himself with the information in paragraph 3.4.1 on communication conventions.

If you are using cassette tapes or the UST, then you must use Standard DOCS. Standard DOCS is used for both 774 and 990-based computers. Standard DOCS consists of four types:

* Maxi DOCS
* Mini DOCS
* Batch Command Stream (BCS) DOCS
* Debug DOCS

The first two types are discussed later in this section. BCS and Debug DOCS are discussed in Sections 9 and 10, respectively.

The following paragraphs discuss DOCS object modules and supported equipment and media.

3.2 DOCS OBJECT MODULES

The load media contains fully linked object (FLO) modules. A DOCS FLO module is the result of DOCS modules linked to a loader module. Each FLO module has a unique identifier which is formed by combining:

1. A one-character designator for the loader type
2. A two-character designator for the DOCS type
3. The DOCS suffix

For example, the unique identifier for Maxi DOCS linked with the unit record (cassette/UST) loader is:

UMXDOCS

where:

U represents the unit record (cassette/UST) loader.

MX indicates Maxi.

DOCS is the suffix.

The identifiers used for Standard DOCS FLO modules for unit record load media are as follows:

Loader Type	DOCS Version
U -- Unit record (cassette, 990 maintenance diagnostic unit (MDU), microfloppy, Universal Systems Terminal (UST))	MX -- Maxi DOCS
	MN -- Mini DOCS
	BC -- Batch Command Stream DOCS
	DB -- Debug DOCS

Table 3-1 summarizes the identifiers for the Standard DOCS FLO modules.

Table 3-1 Standard DOCS FLO Modules

DOCS Type	Load Media
	Unit Record:
Maxi DOCS	UMXDOCS
Mini DOCS	UMNDOCS
BCS DOCS	UBCDOCS
Debug DOCS	UDBDOCS

3.3 SUPPORTED EQUIPMENT

Standard DOCS supports certain computer models, interactive and message terminals, and media, as explained in the following paragraphs.

3.3.1 Computer Models

Standard DOCS runs on Models 774 and 990 Computers. The 990 computers currently include the 990/4, 990/5, 990/10, 990/10A, 990/12, S300, and S300A central processing units (CPUs). DOCS contains code that can determine the CPU on which it is running, since interrupt handling and some system parameters depend on this information.

3.3.2 Interactive and Message Terminals

An interactive terminal allows you to supply DOCS with the information it needs to proceed with a test and receive header and error messages that DOCS generates. Message terminals can only receive output, specifically error messages. If you want a hardcopy of any information other than error messages, you must use either an ASR/KSR, a KSR, or a UST with a printer as your interactive terminal. The following interactive and message terminals are currently supported:

Interactive Terminals/Interfaces	Message Terminals
743 KSR - TTYEIA	743 KSR
733 ASR/KSR - TTYEIA, /5 Port	733 ASR/KSR
820 KSR - TTYEIA	810 Line printer
840 KSR - TTYEIA	820 Line printer
913 VDT - 913 CTL	840 Line printer
911 VDT - 911 CTL	850 Line printer
940 EVT - CI401, CI402, /10A Port, CI421, CI422, EI300	855 Line printer
931 EVT - CI402, /10A Port, CI421, CI422, EI300	880 Line printer
UST -	Optional Printer

Once DOCS is loaded, it determines the type and location of the terminal at which you are responding. Then, DOCS asks you for any additional information required to handle the interactive and message terminals.

3.3.3 Load Media

Standard DOCS and the diagnostic test modules are packaged on cassettes or UST microfloppy diskettes. The structure and types of available load media are also discussed in Section 7.

3.4 MAXI DOCS

Maxi DOCS, the basic type of standard DOCS, can run on all Model 990, S300 and S300A Computers and can communicate with you through interactive terminals. Mini DOCS is a subset of Maxi DOCS. BCS and Debug DOCS consist of Maxi DOCS with the batch command stream and debug features, respectively.

The following paragraphs discuss Maxi DOCS and apply to all types of Standard DOCS. Differences in Mini DOCS are noted in individual paragraphs. Refer to Sections 9 and 10 for information on BCS and Debug DOCS, respectively.

To create Maxi DOCS using Dynamic DOCS, refer to Section 4.

3.4.1 Communication Conventions

The conventions used to communicate with all Standard DOCS are described in the following list. These conventions apply to any supported interactive terminal.

* A hyphen (-) is displayed when DOCS waits for a response. If you are using a VDT, a blinking cursor appears after the hyphen.

* Press the RETURN or NEW LINE key to enter all input. If you are using a VDT, the cursor disappears after you press the RETURN key, indicating that no more input is required.

* A period must precede all DOCS-supported verbs (for example, .IS). Test verbs are not preceded by a period.

* Most prompts are designed with predefined default values that are displayed with the prompts. Accept the default value by pressing the RETURN or NEW LINE key. Otherwise, enter a different value in response to the prompt and then press the RETURN or NEW LINE key. (In most cases, the new value is used as the default value the next time the prompt is issued.)

* The RUB OUT key (733 ASR/KSR), the left arrow (<--) key (911 and 913 VDTs and the UST), and the backspace key (940 - 931, 820 and 840 KSRs, and the UST) void your response and allow you to reenter the response. These keys are only effective before you press the RETURN or NEW LINE key.

* Hexadecimal numbers are used at all times. You can enter from 1 to 64 contiguous hexadecimal digits. You cannot have embedded blanks and leading zeros are not required. The number is internally right-justified with leading zeros to the left. For example, if a memory test requests the ending memory location to be tested and the address is >1FFFF, your response would be:

  ENDING MEMORY ADDRESS? (DEF=13B3E) - 1FFFF

  If the ending address is >730, respond as follows:

  ENDING MEMORY ADDRESS? (DEF=13B3E) - 730

* When you enter a number, DOCS verifies that it is a legal hexadecimal value. If it is not hexadecimal, you receive an error message and must reenter the number.

* Respond to all YES/NO prompts using 1 for YES and 0 for NO. If you enter any other value, you receive an error message and must reenter the response.

* All ASCII data is entered as a string consisting of 1 to 64 contiguous characters.

* The VERB? prompt indicates that DOCS is ready to receive a verb. (Verbs are discussed in Section 6.) Enter any legal verb. If you enter an illegal verb, the ILLEGAL VERB message appears.

* If you press the @ key on any interactive terminal, or the 911 VDT CMD key, 913 VDT HELP key, 940/931 EVT ESC key, or 733/743 ASR ESC key while a test is executing, DOCS terminates the test activity and responds with the VERB? prompt. The test may require reinitialization.

* The 911/931 VDT blank orange key and 913 VDT PRINT key are used as pause keys. These keys allow you to stop test execution so you can view the test output on a VDT. Once you press a pause key, the test stops executing and the computer enters the idle mode. To resume execution, press the pause key again. A pause during execution can affect the testing ability of a diagnostic, depending on what the diagnostic is doing when it is stopped. For example, if you press a pause key during a timing loop, the testing ability is affected. (Note that the pause function cannot work if the interrupt mask of the computer is set to mask the interrupt level of the I/O device. Further note that the 940 EVT has no pause capability.)

3.4.2 Operating Procedures

After you load Standard DOCS, press the character Y on the interactive terminal. (If you are using a 940/931 EVT, the status line will be full duplex (FDPLX) when DOCS is ready for you to press the character Y.) DOCS determines the type of computer on which it is running and locates the interactive terminal you are using by scanning the CRU base addresses >0000 through >1F00, and TILINE addresses >F800 through >FC00 for a Y, and checking the Front Panel (>1FE0). The interactive terminal is not restricted to a predesignated CRU or TILINE base address but can be located at any base address within the specified ranges. Since DOCS scans both address ranges, no time limit exists for you to respond. However, you should allow sufficient time for DOCS to load before pressing Y.

Once DOCS locates the interactive terminal, a display of initialization prompts appears, and you can proceed with Standard DOCS initialization and test module loading and execution. These procedures are discussed in the following paragraphs and illustrated by the flowchart in Figure 3-1.

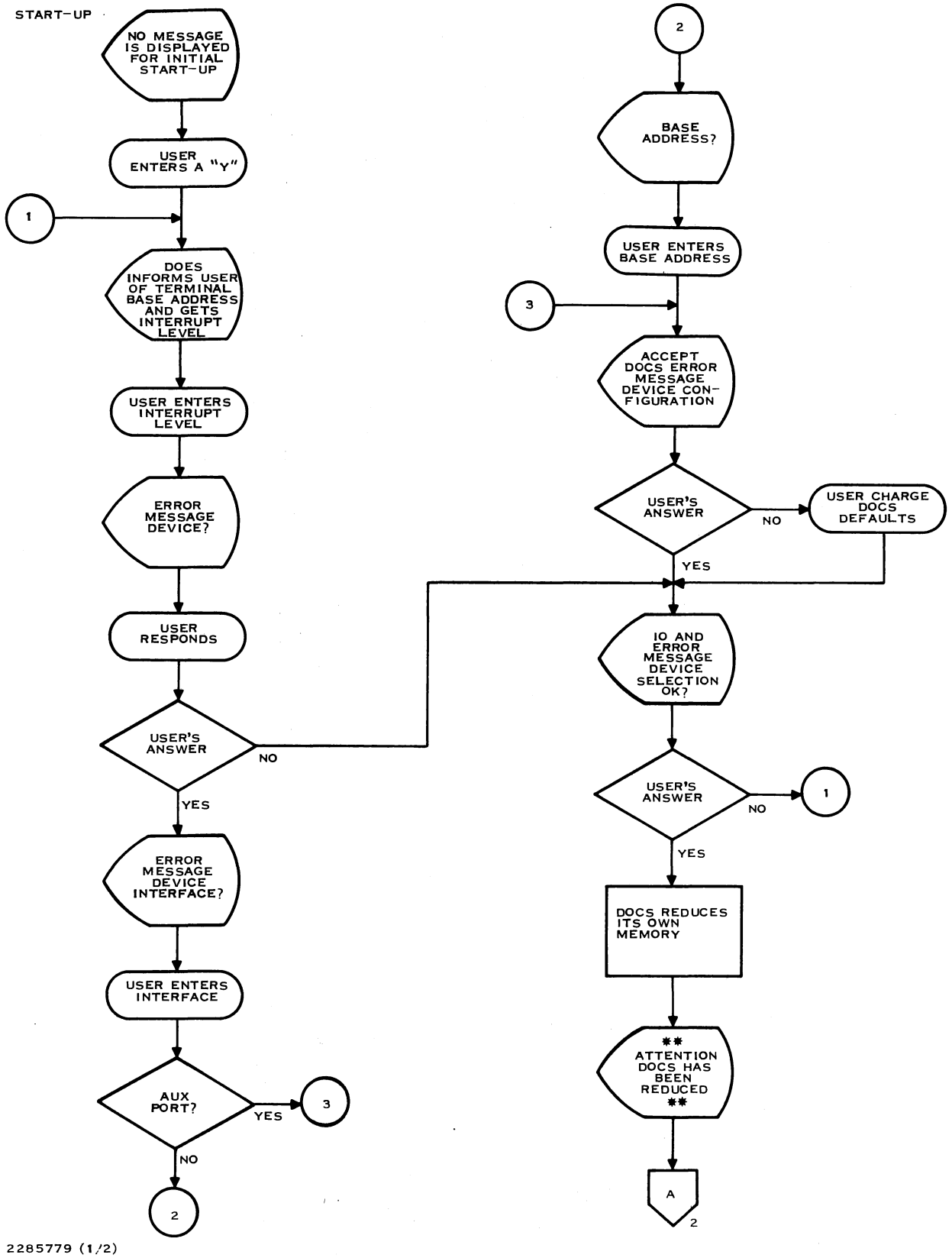
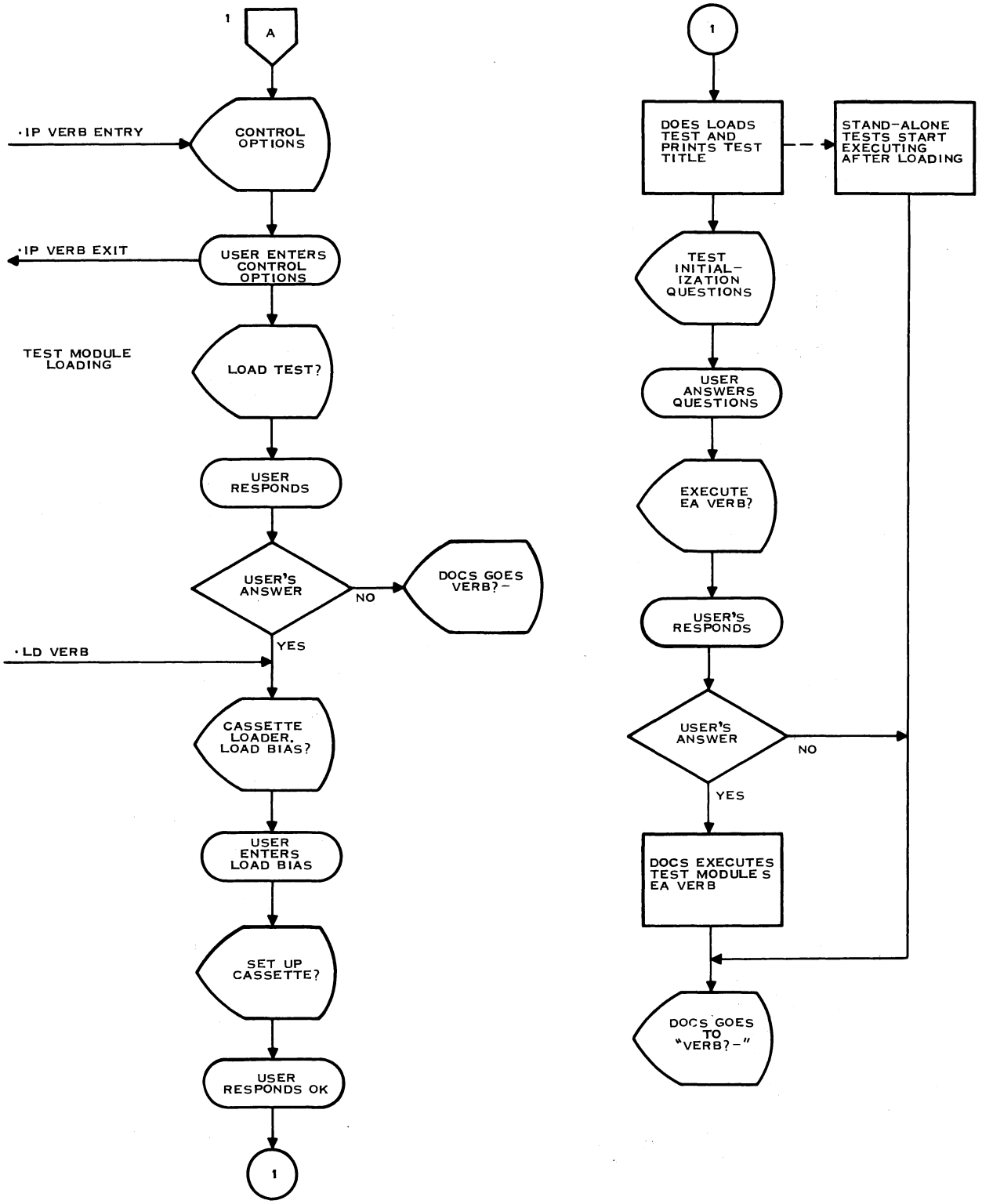


Figure 3-1 Standard Maxi DOCS Flowchart (Sheet 1 of 2)



2285779 (2/2)

Figure 3-1 Standard Maxi DOCS Flowchart (Sheet 2 of 2)

3.4.2.1 Initializing Standard Maxi DOCS. During the initialization process, Maxi DOCS requests the following information:

* The interrupt level of the interactive terminal (at which you entered a Y), if the interactive terminal is not at a fixed interrupt level
* The CRU or TILINE base address and interrupt level of the error message terminal (if different from the interactive terminal)
* Configuration of the message terminal(you can select the DOCS default configuration)
* The system control options

The following discussion explains the prompts and messages that appear during initialization. This message is displayed at the beginning of the initialization process:

```
**** 66666I DOCS   VERSION = JJJ/YY *X  TI 990/XX   ****
```

Then, Maxi DOCS informs you of the interactive terminal CRU or TILINE address, will ask for the interrupt level of the interactive device.

```
TERMINAL BASE ADDRESS - XXXX

TERMINAL INTERRUPT LEVEL  - OX
```

If the terminal is at a fixed interrupt level, the prompt appears as:

```
TERMINAL INTERRUPT LEVEL?  DEF = X
```

A default value is displayed with this prompt. If this value is correct, press the RETURN or NEW LINE key. Otherwise, enter the correct value.

Next, Maxi DOCS prompts you for an error message device:

```
ERROR MESSAGE DEVICE? (O=ME, 1=PRINTER) DEF = O
```

Respond with an appropriate value for the error message device prompt. For example, O indicates that you want to use the interactive device to receive messages. If you press the RETURN or NEW LINE key to accept the default value, the interactive terminal serves as the message terminal. If you enter 1, Maxi DOCS prompts you for the interface to which the error message device is connected.

```
ERROR MESSAGE DEVICE INTERFACE:

  0=990/5     1=990/10A     2=CI402     3=CI403
  4=CI421     5=CI422       6=TTYEIA    7=MY AUX

DEF = 2
```

CAUTION

If a printer is to interface with the EI300 Board, then the user must enter a 5 (CI422) in response to this prompt.

NOTE

You can only use the UST as an error message device if you are also using it as the interactive device.

If you entered other than 7 (the auxiliary port) in response to this prompt, the following prompt is displayed:

```
PRINTER BASE ADDR?    DEF = 0800
```

Enter the correct CRU or TILINE base address for the error message device.

DOCS next informs the user of the DOCS default error message device configuration:

```
DOCS ERROR MESSAGE DEVICE DEFAULT CONFIGURATION:

   SPEED:    4800 BAUD
   PARITY:   EVEN
   DUPLEX:   FD-RC (ON)

CHANGE CONFIGURATION?  DEF = 0
```

You can accept this configuration by taking the default to the above prompt. If you want to change the DOCS error message device configuration enter a 1 and the following configuration prompts will appear:

```
DEVICE SPEED?
 0=110   1=200   2=300   3=600
 4=1200  5=2400  6=3600  7=4800
 8=7200  9=9600  A=19200

 DEF = 7

DEVICE PARITY?
 0=NONE  1=ODD  2=EVEN  3=MARK  4=SPACE

 DEF = 2
```

```
DEVICE DUPLEX?
 0=FD  1=FD-RC (OFF)  2=FD-RC (ON)

 DEF = 2

ARE THE SELECTED I/O & ERROR MESSAGE DEVICES OK?  (DEF=1)
```

Enter the correct value in each prompt for your own error message device configuration.

DOCS now asks if all the selections above are good and valid. If you need to change anything, (error message device, terminal interrupt level, etc.), this is the last chance before DOCS memory reduction takes place. If everything is OK take the default. If changes are necessary, enter 0. This will return the user to the terminal base address prompt, and the potential for entering the correct data. If the default is accepted the following message is displayed:

```
***** ATTENTION *****

   DOCS HAS BEEN REDUCED IN SIZE.  IT IS NECESSARY TO
RELOAD DOCS IN ORDER TO CHANGE I/O OR ERROR MESSAGE DEVICES.
```

The following prompt allows you to select system control options:

```
ENTER CONTROL OPTIONS SEPARATED BY COMMAS:
(E = ERR MSG'S, H = HDR MSG'S, N = ERR #'S, P = PAUSE ON ERR'S)
DEF = Y,E,H,N -
```

The following system control options are available:

* E -- controls whether error messages are output
* H -- controls whether header messages are output
* N -- controls whether error message numbers are output
* P -- causes the computer to idle if an error occurs

You can take the default or enter one or more of the control options in any order. Separate each option with a comma. You can change the options at any time using the Initialize Control Options (.IP) verb described in Section 6.

3.4.2.2 Test Module Loading and Execution. After initialization, DOCS displays a series of prompts that are used to load and execute a test module. (Alternatively, you can load and execute the test module by entering the correct set of verbs.) The following paragraphs show the prompts and messages that appear during the load process and execution of a test module using the unit (cassette) loader.

The first prompt asks if you want to load a test:

```
LOAD TEST? (DEF = 1) -
```

If you want to load a test using Standard Maxi DOCS, enter 1 (accept the default) in response to this prompt. The following prompt then appears:

```
CASSETTE/UST LOADER

LOAD BIAS?  (DEF = XXXX)
```

This prompt requests an address at which to load the test module. The default value is the first location in memory following the end of DOCS.

If you are loading off an ASR then the following prompt appears:

```
PLEASE: PLACE CASSETTE IN TRANSPORT
        POSITION AT LOAD POINT

PRESS 'RETURN' WHEN UNIT IS READY-
```

If you are loading off an MDU (Maintenance Diagnostic Unit) or a UST (Universal Systems Terminal), then one of the following prompts appears:

```
PLEASE: PLACE CASSETTE IN MDU
        PUSH RESET AND REWIND

          ----------OR----------

      INSERT THE UST MICROFLOPPY
    WITH THE DESIRED DIAGNOSTICS


PRESS 'RETURN' WHEN DONE -

PUSH 'LOAD' ON MDU WHEN UNIT IS READY
```

In most cases the suggested default is acceptable. DOCS then loads the test and the following message appears:

```
END OF TEST = XXXX
```

DOCS puts the file address in its memory allocation table, prints the test title, and executes the initialization verb of the test.

If you want to specify a different load point, avoid selecting an address that would cause DOCS to load a test module over itself. If you specify a load bias that precedes the end of DOCS, the following error message appears:

```
YOU CANT OVERLAY DOCS WITH CASSETTE/UST LOADER
```

When you load the test module, DOCS displays the title of the test module and executes the IT verb. Each time the IT verb executes, DOCS prints the appropriate test title. Then, DOCS requests the information necessary for correct execution of the test. If the test initialization process is interrupted for any reason, the following message is displayed:

```
                    * * *  WARNING  * * *
INITIALIZATION NOT COMPLETE--ENTER VERB "IT" FOR RESTART
```

When the test module has been initialized, DOCS asks whether to execute the EA verb.

```
EXECUTE EA VERB? (DEF=1) -
```

This verb executes all the standard tests of a diagnostic. Accept the default value (1) to execute these tests. If you enter 1, the EA verb completes execution of the subtests and then issues the VERB? prompt. If you enter 0, the EA verb is not executed and the VERB? prompt appears immediately.

NOTE

You do not need to reload DOCS each time you load a new test module. DOCS remains in memory and loads a new test whenever you execute the Load Diagnostic (.LD) verb. However, DOCS can communicate with only one test module at a time. If you execute the .LD verb after loading a test, all pointers to the old test are lost.

3.4.3 Interrupt Handling

Maxi DOCS handles all interrupts except those handled by some test modules. DOCS can detect two fault conditions relative to interrupts:

* When an interrupt at a level not being used by DOCS is captured, a message indicating an unexpected interrupt appears.

* When control is passed to DOCS from a test module, DOCS checks that the test module has not changed the interrupt traps.

During the Maxi DOCS initialization process, interrupt vectors are set up for the following:

* Power-up (level 0)

* Power fail (level 1)

* System error (level 2)

* Real-time clock (levels 5, 7, or 15)

* Interactive terminal

All other interrupt vectors are set to branch to the code that handles unused interrupts. If one of the unused interrupts occurs, the following message is displayed:

UNEXPECTED INTERRUPT AT LEVEL = <x>

where:

<x> is an integer that represents an interrupt level.

DOCS then issues the VERB? prompt.

Many of the test modules can take over an interrupt trap. This is usually done during test module initialization. However, if you execute the .LD verb, DOCS resets the interrupt vectors to the original DOCS values.

NOTE

The UST does not use an actual interrupt, so it is unaffected by any setting or resetting of the interrupt traps. Its simulated interrupt (>FF), however, can be masked off by setting the interrupt mask to 0, 1 or 2.

If the test module has changed any interrupt trap and you return control to DOCS by pressing the @, CMD, or HELP key, DOCS detects the change in interrupt traps and the following prompt appears:

```
INTERRUPT TRAPS HAVE BEEN CHANGED--RESET TRAPS TO DOCS ?(DEF=0)
```

Enter 1 to reset all interrupts to DOCS. Otherwise, press the RETURN or NEW LINE key to allow the test module to retain control of its interrupt traps.

Exercise care if you prematurely terminate any of the verbs by pressing the @, CMD, or HELP key, since the test may be unable to restore interrupt traps.

If a system error (level 2) occurs, the following message is displayed (the example shown is a TILINE time-out, TO = 1):

```
LEVEL 2 INTERRUPT -- STATUS = 8000
CRU 15 14 13 12 11 10  9  8  7  6  5  4  3  2  1  0
    TO PO IO ME MP NP EV WV SO BP CK AO
     1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0

AT TIME OF ERROR WP = 1596  PC = 7596  ST = 2602
```

Descriptions of the CRU interrupt bits are listed in the following chart:

CRU Bit	Mnemonic	Description
15	TO	TILINE timeout
14	PO	Privilege instruction fetch -- privilege mode off
13	IO	Illegal operation
12	ME	TILINE memory error
11	MP	Memory mapping error
10	NP	Segment not present
9	EV	Execution violation
8	WV	Write violation
7	SO	Stack overflow/underflow
6	BP	Diagnostic breakpoint
5	CK	12 ms clock
4	AO	Arithmetic overflow
3	--	CPU ID bits
2	--	CPU ID bits
1	--	CPU ID bits
0	--	CPU ID bits

If the status returned is 0, the following message appears:

```
INVALID LEVEL 2 INTERRUPT

LEVEL 2 INTERRUPT -- STATUS = 0000
CRU 15 14 13 12 11 10  9  8  7  6  5  4  3  2  1  0
    TO PO IO ME MP NP EV WV SO BP CK AO
     0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
```

After this message is displayed, the VERB? prompt is issued.

If a power failure occurs (level 0 or 1) in a system with battery backup, the following message is displayed:

```
POWER ON
```

DOCS then returns to the VERB? prompt.

3.5 STANDARD MINI DOCS

Standard Mini DOCS is similar in operation to Maxi DOCS except that Mini DOCS only supports the following verbs:

* Execute Stand-Alone Tests (.ES)
* Factory Test (.FT)
* Initialize Print Options (.IP)
* Load Diagnostic (.LD)
* Load Verb Module (.LV)
* Print Available Verbs (.PV)

NOTE

With the release 5.0.0 and greater, neither the Initialize Test (.IT) verb nor the Initialize System (.IS) verb will initialize the system. They give a message to the user to reload DOCS.

You can also create Mini DOCS using Dynamic DOCS. Refer to Section 4 for an explanation of this feature.

3.6 DOCS RESTART PROCEDURE

Since the 5.0.0 release of DOCS you must reload DOCS in order to restart, (change I/O or error message device).

Section 4

Dynamic Diagnostics Operational Control System (DOCS)

4.1 INTRODUCTION

The Diagnostics Operational Control System (DOCS) is a small operating system that supports and controls all diagnostic tests. Dynamic DOCS differs from Standard DOCS because it allows you to link the verb modules listed below. (The verb modules are already linked when you receive Standard DOCS.)

If you are using a disk or magnetic tape loader, you must use Dynamic DOCS. Once you have loaded Dynamic DOCS, you can build other types of DOCS. These types consist of Dynamic DOCS with the appropriate DOCS verb modules loaded and are as follows:

DOCS Type	DOCS Verb Modules
Mini	--
Maxi	MXVERB1 MXVERB2
BCS Run	MXVERB1 BCS RUN (contains execute list mode, and .LL and .EL verbs)
BCS Edit	MXVERB1 MXVERB2 BCSEDT (contains all BCS verbs)
Debug	MXVERB1 MXVERB2 DBVERB1

Dynamic Mini DOCS contains DOCS and the following verbs:

Initialize Print Options (.IP)
Load Diagnostic (.LD)
Load Verb Module (.LV)
Clear Interactive VDT Screen (.CS)
Execute Stand-Alone Test (.ES)

```
        Print Available Verbs (.PV)
Change Load Subdirectory Name (.CN)
       Modify Load Parameters (.ML)
                 Rewind Tape (.RW)
```

Dynamic Mini DOCS does not contain any verb modules.

Dynamic Maxi DOCS is discussed in this section. BCS RUN, BCS Edit, and Debug DOCS are discussed in Sections 9 and 10.

The DOCS verbs allow you to load, initialize, and execute test modules. Each module supports a set of verbs related to the device that the test evaluates. In addition, DOCS supports a set of verbs that help you diagnose problems.

If you run an SAT under the control of Dynamic DOCS, you must load Dynamic DOCS into memory before loading any module that is part of an SAT.

4.2 DOCS OBJECT MODULES

The load media contains fully linked object (FLO) modules. DOCS FLO module is the result of a DOCS module linked to a loader module. Each FLO module has a unique identifier which is formed by combining:

1. A one-character designator for the loader
2. A two-character designator for the DOCS type
3. The DOCS suffix

For example, the unique identifier for Dynamic DOCS linked with the disk loader is:

DDNDOCS

where:

D represents the disk loader.

DN indicates Dynamic.

DOCS is the suffix.

The identifiers used for Dynamic DOCS FLO modules for disk and magnetic tape load media are as follows:

Loader Type	DOCS Type
D -- Hard disk and double-sided,	DN -- Dynamic DOCS

double-density (DSDD) diskette

T -- Open reel and eight-inch cartridge tapes

DN -- Dynamic DOCS

The identifiers for the Dynamic DOCS FLO modules are as follows:

* DDNDOCS -- DOCS FLO module for hard disk and DSDD diskette
* TDNDOCS -- DOCS FLO module for open reel and eight-inch cartridge tapes.

4.3 SUPPORTED EQUIPMENT

Dynamic DOCS supports certain computer models, interactive and message terminals, and media, as explained in the following paragraphs.

4.3.1 Computer Models

Dynamic DOCS supports the same models as Standard DOCS. Refer to paragraph 3.3.1 for the Standard DOCS computer models.

4.3.2 Interactive and Message Terminals

An interactive terminal allows you to supply DOCS with the information it needs to proceed with a test and receive header and error messages that DOCS writes out. Message terminals can only receive output, specifically error messages. If you want a hardcopy of any information other than header and error messages, you must use either an ASR/KSR or a KSR as your interactive terminal.

Dynamic DOCS supports the same interactive and message terminals as Standard DOCS. Refer to paragraph 3.3.2 on Standard DOCS interactive and message terminals.

4.3.3 Load Media

Dynamic DOCS, DOCS verb modules, and the diagnostic test modules are packaged on several types of load media. The structure and types of load media available are discussed in Section 7. However, Dynamic DOCS only supports hard disk, DSDD diskette, open reel magnetic tape, and eight-inch cartridge magnetic tape load media.

4.4 COMMUNICATION CONVENTIONS

The conventions used by Dynamic DOCS are described paragraph 3.4.1 which deals with the communication conventions of Standard DOCS, which are the same as the conventions of Dynamic DOCS. Paragraph 3.4.1 is extremely important to the first-time user of any form of DOCS.

4.5 OPERATING PROCEDURES

The operating procedures for Dynamic DOCS are the same as those for Standard DOCS. Please refer to paragraph 3.4.2 of this manual which deals with Standard DOCS operating procedures and general DOCS information.

Dynamic DOCS differs from Standard DOCS in that it allows you, after initialization, to link various verb modules. See Figure 4-1 for a clarification of this distinction.

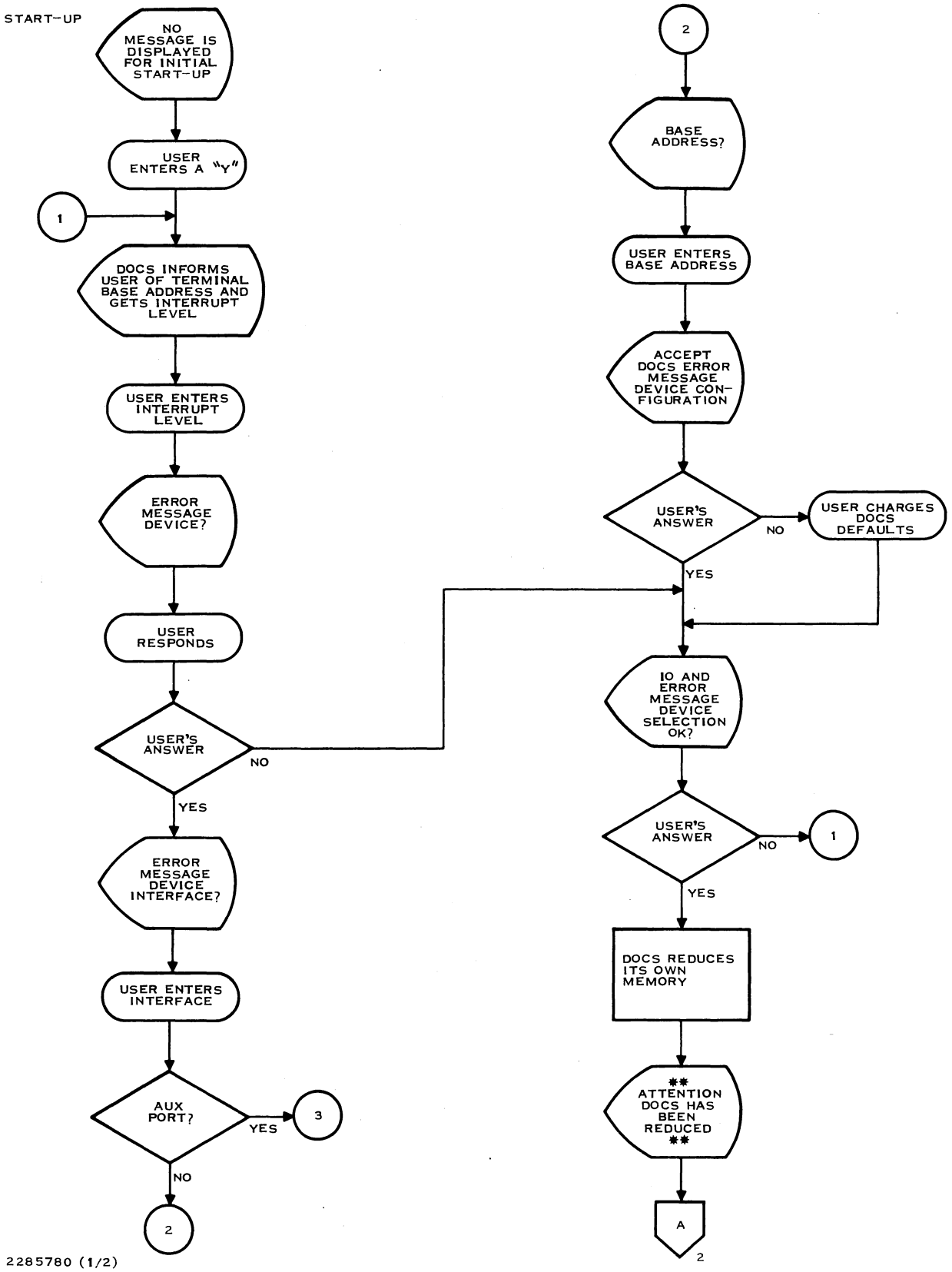


Figure 4-1 Dynamic DOCS Flowchart (Sheet 1 of 2)

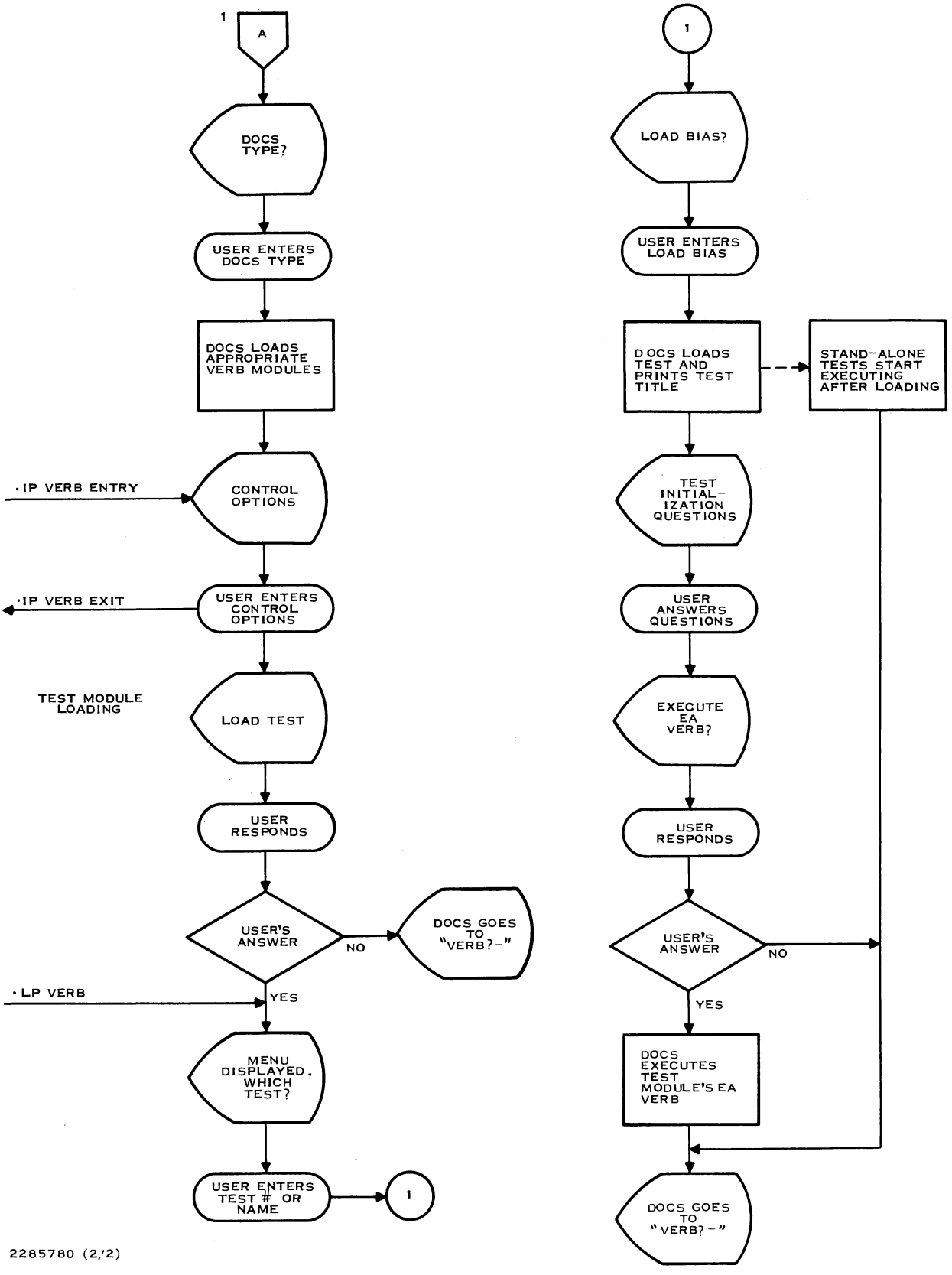


Figure 4-1 Dynamic DOCS Flowchart (Sheet 2 of 2)

4.5.1 Initializing Dynamic DOCS

During the initialization process, Dynamic DOCS differs from Standard DOCS only in that after the DOCS memory reduction the user is prompted for the DOCS type:

```
DOCS TYPE?
(0=MINI, 1=MAXI, 2=BCSRUN, 3=BCSEDT, 4=DEBUG) DEF = 1
```

To load Maxi Docs take the default of 1. Mini Docs is a subset of Maxi DOCS. BCS and DEBUG DOCS consist of Maxi DOCS with the batch command stream and debug features, respectively. Refer to Sections 9 and 10 for information on BCSRUN, BCSEDT, and DEBUG DOCS.

4.5.2 Test Module Loading and Execution

After initialization, DOCS displays a series of prompts that are used to load and execute a test module. (Alternatively, you can load and execute the test module by entering the correct set of verbs.) The following paragraphs show the prompts and messages that appear during the load process and execution of a test module for disk Dynamic DOCS.

The first prompt asks if you want to load a test:

```
LOAD TEST? (DEF=1) -
```

If the test has already been loaded, accept the default value 1 (YES). If you want to run DOCS without a test, enter the value 0 (NO). Then, DOCS either executes the IT verb of the previously loaded test module or issues the VERB? prompt.

If you enter 1 (YES) in response to the LOAD TEST? prompt, DOCS leads you through the test module load procedure. The format of this procedure varies with respect to the DOCS load media. These formats are described with the Load Diagnostic (.LD) verb in Section 6.

If you choose to load a test, the menu listing the DOCS verb modules and diagnostic tests is displayed:

```
MENU FOR VCATALOG DIRECTORY

00) BCSRUN      01) BCSEDT      02) DBVERB1     03) DDNDOCS
04) MXVERB1     05) MXVERB2     06) MEMTST

FILE # OR NAME?
```

This menu lists each of the DOCS test modules. Select the test you want to load by entering the appropriate number or file name from this menu.

Next, the LOAD BIAS? prompt is displayed:

```
LOAD BIAS? DEF = XXXXX -
```

This prompt requests an address at which to load the test module. The default value is the first location in memory following the end of DOCS. In most cases, accept the default value for this prompt. DOCS loads the test and the following message appears:

```
END OF TEST = XXXX
```

DOCS puts the file in its memory allocation table, prints the test title, and executes the initialization verb of the test.

If you want to specify a different load point, avoid selecting an address that would cause DOCS to load a test module over itself. If you specify a load bias that precedes the end of DOCS, the following message appears:

```
WARNING - THIS OVERLAYS DOCS
ARE YOU SURE ?    DEF=0
```

When you must load a test module in the area that contains DOCS, follow these steps:

1. Reload DOCS, specifying a load bias that moves DOCS to a different memory area.
2. Press Y on the interactive terminal.
3. Respond to the initialization prompts.
4. Load the test module.

When you load the test module, DOCS displays the title of the test module and executes the IT verb. Each time the IT verb executes, DOCS prints the appropriate test title. Then, DOCS requests the information necessary for correct execution of the test. If the test initialization process is interrupted for any reason, the following message is displayed:

```
          * * *  WARNING  * * *
INITIALIZATION NOT COMPLETE--ENTER VERB "IT" FOR RESTART
```

When the test module has been initialized, DOCS asks whether to execute the .EA verb:

```
EXECUTE EA VERB? (DEF=1) -
```

If you enter the default value 1 (YES), the EA verb executes selected tests in the test module. The EA verb completes its execution of the tests and then responds with the VERB? prompt.

If you enter 0, the EA verb is not executed and the VERB? prompt appears immediately.

NOTE

You do not need to reload DOCS each time you load a new test module. DOCS remains in memory and loads a new test whenever you execute the .LD verb. However, DOCS can communicate with only one test module at a time. If you execute the .LD or .IS verb after loading a test, all communication with the test is lost.

4.5.3 Interrupt Handling

Dynamic DOCS handles all interrupts in the same manner as Standard DOCS. Refer to paragraph 3.4.3 on Standard DOCS interrupt handling.

4.6 DOCS RESTART PROCEDURE

Since the 5.0.0 release of DOCS and later, you must reload DOCS in order to restart (change I/O or error message devices).

Section 5

S200DOCS

5.1 INTRODUCTION

The Diagnostics Operational Control System (DOCS) is a small operating system that supports and controls all diagnostic tests. S200DOCS is an adaptation of Dynamic DOCS developed exclusively for the S200 system.

The five types of S200DOCS are as follows:

* Mini DOCS
* Maxi DOCS
* Batch Command Stream (BCS) Run DOCS
* Batch Command Stream (BCS) Edit DOCS
* Debug DOCS

Mini DOCS for S200DOCS contains DOCS and the following verbs:

Initialize System	(.IS)
Initialize DOCS Control Parameters	(.IT)
Initialize Print Options	(.IP)
Load Diagnostic	(.LD)
Load Verb Module	(.LV)
Clear Interactive VDT Screen	(.CS)
Execute Stand-Alone Test	(.ES)
Print Available Verbs	(.PV)
Change Load Subdirectory Name	(.CN)
Modify Load Parameters	(.ML)
Rewind Tape	(.RW)

S200DOCS and Maxi DOCS are discussed in this section. S200DOCS BCS Run, BCS Edit, and Debug DOCS are discussed in Sections 9 and 10.

After you load S200DOCS, you can build each of these DOCS types by combining S200DOCS with the appropriate DOCS verb modules. Note that Mini DOCS contains no DOCS verb modules. Table 5-1 lists the five types of DOCS and the DOCS verb modules they contain.

Table 5-1 S200DOCS Verb Modules

DOCS Type	DOCS Verb Modules
Mini	--
Maxi	MXVERB1 MXVERB2
BCS Run	MXVERB1 BCS RUN (contains execute list mode, .LL and .EL verbs)
BCS Edit	MXVERB1 MXVERB2 BCSEDT (contains all BCS verbs)
Debug	MXVERB1 MXVERB2 DBVERB1

5.2 SUPPORTED EQUIPMENT

S200DOCS supports certain computer models, interactive and message terminals, and media, as explained in the following paragraphs.

5.2.1 Computer Models

S200DOCS supports the Model S200 Computers. DOCS contains code that can determine the CPU on which it is running, since interrupt handling and some system parameters depend on this information.

5.2.2 Interactive and Message Terminals

An interactive terminal allows you to supply S200DOCS with the information it needs to proceed with a test and receive header and error messages that S200DOCS generates. Message terminals can only receive output, specifically header and error messages.

S200DOCS currently supports the following interactive and message terminals:

Interactive Terminals	Message Terminals
940 EVT	810 and 820 printers 840 RO printer

Once S200DOCS is loaded, it prompts you for any additional information required to handle the interactive and message terminals.

5.2.3 Load Media

S200DOCS, DOCS verb modules, and the diagnostic test modules for the S200 system are currently packaged on an 8 inch double-sided, double-density (DSDD) diskette and a 5.25 inch DSDD diskette.

5.3 COMMUNICATION CONVENTIONS

The conventions used to communicate with S200DOCS are described in the following list. These apply to any supported interactive terminal.

* A hyphen (-) is displayed when S200DOCS waits for a response. On the 940 EVT, a blinking cursor appears after the hyphen.

* Press the RETURN key to enter all input. The cursor stops blinking after you press the RETURN key, indicating that no more input is required.

* A period must precede all DOCS-supported verbs (for example, .IS). Test verbs are not preceded by a period.

* Most prompts are designed with predefined default values that are displayed along with the prompts. Accept the default value by pressing the RETURN key. Otherwise, enter a different value in response to the prompt and then press the RETURN key. (In most cases, the new value is used as the default value the next time the prompt is issued.)

* The left arrow (<--) key voids your current response and allows you to reenter the response. This key is only effective before you press the RETURN key.

* Hexadecimal numbers are used at all times. You can enter from 1 to 64 contiguous hexadecimal digits, depending on the type of prompt. You cannot have embedded blanks; leading zeros are not required. The number is internally right-justified with leading zeros to the left. For example, if S200DOCS prompts you for the load bias of a test and the default address is >4B30, you can change this value as follows:

  LOAD BIAS? DEF = XXXXXX

  If you want to use the default, press the RETURN key. When you change the value, S200DOCS verifies that the number that you enter is a legal hexadecimal value. If If it is not hexadecimal, you receive an error message and must reenter the number.

* Respond to all YES/NO prompts using 1 for YES and 0 for NO. If you enter any other value, you receive an error message and must reenter the response.

* All ASCII data is entered as a string consisting of 1 to 64 contiguous characters.

* The VERB? prompt indicates that DOCS is ready to receive a verb. (Verbs are discussed in Section 6.) Enter any legal verb. If you enter an illegal verb, the ILLEGAL VERB message appears and you must enter a legal verb.

* If you press the @ or ESC key while a test is executing, S200DOCS terminates the test activity and responds with the VERB? prompt. The test may require reinitialization.

5.4 OPERATING PROCEDURES

Insert the DOCSD200 diskette into the diskette drive and turn the 940 EVT on. If the 940 EVT is already on, turn it off and on again. When the diskette drive access indicator light is off, press the character Y on the interactive terminal. No time limit exists for you to respond. Once you enter Y, the following message and prompt appears:

```
**** TEXAS INSTRUMENTS S200 DOCS  VERSION = JJJ/YY  XX

     USE "RETURN" TO TERMINATE INPUT
     USE "BACKSPACE" TO ERASE INPUT
```

Enter the type of S200DOCS that you need. In most cases, use the default (Maxi DOCS). The following message is displayed:

```
LOADING VERB MODULE MXVERB1   >DONE.
LOADING VERB MODULE MXVERB2   >DONE.
```

Since verb modules merely enhance the normal features and capabilities of DOCS, it is not stricty necessary to load verb modules. If they are needed, however, then after S200DOCS loads the appropriate verb modules, you can proceed with S200DOCS initialization and test module loading and execution. These procedures are discussed in the following paragraphs and illustrated by the flowchart in Figure 5-1.

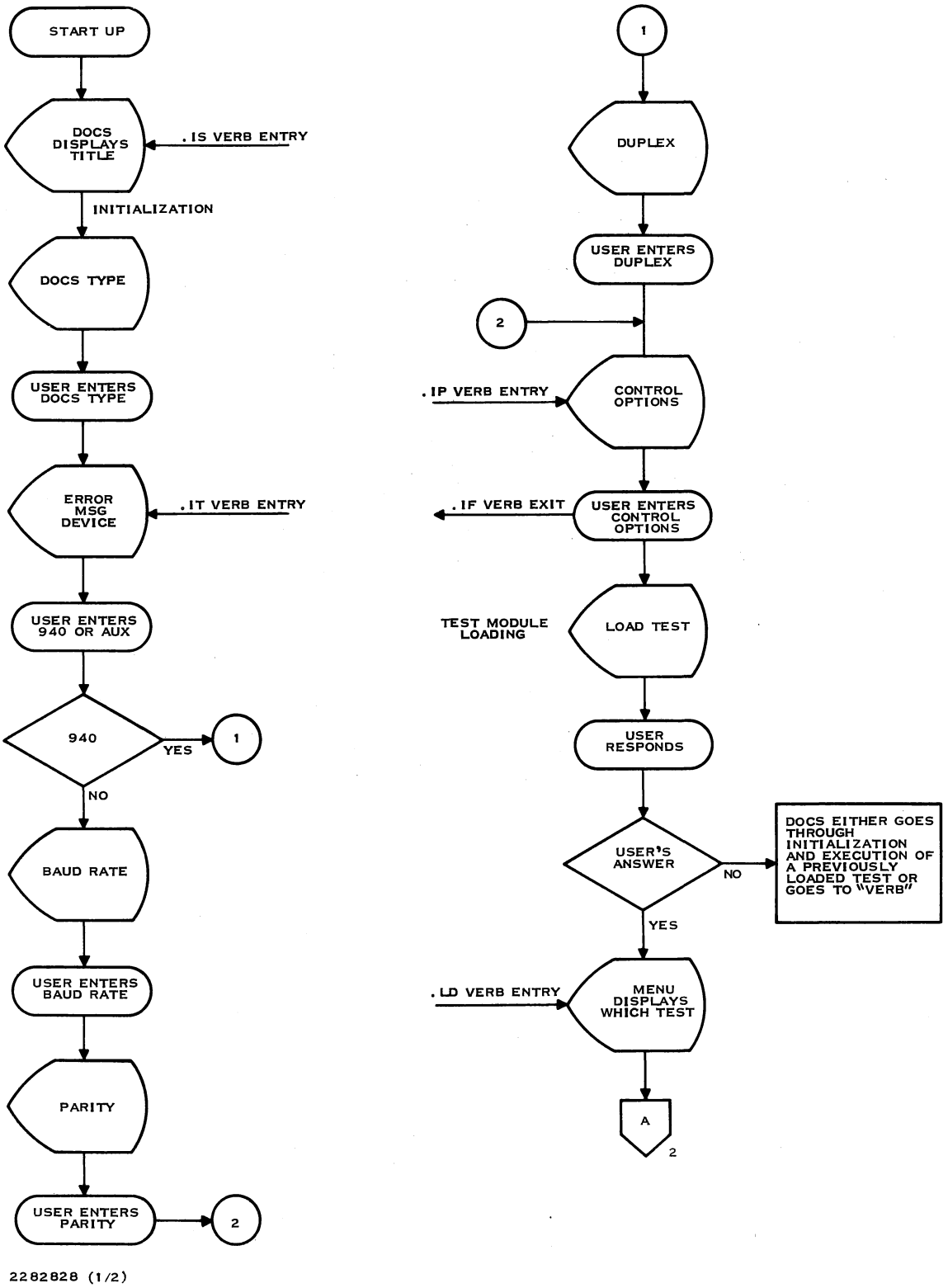


Figure 5-1 S200DOCS Flowchart (Sheet 1 of 2)

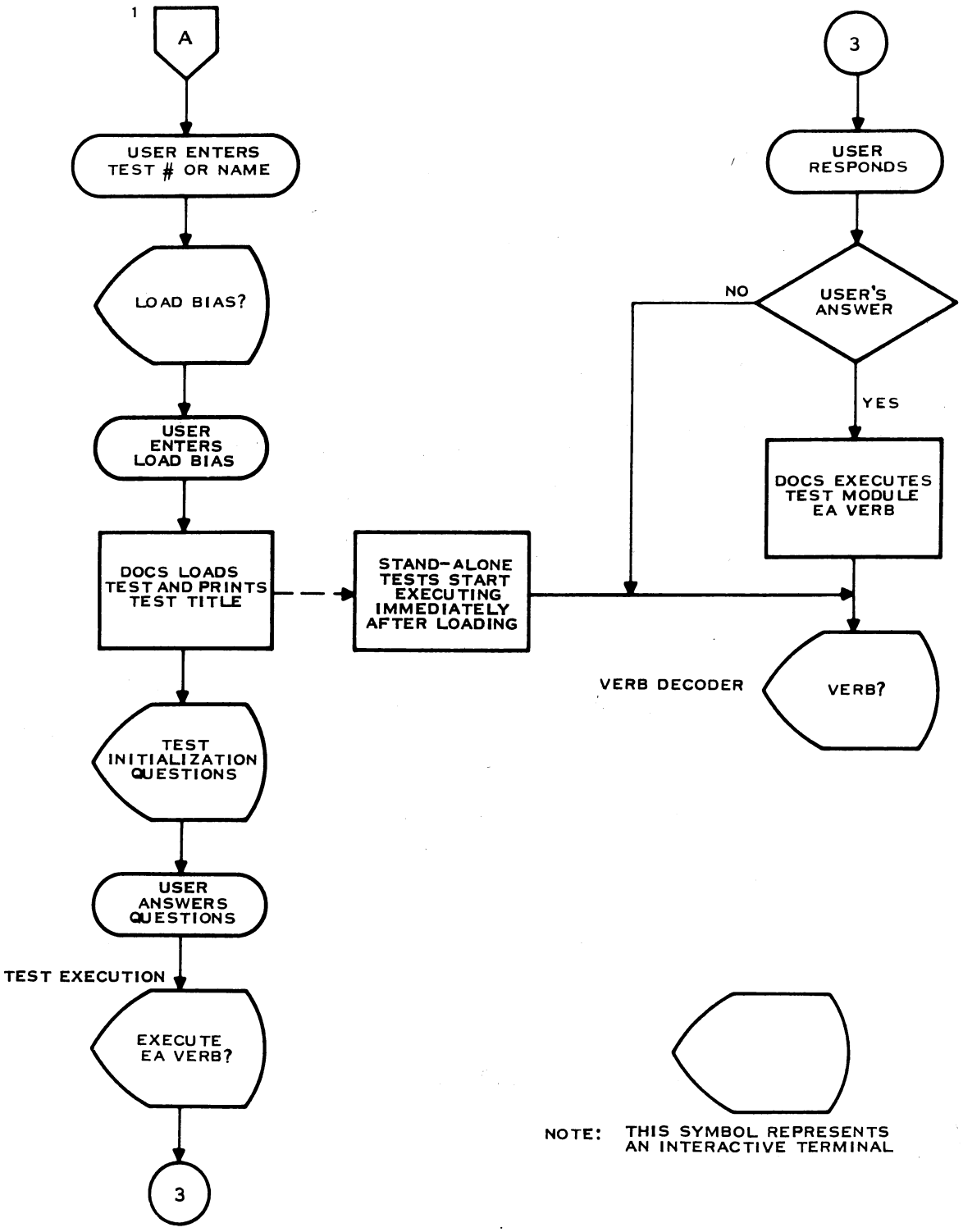


Figure 5-1 S200DOCS Flowchart (Sheet 2 of 2)

5.4.1 Initializing S200DOCS

During the initialization process, S200DOCS requests the following information:

* Type of error message device
* Baud rate, parity, and communications mode (requested only if you specify an auxiliary port)
* System control options

The preceding paragraph instructed you to enter the desired DOCS type. Initialization begins with the following prompt:

```
EVT 940 INT LEVEL ? DEF =04
```

Enter the interrupt level of the 940 EVT. Next, the following prompt is displayed:

```
ERROR MSG DEVICE? (0=810/820  1=940  2=840) DEF=1
```

Enter the type of terminal you want to select for messages. The default for ERROR MSG DEVICE is the interactive terminal. If you press the RETURN key, the interactive terminal also serves as the message terminal and S200DOCS displays no further prompts.

If you specified an 810, 820, or 840 as the error message device, the following prompt is displayed:

```
ERR MSG INTERFACE(0=940 AUX, 1=OTHER)?DEF=0
```

If you accept the default of 0, the following prompts are displayed:

```
BAUD RATE?
    (0=110,  1=200,  2=300,  4=1200, 5=2400,
     6=3600, 7=4800, 8=7200, 9=9600) DEF=2  -

PARITY?
    (0=EVEN, 1=ODD, 2=NONE) DEF=2 -

DUPLEX?
    (0=DC1/DC3, 1=RCON, 2=RCOFF) DEF=0  -
```

When S200DOCS requests control options the following prompt is displayed:

```
ENTER CONTROL OPTIONS SEPARATED BY COMMAS:
(E=ERR MSG'S, H=HDR MSG'S, N=ERR #'S, P=PAUSE ON ERR'S)
DEFAULT=E,H,N
```

Either accept the default or type one or more of the control options in any order. Separate each option from the next by a comma. You can change the options at any time by using the .IP verb as described in Section 6. Interpret the abbreviations for the control options as follows:

* E -- controls whether error messages are written
* H -- controls whether header messages are written
* N -- controls whether error message numbers are written
* P -- causes the computer to idle if an error occurs

You can terminate S200DOCS initialization at any time by pressing the @ or ESC key. The following message appears:

```
* * *  WARNING  * * *
INITIALIZATION NOT COMPLETE--ENTER VERB ".IS" FOR RESTART
```

This message indicates that defaults will be used for prompts not already answered and that interrupt and XOP traps have been initialized to match these defaults. Any test module initialized before this point may require reinitialization.

After you respond to all the initialization prompts, the prompts related to test module loading are displayed.

5.4.2 Test Module Loading and Execution

After initialization, S200DOCS displays a series of prompts that are used to load and execute a test module. (Alternatively, you can load and execute the test module by entering the correct set of verbs.) The following paragraphs show the prompts and messages that appear during the load process and execution of a test module and explain how to respond to each prompt.

The first prompt asks if you want to load a test:

```
LOAD TEST? (DEF=1) - 3
```

If your disk is not write-protected, this message appears:

```
WRITE PROTECT THE DISK AND HIT RETURN-
```

The user must remove the silver tab from the diskette to write protect it. To load a test, enter 1 in response to the LOAD TEST? prompt. The menu for the S$UDIAG directory is displayed. An example is as follows:

```
MENU FOR S$UDIAG DIRECTORY

00) AU200      01) BCSEDT     02) BCSRUN     03) COM90X
04) DBVERB1    05) DSKCOM     06) DSKSA      07) DSKWDX
08) LP810      09) MXVERB1    0A) MXVERBA    0B) S200DOCS
0C) TST8X0     0D) TST940

TYPE: DIAG. # OR FILE NAME ? -
```

The S$UDIAG Directory lists each of the DOCS verb and diagnostic test modules. Select the test you want to load by entering the appropriate number or file name from this menu. For example, to load test TST940, enter the hexadecimal number D or file name TST940 in response to the TYPE: DIAG. # OR FILE NAME? prompt.

Next, the LOAD BIAS? prompt is displayed:

```
LOAD BIAS? DEF=XXXXXX -
```

This prompt requests an address at which to load the test module. The default value is the first location in memory following the end of S200DOCS. In most cases, accept the default for this prompt. S200DOCS loads the test and the following message and prompt appear:

```
END OF TEST = XXXX
TST940 - 940 ELECTRONIC VIDEO TERMINAL VERSION # = MM/YY : *X
INTERFACE DEVICE? (0=CIM, 1=TTY/EIA, 2=TMS990X, 3=SX00 4=CI402)
DEF=0-
```

S200DOCS puts the file in its memory allocation table, prints the test title, and executes the initialization verb of the test. Enter a 3 in response to the INTERFACE DEVICE? prompt. The following prompts appear:

```
ENTER CRU BASE OF INTERFACE DEVICE? DEF=1700 -
INTERRUPT LEVEL OF INTERFACE DEVICE? DEF=6 -
```

Enter the CRU base address and the interrupt level of the interface device. The following prompts appear:

```
IS UUT DOC'S I/O DEVICE? DEF= 0 -
LINE PRINTER ON AUX #1? DEF= 0 -
```

Enter 1 (YES) in response to the IS UUT DOC'S I/O DEVICE? prompt if the unit under test (UUT) is the DOCS I/O device. Otherwise, enter 0 (NO). Respond to the LINE PRINTER ON AUX #1? prompt with a 1 (YES) or 0 (NO).

If you want to specify a different load point, avoid selecting an address that would cause S200DOCS to load a test module over itself. If you specify a load bias that precedes the end of S200DOCS, an error message appears. When you must load a test module in the area that contains S200DOCS, follow these steps:

1. Reload S200DOCS, specifying a load bias that moves S200DOCS to a different memory area.

2. Respond to the initialization prompts.

3. Load the test module.

When you load the test module, S200DOCS displays the title of the test module and executes the IT verb. Each time the IT verb executes, S200DOCS displays the appropriate test title. Then, S200DOCS requests the information necessary for correct execution of the test. If the test initialization process is interrupted for any reason, S200DOCS displays the following message:

```
* * * WARNING * * *
INITIALIZATION NOT COMPLETE--ENTER VERB "IT" FOR RESTART
```

When the test module has been initialized, S200DOCS asks whether to execute the EA verb.

```
EXECUTE EA VERB? (DEF=1) -
```

This verb executes all the standard tests of a diagnostic. If you enter the default value 1 (YES), the EA verb executes selected tests in the test module. The EA verb completes its execution of the tests and then responds with the VERB? prompt.

If you enter 0, the EA verb is not executed and the VERB? prompt appears immediately.

NOTE

You do not need to reload S200DOCS each time you load a new test module. S200DOCS remains in memory and loads a new test whenever you execute the Load Diagnostic (.LD) verb. However, S200DOCS can communicate with only one test module at a time. If you execute the .LD or .IS verb after loading a test, all communication with the test is lost.

5.4.3 Interrupt Handling

S200DOCS handles all interrupts except those handled by a test module. S200DOCS can detect two fault conditions relative to interrupts:

* When an interrupt at a level not being used by S200DOCS is captured, a message indicating an unexpected interrupt appears.

* When control is passed to S200DOCS from a test module, S200DOCS checks that the test module has not changed the interrupt traps.

During the S200DOCS initialization process, interrupt vectors are set up for the following:

* Power-up (level 0)

* Power fail (level 1)

* System error (level 2)

* Real-time clock (level 3)

* Interactive terminal (level 4)

* Peripheral bus interface (level 14)

All other interrupt vectors are set to branch to the code that handles unused interrupts. If one of the unused interrupts occurs, the following message is displayed:

UNEXPECTED INTERRUPT AT LEVEL = <x>

where:

<x> is an integer that represents an interrupt level.

Then, S200DOCS issues the VERB? prompt.

Many of the test modules will take over an interrupt trap. This is usually done during test module initialization. However, if you execute the .IS, .IT, or .LD verb after initializing the test module, S200DOCS resets the interrupt vectors to the original S200DOCS values.

If the test module has changed the S200DOCS interrupt traps for level 0, 2, 3, 4, or 14 and you return control to S200DOCS by pressing the @ or ESC key, S200DOCS detects the change in interrupt traps and the following prompt appears:

```
INTERRUPT TRAPS HAVE BEEN CHANGED--RESET TRAPS TO DOCS ?(DEF=0)
```

Enter 1 to reset all interrupts to S200DOCS. Otherwise, press RETURN to allow the test module to retain control of the interrupt traps.

Exercise care if you prematurely terminate any of the verbs by pressing the @ or ESC key, since the test may be unable to restore interrupt traps.

If S200DOCS receives a system error (level 2), then a message describing the interrupt appears on your screen (the example shown is a TILINE time-out, TO = 1):

```
INVALID LEVEL 2 INTERRUPT

LEVEL 2 INTERRUPT -- STATUS = 8000
CRU 15 14 13 12 11 10  9  8  7  6  5  4  3  2  1  0
    TO PO IO ME MP NP EV WV SO BP CK AO
     1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0

AT TIME OF ERROR WP = 1596  PC = 609A  ST = 2602
```

Descriptions of the CRU interrupt bits are listed in the following chart:

CRU Bit	Mnemonic	Description
15	TO	TILINE time-out
14	PO	Privilege instruction fetch - privilege mode off
13	IO	Illegal operation
12	ME	TILINE memory error
11	MP	Memory mapping error
10	NP	Segment not present
9	EV	Execution violation
8	WV	Write violation
7	SO	Stack overflow/underflow
6	BP	Diagnostic breakpoint
5	CK	12 ms clock
4	AO	Arithmetic overflow
3	--	990/12 CPU ID bits
2	--	990/12 CPU ID bits
1	--	990/12 CPU ID bits
0	--	990/12 CPU ID bits

If the status is returned to 0, the following message appears:

```
INVALID LEVEL 2 INTERRUPT

LEVEL 2 INTERRUPT -- STATUS = 8000
CRU 15 14 13 12 11 10  9  8  7  6  5  4  3  2  1  0
    TO PO IO ME MP NP EV WV SO BP CK AO
     0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
```

Then, S200DOCS issues the VERB? prompt. If the software gets lost and somehow goes to level 0, the following message appears:

```
POWER ON OCCURRED

ENTER A 'Y'
```

To reinitialize DOCS, enter a Y in response to this prompt.

5.5 S200DOCS RESTART PROCEDURE

If you want to use a different type of S200DOCS or if you need to rebuild S200DOCS, issue the .IS verb. This causes S200DOCS to restart as if it had just been loaded.

Section 6

DOCS Verbs

6.1 INTRODUCTION

DOCS verbs are general utility verbs that load modules and help you diagnose problems. The verbs supported by DOCS are dependent on the category of DOCS being used. All DOCS verbs are preceded by a period (.) to distinguish them from the test module verbs. DOCS verbs are divided into the following groups:

* Verbs specific to Dynamic DOCS
* Verbs specific to S200DOCS
* Verbs common to Dynamic DOCS and S200DOCS
* Verbs common to S200DOCS, Standard and Dynamic DOCS

The following paragraphs explain each group of verbs.

6.2 VERBS SPECIFIC TO DYNAMIC DOCS

Table 6-1 lists the verbs that are specific to Dynamic DOCS. The following paragraphs describe the verbs.

Table 6-1 Verbs Specific to Various DOCS Types

Verb Format	DOCS TYPE S200	Standard	Dynamic
.ML Modify Load Parameters			xx
.RW Rewind Tape			xx
.FT Factory Test	xx		
.IS Initialize System	xx		
.IT Initialize DOCS Control Parameters	xx		
.CN Change Load Subdirectory Name	xx		xx
.IP Initialize Print Options	xx	xx	xx
.LD Load Diagnostic Test Module	xx	xx	xx
.LV Load DOCS or External Verb Module	xx	xx	xx
.PV Print Available Verbs	xx	xx	xx
.ES Execute Stand-Alone Test	xx	xx	xx
.MI Memory Initialize	xx	xx	xx
.CW Read/Write CRU Word (looping)	xx	xx	xx

.CS Clear Interactive VDT Screen	xx	xx	xx
.RS TILINE I/O Reset	xx	xx	xx
.DM Dump Memory	xx	xx	xx
.SM Show Memory Allocation	xx	xx	xx
.MM Modify Memory	xx	xx	xx
.MV Move Memory	xx	xx	xx
.MS Memory Search	xx	xx	xx
.GO Go to User-Specified Location	xx	xx	xx
.AD Add Hexadecimal Data	xx	xx	xx

6.2.1 Modify Load Parameters (.ML) Verb

The .ML verb changes the TILINE address and unit select of the load device, and number of retries made to the load device. The following example issues the .ML verb and its prompts:

```
VERB ? - .ML
ENTER TILINE ADDRESS. (DEF= xxxx) -
ENTER LOAD UNIT # (0,1,2,3). (DEF= y) -
ENTER MAXIMUM # OFF RETRYS. (DEF= z) -
```

where:

xxxx indicates the default TILINE address of the load device.

y indicates the default load unit number of the load device.

z indicates the default for the maximum number of retries for the load device.

6.2.2 Rewind Tape (.RW) Verb

The .RW verb rewinds the open reel or cartridge tape that is used as the load device. The following example issues the .RW verb. There are no prompts issued.

```
VERB ? - .RW
```

6.3 VERBS SPECIFIC TO S200DOCS

The Factory Test (.FT) verb is used only with S200DOCS. This verb should only be used during factory test and is of no use to other users. Turning on the factory test causes a "~" to be displayed before and after each error message. Turning off the factory test discontinues the display of the "~" during error messages. The following example issues the .FT verb and displays its prompt:

```
VERB ? - .FT
FACTORY TEST ON/OFF (DEF = 0) - <x>
```

where:

<x> indicates that you enter 1 if you want factory test ON. If you want factory test OFF, enter 0.

6.3.1 Initialize System (.IS) Verb

The .IS verb rebuilds the type of DOCS that you need. When you enter the .IS verb, the following prompt appears:

```
ENTER A "Y"
```

After you enter Y, DOCS deallocates all currently loaded files and DOCS verb modules. Then, DOCS returns to its initialization procedure.

6.3.2 Initialize DOCS Control Parameters (.IT) Verb

The .IT verb modifies the initialization parameters of DOCS without rebuilding DOCS. When you issue the .IT verb, the following prompt appears:

```
ENTER A "Y"
```

After you enter Y, DOCS returns to the initialization portion of DOCS. All previously loaded files are still accessible.

6.4 VERBS COMMON TO DYNAMIC DOCS AND S200DOCS

The Change Load Subdirectory Name (.CN) verb is common to Dynamic DOCS and S200DOCS. This verb enables you to place the DOCS modules under a first-level subdirectory on the disk. This process saves load time and also cleans up the directory. All files should be under this subdirectory.

S$UDIAG is the default subdirectory name. If S$UDIAG exists, DOCS searches this directory for files that are to be loaded. If S$UDIAG does not exist, DOCS searches the top-level directory (VCATALOG) for the files.

NOTE

DDNDOCS and S$LOADER (if applicable) must exist under the top-level directory. You should keep all of the diagnostics under VCATALOG.S$UDIAG.

The following example issues the .CN verb and prompts you for a subdirectory name:

```
VERB ? -.CN

ENTER NEW SUBDIRECTORY NAME:  -  <name>
```

where:

<name> indicates that you enter a subdirectory name. Accept the default (.S$UDIAG) by pressing the RETURN or NEW LINE key.

After you change the subdirectory name, all subsequent disk loads search the new directory for files to be loaded.

6.5 VERBS COMMON TO S200DOCS, STANDARD AND DYNAMIC DOCS

Table 6-2 lists the verbs that are common to S200DOCS, Standard and Dynamic DOCS. The following paragraphs describe these verbs.

Table 6-2 Verbs Common to S200DOCS, Standard and Dynamic DOCS

Verb Format	Meaning
.IP	Initialize Print Options
.LD-<loading parameters>	Load Diagnostic Test Module
.LV-<loading parameters>	Load DOCS or External Verb Module
.PV	Print Available Verbs (both DOCS and test module)
.ES-<loops>	Execute Stand-Alone Test
.MI-<add>-<#words>-<data>	Memory Initialize
.CW-<base>-<#bits>-<data>-<loop>	Read/Write CRU Word (looping)
.CS	Clear Interactive VDT Screen
.RS	TILINE I/O Reset
.DM-<add>	Dump Memory
.SM	Show Memory Allocation
.MM-<add>-<data>	Modify Memory

Table 6-2 Verbs Common to S200DOCS, Standard and Dynamic DOCS (Continued)

Verb Format	Meaning
.MV-<from>-<to>-<#words>	Move Memory
.MS-<cond>-<add>-<#words>-<data>	Memory Search
.GO-<ADD>	Go to User-Specified Location (unmapped)
.AD-<data1>-<oper>-<data2>	Add Hexadecimal Data

NOTE

The .ES verb is included with S200DOCS for future use. However, there are no SATs currently available for the S200 computer.

6.5.1 Initialize Control Options (.IP) Verb

The .IP verb enables you to modify the system control options. One or more options can be modified. If you enter more than one option, separate each with a comma. The following example issues the .IP verb and changes three system control options:

```
VERB ? - .IP

ENTER CONTROL OPTIONS SEPARATED BY COMMAS:
(E=ERR MSG'S, H=HDR MSG'S, N=ERR #'S, P=PAUSE ON ERR'S)
DEFAULT = E,H,N) - E,N,P
```

6.5.2 Load Diagnostic (.LD) Verb

The .LD verb loads diagnostic test modules into memory so they can be executed. The following paragraphs discuss the .LD verb for Standard DOCS, Dynamic DOCS, and S200DOCS.

6.5.2.1 Standard DOCS. Diagnostic tests are available on unit record (cassette/UST) load media for Standard DOCS. Refer to Section 3 for instructions on loading diagnostic tests using Standard DOCS.

6.5.2.2 Dynamic DOCS. Diagnostic tests are available on several types of load media for Dynamic DOCS. The type of prompts generated by the .LD verb depends on the type of media you are using. For example, if you load DOCS from hard disk, you are prompted differently than if you load from tape. Generally, if you load DOCS from one media type, DOCS cannot load diagnostic tests from another media type. However, there is one exception: when you use Tape DOCS, you can load from different tape units using the .ML verb.

The following paragraphs discuss the loading protocols for various media after the .LD verb is entered.

Hard Disk/DSDD Diskette Load.

If you execute the .LD verb when you use a hard disk or DSDD diskette as your load media, DOCS displays the current catalog being loaded and the menu of the tests residing on the disk. Then, you are prompted to enter either the file number or the name of the test to be loaded.

For example, the following menu would be displayed for a hard disk:

```
MENU FOR VCATALOG DIRECTORY

06) ADCHK       00)CRCOMM     01)CRDRDR     07)CRUEXP
04) DS10PD      0A)EMUTST     0B)EROMBT     05)FIVMOD
02) FLPDSK      08)INPMOD     09)IO16       03)OUTMOD
```

If you have more files, the following message is displayed below the menu:

```
"RETURN" SHOWS MORE FILES
```

Each time you press the RETURN or NEW LINE key, more file names of tests residing on the disk are displayed. When you press RETURN or NEW LINE and the beginning of the menu reappears, the entire list has been displayed.

Next, you are prompted to enter the type of diagnostic that you want to load. The following prompt is displayed:

```
FILE # OR NAME?
```

where:

<#/name> indicates that you enter either the number or file name of the diagnostic test module. For this example, you could enter either IO16 or 9.

After you enter a response, you are prompted for the load point of the test module.

```
LOAD BIAS? (DEF= YYYY) - <XXXX>
```

where:

<xxxx> indicates that you enter a load bias that is greater than >YYYY. Otherwise, press the RETURN key to accept the default. The default is the location immediately following DOCS.

NOTE

The default load bias is dependent upon the type of interface(s) you are using for the interactive and error message device.

If you specify a load bias that precedes the end of DOCS, the following message appears:

```
WARNING - THIS OVERLAYS DOCS.
ARE YOU SURE? (DEF=0) -
```

If you enter 1 (YES), DOCS loads the test module. If you accept the default 0 (NO), you are prompted for the load point again. After you specify the load bias, DOCS loads the test module and displays the end location of the module.

Open Reel/Cartridge Tape Load.

If you use open reel or cartridge tape as your load medium, executing the .LD verb causes DOCS to display the current catalog that is being loaded. The menu listing the tests residing on that catalog is also displayed. Then, you are prompted for the number

or filename of the test that you want to load.

For example, the following menu would be displayed for tape:

```
MENU FOR VCATALOG DIRECTORY

06) ADCHK      00)CRCOMM     01)CRDRDR     07)CRUEXP
04) DS10PD     0A)EMUTST     0B)EROMBT     05)FIVMOD
02) FLPDSK     08)INPMOD     09)IO16       03)OUTMOD
```

If you have more files, the following message is displayed below the menu:

```
"RETURN" SHOWS MORE FILES
```

Each time you press the RETURN or NEW LINE key, more file names of the tests residing on the tape are displayed. When you press RETURN or NEW LINE and the beginning of the menu reappears, the entire list has been displayed.

Next, you are prompted to enter the type of diagnostic that you want to load. The following prompt is displayed:

```
FILE # OR NAME?
```

where:

<#/name> indicates that you enter either the number or file name of the diagnostic test module. For this example, you could enter either IO16 or 9.

After you enter a response, you are prompted for the load point of the test module.

```
LOAD BIAS? (DEF= YYYY) - <xxxx>
```

where:

<xxxx> indicates that you enter a load bias that is greater than >YYYY. Otherwise, press the RETURN key to accept the default. The default is the location immediately following DOCS.

If you specify a load bias that precedes the end of DOCS, the following message appears:

```
WARNING - THIS OVERLAYS DOCS.
ARE YOU SURE? (DEF=0) -
```

If you enter 1 (YES), DOCS loads the test module. If you accept the default 0 (NO), you are prompted for the load point again. After you specify the load bias, DOCS loads the test module and displays the end location of the module.

6.5.2.3 S200DOCS. If you are using S200DOCS, the .LD verb loads diagnostic test modules into memory and displays the current catalog being loaded and the menu of tests on that catalog.

```
MENU FOR S$UDIAG DIRECTORY

00) AU200      01) BCSEDT     02) BCSRUN     03) COM90X
04) DBVERB1    05) DSKCOM     06) DSKSA      07) DSKWDX
08) LP810      09) MXVERB1    0A) MXVERB2    0B) S200DOCS
0C) TST8X0     0D) TST940
```

If more files can be displayed, the following message appears below the menu:

```
TYPE RETURN FOR ADDITIONAL FILES
```

Each time you press the RETURN key, more of the tests residing on the diskette are displayed. If you press the RETURN key and the beginning of the menu reappears, all of the files residing on the diskette have been displayed.

Then, the following prompt is displayed:

```
TYPE: DIAG. # OR FILE NAME? - <#/name?>
```

where:

<#/name> indicates that you enter either the file name or number of the test that you want to load.

For example, if you want to load the 840 test, enter either C or TST8X0.

After you specify a test, DOCS prompts you for the load point for the test module:

```
LOAD BIAS? (DEF=006664) - <xxxx>
```

where:

<xxxx> is a value greater than the end of DOCS. If you wish to accept the default value press the RETURN key.

If you need to abort the .LD verb, press the CMD or @ key. If you specify a load bias that precedes the end of DOCS, the following message appears:

```
***WARNING*** THIS MAY OVERLAY DOCS.
DOCS MAY HAVE TO BE RELOADED TO UPPER MEMORY
ARE YOU SURE (DEF=0) -
```

If you enter 1 (YES), DOCS loads the test module. If you accept the default 0 (NO), the verb requests the load point again. After you specify the load bias, DOCS loads the test module and tells you the end location of the module.

6.5.3 Load Verb Module (.LV) Verb

The .LV verb allows DOCS to recognize a module of DOCS or user-written (external) verbs. The .LV verb format is similar to the .LD verb format except that external verbs can be loaded into mapped memory.

When you execute the .LV verb, the current loaded test is unlinked from DOCS. Any diagnostic test can be loaded (using the .LD verb) following the verb module. Then, you can access the DOCS verbs and test verbs in any operational mode. The verb module composition must adhere to the format defined in the following paragraphs.

NOTE

DOCS verb modules cannot execute properly in mapped memory, therefore, they should not be loaded into mapped memory.

The verbs in the verb module have the same format as DOCS verbs. If a verb module verb has a name identical to a DOCS verb, the DOCS verb is recognized first. Verbs are searched in the following order:

1. DOCS verbs
2. External verbs
3. Test verbs

External verbs are written in a specific format. The first generated code of the test module must be the VDPAVM table address. The format of this table is as follows:

```
            DATA  VDPAVM
            DATA  >FFFF
            DATA  >FFFF
VDPAVM
            TEXT  'LVVB'
            DATA  [verb table label]
            DATA  0,0
```

The first three statements must appear exactly as shown. DOCS must find the text LVVB to recognize the new verb table. The [verb table label] is a standard table that must be present within the same verb module. This table follows a specific format so that DOCS can search it for verbs. The format of this table is as follows:

```
[verb table label] DATA  2
                   DATA  4
                   TEXT  '8 CHAR NAME'
*
                   TEXT  -'.xx '
                   DATA  [verb label]
*
                   TEXT  -'.xx '
                   DATA  [verb label]
                     .
                     .
                     .
                   DATA  0,0
```

where:

xx	is a two-character verb mnemonic.
[verb label]	is a label representing the address of a routine within the module that is invoked for the preceding verb text.

DOCS uses the first two DATA words to search for the available verbs. The 8 CHAR NAME must be exactly eight bytes long (it can be blank-filled to the eighth character). This name is usually the name of the user-written verb module and is displayed when you execute the .PV verb.

The TEXT/DATA pairs represent the verb followed by the label of the routine that handles the verb. The verb is in a four-byte text field such that:

1. Byte 1 is a period (.).
2. Bytes 2 and 3 are the verb mnemonic.
3. Byte 4 is a blank.

This four-byte field must be preceded by a minus sign (-) which causes Byte 4 to be represented as the two's complement of a blank character.

DATA 0,0 is the table terminator.

6.5.4 Print Available Verbs (.PV) Verb

The .PV verb displays a list of the DOCS verbs and verbs for the test currently loaded. If no test is loaded, no test verbs are displayed. The following example issues the .PV verb and displays the DOCS verbs and test verbs for Dynamic DOCS. If you are using Standard DOCS or S200DOCS, a similar list is displayed.

```
VERB ? - .PV

DOCS'S  VERBS (Mini DOCS verbs)
.IP  .LD  .LV  .CN  .ML  .RW  .ES  .PV  .FT

MXVERB1 VERBS (Maxi DOCS verbs)
.MI  .CW  .CS  .RS

MXVERB2 VERBS (Maxi DOCS utility verbs)
.DM  .SM  .MM  .MV  .MS  .GO  .AD

BCSEDT  VERBS (verbs in BCSEDT DOCS)
.EL  .AT  .LL  .LG  .LB  .GT  .IF  .CF  .SF  .MF  .OI  .OC
.OD  .BL  .IL  .DL  .RL  .CL  .PL  .TN  .DO  .TS

TEST'S VERBS (example only)
IT  ET  LT  EA  LA  WT  RT  QT  SS  RS  AC  TR  LR  LC  UT
LQ  MZ  MO, etc.
```

6.5.5 Execute Stand-Alone (.ES) Verb

The .ES verb executes the stand-alone test (SAT) that is currently loaded in memory. The test must have been loaded using the .LD verb and completed at least one time. The following example issues the .ES verb and displays its prompt:

```
VERB ? - .ES

NUMBER OF LOOPS? - <x>
```

where:

<x> is the number of times that the SAT is to be executed.

NOTE

No SATs are currently available for the S200.

6.5.6 Memory Initialize (.MI) Verb

The .MI verb initializes memory with specific data. If mapping is available, the data can be stored in locations higher than >FFFF. Otherwise, the addresses wrap around. This verb does not initialize the TILINE peripheral control space (not available on the 990/4), but it does initialize the RAM that is shadowed by it.

```
VERB?-.MI
ADDRESS- (aaaaaa)
# OF WORDS (DEF=0000)- <nnnn>
DATA (DEF=0000) - <dddd>
```

where:

aaaaaa the address where the specified data is stored. This address can be up to a 21-bit, byte-level.

<nnnn> is the number of words of memory with the specified data.

<dddd> is the specified data.

6.5.7 Read/Write CRU Word (.CW) Verb

The .CW verb displays the data present at a specific CRU base address. Then, you are able to write from 1 to 16 bits of data to that CRU base address. (The legal bit count is 0 through >F, where 0 = 16 bits, 1 = 1 bit, 2 = 2 bits, and so on.) You can also loop on this write by entering 1 (YES) for the LOOP ON WRITE? prompt. Otherwise, enter a 0 (NO). To terminate the looping, press the @, CMD or HELP key. The following example issues the .CW verb and displays its prompts:

```
VERB ? - .CW

CRU BASE - (DEF=0000) = <xxxx>  CURRENT DATA = <yyyy>
# BITS - <b>  DATA - <dddd>
LOOP ON WRITE? -1
```

where:

<xxxx> is a CRU address.

<b> is a hexadecimal number from >1 to >F that indicates the number of bits of data to write.

<dddd> is the data that is to be written at the specified CRU base address.

<yyyy> is the data that currently exists at the specified CRU base address.

NOTE

The UST uses CRU base >1FE0 for communication through the Front Panel. If you are using the UST, manipulating the Front Panel CRU bits can lock up both the 990 and the UST.

6.5.8 Clear Interactive VDT Screen (.CS) Verb

The .CS verb clears the interactive VDT screen and moves the cursor to the lower left-hand corner. If you are using a 733 ASR/KSR, or 743 KSR as the interactive terminal, this verb causes a carriage-return line feed.

6.5.9 TILINE I/O Reset (.RS) Verb

The .RS verb issues a hardware reset (>360), clears the FAULT light, and reinitializes the DOCS interactive and error messages interfaces. In addition, the .RS verb issues a PBI reset for the S200.

6.5.10 Dump Memory (.DM) Verb

The .DM verb sends a memory dump (words of memory) to a selected error message terminal. If mapping is available, data can be displayed from above >FFFF. Otherwise, the address wraps around and the data comes from low memory. In the following example, the .DM verb is issued and its prompt is displayed:

```
VERB ? - .DM

ADDRESS (xxxxxx) - aaaaaa
```

where:

xxxxxx is the default address.

aaaaaa is a memory address where the dump begins. This address can be up to 21-bit, byte-level for Standard or Dynamic DOCS. However, this address can be up to 16-bit, byte-level for S200DOCS.

Eight lines of hexadecimal data and ASCII interpretation are displayed at the message terminal. The following example displays data as a result of issuing the .DM verb at location 0000:

```
PRESS - @ - TO STOP
ADDR    0    2    4    6    8    A    C    E

000000 0262 0282 0262 0292 0262 0296 0262 0344.............D
000010 0262 0344 1DAE 1DCE 0262 21F8 0262 0344..D.....!....D
000020 0262 0344 0262 0344 0262 0344 0262 0344..D...D...D...D
000030 0262 0344 0262 0344 0262 0344 0262 0344..D...D...D...D
000040 0262 0258 0262 0258 0262 0258 0262 0258..X...X...X...X
000050 0262 0258 0262 0258 0262 0258 0262 0258..X...X...X...X
000060 0262 0258 0262 0258 0262 0258 0262 0258..X...X...X...X
000070 0262 0258 0262 0258 0262 0258 0262 0258..X...X...X...X

TO CONTINUE ENTER (F=FORWARD, B=BACKWARD, DEFAULT=F) -
```

The TO CONTINE ENTER prompt allows you to scroll forward or backward through memory. If you enter F, eight more lines of data are displayed. If you enter B, you scroll backward. This continues until you press the @, CMD, or HELP key.

6.5.11 Show Memory (.SM) Verb

The .SM verb displays the memory allocation table of DOCS. This table lists all of the available memory in the chassis and the files that are dedicated to memory. If memory is available and no file is allocated to that memory address space, a blank line follows the address. The following example issues the .SM verb and displays the memory allocation table for Dynamic DOCS. If you are using S200DOCS or Standard DOCS, this table will contain similar information in this same format.

```
VERB ? - .SM

000000    TO    00009E
0000A0    TO    0044BE      DOCS
0044C0    TO    00476C      MXVERB1
00476E    TO    004E1E      MXVERB2
004E20    TO    007EA2      RAMTST
007EA4    TO    017FFE
```

6.5.12 Modify Memory (.MM) Verb

The .MM verb modifies memory. Press the @, CMD, or HELP key to terminate this verb. The following example modifies memory at the address specified:

```
VERB ? - .MM
ADDRESSES(xxxxxx) - aaaaaa
```

where:

xxxxxx is the default address.

aaaaaa is the address of the memory that you want to modify.

Then, the data at the specified address is displayed:

```
aaaaaa - dddd - yyyy
   .       .
   .       .
```

where:

aaaaaa is the address of the memory that you want to modify.

dddd is the data that you want to modify.

yyyy represents the data modification.

With Standard and Dynamic DOCS, you can also look at any TILINE controller by entering its TILINE address. For example, enter F800 to address a disk controller with the TILINE peripheral control space (TPCS) >F800. (This does not apply for the S200.)

6.5.13 Move (.MV) Verb

The .MV verb moves data from one address to another. The following example issues the .MV verb and displays its prompt:

```
VERB ? - .MV
FROM - xxxxxx  TO - yyyyyy  # - zzzzzz
```

where:

xxxxxx represents the address of the current location of the data.

yyyyyy represents the address of the location where you want to move the data.

zzzzzz represents the number of words to be moved.

CAUTION

When you use the .MM and .MV verbs, be sure not to write over DOCS or the diagnostic.

6.5.14 Memory Search (.MS) Verb

The .MS verb searches memory for one of the following comparative conditions:

* EQ -- Search until DATA equals a word in memory
* NE -- Search until DATA does not equal a word in memory
* GT -- Search until DATA is greater than a word in memory
* LT -- Search until DATA is less than a word in memory

In the following example, the .MS verb is issued and its prompts are displayed:

```
VERB ? - .MS

COND (EQ NE GT LT  DEF = EQ) -
ADDRESS - (aaaaaa)  # -nnnn  DATA-dddd
# OF WORDS (DEF = nnnn) - DATA (DEF = dddd) -
```

where:

aaaaaa indicates the address where the memory search begins.

nnnn is the number of words to search. If the comparative condition is found before the specified number of words is searched, the search terminates.

dddd is the data that is being searched for.

For example, to search memory from location >00F0 to >01FF (10F words) for a word that is arithmetically greater than >CCEC, your responses would be as follows:

```
VERB ? - .MS

COND (EQ NE GT LT  DEF = EQ) - GT
ADDR (00000) - F0
# OF WORDS (DEF = 0000) - 10F  DATA (DEF=0000) - CCEC
```

If the condition is found, DOCS displays the address of the word and the data. If the condition is not found, the following message appears:

```
CONDITION NOT FOUND
```

Note that since the most significant bit of all negative numbers is 1, the comparative value of a negative number becomes greater as its absolute value increases.

6.5.15 Go to User-Specified Location (.GO) Verb

The .GO verb branches to the address that you specify. A Branch and Link (BL) instruction is used to branch to the specified location. If the code at that location executes a RETURN (B *R11) instruction, control returns to DOCS. The .GO verb does not handle addresses above >F7FE.

```
VERB ? -.GO

ADDR -<xxxx>
```

where:

<xxxx> is the location to which you want to branch.

6.5.16 Add Hexadecimal Data (.AD) Verb

The .AD verb adds two hexadecimal numbers. In the following example, the .AD verb is issued and its prompt is displayed:

```
VERB ? - .AD

A(xxxxxxxx)-<iiii> OPER(+)-<s> B(zzzzzzzz)-<jjjj>=vvvvvvvv
```

where:

xxxxxxxx	is the default value for A. This value can be accepted or changed to any hexadecimal value.
zzzzzzzz	is the default value for B. This value can be accepted or changed to any hexadecimal value.
<s>	is the operand (+/-, the operation to be performed)
<iiii>	indicates a hexadecimal value.
<jjjj>	indicates a hexadecimal value that is to be added to or subtracted from the A value. (Subtraction uses the two's complement form.)
vvvvvvvv	represents the sum or difference of values A and B.

The OPER prompt indicates whether to add or subtract A and B. To add the two values, accept the default. If you want to subtract B from A, enter a minus sign (-) in response to this prompt in location <s>.

Section 7

Diagnostic Loading Procedures

7.1 INTRODUCTION

This section contains the procedures required to load either the Diagnostics Operational Control System (DOCS) or stand-alone tests (SATs) into the 990-based computer and start their execution. Tests are available on a variety of load media, which are discussed in the following paragraphs.

7.2 LOAD MEDIA AND PACKAGING

The diagnostic tests have been packaged according to load media (see Figure 7-1). Appendix A lists the part numbers for these packages. Both DOCS and the diagnostic test modules are currently available on the following types of media:

* Cassette and UST microfloppy diskette (unit record)
* Hard disk -- DS10, DS25, DS31, DS50, DS80, DS200, DS300, and CD1400
* Double-sided, double-density (DSDD) diskette
* One/half-inch open reel magnetic tape
* Eight-inch magnetic cartridge tape

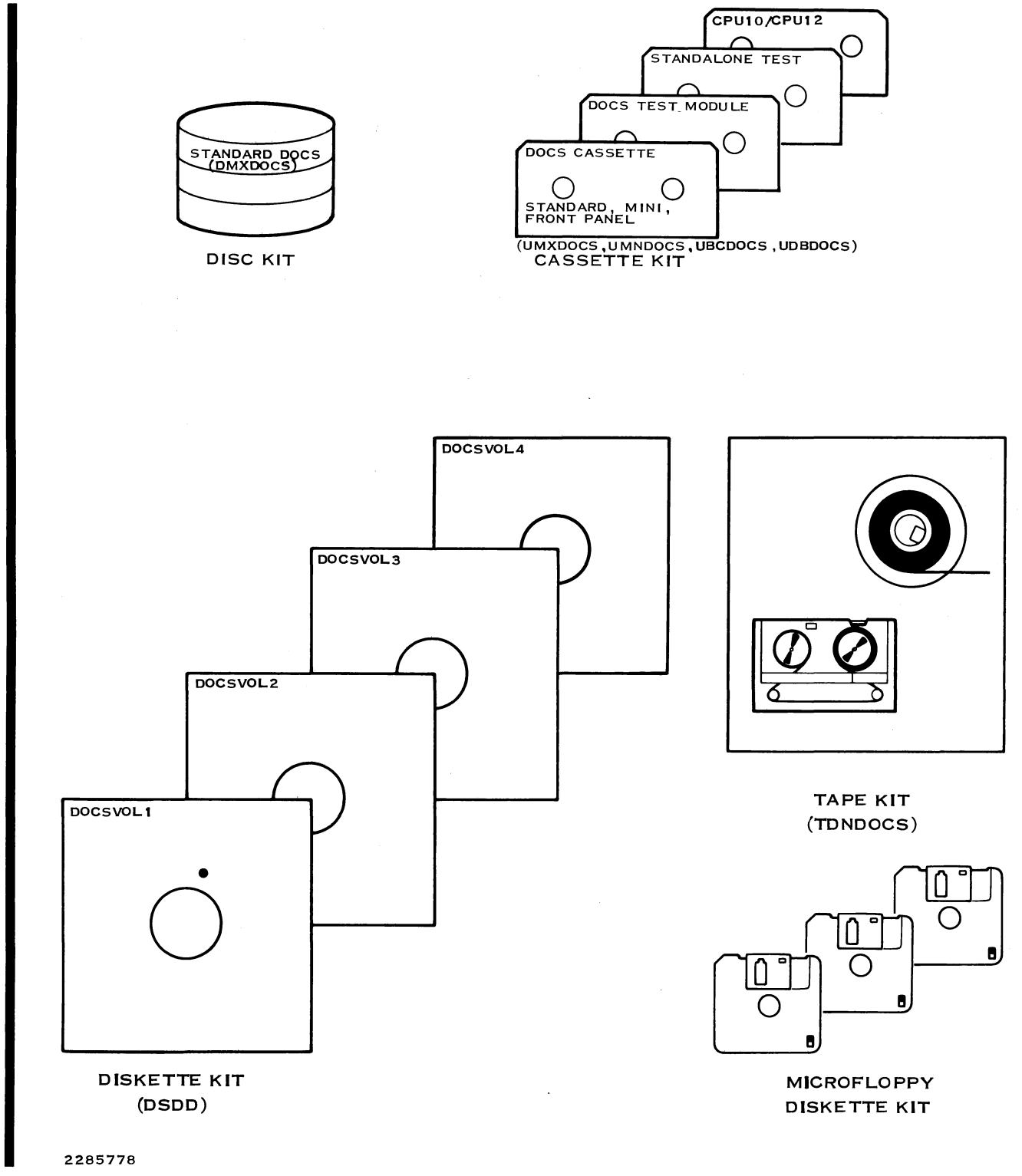


Figure 7-1 Diagnostic Media Packages

7.2.1 Cassette Package

Cassette (unit record) media are controlled by Standard DOCS. Each cassette in the package contains two tests: one test on side A and one test on side B. One cassette contains two types of Standard DOCS:

* Maxi DOCS (side A)
* Mini DOCS (side B)

7.2.2 Disk Package

All disk packages are DX10-file compatible. Each disk package contains all of the tests on a single disk. These tests are controlled by Dynamic DOCS. DOCS begins execution when you perform an initial program load (IPL). The procedures in paragraph 3.3.3 explain how to load a DOCS test module or SAT.

7.2.3 DSDD Diskette Package

The DSDD diskette package is controlled by Dynamic DOCS and contains four DSDD diskettes. These diskettes contain Dynamic DOCS and DOCS verb modules and the following groups of diagnostics:

* DOCSVOL1 contains diagnostics for:
  - AMPL
  - Memory
  - Processors

7.2.4 Open Reel/Cartridge Magnetic Tape Package

All tape media are controlled by Dynamic DOCS. The first file on the open reel and cartridge magnetic tapes is DOCS. The files following DOCS contain the verbs.

NOTE

Because of the difference in loaders, DOCS loaded from one media type cannot, in general, load test modules from another media type. For example, if you load DOCS from a disk or diskette, you cannot load test modules from cassette.

7.2.5 UST Microfloppy Package

UST diskettes contain Standard DOCS on a high-capacity microfloppy. Volume contents are as follows:

DISK # 1 Contains

Diagnostic Operational Control System (DOCS)

UBCDOCS	UDBDOCS	UMNDOCS	UMXDOCS

CPU Tests

AU04	CPU10	CPU12C	CRUTIC
AU04TST	CPU12A	CPU12D	MAPTIC
AU05	CPU12B	CPU12E	

Memory Tests

CTLTST	MAPTST	MEMTST	
MAP12	MEMPRT	RAM04	

Special Tests

CRDRDR	EMU940	ONPMOD	TFMTST
CRUEXP	EMUTST	I016	TNXTST
DACHK	EROMBT	OUTMOD	TRACE
EMU900	FIVMOD	PROMPG	WATDOG

DISK # 2 Contains

Printer Tests

CPTEST	LP810	LQPTST	TST8X0
FLPTST	LPTEST	PRTIF	

Terminal Tests

CRT911	TST733	TST940
CRT913	TST931	

Communications Board Tests

ACUTST	CRCOMM	FC3TST	RMTEIA
BCAIMT	ENETST	LLMTST	TTYEIA
COM90X	EXTACU	MUXTST	

DISK # 3 Contains

Disk Drive Tests

DDFLOP	DSKCOM	DSKWD5	WDFLPY
DS10PD	DSKM3X	DSKWD8	DS990R
DSKCD1	DSKSA	FLPDSK	
DSKCD2	DSKTRI	RMTFLP	

Tape Drive Tests

MTCTST1	MTCTST2	TAPTST

TILINE Bus Tests

CONVER	TILCOU	TPBITS
HSTSLV	TLCPLR	

7.3 LOADING DOCS FROM CASSETTE ON THE 733 ASR (UNIT LOADER)

DOCS test modules can be loaded from cassette on the 733 ASR using the following procedure. The 733 ASR is expected to be located off port 1 on a /5 CPU system or off an EIA card located at CRU address 0 on other systems.

1. Select the correct side of the cassette (A or B).
2. Place the cassette on either cassette unit of the 733 ASR. Place that unit in the playback mode using the PLAYBACK/RECORD switch. The cassette tape must be positioned at the load point by rewinding to the beginning of the tape and pressing the LOAD switch for the cassette unit.
3. Set the TAPE FORMAT switch to the line mode.
4. Press HALT/SIE on the programmer panel.
5. Press CLR and the bit 8 switch on the front panel to change the display value to >0080.
6. Press MA ENTER.
7. Press CLR to change the display to >0000.
8. If you are using the diskette loader ROMs, set the display to >0001. If you are using the universal loader ROMs, skip this step.
9. Press MDE.

    * If you are using the diskette loader ROMs, this step enters >0001 at location >0080.
    * If you are using the universal loader ROMs, this step enters >0000 at location >0080.

    If the test is to be loaded at the default load point address (>A0), go to step 11. Otherwise, proceed with the next step.

10. To change the load point, press CLR and enter >92 on the programmer panel. Then press MA and CLR, and enter the new load point on the programer panel. Press MDE.

11. Press RESET.

12. Press LOAD.

When loading DOCS the loader ROM will load the DOCS loader into 990 computer memory and then the tape will stop. The tape will automatically start again as the DOCS loader loads the rest of DOCS into memory.

7.4 LOADING DOCS FROM CASSETTE ON THE MDU (UNIT LOADER)

DOCS tests or SATs can be loaded from cassette using the 990 Maintenance Diagnostic Unit (MDU) as outlined in the following procedure. Appendix B describes the MDU in detail.

1. Select the correct side of the cassette (A or B).

2. Place the cassette in the MDU and press MDU RESET and then REWIND.

3. Press HALT/SIE on the MDU programmer panel.

4. Press RESET, (On the MDU, not the MDU programmer panel).

5. Press LOAD, (On the MDU, not the MDU programmer panel).

The loader ROM will load the DOCS loader into 990 Computer memory and then the tape will stop. Hit RESET on the MDU, then LOAD on the MDU. On a 990/5, you must do this twice. Then the DOCS loader will load the rest of DOCS into memory at address >A0.

NOTE

SATS (Stand Alone Tests) must be loaded by DOCS when using the cassette loader.

7.4A LOADING DOCS FROM MICROFLOPPY ON THE UST (UNIT LOADER)

The Universal Systems Terminal (UST) may be used to load DOCS tests as well as SATs by following the procedure outlined below. Further details may be found in Appendix K.

1. Start up the Front Panel Monitor program in LOADER mode.
2. Insert the DOCS microfloppy (DISK # 1) into the drive and press the space bar.
3. Use the Arrow keys on the UST keyboard to select the desired DOCS, and press the RETURN key.
4. The load process occurs, taking between 20 and 90 seconds, depending on the 990 CRU type and the chosen DOCS.
5. When loading is complete, the UST will ask if you wish to use it as an interactive device as well. If so, press the "Y" key, and DOCS will begin to execute. If you wish to use a different device as the interactive terminal, you must press the "N" key. Since the UST can perform both functions, it must be told that it will not be the interactive device. Pressing the "N" key on the UST, followed by a "Y" on an alternate 990 terminal, brings up a menu of the DOCS tests on the current disk of the UST.

NOTE

Stand Alone Tests (SATs) must be loaded by DOCS when you are using the UST as a loading device.

7.5 LOADING DOCS FROM DISK OR DSDD DISKETTE (DISK LOADER)

The following procedure explains how to load DOCS from DSDD diskettes and hard disk media (DS10, DS25, DS31, DS50, DS80, DS200, DS300, or CD1400).

1. Insert the disk into the system disk drive (location >F800, unit 0). Turn on the drive and enable the write-protect. If you are loading DSDD diskettes, insert one diskette (usually DOCSVOL1) in the diskette drive and write protect it.

NOTE

You can use the removable pack on the DS10 drive as the system disk if you connect a jumper plug at the rear of the drive.

2. Perform a DOCS load using the standard DX10 loading procedures:

   * If the computer has a disk initial program load (IPL) ROM, press HALT and LOAD. If a programmer panel is not available, cycle the system power.

   * If a disk IPL ROM is not available, load from the disk using a DX10 IPL card deck or cassette.

   * If a DX10 3.X disk IPL ROM is installed, an IPL can be performed from a nonstandard location. If the following disk IPL ROM locations contain the indicated data, it is a DX10 3.X disk IPL ROM.

Location	Data	Meaning
>80	>FFFF	TILINE device
>82	>F800	TILINE address
>84	>0800	Unit select = 0

If the disk is nonstandard, modify locations >82 and >84 to reflect the proper locations. A 1 in bits 4, 5, 6, and 7 at location >84 indicates disk units 0, 1, 2, and 3, respectively.

3. DOCS waits for a Y response to be entered on the interactive terminal. Once you enter a "Y", refer to the paragraph on Dynamic DOCS 4.5 for details on DOCS operating procedures.

If you are loading from a DSDD diskette (usually DOCSVOL1) and need to access one of the other diskettes, remove the diskette and insert the next one. It is not necessary to repeat the load procedure.

7.6 LOADING DX10 WHEN CORESIDENT WITH DOCS

The DX10 operating system software, DOCS, and tests may be packaged on the same disk. In this case, the system file will be set so that Disk Dynamic DOCS (DDNDOCS) will load initially (you must press Y). This allows the tests to verify system operation. After loading DOCS and running the required tests, you may perform an IPL of DX10 by following this procedure:

1. Load DOCS as described in paragraph 7.6.

2. Initiate a test module load using the .LD verb. Since DX10 is on the same disk, the DOCS menu will show S$LOADER (the DX10 system loader). Note the menu number for this module.

3. Perform a DX10 IPL by entering S$LOADER or the menu number for the module. The disk must be write-protected during the IPL of DX10.

4. The load bias for S$LOADER must be >A0 for a correct IPL. Ignore any DOCS overlay warning message.

5. When the IPL is performed and the IDLE light is lit, disable the write-protect. This must be done before the system can be initialized.

7.7 MODIFYING THE IPL PROCESS

Use the Modify Volume Information (MVI) DX10 command to make changes for the IPL procedure (see paragraph 8.2.3). This command is used to identify S$LOADER as the primary loader on the disk (in order to load the DX10 system by performing an IPL) when tests are no longer required. The MVI command can also be used to identify DDNDOCS as the secondary system, at a later time, when tests are needed.

7.8 LOADING DOCS FROM TAPE (TDNDOCS)

You can load DOCS test modules or SATs from tape by using Tape Dynamic DOCS (TDNDOCS). The following paragraphs discuss the load procedures, initialization, and error messages associated with TDNDOCS.

7.8.1 TDNDOCS Loading Procedure

To load TDNDOCS using the programmer panel, perform the following procedure.

1. Mount the tape on the loading device. Be sure to write-protect the tape. If you are using an S300/S300A computer, proceed with Step 2. Otherwise, go on to Step 3.

2. To load TDNDOCS on the S300/S300A computer, cycle the power on the main EVT (that is, if the computer is already on, turn it off and on again). The TDNDOCS load process is complete. Skip the remaining steps.

3. Press HALT.

4. Press RESET.

5. Enter the hexadecimal value of the TILINE address at location >82 on the programmer panel and press MA.

6. If you are using the standard values for a tape device, enter the TILINE address >F880 on the programmer panel. If you are using a nonstandard TILINE address, enter the hexadecimal value of the address in location >82.

7. Press MDE.

8. Enter the hexadecimal location of >84 on the programmer panel and press MA.

9. Enter the unit select >8000 to specify unit 0 and press MDE. The nonstandard unit selects are as follows:

   * >4000 specifies unit 1
   * >2000 specifies unit 2
   * >1000 specifies unit 3

10. Press LOAD.

11. Enter the character Y on an available terminal.

12. Go to Section 7.8.2 for Initialization.

For further information about the tape loading procedures, refer to the ROM Loader User's Guide, part number 2270534-9701.

If you are using the operator panel to load the tape, check all of the units on the controller at TILINE address >F880. If any unit has a device number lower than that of the loading device, place the unit in a not ready state. To load from the tape, turn the key switch to the load position.

7.8.2 Initialization

After you load TDNDOCS, initialization prompts appear. A subset of the prompts that appear is similar to Disk Dynamic DOCS. The following example displays the initialization prompts (refer to paragraph 3.4.2.1 for instructions on how to answer these prompts):

```
      ****DYNAMIC DOCS   VERSION *T    096/83   TI 990/10****

          USE "RETURN" OR "NEW LINE" TO TERMINATE INPUT
          USE "<---" (LEFT ARROW) OR "RUB OUT" TO ERASE INPUT

ENTER DOCS TYPE <0=MINI 1=MAXI 2=BCS RUN 3=BCS EDIT 4=DEBUG>?
DEF= 1 -

CRT 911 INT LEVEL ? DEF = 0A    -

ERR MSG DEVICE (0=ASR/KSR 1=913 2=911 3=810/820 4=940 5=840)?
DEF=2 -

ERR MSG INTERFACE(0=940 AUX, 1=OTHER)? DEF=0

ENTER CONTROL OPTIONS SEPARATED BY COMMAS:
<E = ERR MSG'S, H = HDR MSG'S, N = ERR #'S, P = PAUSE ON ERR'S)
DEFAULT = E,H,N -
```

You are finished loading and initializing TDNDOCS and can load any additional diagnostic test. The following prompt appears:

```
     LOAD TEST? (DEF=1) -
```

If you want to load a test, press RETURN to accept the default value 1 (YES). Otherwise enter the value 0 (NO), and the following prompt appears:

```
     VERB? -
```

In response to this prompt, you can select any Standard DOCS verb or the .ML verb. (Refer to paragraph 7.8.4 for information about the .ML verb).

If you entered 1 to load a test, the TDNDOCS loader is invoked. The TDNDOCS loader is also invoked if you issue the .LD verb in response to the VERB? prompt. When the loader is invoked, follow the procedure for loading additional tests discussed in the next paragraph.

7.8.3 Loading Procedure for Additional Tests

After the TDNDOCS loader is invoked, the following prompt appears:

READY TAPE AT UNIT <x> AND HIT RETURN -

where:

<x> is a digit representing the unit number.

This prompt continues to appear until the tape unit is ready or the load process is aborted by pressing the @ or CMD key.

If the write ring is installed on a magnetic tape or the cartridge tape unit is not write protected, the following message appears:

WRITE PROTECT THE TAPE AND HIT RETURN -

When the unit is ready to load, press RETURN. Then, TDNDOCS searches for a catalog and the following message is displayed:

SEARCHING FOR CATALOG

TDNDOCS loads the catalog and displays its contents in alphabetical order. Each file name contains a corresponding file number. If the output device is a hard-copy terminal, the following message and all of the file names are displayed:

TYPE: @ TO INTERRUPT DISPLAY

If the output device is a video display terminal (VDT), TDNDOCS only displays 16 file names and numbers at a time. The names of the files and their corresponding numbers vary, depending on the tape build process that you are using. The following example displays 16 file names and numbers:

20) ADCHK	01) AUO5	0F) CPU10	02) CPU12P7
24) CPU12P8	05) CPU12P9	0B) CPU12P10	11) CPU12P11
15) COMM05	09) CRCOMM	07) CRDRDR	09) CRT911
0A) CRT913	0D) CRUEXP	03) DACHK	25) DDFLOP

Each set of file names is followed by a message and prompt:

"RETURN" SHOWS MORE FILES

FILE # OR NAME?

Press RETURN to display another set of 16 file names, or enter a file name (or number) to select a file. If you enter an invalid file name or number, the FILE # OR NAME? prompt appears again.

If you press RETURN and the first set of file names appears again, all of the file names have been displayed.

Once you select a file, the following prompt appears:

```
LOAD BIAS?  (DEF= xxxx) -
```

where:
xxxx is the default value.

The first address available beyond DOCS in the memory map becomes the default address. You can select the default address or enter another load bias. However, if you choose a load bias lower than the default address, the following message appears:

```
WARNING - THIS OVERLAYS DOCS.
ARE YOU SURE?   DEF=0
```

You can either select another load bias or choose to overlay DOCS. If you overlay DOCS with a test that is not a SAT, the load procedure fails. If you overlay DOCS with a SAT, the test runs properly. However, upon completion, you need to perform an IPL of DOCS.

TDNDOCS displays the following message:

```
SEARCHING FOR FILE
```

When the file is located, FOUND appears next to the message:

```
SEARCHING FOR FILE-FOUND
```

The actual load process now begins. The light-emitting diodes (LEDs) on the programmer panel display the initial memory address of the test file that is being loaded.

If the test file is a SAT, the controller gives control to the SAT and the SAT independently executes the test one time. After execution, the SAT returns control to TDNDOCS and the following prompt is displayed:

```
VERB? -
```

If the test file is not a SAT, the controller gives control to TDNDOCS, which displays the title of the test and executes the test's IT verb. The following prompt appears next:

```
EXECUTE EA VERB ?  (DEF=1) -
```

The prompts that appear during the execution of a test loaded by TDNDOCS are identical to those of a test loaded by another media.

Aside from differences in error messages (which are described in paragraph 7.8.5), the performance of TDNDOCS is identical to that of DDNDOCS.

7.8.4 Modify Load Parameters (.ML) Verb

All loads take place on the load device that you specified in the IPL procedure. If you want to modify any load parameters, you can use the DOCS .ML verb. This verb allows you to specify an alternate TILINE address, unit select address, or retry limit for further tape loads. A retry limit specifies the number of times you can attempt a procedure that is receiving error messages against it without causing the procedure to abort. The .ML verb prompts are:

```
ENTER TILINE ADDRESS. DEF = xxxx -
ENTER LOAD UNIT (0,1,2,3). DEF = z -
ENTER MAXIMUM # OF RETRIES. DEF = 5 -
```

where:

xxxx is the TILINE address.
z is the load unit number.

After you enter your responses, TDNDOCS issues the following prompt:

```
VERB? -
```

Enter a standard DOCS verb. Any subsequent tape loads will use the modified parameters to load from the specified tape device. Reexecute the .ML verb if you need to change the parameters again.

7.8.5 Error Messages

TDNDOCS checks for errors while searching and loading files from the tape. Some errors result in immediate termination of the load process; other errors allow you to use a procedure again until you exceed the retry limit. All fatal errors are preceded by an associated error number that appears on the interactive device and front panel.

7.8.5.1 Nonfatal Error Summary Messages. When TDNDOCS detects any errors during the load process that forces it to reread data, it keeps a summary of all retry attempts. Upon completion of the load, TDNDOCS displays the error summary messages at the I/O device. This information may be useful in determining the validity of the media, since a retry attempt does not necessarily mean that the load failed.

NOTE

Only a numbered fatal error message denotes that a failure occurred and that the load process was aborted.

A summary indicates the total number of retry attempts made during the entire load process. If no attempts were made, no summary appears.

The following example illustrates a retry summary. The corresponding fatal error message numbers are shown in parentheses. Actual summaries can contain any or all of the following messages:

SUMMARY OF RETRYS DURING LOAD:

VRC/RAW ERROR - # RETRYS = <x>
Either the vertical redundancy check failed or the read after write failed. (10A)

LRC ERROR - # RETRYS = <x>
The longitudinal redundancy check failed. (10B)

CRC/PE/DATA ERROR - # RETRYS = <x>
Either the cyclical redundancy check failed or a parity or data error occurred. (10C)

MEMORY PARITY ERROR - # RETRYS = <x>
A parity error occurred in one or more of the bits in memory. (10D)

TAPE TIMING ERROR - # RETRYS = <x>
A tape timing error occurred. (10E)

where:

<x> is the number of retry attempts. The default limit for one read is five.

You can modify the retry attempt limit by using the .ML verb. The number of retry attempts listed for a particular type of error can exceed the limit for attempts made during one read since the summary counts the number of retry attempts made during the entire load process. Exceeding the retry attempt limit for any particular error during one read results in a fatal error message.

7.8.5.2 Fatal Error Messages. If any of the following errors occurs, the loading procedure terminates.

```
***ERROR 101
CONTROLLER ABNORMAL COMPLETION - LOAD ABORTED
     Probable cause:
       * An I/O reset
       * A power failure

***ERROR 102
UNIT OFFLINE - LOAD ABORTED
     Probable cause:
       * An OFFLINE status
       * A NOT READY status

***ERROR 103
TAPE REWINDING - LOAD ABORTED
     Probable cause:
       * Load was attempted while tape was rewinding

***ERROR 104
TAPE CONTROLLER TIME OUT - LOAD ABORTED
     Explanation:
       * Tape controller is not responding within
         allowable time limits

***ERROR 105
UNEXPECTED BOT - LOAD ABORTED
     Explanation:
       * Unexpected beginning of tape mark detected

***ERROR 106
UNEXPECTED EOT - LOAD ABORTED
     Explanation:
       * Unexpected end of tape mark detected

***ERROR 107
TILINE TIMEOUT DURING TAPE I/O - LOAD ABORTED
     Explanation:
       * Tape drive not responding
       * Interface malfunctioning

***ERROR 108
TAPE FORMAT ERROR - LOAD ABORTED
     Explanation:
       * Tape drive does not recognize format of the
         mounted tape
```

```
***ERROR 10A
VRC/RAW ERROR - LOAD ABORTED
DATA READ IS FAULTY AND RETRY ATTEMPTS HAVE REACHED LIMIT
     Probable cause:
       * Vertical redundancy check failed
       * Read after write failed
     (A summary of the retry attempts follows the
     error message.)

***ERROR 10B
LRC ERROR - LOAD ABORTED
DATA READ IS FAULTY AND RETRY ATTEMPTS HAVE REACHED LIMIT
     Probable cause:
       * Longitudinal reduncancy check failed
     (A summary of the retry attempts follows the
     error message.)

***ERROR 10C
CRC/PE/DATA ERROR - LOAD ABORTED
DATA READ IS FAULTY AND RETRY ATTEMPTS HAVE REACHED LIMIT
     Probable cause:
       * Cyclical redundancy check failed
       * Parity error occurred
     (A summary of the retry attempts follows the
     error message.)

***ERROR 10D
MEMORY PARITY ERROR
DATA READ IS FAULTY AND RETRY ATTEMPTS HAVE REACHED LIMIT
     Explanation:
       * Parity error occurred in one or more of
         the words in memory
     (A summary of the retry attempts follows the
     error message.)

***ERROR 10E
TAPE TIMING ERROR
DATA READ IS FAULTY AND RETRY ATTEMPTS HAVE REACHED LIMIT
     Probable cause:
       * Unknown
     (A summary of the retry attempts follows the
     error message.)

***ERROR 10F
UNEXPECTED EOR - LOAD ABORTED
     Probable cause:
       * Controller detected an unexpected end of
         record
       * Catalog possibly bad
```

***ERROR 110
UNEXPECTED EOF - LOAD ABORTED
Probable cause:
* Controller detected an unexpected end of file
* Catalog possibly bad

***ERROR 111
MISSING COLON RECORD IN OBJECT FILE - LOAD ABORTED
ERROR OCCURRED DURING PROCESSING OF DATA PREVIOUSLY READ FROM TAPE.
Probable cause:
* File not written to the tape correctly

***ERROR 112
EOF EXPECTED BUT NOT FOUND - LOAD ABORTED
ERROR OCCURRED DURING PROCESSING OF DATA PREVIOUSLY READ FROM TAPE.
Probable cause:
* File not written to the tape correctly

***ERROR 113
CHECKSUM ERROR - LOAD ABORTED
ERROR OCCURRED DURING PROCESSING OF DATA PREVIOUSLY READ FROM TAPE.
Probable cause:
* File not written to the tape correctly

*** ERROR 114
NEED MORE MEMORY FOR LOAD - LOAD ABORTED
Probable cause:
* Data being read would load into memory in such a way that it would overlay the TDNDOCS loader
* Data being read would load into memory in such a way that it would overlay the TILINE peripheral control space (TPCS)

***ERROR 115
END OF TAPE FILES DETECTED - LOAD ABORTED
Probable cause:
* TDNDOCS has completed its search of the tape and has found two consecutive end of file marks without encountering any file numbers

7.8.5.3 Wrong File Error Messages. When seeking a specified file, TDNDOCS skips a calculated number of end-of-file (EOF) marks according to the number of files preceding the specified file. However, some tape errors can either mask EOF marks or indicate extra EOF marks on the tape. Occurrence of these errors could cause problems in loading a specified catalog or test file.

Each catalog is built with an entry that marks it as a catalog and notes the number of the file with which it is associated. If TDNDOCS cannot find the catalog identifier when searching for a specified catalog, it skips EOF marks until the next catalog is found.

Using a defective tape could lead to a special problem. While loading a specified file, TDNDOCS might detect a catalog where it expects a file.

```
ATTEMPT TO LOAD MENU FOR FILE # <xx>
ENTER 0 TO LOAD PRIOR FILE, A 1 FOR NEXT FILE -
```

where:

<xx> represents a file number.

If you enter 0, the previous file is loaded. If you enter 1, the following file is loaded. For simple errors, this procedure allows you to load the file that you want. However, when this message appears, the tape is defective and you should replace it.

If a tape is so defective that neither the catalog nor the specified file can be loaded, the following message appears:

```
***ERROR 116
CAN'T FIND FILE - LOAD ABORTED
```

Section 8

DOCS Building Procedures

8.1 INTRODUCTION

This section explains how to build DOCS on a hard disk, double-sided, double-density (DSDD) diskette, and how to build S200DOCS on a DSDD diskette. Before you attempt to build a DOCS disk, or S200DOCS diskette, refer to Appendix A for part numbers and descriptions of DOCS and test modules. You should also be familiar with the DX10, or DX Micro (DXM) operating systems.

For release 5.0.0 and later single-sided, single-sensity (SSSD) diskettes are no longer supported by DOCS as load media. For release 5.2.0 and later, the UST microfloppy diskette is supported by DOCS, but since it is an MS(R)-DOS formatted diskette, it cannot be built using the following procedures.

8.2 DOCS DISK BUILDING PROCEDURE

The DOCS disk is a DX10-compatible, initialized disk with DOCS installed and is available on the following types of media: DSDD diskette, DS10, DS25, DS31, DS50, DS80, DS200, DS300, DS300A, CD1400, WD300, WD500, and WD800. All diagnostics (including Disk Dynamic DOCS) that are to be displayed in the DOCS menu are installed as files at the second level of the disk directory. In other words, the diagnostic files must have a pathname with the following format:

<volume name>.S$UDIAG.<diagnostic file name>

In order for a diagnostic file to be displayed in the menu, it must be installed on the disk as a relative record file with a nonexpandable logical record length of 72 records.

The DX10 operating system, Release 3.2.1 or later, is required to build a DOCS disk. However, any DX10 Release 3.X can add, delete, or replace diagnostic files once the initial DOCS disk has been created. Use the Disk Copy/Restore (DCOPY) command to copy an existing DOCS disk to another disk of the same type.

MS is a registered trademark of the Microsoft Corporation.

To create a DOCS disk while running under DX10, perform the following steps:

1. Initialize a disk using the Initialize New Volume (INV) command, specifying the disk as a system disk. (If you are using a new disk, issue the Initialize Disk Surface (IDS) command.)

2. Create the subdirectory .S$UDIAG using the Create Directory File (CFDIR) command.

3. Create relative record files for DOCS and all tests to be installed on the disk by issuing the Create Relative Record File (CFREL) command.

4. Copy the diagnostics to these files using the Copy/Concatenate (CC) command.

5. Assign the DOCS file as the disk loader using the Modify Volume Information (MVI) command.

The last step brings DOCS into memory and begins execution when the disk is loaded.

In order to conserve disk space and reduce loading times, factory-created DOCS disks are produced in a compressed ASCII object format. ASCII object files are usable but less efficient. Creating compressed object files is explained at the end of this section.

8.2.1 Initializing the Disk

Initialize a disk by issuing the INV or IDS command. The initialization process formats the disk properly and places a primitive loader on track 1. Enter the following responses for the INV command:

```
[] INV
INITIALIZE NEW VOLUME
                    UNIT NAME: DSxx
                  VOLUME NAME: <volume name>
  NUMBER OF VCATALOG ENTRIES: 150
        BAD TRACK ACCESS NAME:
DEFAULT PHYSICAL RECORD SIZE: 768
HARDWARE INTERLEAVING FACTOR: 1
     FORCE CLEARING OF DISK?: NO
      USED AS A SYSTEM DISK?: YES
         LISTING ACCESS NAME:

          LOADER ACCESS NAME:
```

where:

DSxx indicates the disk drive you are using.

<volume name> is the name of the disk.

The initialized disk has the SYSLD primitive loader installed on track 1, sector 0.

If a previously built DOCS disk is available, copy that disk to another disk of the same type by issuing the DCOPY command. This performs the initialization procedure.

8.2.2 Creating the Subdirectory

Create the DOCS subdirectory using the CFDIR command on DX10 with the following responses:

```
[] CFDIR
CREATE DIRECTORY FILE
                    PATHNAME: <volume name>.S$UDIAG
                 MAX ENTRIES: 101
DEFAULT PHYSICAL RECORD SIZE:
```

where:

<volume name> is the name of the disk you are using.

The .S$UDIAG directory will contain all of the DOCS files except the initial DOCS load file.

NOTE

When creating the relative record file during the installation of the fully linked object (FLO) module on a DOCS disk, place the initial DOCS file (DDNDOCS and all associated DOCS verb modules) under the first- and second-level directories. The first-level directory is at the same level as S$UDIAG, under the directory <volume name>. The second-level directory is under the directory S$UDIAG.

8.2.3 Installing a FLO Module on a DOCS Disk

Follow this procedure to install a FLO module or compressed object module (that is, DDNDOCS) on the DOCS disk:

1. Create each relative record file for the module by issuing the DX10 CFREL command with the following responses:

```
[] CFREL
CREATE RELATIVE RECORD FILE
                    PATHNAME: <volume name>.S$UDIAG.<filename>
 LOGICAL RECORD LENGTH: 72
PHYSICAL RECORD LENGTH: 256
     INITIAL ALLOCATION: <xx>
  SECONDARY ALLOCATION: 0
              EXPANDABLE?: NO
           FORCED WRITE?: NO
```

where:

<volume name> is the name of the disk you are using.

<filename> is the IDT name of the module.

<xx> is the number of records in the module.

Next, issue the Map Disk (MD) command to check the attributes of the file created.

```
[] MD
MAP DISK
                  PATHNAME: <volume name>.S$UDIAG.<filename>
  LISTING ACCESS NAME:
            SHORT FORM?: NO
       TOP LEVEL ONLY?: NO
DIRECTORY NODES ONLY?: NO
```

where:

<volume name> is the name of the disk you are using.

<filename> is the IDT name of the module.

The file attributes should resemble the following example:

```
ALLOC: PRI=118 SEC=0 #SECS=0 LRECL=72 PRECL=256 ADU/BLK=1
WPT/DPT/BLK/PERM/FORCED/=NNYYN  DATA-FMT=BINARY
```

2. Copy the object or compressed object diagnostic file to the relative record file by issuing the CC command with the following responses:

```
[] CC
COPY/CONCATENATE
 INPUT ACCESS NAME(S): <input name>
   OUTPUT ACCESS NAME: <output name>
              REPLACE?: YES
MAXIMUM RECORD LENGTH:
```

where:

<input name> is the pathname of the diagnostic file.

<output name> is the pathname of the relative record file.

Verify that the diagnostic file is a relative record file by issuing the MD command.

3. Assign the system file to the module using the MVI command. (You can use the system file that usually brings in the DX10 operating system to bring in DOCS.) Issue the MVI command with the following responses:

```
[] MVI
MODIFY VOLUME INFORMATION
CONTROL ACCESS NAME: ME
MVI
DISK? DS01
COMMAND (L,C,Q)? L
```

The following list is displayed:

```
     PRIMARY     SECONDARY SELECT
 SYSTEM IMAGE: DD33A       DD33B      P      P, S, T
 PROGRAM FILE: S$PROGA                P
 OVERLAY FILE: S$OVLYA                P
  LOADER FILE: S$LOADER    DDNDOCS    S      P, S, T
    WCS FILE:                         P
  DIAGNOSTIC: DDNDOCS      --------   N      N, Y
 VOLUME NAME: SYSTEM32
COMMAND (L,C,Q)? Q
```

Change the secondary loader file on the DOCS disk by issuing the MVI command again. Enter the following responses:

```
[] MVI
MODIFY VOLUME INFORMATION
CONTROL ACCESS NAME: ME
MVI
DISK? DSxx
COMMAND (L,C,Q)? C
WHICH ITEM (S,O,P,L,D,V)? L
PRIMARY:
SECONDARY: <IDTname>
SELECT: <T/Y>
COMMAND (L,C,Q)? Q
MVI TERMINATED
```

where:

DSxx	is the device or volume name where the DOCS is located. For example, DS02.
<IDTname>	is the IDT name of the DOCS relative record file.
<T/Y>	indicates that you enter T (TEST) to specify the secondary loader file (DDNDOCS). Otherwise, enter Y (YES) to specify the diagnostics file (DDNDOCS).

Whether you enter T or Y in response to the SELECT prompt, you are allowed a one-time load of DDNDOCS on a HALT-RESET-LOAD sequence while the disk is write protected. In either case, DDNDOCS must reside on the disk.

Once DOCS is initialized and running, you can load and execute one or more test modules.

To restore DX10, repeat the HALT-RESET-LOAD sequence, or specify the primary system loader S$LOADER (at load bias >A0) as the DOCS load file. DOCS can be loaded repetitively by selecting DDNDOCS as the secondary system file. This can be accomplished by entering S in response to the SELECT prompt of the MVI command.

A test module, another DOCS FLO module, or a compressed object module can be installed on the DOCS disk using DX10 by following this procedure:

1. If necessary, create a relative record file for the module using the CFREL command.
2. Copy the object or compressed object file module using the CC command.
3. Verify the proper file format using the MD command.

NOTE

The track 1 loader of the disk can only load files that are in the top directory. This means that the IPL DOCS (DDNDOCS and all associated DOCS verb modules) must be under the VCATALOG.

8.3 S200DOCS BUILDING PROCEDURE

The following paragraphs explain how to build S200DOCS on a DSDD diskette. Before attempting to build an S200DOCS diskette, refer to Appendix A for part numbers and descriptions of DOCS test modules. Familiarity with the DXM operating system is also helpful.

The S200DOCS diskette is a completely compatible, DXM-type, initialized diskette with S200DOCS installed. All diagnostics to be displayed in the S200DOCS menu are installed as files at the second level of the disk directory. In other words, all diagnostic files must have a pathname with the following format:

<volume name>.S$UDIAG.<diagnostic file name>

For a diagnostic file to be displayed in the menu, it must be installed on the diskette as a relative record file with a nonexpandable logical record length of 72 records.

You need DXM Release 1.0.0 or later to build an S200DOCS diskette. However, any DX10 Release 3.X or later can add, delete, or replace diagnostic files after you create the initial S200DOCS diskette. On DX10, you can copy an S200DOCS diskette to another diskette (of the same type) using the Disk Copy/Restore (DCOPY) System Command Interpreter (SCI) command.

Perform the following steps to create an S200DOCS diskette under DXM:

1. Initialize a scratch disk using the Initialize New Volume (INV) SCI command specifying the disk as a system disk. (If the disk is new, use the Initialize Disk Surface IDS command.)

2. Create a sequential file using the Create Sequential File (CFSEQ) command.

3. Copy S200DOCS to the sequential file.

4. Create the subdirectory S$UDIAG using the Create Directory File (CFDIR) command.

5. Create relative record files for all tests to be installed on the disk using the Create Relative Record File (CFREL) command.

6. Copy the diagnostics to these files using the Copy/Concatenate (CC) command.

7. Assign the S200DOCS file as the system loader file using the Modify Volume Information (MVI) command.

In order to conserve disk space and reduce loading times, factory-created S200DOCS diskettes are produced in a compressed ASCII object format. Although you can use standard ASCII object files, they are less efficient. Paragraph 8.4 explains how to create compressed object files. The paragraphs that follow describe in detail each of the steps needed to create an S200DOCS diskette.

8.3.1 Initializing the Disk

Initialize a scratch disk using the INV command or the IDS command. The initialization process formats the disk properly and places a primitive loader on track 1. You need DXM Release 1.0.0 or later to perform this step. The format for the INV command is as follows:

```
[]INV
INITIALIZE NEW VOLUME
                    UNIT NAME: <drive number of unit>
                  VOLUME NAME: <name given to volume>
  NUMBER OF VCATALOG ENTRIES: 101
       BAD TRACK ACCESS NAME: <RETURN>
DEFAULT PHYSICAL RECORD SIZE: 288
HARDWARE INTERLEAVING FACTOR: 6
USED AS A SYSTEM DISK?: YES

          LOADER ACCESS NAME: <RETURN>
```

Be sure to specify the disk as a system disk. The responses above are correct for a DSDD diskette. If you are using a WD500, specify 256 for DEFAULT PHYSICAL RECORD SIZE and 16 for HARDWARE INTERLEAVING FACTOR.

The initialized disk has the SYSLD primitive loader installed on track 1, sector 0. You cannot use DX10 to create an S200DOCS diskette because its primitive loader is different. If a previously built S200DOCS diskette is available, you can initialize it under DX10 by executing a DCOPY command from the initialized disk to the scratch disk.

8.3.2 Creating a Sequential File

Create the S200DOCS sequential file by executing the CFSEQ command:

```
[ ] CFSEQ
CREATE SEQUENTIAL FILE
              PATHNAME: <volume name>.S200DOCS
 LOGICAL RECORD LENGTH: 72
PHYSICAL RECORD LENGTH: 288 (288-DSDD FLOPPY, 256-WD500)
    INITIAL ALLOCATION: <number of records in module>
  SECONDARY ALLOCATION: 0
           EXPANDABLE?: NO
       BLANK SUPPRESS?: YES
         FORCED WRITE?: NO
```

8.3.3 Copying S200DOCS to the Disk

Copy S200DOCS to the sequential file using the CC command:

```
[] CC
COPY/CONCATENATE
 INPUT ACCESS NAME(S): <pathname of S200DOCS>
   OUTPUT ACCESS NAME: <volume name.S200DOCS>
             REPLACE?: YES
MAXIMUM RECORD LENGTH: <RETURN>
```

8.3.4 Creating the Subdirectory

Create the DOCS subdirectory, under which all S200DOCS files will be kept; to do so, use the CFDIR command:

```
[] CFDIR
CREATE DIRECTORY FILE
                  PATHNAME: <volume name>.S$UDIAG
                MAX ENTRIES: 101
DEFAULT PHYSICAL RECORD SIZE: <RETURN>
```

8.3.5 Creating a Relative Record File

Create each relative record file by executing the CFREL command:

```
[] CFREL
CREATE RELATIVE RECORD FILE
               PATHNAME: <volume name>.S$UDIAG.<IDT name of module>
 LOGICAL RECORD LENGTH: 72
PHYSICAL RECORD LENGTH: 288 (288-DSDD; 256-WD500)
    INITIAL ALLOCATION: <number of records in module>
  SECONDARY ALLOCATION: 0
           EXPANDABLE?: NO
         FORCED WRITE?: NO
```

Use the Map Disk (MD) command to check the attributes of each file created. The format for the MD command is as follows:

```
[] MD
MAP DISK
             PATHNAME: <volume name>.S$UDIAG.<IDT name of module>
  LISTING ACCESS NAME: <RETURN>
         SHORT FORM?: NO
      TOP LEVEL ONLY?: NO
DIRECTORY NODES ONLY?: NO
```

The attributes of each file that is created should be similar to those in the following example:

```
 ALLOC: PRI=118  SEC=0  #SECS=0  LRECL=72  PRECL=288  BLK/ADU=1
 WPT/DPT/BLK/PERM/FORCED/=NNYYN  DATA-FMT=BINARY
```

8.3.6 Copying the Diagnostic Files

Copy each object or compressed object diagnostic file to the relative record file by executing the CC command:

```
[] CC
COPY/CONCATENATE
 INPUT ACCESS NAME(S): <pathname of diagnostic file>
   OUTPUT ACCESS NAME: <pathname of relative record file>
              REPLACE?: YES
MAXIMUM RECORD LENGTH: <RETURN>
```

Verify the existence of the diagnostic as a relative record file by executing the MD command, with <vol name>.S$UDIAG.<IDT name of module> as your response to the prompt PATHNAME.

8.3.7 Assigning the System File

You can use the system file that normally brings in the DXM operating system to bring in DOCS. To do so, use the MVI command. Respond to the SELECT prompt in either of two ways:

* Enter T (for test) to specify the secondary system image (S200DOCS).
* Enter Y (for yes) to specify the diagnostic file (S200DOCS).

Both responses permit a one-time load of S200DOCS on a power-up sequence while the disk is unprotected. In both cases, S200DOCS must reside on the disk.

NOTE

Since MVI operates in TTY mode, prompts appear one line at a time during actual execution of the MVI procedure.

```
[] MVI
MODIFY VOLUME INFORMATION
CONTROL ACCESS NAME: ME
MVI
DISK? DS01
COMMAND (L,C,Q)? L
                      PRIMARY    SECONDARY SELECT
       SYSTEM IMAGE: <SYSTEM>   <S200DOCS>    P    (P, S, T)
       PROGRAM FILE: <S$PROGA>                P
       OVERLAY FILE: <S$OVLYA>                P
        LOADER FILE:                          S    (P, S, T)
           WCS FILE:                          P
         DIAGNOSTIC: <S200DOCS>               N    (N, Y)
        VOLUME NAME: <SYSTEM32>
COMMAND (L,C,Q)?
```

To change the secondary load file to S200DOCS using the MVI command, respond as follows:

```
[] MVI
MODIFY VOLUME INFORMATION
CONTROL ACCESS NAME: ME
MVI
DISK? <device or volume name of S200DOCS disk, e.g. DS02>
COMMAND (L,C,Q)? C
WHICH ITEM (S,O,P,L,D,V,W)? S
              PRIMARY: <return>
            SECONDARY: <S200DOCS>
               SELECT: <T>
COMMAND (L,C,Q)? Q
MVI TERMINATED
```

Next, repeat the initial program load (IPL) for the system. S200DOCS is now loaded.

Once DOCS is initialized and executing, you can load and execute one or more test modules. To restore DXM, repeat the power-up sequence.

You can load S200DOCS repeatedly by selecting S200DOCS as the secondary system file. To do so, enter S in response to the SELECT prompt of the MVI command. Do not do this to your system disk; if you do, you will not be able to load DXM from your system disk.

NOTE

Currently, the S200 track 1 loader supports only those sequential DOCS files that are in the top-level directory. Consequently, S200DOCS must be in sequential format and in the top-level directory. A relative version of S200DOCS may be placed under the S$UDIAG directory in order to load S200DOCS using S200DOCS.

8.4 CREATING A COMPRESSED OBJECT FILE MODULE

You can create a compressed object file module by issuing the Execute Linkage Editor (XLE) command on DX10 with the following link control file:

```
NOSYMT
FORMAT COMPRESSED
PHASE 0,<name>
INCLUDE <pathname of standard object file>
END
```

where:

<name> is any meaningful name (for example, the IDT name of the module).

<pathname> is the name of the standard object file.

Section 9

Batch Command Stream DOCS

9.1 INTRODUCTION

Batch Command Stream DOCS is a type of DOCS consisting of Standard DOCS, Dynamic DOCS, or S200DOCS with the batch command stream (BCS) verbs. This type allows you to build a list of verbs and their associated input data and then execute the list as though it were one verb. Since BCS verbs are common to all three categories of DOCS, the term DOCS will refer to the category you are using throughout this section.

Standard DOCS contains all of the BCS verbs. S200DOCS and Dynamic DOCS consists of two types: BCS Run DOCS and BCS Edit DOCS. Refer to Sections 4 and 5 for an outline of building these two types with Dynamic DOCS and S200DOCS, respectively.

Hardware required for Batch Command Stream DOCS includes a 990-based computer with 64K bytes of memory, an interactive terminal, and a loading device.

9.2 BCS DOCS Type

The BCS DOCS type has the following additional characteristics:

* Three operational modes
* A well-defined structure for the BCS list
* Ability to load as many as three data files in memory in addition to BCS verbs:
  - Batch command stream list
  - User-defined verb module
  - Diagnostic
* A group of verbs that enable you to build and execute a BCS list

The first three characteristics are explained in the following paragraphs. The BCS verbs are discussed in paragraph 9.4.

9.2.1 Modes of Operation

Batch Command Stream DOCS features three operational modes:

* Normal mode
* Build list mode
* Execute list mode

Depending on the mode of operation, some BCS verbs are legal and some are illegal. The Maxi DOCS verbs are legal in any BCS mode. The three operational modes also allow DOCS to determine when you have entered a verb that is out of context or illogical. For example, the Execute List (.EL) verb cannot be entered while DOCS is in build list mode. If this is attempted, DOCS responds with the message ILLEGAL INPUT. Table 9-1 lists the DOCS verbs and indicates which verbs may be used in each mode.

9.2.2 Normal Mode

In normal mode, DOCS accepts and processes verbs in the usual manner. DOCS is in normal mode when it is loaded and begins executing. It returns to normal mode from either of the other two modes whenever you press the @, CMD, or HELP key, or when an error has occurred in another mode.

9.2.3 Build List Mode

Build list mode allows you to build a BCS list in memory when you execute the Build List (.BL) verb. DOCS then displays the message ENTERING BUILD LIST MODE. Operationally, DOCS appears to function as it does in normal mode but is actually placing a copy of every user input (except some of the execution control and list editing verbs) in a special table until build list mode terminates. The batch command stream normally terminates when it reaches a plus (+) sign at the end of the list. If you need to interrupt the batch stream during execution, press the @, CMD, HELP, or ESC key. This returns DOCS to normal mode and the BCS list remains in memory.

If you are using BCS Edit DOCS, you can modify this BCS list using the list editing verbs described in paragraph 9.4.2. You can execute these verbs in either build list or normal mode.

Table 9-1 DOCS BCS Verbs

DOCS BCS Verbs	Description	Normal Mode	Build List Mode	Execute List Mode
.BL	Build List	x		
.LL	Load List	x		
.EL	Execute List	x		
.IL	Insert Line	x	x	
.DL	Delete Line	x	x	
.RL	Replace Line	x	x	
.CL	Compress List	x	x	
.PL	Print Line (or list)	x	x	
.LG	Load and Go	x	x	x
.AT	Automatic Terminate	x	x	x
.CF	Clear Flag	x	x	x
.SF	Set Flag	x	x	x
.MF	Modify Flag	x	x	x
.LB	Establish Label		x	x
.GT	Unconditional Transfer		x	x
.IF	Conditional Transfer		x	x
.OI	User Input		x	x
.OC	User Comment		x	x
.OD	User Decision		x	x
.TS	Transfer on Space Bar		x	x
.TN	Transfer on Number of Errors		x	x
.DO	Do Loop		x	x
+	Exit to Normal Mode		x	x

9.2.4 Execute List Mode

DOCS goes into execute list mode when you invoke the Execute List (.EL) verb. DOCS then displays the message ENTERING EXECUTE LIST MODE. From this point on, DOCS receives all inputs from the BCS list and no user intervention is required. However, there are some cases when user intervention is desirable.

When a BCS list is designed to automatically test a TTY/EIA card, you can build a list that loads the test, responds to most of the initialization prompts, and executes the test. However, some prompts may need to be answered directly by the verb.

When you want to specify responses, you can place the Operator Input (.OI) and Operator Decision (.OD) verbs at appropriate places in the BCS list. In this way, DOCS obtains input from the list until:

* The .OI and .OD verb are encountered
* The end of the list is reached
* You press the @, CMD, HELP or ESC key
* A major error condition is encountered

9.2.5 BCS List Structure

When a BCS list is being built, it is placed immediately following DOCS. The structure and format of this list are shown in Figure 9-1 and Figure 9-2.

Line No.	User Input	Hexadecimal Memory Content	Comments
1	.LD<CR>	2E4C440D	Load verb module
2	TST940<CR>	5453543934300D	TST940 diagnostic
3	<CR>	0D	Default load bias
.			Responses to IT
.			verb prompts
.			
7	ET<CR>	45540D	Execute test
8	1<CR>	310D	Test number 1
9	ET<CR>	45540D	Execute test
A	3<CR>	330D	Test number 3
B	ET<CR>	45540D	Execute test
C	5<CR>	350D	Test number 5
D	+<CR>	2B0D	List terminator

Figure 9-1 Example of a BCS List

Figure 9-1 illustrates the organization of a simple BCS list. Each element or line of the list consists of a single user input (character string) that includes a set of ASCII characters followed by a RETURN or NEW LINE character (CR). Each line in the list has an associated line number (line #). Using these line numbers, you can modify specific elements of the list with special list editing verbs (.IL, .DL, and .RL). You can also use the line numbers to begin execution of the batch stream at any place within the list. When the list is printed (using the .PL verb), the line numbers appear next to each element.

The list terminator is designated by the plus (+) sign. When DOCS is directed to execute the BCS list, it starts at the line number specified by the Execute List (.EL) verb. For every prompt or request by DOCS or the test module, user responses are obtained from the list in the order in which they have been entered. This normally continues until the list reaches the + sign, which causes DOCS to terminate execute list mode and return to normal mode. The first prompt following execution of the .EL verb is VERB?.

In the example in Figure 9-1, when you enter the .EL verb and specify line 1 as the starting point, DOCS loads the TST940 test module, responds to the IT prompts, and executes parts 1, 3, and 5. It then terminates execute list mode and returns to normal mode with the VERB? prompt.

Figure 9-2 gives a formal definition of a BCS list.

<BCS List>=[<char string input>][<input list>]<list term><CR>

where:

Items in [ ] are optional

<char string input>	= [<char string>]<CR>
<input list>	= <char string>[<input list>]
<char string>	= Any set of ASCII characters up to 50 characters long. Illegal character strings include @, +, .LL, .IL, .DL, .RL, .CL, and .PL. <char string> can also be null.
<CR>	= >0D that is generated by the NEW LINE or RETURN key in build list mode.
<list term>	= +
<insert>	= >F1 (list editing insert character)
<delete>	= >F2 (list editing delete character)

Figure 9-2 Formal Definition of a BCS List

An input list cannot exceed 50 characters and certain character strings are not allowed in the list. These illegal strings are typically verbs that are illegal in execute list mode or verbs that must be terminated by the @, CMD, HELP or ESC key, since these keys terminate the list.

Batch stream data can be placed in the BCS list by one of three methods:

* The first method places DOCS in build list mode through the execution of the Build List (.BL) verb. User inputs (except for some of the execution control and list editing verbs) are then placed in the BCS list until build list mode is terminated by pressing the @, CMD, HELP, or ESC key.

* The second method involves executing either the load list (.LL) verb or Load and Go (.LG) verb, which brings in a file from the DOCS load media and places it in the BCS list. The .LG verb differs from the .LL verb because it begins execution of the list immediately after the list is loaded.

  If you use this method to build a BCS list, the data file must first be created by the DX10 (990 computers) or DXM (S200 computers) text editor and assembler. The text editor is used to create an assembly language program consisting entirely of text statements that contain the character strings comprising the BCS list (see Figure 9-2). The text edited program is then assembled and placed on the DOCS load media in the correct file format. When this process is complete, the .LL or .LG verb can load the batch stream into the BCS list. Refer to Figure 9-7 for a sample program to build a BCS list using the DXM assembler.

* The third method uses list editing commands (.IL, .DL, .RL, and .CL) in normal or build list mode to modify an existing list or create a new one.

9.2.6 Memory Usage

DOCS allows the loading of up to three files in memory at the same time:

* A BCS list
* A user-written verb module
* A diagnostic test

The BCS list should always be loaded immediately following DOCS and can be any length (up to the end of available memory). Use the following order when loading:

1. DOCS & BCS verbs
2. Batch command streams
3. User-written verb modules
4. Diagnostics

CAUTION

You must observe this hierarchy when loading files. If you load in any other order, you may lose one or more files.

DOCS controls the location of the files. It is recommended that you accept the default load bias for all load procedures.

The memory map in Figure 9-3 illustrates DOCS memory usage for 990 computers and Figure 9-4 illustrates DOCS memory usage for S200 computers.

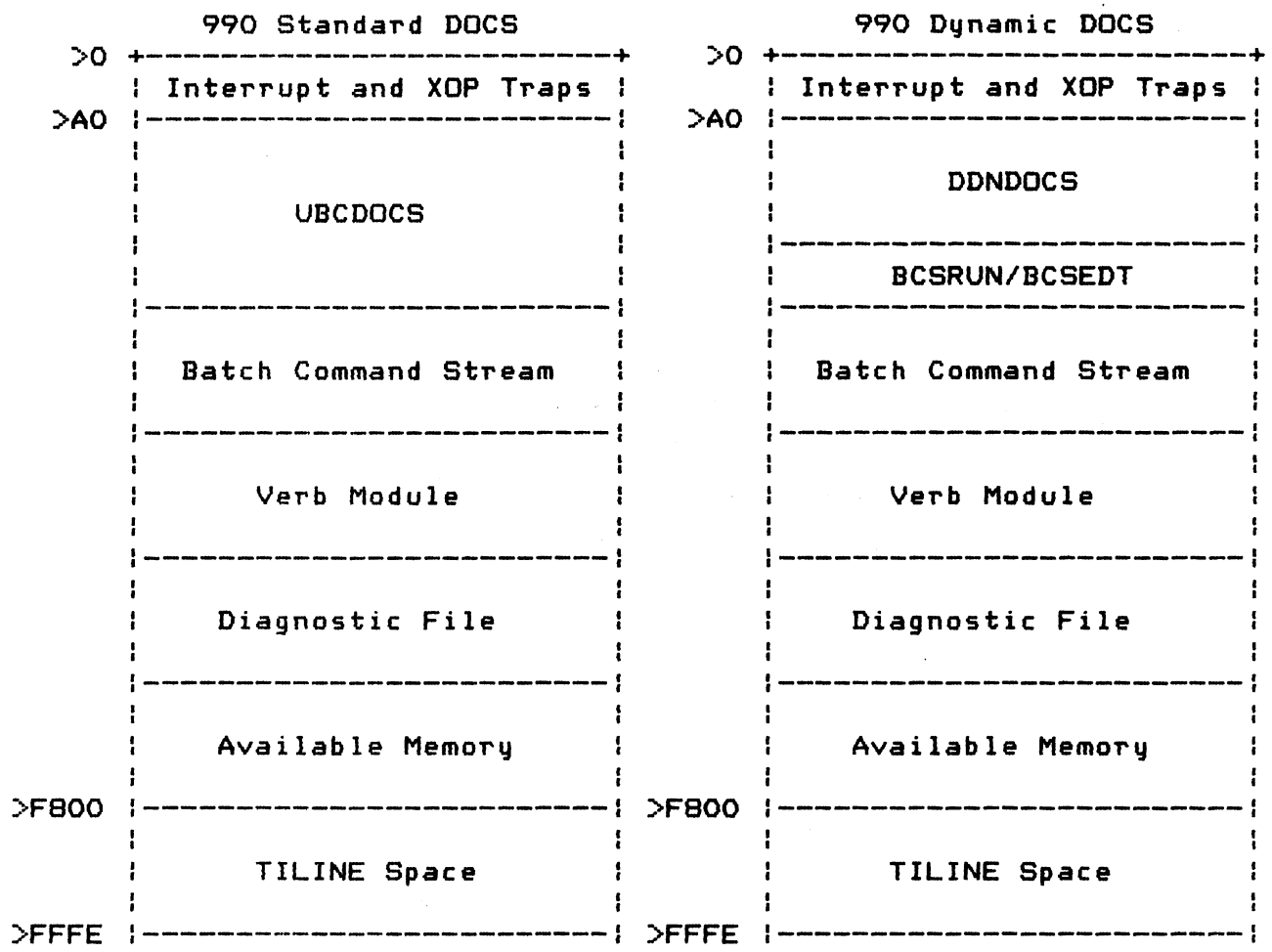


Figure 9-3 BCDDCS Memory Usage for 990 Computers

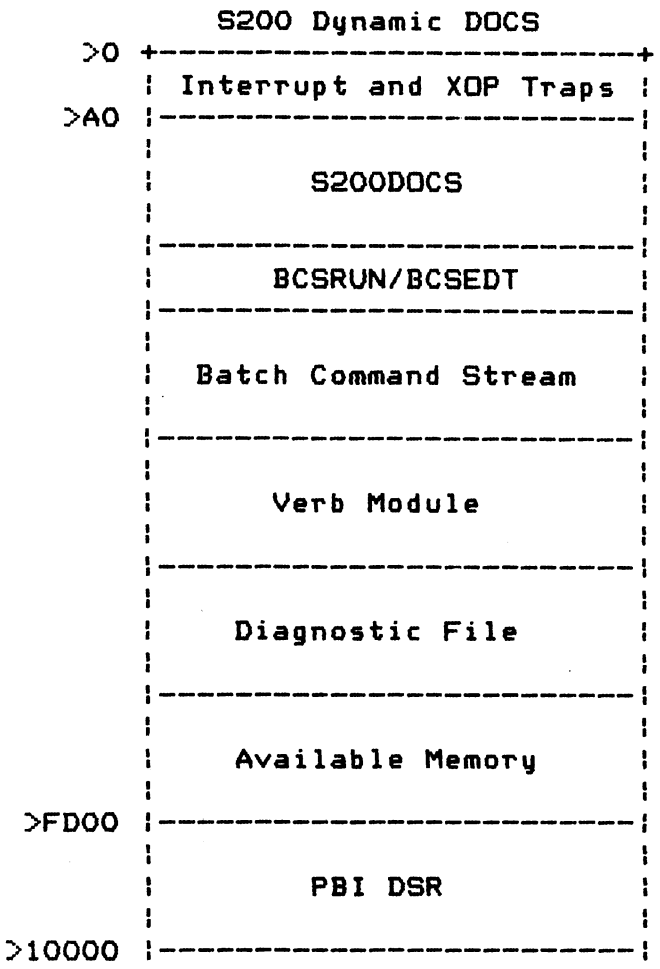


Figure 9-4 DOCS Memory Usage for S200 Computers

9.3 BCDOCS LOAD PROCEDURES

Refer to Section 3, 4, or 5 for initialization procedures for Standard DOCS, Dynamic DOCS, or S200DOCS respectively. For S200DOCS and Dynamic DOCS, enter 2 or 3 in response to the DOCS TYPE? prompt, respecitvely. BCSRUN contains only those verbs required to execute a batch stream. BCSEDT, which contains all of the BCS and DOCS verbs, should be used under most circumstances. After these responses are made, you can build a list of batch stream commands for a test. (Since Standard DOCS already contains the BCS verbs, this prompt does not appear.)

NOTE

If you use Standard BCDOCS frequently, you can set up BCDOCS as the system file. In this case, BCDOCS is loaded during the initial system load and you can proceed directly with BCDOCS initialization. Refer to Section 8 for system file assignment procedures.

9.4 BCDOCS VERBS

The verbs supporting the batch command stream capability are divided into three categories:

* Mode control
* List editing
* Execution control

In addition to these verbs, a set of 128 flags (numbered 0 through >7F) are available for batch command stream program development. These flags can be set, cleared, or tested by a number of DOCS verbs. Flag 0 is designated as the system error flag and is set any time an error message is displayed.

9.4.1 Mode Control Verbs

The mode control verbs (.BL, .EL, .LL, .LG, and the @ and + characters) control the mode of operation and provide a means of building the BCS list. The following paragraphs describe these verbs.

9.4.1.1 Build List (.BL) Verb. The .BL verb initializes the BCS list and places DOCS in build list mode. It is an illegal input when DOCS is in execute list mode and, if executed, causes a return to normal mode. While DOCS is in build list mode, all inputs are stored in the BCS list and all verbs are executed as they are entered. Entering the plus (+) sign terminates the list and build list mode. The list is built in a 1024-byte segment of memory immediately following DOCS and remains available until you execute another .BL, .LL, or .LG verb.

NOTE

Execution of the .BL verb unlinks all diagnostics, user-written verb modules, and batch command streams that are currently loaded.

9.4.1.2 Execute List (.EL) Verb. The .EL verb places DOCS in execute list mode and starts batch stream execution at the line specified, which is represented in the following format by <line #>.

.EL-<line #>

If you do not specify the line number, execution resumes at the default line number. The default line number is either 0 or the number of the line that was last executed when a prior execution of the list was terminated by the @, CMD, or HELP key. When 0 is specified or is the default line number, execution begins at the beginning of the list. The .EL verb is an illegal input when DOCS is in build list mode and, if executed, causes a return to normal mode and the question VERB?. Execution of this verb clears all the BCS flags.

9.4.1.3 Load List (.LL) Verb. The .LL verb causes DOCS to display the diagnostics menu, load the specified file into the BCS list, and initialize the list pointers. The list is loaded from the media into memory at the location that you specified. The BCS list can be loaded anywhere in available memory using DOCS. The BCS list can be any length that the memory size permits. To prevent loss of important data, it is recommended that you accept the default load address specified by DOCS. The .LL verb is an illegal input when DOCS is in build list mode and, if executed, causes a return to normal mode. The format for this verb is:

.LL-<DOCS loading parameters>

9.4.1.4 Load and Go (.LG) Verb. Execution of the .LG verb is equivalent to executing an .LL verb and then the .EL verb. Execution starts at the beginning of the list. The .LG verb and its parameters can be the last entries in a list that is currently loaded. When .LG is executed, a new list can be entered. This verb permits chaining and overlaying the execution of discrete BCS lists. If a number of BCS lists are to be loaded with the same user-written verb module, load the largest list first. This step ensures that the verb module remains available to DOCS. It is recommended that you accept the default load address to prevent loss of important data. Execution of this verb does not change the status of any of the BCS flags. The .LG verb is a legal input in both build list and execute list modes.

The format for this verb is:

.LG-<DOCS loading parameters>

9.4.1.5 Exit to Normal Mode (+). Entry of the plus (+) sign (whether by you or from the BCS list) causes DOCS to return to normal mode. If DOCS is in execute list mode when you press the + sign, the execution pointer is reset to 0, which is the beginning of the BCS list. This is the normal procedure for terminating the batch stream. The @ or ESC key is used to interrupt the batch stream when necessary.

9.4.2 List Editing Verbs

The list editing verbs (.IL, .DL, .RL, .CL, and .PL) modify the BCS list. These verbs are included only if running the BCS EDIT type of DOCS. They are legal only in build list and normal modes. While modifications are being made, line numbers associated with unmodified list entries remain unchanged. Line numbers are reassigned only when the compress list (.CL) verb is executed. The list editing verbs are described in the following paragraphs.

9.4.2.1 Insert Line (.IL) Verb. The .IL verb inserts the character string followed by <CR> prior to the specified line number, where <CR> is the RETURN or NEW LINE character. The character string can be from 0 to 50 ASCII characters long. Since the list terminator has a line number, insertions can be made throughout the list. The format for the .IL verb is as follows:

.IL-<line #>-<char string>

9.4.2.2 Delete Line (.DL) Verb. The .DL verb deletes any character string input with an associated line number from the BCS list, but it does not delete the list terminator. The format for the .DL verb is as follows:

.DL-<line #>

9.4.2.3 Replace Line (.RL) Verb. The .RL verb replaces any character string input that has an associated line number with a specified character string followed by <CR>. The list terminator cannot be replaced. The format for this verb is as follows:

.RL-<line #>-<char string>

9.4.2.4 Compress List (.CL) Verb. The .CL verb reassigns line numbers in the BCS list. (Line numbers are determined by the relative position of a character string within the BCS list.) The .CL verb is useful for renumbering an existing list following a number of insertions or deletions.

9.4.2.5 Print List (.PL) Verb. The .PL verb outputs one or more character strings from the BCS list to the interactive terminal. It can also be used to print the entire list. Each line that is printed contains the line number followed by its corresponding character string. All deleted lines are denoted by the * character. The character strings of inserted lines are printed without line numbers. If the line is a null character string, <CR> is printed.

In the following format, <output dev> refers to either the interactive or message terminal specified during the initialization of DOCS. If you enter 1, output is sent to the interactive terminal; if you enter 0, the message terminal receives the output. The .PL verb has two formats: the first prints one or more specified lines; the second prints the entire BCS list.

* To print one or more lines:

  .PL-<line #>-<# of lines>-<output dev>

* To print the BCS list:

  .PL-0-<output dev>

9.4.3 Execution Control Verbs

The execution control verbs (.LB, .GT, .IF, .CF, .SF, .MF, .OI, .OC, .OD, .AT, .TS, .TN, .DO, and .LV) control the flow of the BCS list during execution. They also permit user input and the output of messages during batch execution. The execution control verbs are described in the following paragraphs.

9.4.3.1 Establish Label (.LB) Verb. The .LB verb establishes a label of up to four characters in length at the position where it appears in the BCS list. This label is recognized by the routines servicing the transfer-type verbs (.GT, .IF, .OD, .TS, .TN, and .DO). These routines permit branching and looping within the BCS list. Duplicate labels should be avoided, since only the first one in the list will be detected by the service routines. The label must be placed immediately before a verb. The format for the .LB verb is as follows, with <label value> representing a character string consisting of from one to four ASCII characters.

.LB-<label value>

9.4.3.2 Unconditional Transfer (.GT) Verb. The .GT verb causes BCS execution to resume at the verb following the label indicated by <label value> in the following format. This verb is only effective in execute list mode. If the label specified does not exist, DOCS terminates execute list mode and enters normal mode. The .GT verb is an illegal verb in normal mode.

.GT-<label value>

9.4.3.3 Conditional Transfer (.IF) Verb. The .IF verb causes BCS execution to resume at the verb following a specified label if the flag number is set. If either the string or flag number is nonexistent, DOCS exits to normal mode. This verb is effective only in execute list mode and is an illegal input in normal mode. In the following format, <label value> refers to the label prior to the verb and <flag #> indicates the flag number.

.IF-<flag #>-<label value>

9.4.3.4 Flag Verbs. The flag verbs (.CF, .SF, and .MF) clear, set, or modify any of the >7F BCS flags. A flag is user-assigned, and is specified by a flag number (<flag #>). The flag verbs are legal in any mode, but if a nonexistent flag number is specified, DOCS goes into normal mode.

The flag numbers are >0 through >7F. Flag >0 is reserved for the system. This flag may be referenced, but its state should not be modified. All other flags may be manipulated using the following verbs.

Clear Flag (.CF) Verb

The .CF verb clears the specified BCS flag to 0. The format for this verb is as follows:

.CF-<flag #>

Set Flag (.SF) Verb

The .SF verb sets the specified BCS flag to 1. The format for this verb is as follows:

.SF-<flag #>

Modify Flag (.MF) Verb

The .MF verb modifies the specified BCS flag to either 0 or 1. In the following format, <flag value> can be either 0 or 1.

.MF-<flag #>-<flag value>

9.4.3.5 Operator Input (.OI) Verb. The .OI verb allows you to alter the input operation that normally occurs in execute list mode. This is useful when data cannot be predetermined and placed in the BCS list during the build list process, or when you want to control the execution path through the BCS list. An example using the .OI verb is shown in Figure 9-5.

During the build list process, the .OI verb is placed in the BCS list. When the list is later executed, the .OI verb causes DOCS to prompt you for input that is not placed in the list.

The data that you enter is checked for syntax errors. For example, the verb verifies that hexadecimal entries are preceded by a valid hexadecimal sign (>) and that ASCII entries follow standard ASCII/Hollerith Character codes (range 00 through 7F).

9.4.3.6 Operator Comment (.OC) Verb. The .OC verb provides a means of outputting text to you during the execution of a BCS list. In the following format, <text> is a string of from 0 to 50 ASCII characters. This verb is an illegal input during normal mode.

.OC-<text>

9.4.3.7 Operator Decision (.OD) Verb. The .OD verb allows you to implement a decision based on information contained in an output message during BCS execution. The format for the .OD verb is as follows:

.OD-<label value>-<text>-<rspn>

The information in the output message is represented in the format by <text>. If your response (<rspn>) is NO (0), transfer is made to <label value> and execution resumes. If your response is YES (1), execution resumes on the next statement. The .OI verb must be placed in the response field when the BCS list is being built in order to permit user response.

9.4.3.8 Automatic Terminate (.AT) Verb. Execution of the .AT verb causes the automatic termination of execute list mode whenever an error message is output. The system error flag (#0) is set and termination takes place whenever execution of the current verb ends. Execution of the .AT verb causes the BCS flag number 0 to be cleared. This verb is not effective in normal and build list modes.

9.4.3.9 Transfer on Space Bar (.TS) Verb. The .TS verb allows you to transfer to the specified label within the BCS list whenever you press the space bar on the interactive terminal. After the transfer is made, the .TS verb is deactivated and another .TS must be executed to reactivate it. This verb is useful when you are writing a list that will loop on a number of commands until the space bar is pressed.

.TS-<label value>

9.4.3.10 Transfer on Number of Errors (.TN) Verb. The .TN verb causes the BCS list to transfer execution to a specified label when a given number of errors is encountered. Each time an error message is output, a counter is bumped and compared to the given number of errors. When the two values are equal, execution will continue to the specified label. After the transfer is made, the .TN verb is deactivated and another .TN verb must be executed to reactivate it. This verb is useful when a test should recognize a number of errors before taking any action.

.TN-<number of errors>-<label value>

9.4.3.11 Do Loop (.DO) Verb. The .DO verb allows the BCS list to loop a given number of times on a section of commands. When this iteration is complete, the next batch stream entry following the end label is executed. The range of the .DO verb includes the BCS entry following the .DO verb through the label valued specified in the format for the verb as illustrated by the following:

.DO-<label value>-<# iterations>

9.5 BCS LIST EXAMPLE BUILD PROCEDURE

Figure 9-5 illustrates how to build a BCS list that asks if you want to run the TST940 test. If you answer no, DOCS returns to normal mode by going to the table terminator at the end of the list. If you answer yes, the following sequence of events occurs in execute list mode:

1. The diagnostic is loaded.
2. The IT verb prompts are answered from the BCS list.
3. The diagnostic test is executed.

```
VERB? -.BL<CR>

ENTERING BUILD LIST MODE

VERB? -.OC<CR>

 TEXT -TEST 940 BATCH COMMAND STREAM - 9/82 REV *A<CR>

VERB? -.OD<CR>

 LABEL VALUE -BB<CR>
 TEXT  -DO YOU WANT TO RUN THE TST940 DIAGNOSTIC?<CR>
 (1=YES, 0=NO)  -.OI<CR> -<CR>

VERB? -.LD<CR>

MENU FOR S$UDIAG DIRECTORY

00) BCSEDT   01) BCSRUN   02) COM90X   03) DBVERB1
04) DSKCOM   05) DSKSA    06) DSKWDX   07) MXVERB1
08) MXVERB2  09) S200DOCS 0A) TST8X0   0B) TST940

TYPE: DIAG. # OR FILE NAME ? -TST940<CR>
LOAD BIAS?  DEF= XXXXXX  -<CR>
END OF TEST = XXXX

TST940 - 940 ELECTRONIC VIDEO TERMINAL VERSION *A : 09/82

INTERFACE DEVICE?(0=CIM, 1=TTY/EIA, 2=TMS990X, 3=SX00)DEF=0-3<CR>
LINE PRINTER ON AUX #1? DEF = 0 -<CR>
EXECUTE EA VERB? (DEF=1) -0<CR>

VERB? -ET<CR>
TEST # -1
  TEST 01 - COMMUNICATIONS TEST - < 1 MIN
 BAUD RATE PASSED

TEST COMPLETE

VERB? -.LB<CR>

 LABEL VALUE  -BB<CR>

VERB? -+<CR>
ENDING BUILD LIST MODE
```

Figure 9-5 Example of BCS List Build Procedure

The example in Figure 9-6 assumes that the batch stream is always run on an S200 and that the interface device default is always SX00 940 EVT. The batch is designed to run only the communications test (test 1) of TST940. The addresses for the LOAD BIAS? and END OF TEST prompts will vary depending on the category of DOCS you are using.

Figure 9-6 illustrates the internal representation of the list after it has been built.

The list can also be built using the DX10 (990 computers) or DXM (S200 computers) assembler. In this case, the TEXT statement must be used to set up the list as it should appear in the BCS list. Figure 9-7 shows the program necessary to build the list in Figure 9-5.

The user building the list uses a DXM system and performs the DXM Execute Macro Assembler (XMA) command, giving the file pathname of the program in Figure 9-5 as the SOURCE ACCESS NAME. The specified OBJECT ACCESS NAME is the file that should be placed on the DOCS load media in relative format as the list to be loaded by the .LL verb. Once the list is loaded, it can be executed by the .EL verb. (See Section 8 of this volume for information on building files in relative format.)

The DX10/DXM assembler method of building lists is more practical for lists that are to be used often. The list can be loaded much faster than it can be built using the .BL verb. For special lists, use of the .BL verb is more practical. To develop a new list, use the .BL verb to organize and initially develop and debug the list.

After the list is printed, use the assembler to make a permanent copy, which can be loaded later by the .LL verb. Figure 9-6 illustrates the running of the BCS list built by either of the two procedures discussed. (Note that the addresses for the LOAD BIAS? and END OF TEST prompts will vary depending on the category of DOCS you are using.)

Line No.	Character String	Action
1	.OC	Operator comment
2	TST940 BATCH COMMAND STREAM - DATE 9/82 REV *A	Text of comment
3	.OD	Operator decision
4	BB	Transfer label on 'no'
5	DO YOU WANT TO RUN THE TST940 DIAGNOSTIC?	Text of operator decision
6	.OI	Prompt user for yes/no
7	.LD	Load diagnostic
8	TST940	Diagnostic name
9	<CR>	Default load bias
10	3	SX00 interface type
11	<CR>	Default no AUX device
12	0	Execute all - no
13	ET	Execute test
14	1	Test number 1
15	.LB	Establish label
16	BB	Label value
17	+	BCS terminator - Causes DOCS to exit execute list mode, enter normal mode, and issue the VERB? prompt.

Memory Contents								ASCII Interpretation
2E4F	440D	5453	5439	3430	2041	4054	4348	.OC.TST940 BATCH
434F	4D4D	414E	4420	5354	5245	414D	202D	COMMAND STREAM -
4441	5445	2034	2F38	3220	5245	5620	2A2A	DATE 4/82 REV **
0D2E	4F44	0D42	420D	444F	2059	4F55	2057	..OD.BB.DO YOU W
414E	5420	544F	2052	554E	2054	4845	2054	ANT TO RUN THE T
5354	3934	3020	4449	4147	4E4F	5354	4943	ST940 DIAGNOSTIC
3F0D	2E4F	490D	2E4C	440D	5453	5439	3430	?..OI..LD.TST940
0D0D	330D	0D30	0D45	540D	310D	254C	420D	..3..0.ET.1..LB.
4242	0D2B	0D						BB.+.

Note: The hexadecimal OD in the memory dump has an ASCII interpretation of ".". The + represents the list terminator.

Figure 9-6 Internal Representation of TST940 Build List

```
      IDT   'BCS TEST'
CR    EQU   >OD                     EQUATE CR WITH ASCII VALUE
*
      TEXT  '.OC'                   OPERATOR COMMENT
      BYTE  CR
      TEXT  'TST940 BATCH COMMAND STREAM -'
      TEXT  ' DATE 9/82 REV *A'
      BYTE  CR
      TEXT  '.OD'                   OPERATOR DECISION
      BYTE  CR
      TEXT  'BB'                    LABEL TO BRANCH TO ON 'NO'
      BYTE  CR
      TEXT  'DO YOU WANT TO RUN THE '
      TEXT  'TST940 DIAGNOSTIC?'
      BYTE  CR
      TEXT  '.OI'                   OPERATOR INPUT REQUEST
      BYTE  CR
      TEXT  '.LD'                   LOAD DIAGNOSTIC
      BYTE  CR
      TEXT  'TST940'                DIAGNOSTIC NAME
      BYTE  CR
      BYTE  CR                      DEFAULT LOAD BIAS
      TEXT  '3'                     INTERFACE TYPE = SX00
      BYTE  CR
      BYTE  CR                      NO PRINTER ON AUX PORT
      TEXT  'O'                     EXECUTE ALL? = NO
      BYTE  CR
      TEXT  'ET'                    EXECUTE TEST
      BYTE  CR
      TEXT  '1'                     TEST NUMBER
      BYTE  CR
*
      TEXT  '.LB'                   ESTABLISH LABEL
      BYTE  CR
      TEXT  'BB'                    LABEL VALUE
      BYTE  CR
      TEXT  '+'                     TERMINATOR
      BYTE  CR
```

Figure 9-7 Program to Build BCS List Using DXM Assembler

```
VERB? -.EL
LINE # (DEF=0000) -0
ENTERING EXECUTE LIST MODE

VERB? -.OC

 TEXT -TST940 BATCH COMMAND STREAM - DATE 9/82 REV *A

VERB? -.OD

 LABEL VALUE -BB
 TEXT -DO YOU WANT TO RUN THE TST940 DIAGNOSTIC?
 (1=YES, 0=NO) -.OI -1

VERB? -.LD

 MENU FOR S$UDIAG DIRECTORY

 00) BCSEDT     01) BCSRUN     02) COM90X     03) DBVERB1
 04) DSKCOM     05) DSKSA      06) DSKWDX     07) MXVERB1
 08) MXVERB2    09) S200DOCS   0A) TST8X0     0B) TST940

 TYPE: DIAG. # OR FILE NAME ? -TST940
 LOAD BIAS?  DEF=XXXXXX -
 END OF TEST = XXXX

 TST940 - 940 ELECTRONIC VIDEO TERMINAL VERSION *A : 09/82

 INTRFCE DEVICE?(0=CIM, 1=TTY/EIA, 2=TMS990X, 3=SX00)DEF=0-3
 LINE PRINTER ON AUX #1? DEF = 0 -
 EXECUTE EA VERB? (DEF=1) -0

VERB? -ET
 TEST # -1
   TEST 01 - COMMUNICATIONS TEST - < 1 MIN
  BAUD RATE PASSED

 TEST COMPLETE

VERB? -.LB

 LABEL VALUE -BB

VERB? -+
 ENDING BUILD LIST MODE

VERB? -
```

Figure 9-8 Running the BCS List

Section 10

Debug DOCS

10.1 INTRODUCTION

Debug DOCS allows you to establish software breakpoints in the diagnostic tests run under DOCS. This type of DOCS is especially useful when you want to modify a test to particular specifications.

Your 990-based computer must contain 64K bytes of memory, and appropriate loading and interactive devices. You should be familiar with TI 9900 Assembly Language. If you are using a Model 990 Computer, be familiar with the Standard DOCS information in Section 3 before implementing Debug DOCS. If you are using an S200 computer, be familiar with the S200DOCS information in Section 5.

Refer to Appendix A for a list of the object modules required to create a fully linked object module of Debug DOCS.

The following paragraphs discuss the operating procedures, Debug DOCS verbs, and limitations of Debug DOCS.

10.2 OPERATING PROCEDURES

The Debug DOCS operating procedures for 990 and S200 computers differ, as described in the following paragraphs.

10.2.1 Operating Procedures for Model 990 Computers

The DOCS version that is the system file must be loaded and initialized before Debug DOCS can be implemented, unless Debug DOCS is identified as the system file. (Refer to Section 3 or 4 for initialization procedures for Standard or Dynamic DOCS, respectively.) When these procedures are performed, you can then load and execute Debug DOCS.

If Dynamic DOCS was loaded, enter a 4 in response to the 'ENTER DOCS TYPE?' prompt. After you have entered these responses, you can assign breakpoints in either DOCS or in a specific test by executing the Assign Breakpoint (.AB) verb as described later in this section.

NOTE

If you use Standard Debug DOCS frequently, you can set up Debug DOCS as the system file. In this case, Debug DOCS is loaded during the initial load and you can proceed directly with Debug DOCS initialization. Refer to Section 8 for system file assignment procedures.

10.2.2 Operating Procedures for Model S200 Computers

After you load and initialize S200DOCS, you must specify which type of DOCS you want to use. To select Debug DOCS, enter 4 in response to the prompt as shown in the following example:

```
**** TEXAS INSTRUMENTS  S200 DOCS          VERSION *R   9/82   ****

     USE "RETURN" TO TERMINATE INPUT
     USE "BACKSPACE" TO ERASE INPUT

ENTER DOCS TYPE(0=MINI 1=MAXI 2=BCS RUN 3=BCS EDIT 4=DEBUG)?DEF=1-4
     LOADING VERB MODULE MXVERB1   >DONE.
     LOADING VERB MODULE MXVERB2   >DONE.
     LOADING VERB MODULE DBVERB1   >DONE.
```

You can now assign breakpoints in either DOCS or a specific test by executing the assign breakpoint (.AB) verb as described later in this section.

10.3 DEBUG DOCS VERBS

In addition to the standard set of DOCS verbs, Debug DOCS provides six special verbs that allow you to set breakpoints in a test. These debug verbs and their functions are summarized in table 10-1 and subsequent paragraphs. The Debug verbs use a table containing the necessary data values associated with each location. The breakpoint table permits 20 possible entries; therefore, 20 breakpoints may be established at one time in the test being debugged.

Table 10-1 Debug DOCS Verbs

Debug DOCS Verbs	Description
.AB	Assign Breakpoint
.DB	Delete Breakpoint
.PB	Proceed from Breakpoint
.SB	Show Breakpoints
.DA	Delete All Breakpoints
.DP	Delete Current Breakpoint and Proceed

10.3.1 Assign Breakpoint (.AB) Verb

The .AB verb inputs data to the breakpoint table and sets the XOP 14 interrupt trap. The format for using the .AB verb is as follows, with AAAA representing the hexadecimal address associated with the breakpoint, and X indicating the number of words contained by the instruction where the breakpoint should be established.

VERB? -.AB

ADDR -AAAA

NUMBER OF WORDS IN INSTRUCTION - X

10.3.2 Delete Breakpoint (.DB) Verb

The .DB verb deletes the breakpoint in the test being debugged and clears the entry in the breakpoint table. In the following format, AAAA represents the hexadecimal address associated with the breakpoint.

VERB? - .DB

ADDR -AAAA

10.3.3 Proceed from Breakpoint (.PB) Verb

The .PB verb executes the instruction at the current breakpoint and continues execution until the end of the test, or until the next breakpoint occurs.

10.3.4 Show Breakpoints (.SB) Verb

The .SB verb outputs the currently established breakpoints on the interactive terminal. The breakpoints are displayed as four-digit hexadecimal numbers in the order in which they appear in the breakpoint table.

10.3.5 Delete All Breakpoints (.DA) Verb

The .DA verb deletes all currently assigned breakpoints.

10.3.6 Delete Current Breakpoint and Proceed (.DP) Verb

The .DP verb combines the functions of the .DB and .PB verbs by deleting the breakpoint where the test currently is and then executing the test from that point.

10.4 LIMITATIONS TO DEBUG DOCS

There are two restrictions on the use of Debug DOCS that arise when an instruction is replaced by a breakpoint, and then moved to a different memory location and executed from there.

* Debug DOCS does not set a breakpoint at an instruction that depends on more than three words for correct execution. For example, a BLWP instruction has several parameters that must be passed to the next subroutine. Since only three words can be transferred to the breakpoint table, the additional words of data cannot be located by the subroutine.

* Debug DOCS cannot set a breakpoint at a jump instruction. All jump instructions are relative to the current program counter, which points to the table instead of the address where the instruction originally existed. Therefore, any jump is relative to the address in the table, and will conform to the expected flow of the program. Breakpoints at any type of jump instruction must be avoided.

Appendix A

Diagnostic Part Numbers

A.1 INTRODUCTION

This appendix identifies all of the diagnostic information by part number. A two-character revision code follows most part numbers. The revision code for an original release is **, for a first revision is *A, for a second revision is *B, and so on. All data in this appendix is accurate for the 5.2.0 field release. Consult the latest product documentation package for subsequent diagnostic revisions.

The following abbreviations are used to indicate types of media and DOCS in the listings in this appendix:

U = Unit Record Media (Cassette/UST)
D = Hard Disk or Double-Sided, Double-Density (DSDD) Diskette
T = Open Reel/Cartridge Tape
MXDOCS = Maxi DOCS
MNDOCS = Mini DOCS
BCDOCS = Batch Command Stream DOCS
DBDOCS = Debug DOCS
DNDOCS = Dynamic DOCS

Table A-1 lists the maximum byte sizes (in hexadecimal) for the different types of DOCS. Note that the size of DOCS (excluding S200 DOCS) is dependent on the interactive and error message devices that you use.

Table A-1 Maximum Hexadecimal Byte Sizes for DOCS Types

DOCS Type	Byte Size
BCSEDT	14D8
BCSRUN	DAA
DBVERB1	4F4
DDNDOCS	3EBE
MXVERB1	2AC
MXVERB2	6B8
S200DOCS	4A22
TDNDOCS	3EF4
UBCDOCS	50E6
UDBDOCS	4004
UMNDOCS	3216
UMXDOCS	3B70

This appendix contains lists for the following:

* Package Inventory of 990 Unit Diagnostics (Object)
* Package Inventory of 990 Unit Diagnostics -- Cassette Kit (Object)
* Package Inventory of 990 Unit Diagnostics -- Universal Systems Terminal (UST)
* Index of 990 Unit Diagnostics and DOCS Systems

The following paragraphs discuss each of these lists.

A.2 PACKAGE INVENTORY OF 990 UNIT DIAGNOSTICS

The following list describes the media and gives the part numbers for the standard diagnostic packages and add-on diagnostic packages. The standard diagnostic packages are self-contained (that is, they are not combined with DX10).

The add-on diagnostic packages are on the same disk as the DX10 system. These packages are shipped as part of the DS990 package systems (DS990 Models 6, 8, 20, and 30).

PACKAGE INVENTORY

990 Unit Diagnostics
(Object)

Package Base Part Numbers:

License 2250536 includes Software Package 937782AY

Media choices:

Media Type	Standard Packages: Dash Number	Quantity	Add-on Packages: Dash Number
1600 bpi Mag. Tape	-0008	3	none
8" DSDD Diskette	-0021	4	none
CD1400	-0022	1	-0074
DS80	-0024	1	-0070
DS300	-0025	1	-0071
Cartridge Tape (CT)	-0026	3	none

The full package part number includes the base part number plus the dash number for the appropriate media.

EXAMPLE: 2250536-0008 is the full part number for the License on 1600 BPI Magnetic Tape media.

Other related parts/services available on price list:

NONE	Additional Use Fee
2250537-0001	Customer Support Line Service
2250537-nnnn	Software Package Subscription/Update Service (same dash numbers as for Software Package, except Add-on media not available)
2261797-0001	Documentation Kit (contents listed in this document)
2308268-0001	Documentation Kit Subscription/Update Service

Software Package contents are as follows:

Media:

Mag. Tape and Cartridge Tape (-0008,0026):
Part Number: 2270499-1301*K, TAPVOLM (executable from tape)
2270499-1601*K, DOCSVOLM (backup directory of disk)

8" DSDD Diskette (-0021):
Part Number: 2270499-1602*K, DOCSVOL1
2270499-1603*K, DOCSVOL2
2270499-1615*K, DOCSVOL3
2270499-1644*K, DOCSVOL4

Disk (-0022,0024,0025):
Part Number: 2270499-1601*K, DOCSVOLM

Documentation Kit:

	Description	Part Number/Rv	Qty
1.	Package Inventory (this document)	2261797-9901*L	1
2.	Index of 990 Unit Diagnostics and DOCS Systems Release 5.2.0	2270499-9935*K	1
3.	Release and Update Information, Release 5.2.0	2250541-9901*R	1
4.	Unit Diagnostics Handbook,		
	a. Volume 1, General Diagnostic Information	945400-9701*M	1
	b. Volume 2, Diagnostics for 990 Processors and Memories	945400-9702*L	1
	c. Volume 3, Diagnostics for 990 Mass Storage Devices	945400-9703*L	1
	d. Volume 4, Diagnostics for 990 Printers, Terminals, and Interface Modules	945400-9704*N	1
	e. Volume 5, Diagnostics for 990 Industrial Systems	945400-9705*D	1
	f. Volume 6, Diagnostics for 990 Communications Interfaces	945400-9706*H	1
	g. Volume 7, Diagnostics for 990 AMPL Systems	945400-9707*C	1

A.3 PACKAGE INVENTORY OF 990 UNIT DIAGNOSTICS FOR CASSETTE MEDIA

The package for the cassette media diagnostics consists of

PACKAGE INVENTORY

Unit Diagnostics - Cassette Kit
(Object)

Package Base Part Numbers:

License 2250719 includes Software Package 2250721*K

Media choices:

Media Type	Package Dash Number	Media Quantity
Cassette	-0001	41

The full package part number includes the base part number plus the dash number for the appropriate media.

EXAMPLE: 2250719-0001 is the full part number for the License.

Other related parts/services available on price list:

NONE	Additional Use Fee
NONE	Customer Support Line Service
2250720-0001	Software Package Subscription/Update Service
2250717-0001	Documentation Kit (contents listed in this document)
NONE	Documentation Kit Subscription/Update Service

Software Package contents are as follows:

Documentation Kit:

	Description	Part Number/Rv	Qty
1.	Package Inventory (this document)	2250717-9901*M	1
2.	Release and Update Information, Rel 5.2.0	2250541-9901*U	1
3.	Unit Diagnostics Handbook,		
a.	Volume 1, General Diagnostic Information	945400-9701*N	1
b.	Volume 2, Diagnostics for 990 Processors and Memories	945400-9702*M	1
c.	Volume 3, Diagnostics for 990 Mass Storage Devices	945400-9703*P	1
d.	Volume 4, Diagnostics for 990 Printers, Terminals, and Interface Modules	945400-9704*P	1
e.	Volume 5, Diagnostics for 990 Industrial Systems	945400-9705*D	1
f.	Volume 6, Diagnostics for 990 Communications Interfaces	945400-9706*J	1
g.	Volume 7, Diagnostics for 990 AMPL Systems	945400-9707*C	1

Media:

	Item	Cassette P/N,Rev	Cassette Contents Side A	-	Side B	Volume
C	1	2250501-0002-*W	UMXDOCS,158/84 *W	-	UMNDOCS,158/84 *W	1
C	2	2250501-0003-*W	UDBDOCS,158/84 *W	-	UBCDOCS,158/84 *W	1
C	3	225XXXX-0004-*W	ENETST,150/84 **	-	TNXTST,156/84 **	6-2
	4	2250527-0001-*C	CONVER,309/82 **	-	LLMTST,02/80 *A	1-6
C	5	2250502-0001-*L	CPU10,158/84 *A	-	MAPTST,03/79 **	2
	6	2250503-0001-*D	AU04TST,07/81 *H	-	AU04,07/81 *H	2
	7	2250504-0001-*K	AU05,060/83 *C	-	CTLTST,349/83 *A	2
	8	2250505-0002-*B	CRUTIC,024/84 *B	-	MAPTIC,024/84 *B	2
	9	2250506-0001-*B	RAM04,08/78 *A	-	MEMPRT,04/78 *C	2
C	10	2250525-0001-*L	CPU12A,157/84 *A	-	CPU12B,157/84 *A	2
C	11	2250525-0002-*L	CPU12C,157/84 *A	-	CPU12D,157/84 *A	2
C	12	2250525-0003-*L	CPU12E,157/84 *A	-	MAP12,052/83 *B	2
	13	2250526-0001-*B	HSTSLV,05/82 *B	-	TILCOU,05/78 *C	2
	14	2250531-0001-*C	MEMTST,346/83 *F	-	PRTIF,333/82 *A	2-4
	15	2239925-0001-**	DSKCD2,231/83 *A	-	TPBITS,10/82 *C	3
	16	2250521-0001-*G	FLPDSK,04/81 *G	-	RMTFLP,08/77	3
	17	2250522-0001-*G	DDFLOP,08/80 *E	-	WATDOG,03/81 *D	3-5
	18	2250523-0001-*H	DSKM3X,10/79 *C	-	TAPTST,074/83 *F	3
	19	2250524-0001-*G	DSKTRI,221/83 *E	-	DS10PD,04/80 *C	3
C	20	2239926-0001-*B	MTCTST1,278/83 *A	-	MTCTST2,110/84 *B	3
C	21	2250532-0001-*D	DSKCD1,231/83 *A	-	DSKCOM,117/84 *K	3
	22	2250533-0001-*C	DSKSA,294/83 *F	-	DSKWD8,220/83 *A	3
	23	2250592-0001-*C	DSKWD5,318/83 *C	-	WDFLPY,229/83 *A	3
	24	2250507-0001-*D	LPTEST,01/79 *A	-	TTYEIA,07/79 *B	4
C	25	2250508-0001-*L	LP810,108/84 *J	-	FLPTST,04/81 *D	4
	26	2250509-0001-*D	TST733,03/82 *B	-	CRDRDR,04/78 **	4
C	27	2250510-0001-*H	CRT911,086/84 *J	-	CRT913,07/78 *D	4
C	28	2250512-0002-*F	COM90X,158/84 *E	-	TST8X0,026/84 *D	6-4
	29	2250513-0001-*G	CRCOMM,05/82 *D	-	RMTEIA,08/78 **	6-4
	30	2250514-0001-*D	IO16,01/81 *C	-	CRUEXP,045/83 *F	4
C	31	2250530-0001-*L	LQPTST,07/81 *C	-	TST940,133/84 *H	4
C	32	2239927-0001-*B	CPTEST,142/84 *B	-	TST931,150/84 *B	4
	33	2250515-0001-*D	PROMPG,08/80 *D	-	EROMBT,08/80 *E	5
	34	2250518-0001-*B	ADCHK,02/78 *A	-	DACHK,02/78 *A	5
	35	2250519-0001-*B	INPMOD,05/81 *C	-	OUTMOD,05/81 *C	5
	36	2250520-0001-*C	FIVMOD,05/81 *C	-	TFMTST,05/81 *B	5
C	37	2250511-0001-*E	ACUTST,061/84 *H	-	EXTACU,047/83 *C	6
	38	2250528-0001-*J	FC3TST,031/83 *D	-	BCAIMT,231/83 *F	6
	39	2239928-0001-*A	MUXTST,300/83 *B	-	TLCPLR,280/83 **	6-2
	40	2250516-0001-*B	EMUTST,08/78 *A	-	TRACE,08/78 *A	7
	41	2250517-0001-*F	EMU900,04/80 *E	-	EMU940,03/79 *C	7
	42	2539097-0001-*N	DS990R,136/85			3

NOTE: Format for "Cassette Contents" for each side is:
TTTTTTT,DATE RR TTTTTT = Test Name
DATE RR = Test Version (RR=revision, and DATE=Month/Year or Day/Year)

C = NEW OR REVISED (ONE OR BOTH SIDES) THIS RELEASE.

A.4 PACKAGE INVENTORY OF 990 UNIT DIAGNOSTICS -- UNIVERSAL SYSTEMS TERMINAL (UST)

The following list defines the media and their part numbers for the Universal Systems Terminal (UST) diagnostics packages.

```
+-------------------------------------------------------------------+
|                                                                   |
|            Index of 990 Unit Diagnostics and DOCS Systems         |
|                for the Universal System Terminal (UST)            |
|                                                                   |
|                    Unit Diagnostics Release 5.2.0                 |
|                                                                   |
|                                                                   |
| Identification and Media Information:                             |
|                                                                   |
|     Name  -- Menu Name                                            |
|                                                                   |
|     IDENT -- Date (month/year) and revision                       |
|                                                                   |
|     VOL   -- Diagnostic Handbook volume number:                   |
|                                                                   |
|              1 = 945400-9701, General Diagnostic Information      |
|              2 = 945400-9702, Diagnostics for 990 Processors and  |
|                               Memories                            |
|              3 = 945400-9703, Diagnostics for 990 Mass Storage    |
|                               Devices                             |
|              4 = 945400-9704, Diagnostics for 990 Printers,       |
|                               Terminals, and Interface Modules    |
|              5 = 945400-9705, Diagnostics for 990 Industrial Systems |
|              6 = 945400-9706, Diagnostics for 990 Communications  |
|                               Interfaces                          |
|              7 = 945400-9707, Diagnostics for 990 AMPL Systems    |
|                                                                   |
|     Note: The UST/990 Unit Diagnostics Software kit               |
|           (part number 2534004-0001) contains Handbook            |
|            volume 1 only                                          |
|                                                                   |
|     T     -- Type code:                                           |
|              S = Stand-alone test                                 |
|              D = DOCS test                                        |
|              + = DOCS System part                                 |
|              U = Utility                                          |
|                                                                   |
|     DISKETTES --  3.5" (PRO-LITE)                                 |
|              USTDOCS1 2534005-0001, UTS/990 DIAGNOSTICS VOLUME1   |
|              USTDOCS2 2534005-0002, UTS/990 DIAGNOSTICS VOLUME2   |
|              USTDOCS3 2534005-0003, UTS/990 DIAGNOSTICS VOLUME3   |
|                                                                   |
+-----------------------------------+--------------+----------+-----+
```

UST diskettes contain Standard DOCS, but on a high-capacity microfloppy. Volume contents are as follows:

DISK # 1 Contains

Diagnostic Operational Control System (DOCS)

UBCDOCS	UDBDOCS	UMNDOCS	UMXDOCS

CPU Tests

AU04	CPU10	CPU12C	CRUTIC
AU04TST	CPU12A	CPU12D	MAPTIC
AU05	CPU12B	CPU12E	

Memory Tests

CTLTST	MAPTST	MEMTST	
MAP12	MEMPRT	RAM04	

Special Tests

CRDRDR	EMU940	ONPMOD	TFMTST
CRUEXP	EMUTST	I016	TNXTST
DACHK	EROMBT	OUTMOD	TRACE
EMU900	FIVMOD	PROMPG	WATDOG

DISK # 2 Contains

Printer Tests

CPTEST	LP810	LQPTST	TST8X0
FLPTST	LPTEST	PRTIF	

Terminal Tests

CRT911	TST733	TST940	
CRT913	TST931		

Communications Board Tests

ACUTST	CRCOMM	FC3TST	RMTEIA
BCAIMT	ENETST	LLMTST	TTYEIA
COM90X	EXTACU	MUXTST	

DISK # 3 Contains

Disk Drive Tests

DDFLOP	DSKCOM	DSKWD5	WDFLPY
DS10PD	DSKM3X	DSKWD8	DS990R
DSKCD1	DSKSA	FLPDSK	
DSKCD2	DSKTRI	RMTFLP	

Tape Drive Tests

MTCTST1	MTCTST2	TAPTST

TILINE Bus Tests

CONVER	TILCOU	TPBITS
HSTSLV	TLCPLR	

A.5 INDEX OF 990 UNIT DIAGNOSTICS AND DOCS SYSTEMS

The following information is an index of diagnostic tests and DOCS systems. It cross-references the identification information (name of the diagnostic test, volume in which the diagnostic test can be found, and last revision) and the media information (type of media on which the diagnostic is available). The media information excludes availability on cassette, since that information is listed with the package inventory for the cassette kit (see paragraph A.3).

You can use this index to tell which media are available for a particular diagnostic. You can then refer to the package inventory for the 990 Unit Diagnostics (see paragraph A.2) to determine the part number for the media on which the diagnostic is available. For example, this index indicates that the AU04 diagnostic is available on the following media: DOCSVOL1, DOCSVOLM, and TAPVOLM.

You can also use this index to determine which diagnostic tests are contained on specific media. For example, to determine which tests are contained on the DOCSVOL2 media, look at the column entries under DOCSVOL2.

```
+-----------------------------------------------------------------+
|                                                                 |
|            Index of 990 Unit Diagnostics and DOCS Systems       |
|                                                                 |
|                            Release 5.2.0                        |
|                                                                 |
|                                                                 |
| Identification and Media Information:                           |
|                                                                 |
|     Name  -- Menu Name                                          |
|                                                                 |
|     IDENT -- Date (month/year) and revision                     |
|                                                                 |
|     VOL   -- Diagnostic Handbook volume number:                 |
|              1 = 945400-9701, General Diagnostic Information    |
|              2 = 945400-9702, Diagnostics for 990 Processors and|
|                               Memories                          |
|              3 = 945400-9703, Diagnostics for 990 Mass Storage  |
|                               Devices                           |
|              4 = 945400-9704, Diagnostics for 990 Printers,     |
|                               Terminals, and Interface Modules  |
|              5 = 945400-9705, Diagnostics for 990 Industrial Systems
|              6 = 945400-9706, Diagnostics for 990 Communications|
|                               Interfaces                        |
|              7 = 945400-9707, Diagnostics for 990 AMPL Systems  |
|                                                                 |
|     T     -- Type code:                                         |
|              S = Stand-alone test                               |
|              D = DOCS test                                      |
|              + = DOCS System part                               |
|              U = Utility                                        |
|                                                                 |
|     DISKETTES --  8" DSDD                                       |
|              DOCSVOL1 = 2270499-1602, DSDD Volume 1, Sys = DDNDOCS
|              DOCSVOL2 = 2270499-1603, DSDD Volume 2, Sys = DDNDOCS
|              DOCSVOL3 = 2270499-1615, DSDD Volume 3, Sys = DDNDOCS
|              DOCSVOL4 = 2270499-1644, DSDD Volume 4, Sys = DDNDOCS
|                                                                 |
|     DISK/MAG. TAPE (*) --                                       |
|              DOCSVOLM = 2270499-1601, Disk/Mag. Tape (BD),      |
|                           System = DDNDOCS                      |
|              TAPVOLM  = 2270499-1301, Mag. Tape, Sys = TDNDOCS  |
|                                                                 |
|              (*) Mag. Tape media types available:               |
|                   800 BPI                                       |
|                   1600 BPI                                      |
|                   Cartridge Tape                                |
+-----------------------------------+-----------+---------+-------+
```

IDENTIFICATION INFORMATION				MEDIA INFORMATION						
				DISKETTES				DISK/MAG.		NEW OR
VOL	T	Name	IDENT	DOCS VOL1	DOCS VOL2	DOCS VOL3	DOCS VOL4	DOCS VOLM	TAP VOLM	REV'D 5.2.0
1	+	BCSEDT	266/85 *Z	X	X	X	X	X	X	X
1	+	BCSRUN	266/85 *Z	X	X	X	X	X	X	X
1	+	DBVERB1	266/85 *Z	X	X	X	X	X	X	X
1	+	DDNDOCS	266/85 *Z	X	X	X	X	X		X
1	+	FBCDOCS	266/85 *Z				X	X		X
1	+	FDBDOCS	266/85 *Z				X	X		X
1	+	FMNDOCS	266/85 *Z				X	X		X
1	+	FMXDOCS	266/85 *Z				X	X		X
1	+	MXVERB1	266/85 *Z	X	X	X	X	X	X	X
1	+	MXVERB2	266/85 *Z	X	X	X	X	X	X	X
1	+	S200DOCS	096/83 *T				X	X		
1	+	TDNDOCS	266/85 *Z						X	X
1	+	UBCDOCS	266/85 *Z				X	X		X
1	+	UDBDOCS	266/85 *Z				X	X		X
1	+	UMNDOCS	266/85 *Z				X	X		X
1	+	UMXDOCS	266/85 *Z				X	X		X
1	U	CONVER	309/82 **	X				X	X	
1	U	FFCOPY	05/78 **			X		X	X	
2	S	AU04	07/81 *H	X				X	X	
2	S	AU04TST	07/81 *H	X				X	X	
2	S	AU05	060/83 *C	X				X	X	
2	S	CPU10	123/85 *B	X				X	X	X
2	S	CPU12A	202/84 *B	X				X	X	
2	S	CPU12B	202/84 *B	X				X	X	
2	S	CPU12C	202/84 *B	X				X	X	
2	S	CPU12D	202/84 *B	X				X	X	
2	S	CPU12E	202/84 *B	X				X	X	
2	D	AU200	08/82 *A	X				X		
2	D	CRUTIC	089/85 *C	X				X	X	X
2	D	CTLTST	142/85 *B	X				X	X	X
2	D	HSTSLV	05/82 *B	X				X	X	
2	D	MAP12	052/83 *B	X				X	X	
2	D	MAPTIC	135/85 *C	X				X	X	X
2	S	MAPTST	03/79 **	X				X	X	
2	D	MEMPRT	04/78 *C	X				X	X	
2	D	MEMTST	208/85 *G	X				X	X	X
2	D	RAM04	08/78 *A	X				X	X	
2	D	TILCOU	05/78 *C	X				X	X	
2	D	TLCPLR	280/83 **	X				X	X	
2	D	TNXTST	156/84 **	X				X	X	

IDENTIFICATION INFORMATION				MEDIA INFORMATION						
				DISKETTES				DISK/MAG.		NEW OR REV'D
VOL	T	Name	IDENT	DOCS VOL1	DOCS VOL2	DOCS VOL3	DOCS VOL4	DOCS VOLM	TAP VOLM	5.2.0
3	D	DDFLOP	181/84 *F		X			X	X	
3	D	DS10PD	070/85 *D		X			X	X	
3	D	DS990R	136/85 *N		X			X	X	X
3	D	DSKCD1	101/85 *D		X			X	X	X
3	D	DSKCD2	067/85 *C		X			X	X	
3	D	DSKCOM	008/85 *M		X			X	X	
3	D	DSKM3X	10/79 *C		X			X	X	
3	D	DSKSA	193/85 *L		X			X	X	X
3	D	DSKTRI	221/83 *E		X			X	X	
3	D	DSKWD5	193/84 *D		X			X	X	
3	D	DSKWD8	160/85 *C		X			X	X	X
3	D	FLPDSK	04/81 *G		X			X	X	
3	D	MTCTST1	278/83 *A		X			X	X	
3	D	MTCTST2	075/85 *D		X			X	X	X
3	D	RMTFLP	08/77		X			X	X	
3	D	TAPTST	333/84 *H		X			X	X	
3	D	TPBITS	10/82 *C		X			X	X	
3	D	WDFLPY	229/83 *A		X			X	X	
4	D	CPTEST	056/85 *D			X		X	X	X
4	D	CRDRDR	004/78 **			X		X	X	
4	D	CRT911	086/84 *J			X		X	X	
4	D	CRT913	007/78 *D			X		X	X	
4	D	CRUEXP	045/83 *F			X		X	X	
4	D	FLPTST	004/81 *D			X		X	X	
4	D	I016	001/81 *C			X		X	X	
4	D	LP810	108/84 *J			X		X	X	
4	D	LPTEST	01/79 *A			X		X	X	
4	D	LQPTST	07/81 *C			X		X	X	
4	D	PRTIF	333/82 *A			X		X	X	
4	D	RMTEIA	08/78 **			X		X	X	
4	D	TST733	03/82 *B			X		X	X	
4	D	TST8X0	026/84 *D			X		X	X	
4	D	TST931	221/84 *C			X		X	X	
4	D	TST940	133/84 *H			X		X	X	
4	D	TTYEIA	07/79 *B			X		X	X	

IDENTIFICATION INFORMATION				MEDIA INFORMATION						
				DISKETTES				DISK/MAG.		NEW OR
VOL	T	Name	IDENT	DOCS VOL1	DOCS VOL2	DOCS VOL3	DOCS VOL4	DOCS VOLM	TAP VOLM	REV'D 5.2.0
5	D	ADCHK	02/78 *A				X	X	X	
5	D	DACHK	02/78 *A				X	X	X	
5	D	EROMBT	08/80 *E				X	X	X	
5	D	FIVMOD	05/81 *C				X	X	X	
5	D	INPMOD	05/81 *C				X	X	X	
5	D	OUTMOD	05/81 *C				X	X	X	
5	D	PROMPG	08/80 *D				X	X	X	
5	D	TFMTST	05/81 *B				X	X	X	
5	D	WATDOG	03/81 *D				X	X	X	
6	D	ACUTST	061/84 *H			X		X	X	
6	D	BCAIMT	231/83 *F			X		X	X	
6	D	COM90X	202/84 *F			X		X	X	
6	D	CRCOMM	05/82 *D			X		X	X	
6	D	EXTACU	047/83 *C			X		X	X	
6	D	FC3TST	031/83 *D			X		X	X	
6	D	LLMTST	02/80 *A			X		X	X	
6	D	MUXTST	214/84 *C			X		X	X	
6	D	ENETST	150/84 **			X		X	X	
7	D	EMU900	04/80 *E	X				X	X	
7	D	EMU940	03/79 *C	X				X	X	
7	D	EMUTST	08/78 *A	X				X	X	
7	D	TRACE	08/78 *A	X				X	X	

IDENTIFICATION INFORMATION				MEDIA INFORMATION - 3.5" DISKETTES				
VOL	T	Name	IDENT	USTDOCS1	USTDOCS2	USTDOCS3		NEW OR REV'D FOR 5.2.0
1	+	UBCDOCS	266/85 *Z	X				X
1	+	UDBDOCS	266/85 *Z	X				X
1	+	UMNDOCS	266/85 *Z	X				X
1	+	UMXDOCS	266/85 *Z	X				X
1	+	UNITLDR	266/85 *Z	X				X
5	D	ADCHK	02/78 *A	X				
2	S	AU04	07/81 *H	X				
2	S	AU04TST	07/81 *H	X				
2	S	AU05	060/83 *C	X				
2	S	CPU10	123/85 *B	X				X
2	S	CPU12A	202/84 *B	X				
2	S	CPU12B	202/84 *B	X				
2	S	CPU12C	202/84 *B	X				
2	S	CPU12D	202/84 *B	X				
2	S	CPU12E	202/84 *B	X				
4	D	CRDRDR	004/78 **	X				
4	D	CRUEXP	045/83 *F	X				
2	D	CRUTIC	089/85 *C	X				X
2	D	CTLTST	142/85 *B	X				X
5	D	DACHK	02/78 *A	X				
7	D	EMU900	04/80 *E	X				
7	D	EMU940	03/79 *C	X				
7	D	EMUTST	08/79 *A	X				
5	D	EROMBT	08/80 *E	X				
5	D	FIVMOD	05/81 *C	X				
5	D	INPMOD	05/81 *C	X				
4	D	IO16	01/81 *C	X				
2	D	MAP12	052/83 *B	X				
2	D	MAPTIC	135/85 *C	X				X
2	S	MAPTST	03/79 **	X				
2	D	MEMPRT	04/78 *C	X				
2	D	MEMTST	208/85 *G	X				X
5	D	OUTMOD	05/81 *C	X				
5	D	PROMPG	08/80 *D	X				
2	D	RAM04	08/78 *A	X				
5	D	TFMTST	05/81 *B	X				
2	D	TNXTST	156/84 **	X				
7	D	TRACE	08/78 *A	X				
5	D	WATDOG	03/81 *D	X				
6	D	ACUTST	061/84 *H		X			
6	D	BCAIMT	231/83 *F		X			
6	D	COM90X	202/84 *F		X			

IDENTIFICATION INFORMATION				MEDIA INFORMATION - 3.5" DISKETTES				
VOL	T	Name	IDENT	USTDOCS1	USTDOCS2	USTDOCS3		NEW OR REV'D FOR 5.2.0
4	D	CPTEST	056/85 *D		X			X
6	D	CRCOMM	05/82 *D		X			
4	D	CRT911	086/84 *J		X			
4	D	CRT913	07/78 *D		X			
6	D	ENETST	150/84 **		X			
6	D	EXTACU	047/83 *C		X			
6	D	FC3TST	031/83 *D		X			
4	D	FLPTST	04/81 *D		X			
6	D	LLMTST	02/80 *A		X			
4	D	LP810	108/84 *J		X			
4	D	LPTEST	01/79 *A		X			
4	D	LQPTST	07/81 *C		X			
6	D	MUXTST	214/84 *C		X			
4	D	PRTIF	333/82 *A		X			
4	D	RMTEIA	08/78 **		X			
4	D	TST733	03/82 *B		X			
4	D	TST8X0	026/84 *D		X			
4	D	TST931	221/84 *C		X			
4	D	TST940	133/84 *H		X			
4	D	TTYEIA	07/79 *B		X			
1	U	CONVER	309/82 **			X		
3	D	DDFLOP	181/84 *F			X		
3	D	DS10PD	070/85 *D			X		
3	D	DS990R	136/85 *N			X		X
3	D	DSKCD1	101/85 *D			X		X
3	D	DSKCD2	067/85 *C			X		
3	D	DSKCOM	008/85 *M			X		
3	D	DSKM3X	10/79 *C			X		
3	D	DSKSA	193/85 *L			X		X
3	D	DSKTRI	221/83 *E			X		
3	D	DSKWD5	193/84 *D			X		
3	D	DSKWD8	160/85 *C			X		X
3	D	FLPDSK	04/81 *G			X		
2	D	HSTSLV	05/82 *B			X		
3	D	MTCTST1	278/83 *A			X		
3	D	MTCTST2	075/85 *D			X		X
3	D	RMTFLP	08/77			X		
3	D	TAPTST	333/84 *H			X		
2	D	TILCOU	05/78 *C			X		
2	D	TLCPLR	280/83 **			X		
3	D	TPBITS	10/82 *C			X		
3	D	WDFLPY	229/83 *A			X		

Appendix B

Operation of the 990 Maintenance Diagnostic Unit

B.1 INTRODUCTION

This appendix describes the operation of the 990 maintenance diagnostic unit (MDU), which loads diagnostics into a system when no other diagnostic loader device is available.

B.2 DESCRIPTION

The MDU contains a 990 programmer panel and a 733 ASR/KSR tape cassette transport housed in a portable aluminum carrying case (see Figure B-1). This unit contains a power supply and the electronics necessary to control the tape transport and interface to the Model 990 Computer. The 990 maintenance unit has storage space for manuals, CPU ROMs, and interface cables. A special CRU adapter board is available, which allows the 990 maintenance unit to communicate with 960 systems via the CRU interface.

B.2.1 Test Configurations

The MDU accomodates two different test configurations, depending on the system under test. In the first configuration (Figure B-2), the MDU ties into the system under test through the operator or programmer panel interface board (after removal of the existing interface cable from the chassis-mounted operator or programmer panel). In this configuration, diagnostic tests are loaded into program memory from the cassette tape transport on the maintenance unit and the test results are displayed on the data LEDs on the programmer panel in the maintenance unit. This configuration is used when the system is not equipped with an operational 733 ASR data terminal.

B.2.1.1 Stand-Alone Programmer Panel. The programmer panel may also be detached from the MDU and used as a stand-alone unit. In this case, the existing interface cable from the chassis-mounted operator or programmer panel is removed from P7 on the interface board, and the programmer panel interface cable of the MDU is connected to P7 (see Figure B-3). In this configuration, the key switch on the chassis-mounted panel controls the application of AC power to the computer, but all other functions are controlled by the stand-alone programmer panel.

This configuration is used when the system under test is equipped with an operator panel or inoperative programmer panel but contains an operative 733 ASR data terminal. In this case, the diagnostic cassettes are loaded into the system from the 733 ASR and the programmer panel displays test data from selected memory locations on any of the boards containing memory storage.

B.2.2 Operating Controls and Indicators

The operating controls and indicators of the MDU are shown in Figure B-4 and listed and described in Table B-1. Basically, the MDU contains a POWER ON/OFF switch that controls power to the maintenance unit, a RESET switch that initializes the controller board in the maintenance unit, a REWIND switch that rewinds cassette tapes, a LOAD switch that initiates diagnostic load operations, and the conventional controls and indicators found on the 990 programmer panel. However, the functions of three of the programmer panel controls and indicators are slightly different when the programmer panel is used as a part of the MDU. These differences include:

* Key switch -- Does not affect AC power to the computer or MDU, but otherwise exercises the same key control over program intervention in the computer.

* Power LED -- Indicates the status of the power supply in the MDU rather than the power supply in the computer chassis.

* Fault LED -- May be lit by either the computer (self-test failure) or by the maintenance controller board in the MDU in the event of a faulty tape read operation.

The functions of all other programmer panel controls and indicators are the same as those of a conventional chassis-mounted programmer panel (see Figure B-4 and Table B-1).

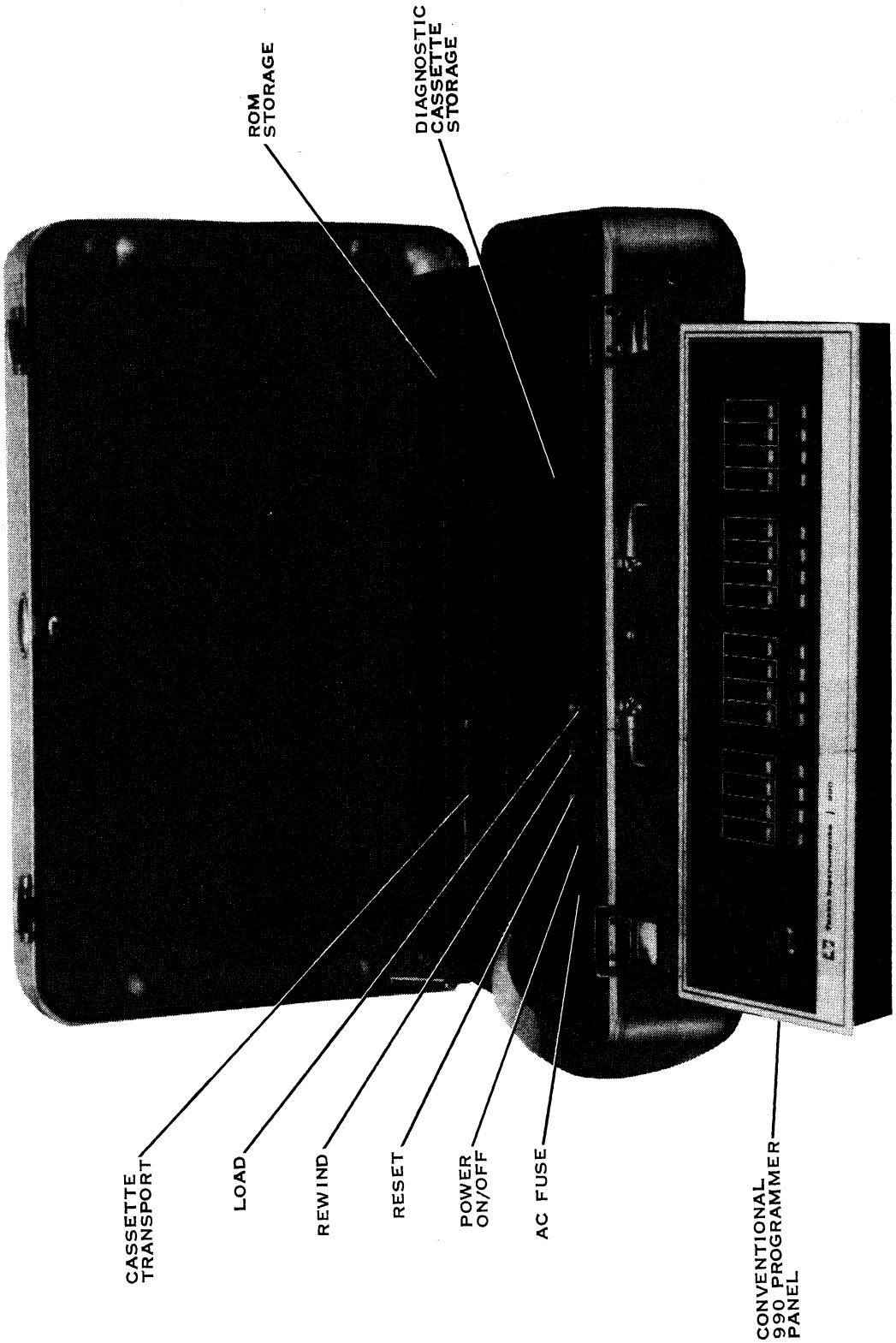


Figure B-1 990 Maintenance Diagnostic Unit

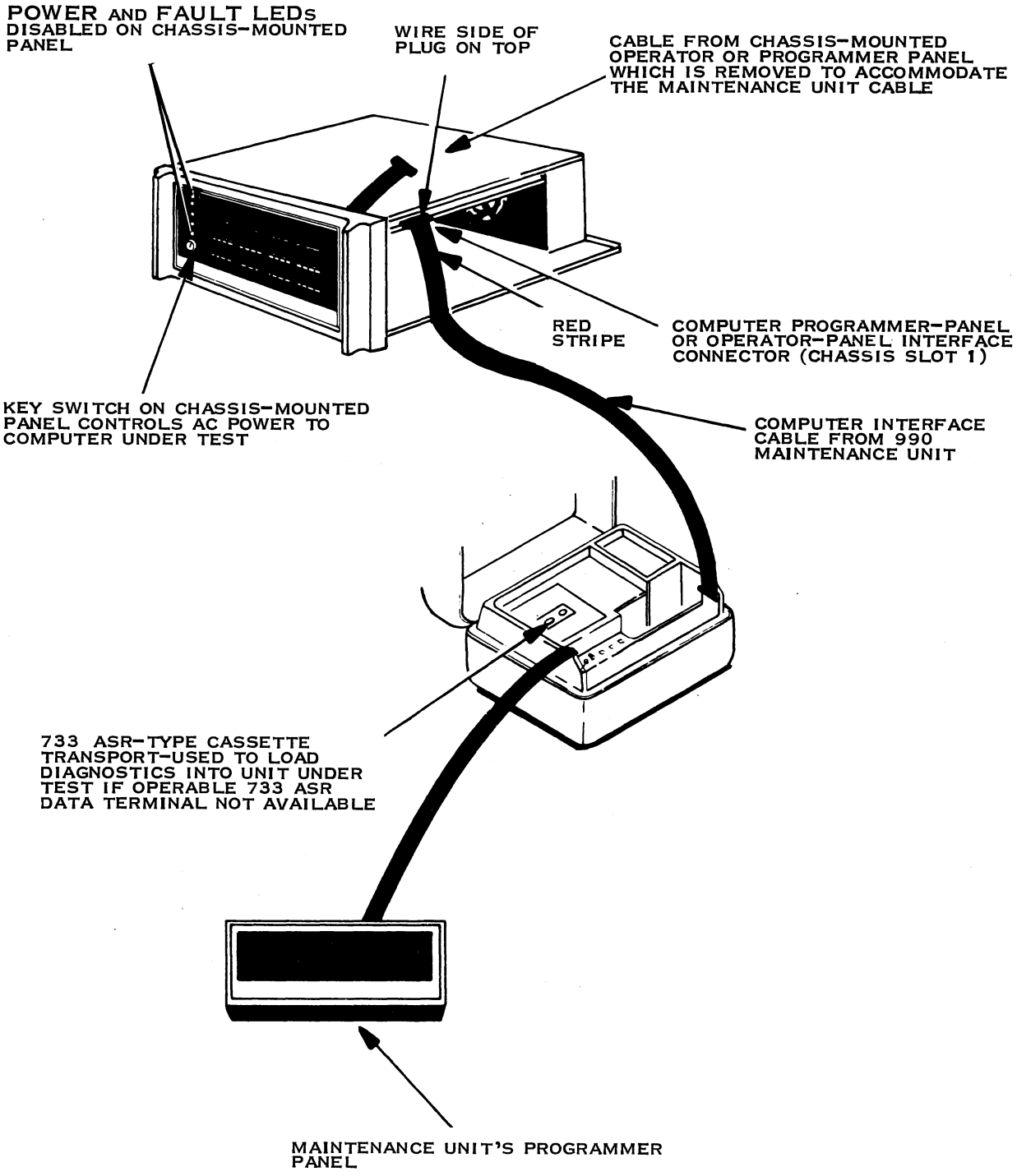


Figure B-2 Test Setup for System Without 733 ASR Data Terminal

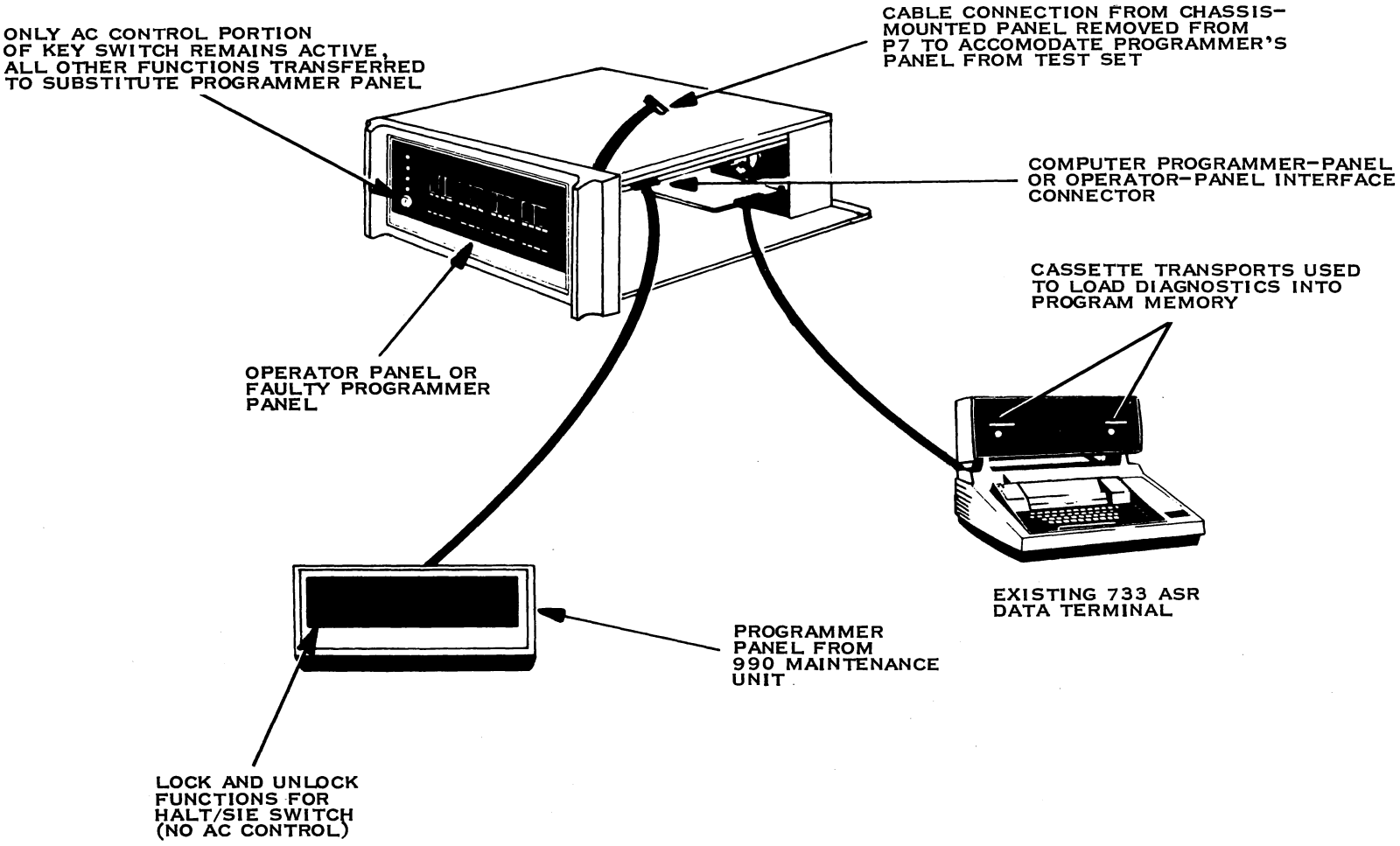
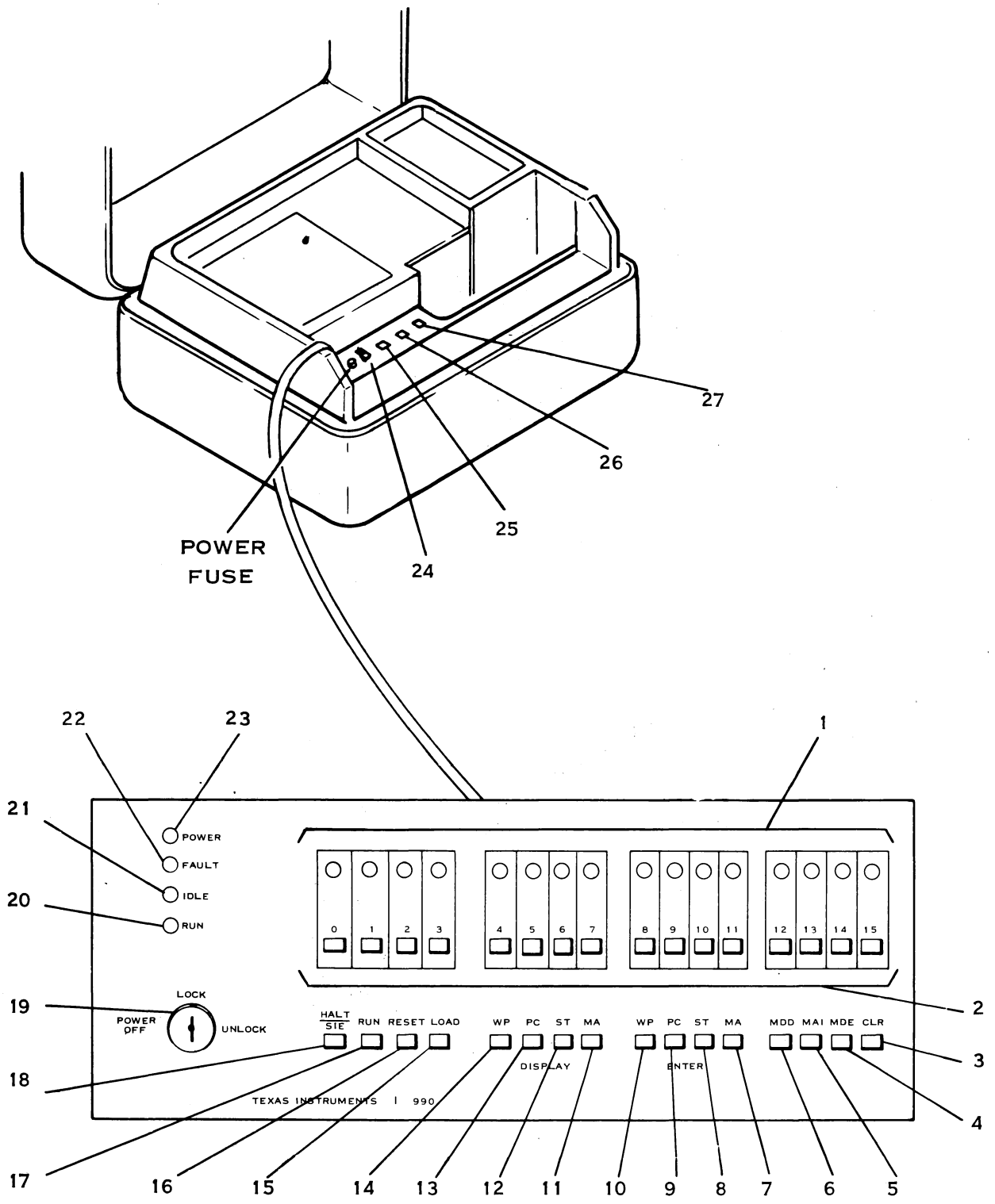


Figure B-3 Alternate Test Setup Using Programmer Panel Only



2280705

Figure B-4 MDU Controls and Indicators

Table B-1 MDU Controls and Indicators

Reference Number	Control or Indicator	Function
1	Data LEDs	Display data being entered into a CPU register or memory location. During test execution, error numbers are displayed in hexadecimal format on the right byte. The left byte is forced to all ones. An LED that is lit represents a logic one. An LED that is not lit represents a logic zero. The least significant bit is displayed on the far right of the LEDs. In the HALT mode, the LEDs display a computer register contents, memory contents, or value entered into memory via the data entry switches, depending on which switches are pressed (see references 5, 7, 9, 11, 13, and 15).
2	Data entry switches	Used in conjunction with the ENTRY switches on the panel to enter data and addresses into selected computer registers and memory locations (active only when the panel is in the HALT mode). In the HALT mode, the data LED located immediately above each data entry switch lights as each switch is pressed. The value indicated by the data LEDs is then stored in the register or memory address selected by the entry switches.
3	CLR	Clears the data LED displays.
4	MDE	Transfers a value displayed on the data LEDs to the memory location defined by the contents of the memory address (MA) register in the computer.
5	MAI	Increments the value stored in the memory address register by 2.

Table B-1 MDU Controls and Indicators (Continued)

Reference Number	Control or Indicator	Function
6	MDD	Causes the contents of the memory location defined by the contents of the memory address register to be displayed on the data LEDs.
7	ENTER MA	Causes the value displayed by the data LEDs to be entered into the memory address register of the computer.
8	ENTER ST	Enters the value displayed on the data LEDs into the status register of the computer.
9	ENTER PC	Loads the value displayed on the data LEDs into the program counter of the computer.
10	ENTER WP	Loads the value displayed on the data LEDs into the workspace pointer of the computer.
11	DISPLAY MA	Displays the value stored in the memory address register on the data LEDs.
12	DISPLAY ST	Displays the contents of the status register on the data LEDS.
13	DISPLAY PC	Displays the contents of the program counter on the data LEDS.
14	DISPLAY WP	Displays the contents of the workspace pointer register on the data LEDs.
15	LOAD switch	Causes the computer to trap to the ROM loader.
16	RESET	Results in an IDRESET: the generation of a pulse that resets all units in the system.
17	RUN	When the computer is halted (programmer panel is active), pressing RUN returns the computer to the RUN mode and deactivates the panel.

Table B-1 MDU Controls and Indicators (Continued)

Reference Number	Control or Indicator	Function
18	HALT/SIE	When the computer is in the run mode (RUN LED is lit), pressing HALT/SIE causes the computer to halt and begin processing the front panel software if the key switch is set to the UNLOCK position. Pressing the switch when the computer is not in the run mode causes the computer to execute a single instruction at the present program counter address. The contents of the program counter are incremented by two and displayed on the data LEDs.
19	Key switch (OFF/LOCK /UNLOCK)	Prevents unauthorized computer program intervention. The key must be inserted into the switch and turned to the UNLOCK position in order to send the HALT/SIE output to the computer. The AC power control function for the 990 computer is controlled by the key switch on the chassis-mounted operator or programmer panel.
20	RUN LED	Lights when a low-active run signal is generated by the computer, indicating that the computer is in run mode. When this LED is lit, all switches on the panel (except HALT/SIE) are disabled and the data LEDs are driven under program control.
21	IDLE LED	Lights when the computer is executing an idle instruction, which indicates computer inactivity for most interrupt-driven software.
22	FAULT LED	Lights when the computer has detected a self-test diagnostic failure or the MDU has detected a tape data fault.

Table B-1 MDU Controls and Indicators (Continued)

Reference Number	Control or Indicator	Function
23	POWER LED	Lights when POWER ON/OFF on the MDU is set to the ON position and the MDU internal power supply is functioning properly.
24	POWER ON/OFF	Controls AC power to the MDU.
25	RESET	Initializes the logic on the maintenance controller board of the MDU. Has no effect on the system under test.
26	REWIND	Causes the cassette tape in the cassette transport to rewind to the beginning of the tape (tape motion stops when clear leader is sensed).
27	LOAD	Causes the program on cassette tape to be loaded into memory. When the load is complete, test execution begins. When LOAD is pressed, it clears CRU output bit E (bit E=0) before the test is loaded into memory.

B.2.2.1 MDU Operating Procedures. Some of the more common operating procedures are briefly described in the following paragraphs. These procedures include:

* Changing the panel mode of operation
* Mounting cassette tapes in the MDU
* Loading a diagnostic from the cassette transport in the MDU
* Entering data into CPU registers or memory locations
* Displaying data from CPU registers or memory locations
* Single instruction execution

--CHANGING PANEL MODE OF OPERATION

The programmer panel in the 990 MDU can be operated in one of two modes: run and halt. The computer initially comes up in run mode when AC power is applied through the key switch on the chassis-mounted front panel. During this time, the RUN LED and all data LEDs on the programmer panel in the MDU light and remain lit until the mode of operation changes. If the key switch on the programmer panel in the MDU is set to the LOCK position, all controls on the panel are disabled.

To change to halt mode, insert the key in the switch and turn to the UNLOCK position. Only HALT/SIE on the panel is enabled. When you press HALT/SIE, the computer ceases normal test execution and traps to the panel software utility, which is located in ROM. The RUN LED on the programmer panel then goes out, and the outputs of the programmer panel switches are constantly monitored by software through the programmer panel CRU-type interface. At this point, the panel is operating in halt mode.

In halt mode, a diagnostic tape can be loaded from the MDU or information may be entered into or displayed from selected CPU registers or program memory locations.

To switch from halt to run mode, press RUN on the programmer panel. This causes the CPU on the processor board to begin program execution at the memory address indicated by its program counter.

--MOUNTING AND REMOVING CASSETTES

To load a diagnostic cassette into the MDU, open the cassette transport door and insert the cassette with the tape end up. The title of the diagnostic to be run should be facing you, as shown in Figure B-5. Press the cassette firmly into the transport so that the capstan and reel motors properly engage the cassette tape and reels. Close the transport door to complete installation of the cassette.

To remove a cassette from the transport, open the door to the first stop and then open it the rest of the way using a quick downward motion. This ejects the cassette from the tape transport. Keep the transport door closed when not in use to prevent accumulation of dust or dirt in the tape drive mechanism and read head.

--LOADING DOCS AND DIAGNOSTICS INTO PROGRAM MEMORY

DOCS, for cassettes, is divided into two files. The first file is the DOCS cassette loader (UNITLDR) and is a standard 990 ASC file. The second file is DOCS itself (UMNDOCS, UMXDOCS, UBCDOCS or UDBDOCS), and is a compressed file. The ROM loader loads up the DOCS loader (UNITLDR) and the UNITLDR loads up the version of DOCS that you are using.

Before initiating a load from cassette, set the POWER ON/OFF switch to the ON position and fully rewind the tape by pressing the REWIND switch on the MDU (refer to item 26 in Table B-1). The tape motion stops automatically when the clear leader at the beginning of the tape is sensed by the maintenance controller board.

Set the KEY switch on the 990 computer chassis-mounted operator or programmer panel to one of the ON positions (ON, LOCK, or UNLOCK) before applying AC power to the computer. At this point, the RUN and POWER LEDs light on the programmer panel in the MDU. Change the panel operation mode to halt by setting the KEY switch on the programmer panel to to the UNLOCK position and pressing HALT/SIE. The RUN LED on the panel will go out, indicating that the CPU is now processing the panel software utility. At this point, the panel software begins examining the switch outputs from the programmer panel. The DOCS UNITLDR ASCII load operation initiated by pressing LOAD (refer to item 27 in Table B-1) on the maintenance unit now begins. The panel software recognizes the combination of the load signal with the maintenance unit present signal as a load from MDU cassette command (refer to the flowchart in Figure B-6). As a result, the panel software branches to the self-test program (if the loader ROM is equipped with self-test). If the self-test fails to execute correctly, the CPU lights the FAULT LED on the programmer panel in the MDU and inhibits the DOCS UNITLDR load operation.

However, if the self-test executes satisfactorily (or if a self-test program is not present), the CPU branches to the ASR loader program, which is also stored in ROM. Since the load signal occurred with the maintenance unit present, the software retains the CRU base address of the front panel, and the DOCS UNITLDR load operation starts.

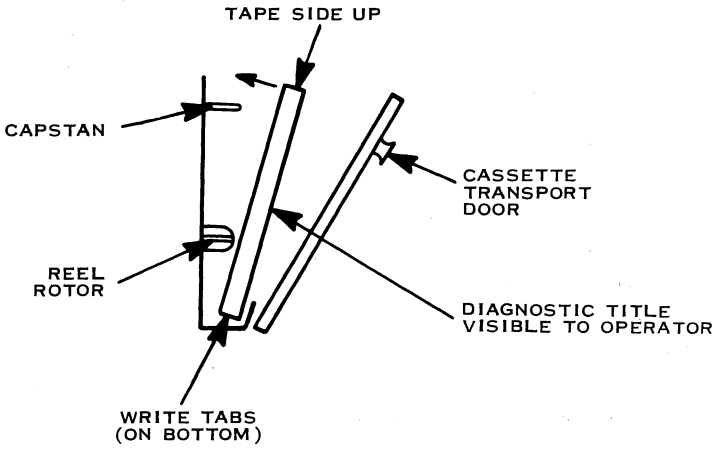
At this point, the MDU waits to go into run mode and then begins transmitting tape read data over the programmer panel CRU interface under control of the 733 loader program. Each time the maintenance controller board in the MDU has a byte of data ready for the computer, it sets CRU bit 12, which is interpreted by software as a read request. When the 733 loader recognizes the read request, it serially transfers the next byte of data from the MDU and resets CRU bit 12, freeing the MDU to ready the next data byte. If the MDU has another byte of data available before the loader has accepted the previous data byte (CRU bit 12 is still set to the computer), the load operation is aborted and the FAULT LED lights on the programmer panel of the maintenance unit. When the 733 loader decodes an end-of-file tag, it sets clear write request via CRU output bit 13, which causes the MDU to stop the cassette transport and return to an idle state.

After the ROM loader loads up the DOCS UNITLDR file, control is transferred to this DOCS loader. You will now need to press again RESET and LOAD on the MDU in order to start the loading of DOCS by the DOCS Unit Loader.

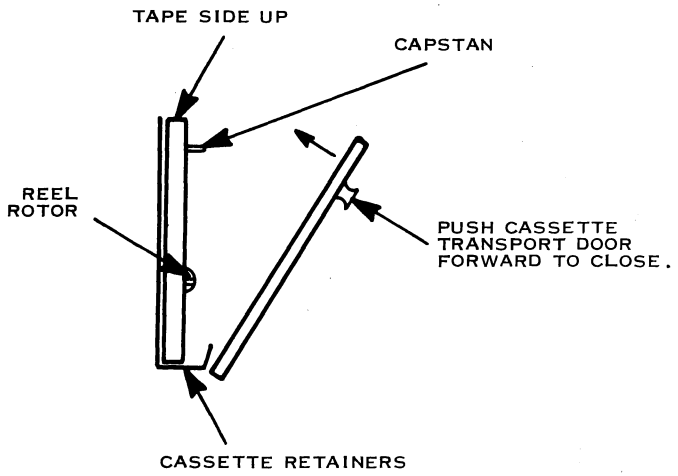
On the 990/5 CPU you will need to hit RESET and LOAD twice on the MDU. This is because the 990/5 ROM Loader does not finish reading "extra" data on the last record. Therefore, the first RESET and LOAD merely positions the cassette correctly.

DOCS is now being loaded by its own loader (UNITLDR). When the load is over the front panel will cease to be updated and you can hit "Y" on your chosen interactive terminal.

The loading of diagnostics is straight forward. All you need do is execute the .LD (Load Diagnostic) verb and DOCS will prompt you for the test load bias, and then direct you to insert the cassette and press <RETURN> and <REWIND> on the MDU. Once you have done this, DOCS will ask you to hit LOAD on the MDU to start the diagnostic load procedure.



STEP 1. CASSETTE IS PROPERLY ORIENTED AND ALIGNED IN TRANSPORT



STEP 2. CASSETTE IS PRESSED INTO POSITION AND DOOR IS CLOSED.

2283761

Figure B-5 Cassette Installation, Simplified Diagram

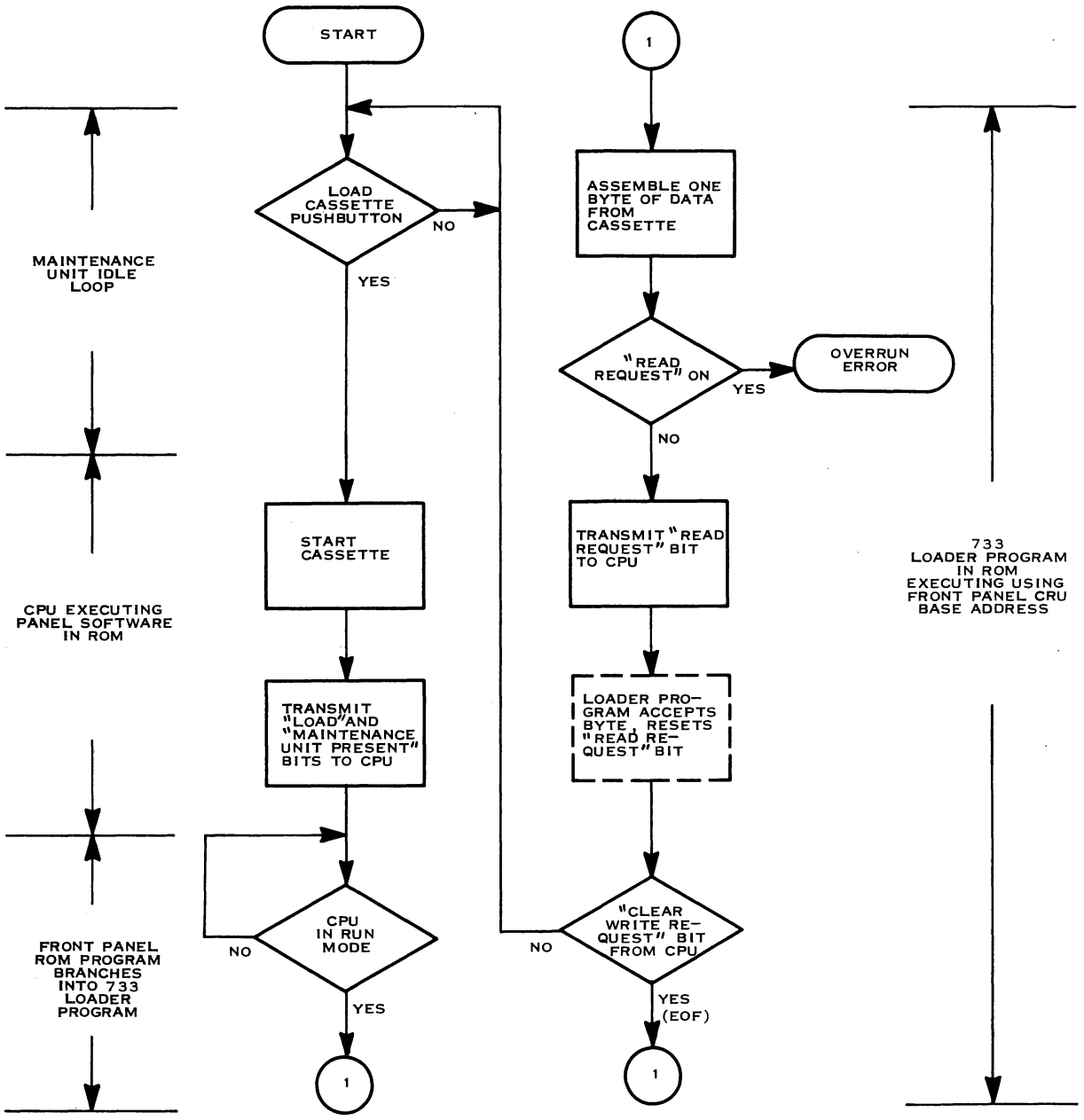


Figure B-6 Flowchart of Diagnostic Load from MDU

Appendix C

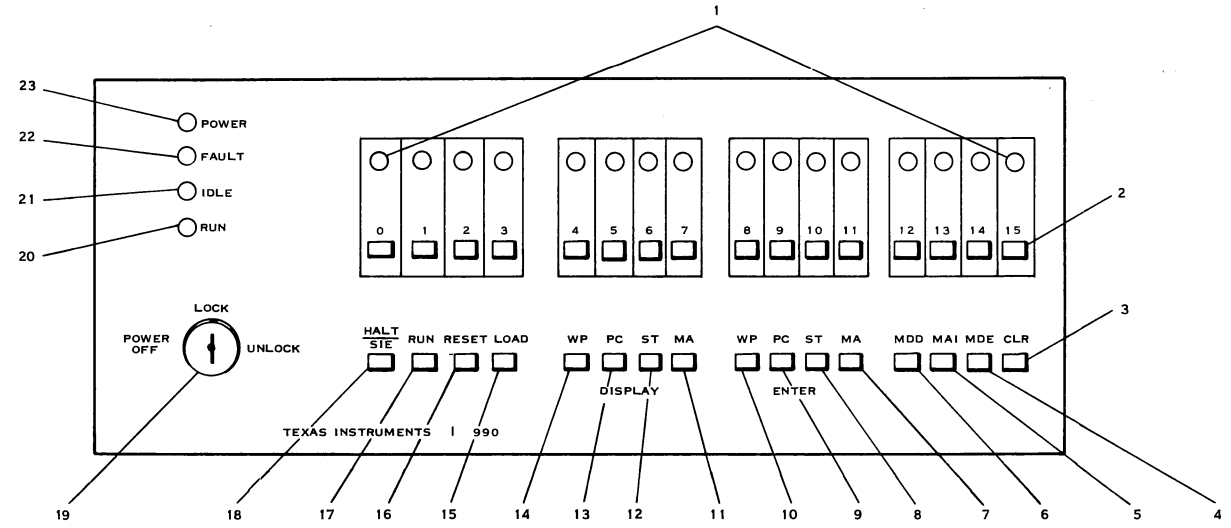
Operation of the 990 Programmer Panel

C.1 INTRODUCTION

The programmer panel controls and indicators are shown in Figure C-1 and listed and described in Table C-1. Basically, the panel functions in one of two modes: run or halt.

When power is initially applied to the computer, the system comes up in the run mode, which locks out the programmer controls. In this mode of operation, the RUN LED and all data LEDs on the panel light and remain lit.

In order to set the programmer panel to halt mode, insert the key in the key switch and turn to the UNLOCK position. Press HALT/SIE to halt the computer and activate the controls on the programmer panel. To return the panel to run mode, press RUN.



2276920

Figure C-1 990 Programmer Panel Controls and Indicators

Table C-1 990 Programmer Panel Controls and Indicators

Reference Number	Control or Indicator	Function
1	Data LEDs	In the run mode of operation, all data LEDs light except when the computer halts. At this point, the contents of the program counter of the CPU are displayed. A lighted LED denotes logic 1, an extinguished indicator denotes logic 0. A data LED changes to the opposite value when the switch is pressed. The least significant bit is displayed on the far right of the LEDs. In halt mode, the LEDs display a computer register contents, memory contents, or a value entered into computer memory via the data entry switches, depending on which switches are pressed (see references 5, 7, 9, 11, 13, and 15).
2	Data entry switches	Used in conjunction with the ENTRY switches on the panel, these switches enter data and addresses into selected computer registers and memory locations only when the panel is in halt mode. In halt mode, the data LED located immediately above each data entry switch lights as each switch is pressed. The value indicated by the data LEDs is then stored in the register or memory address selected by the entry switches.
3	CLR	When CLR is pressed, the data LED displays are cleared.

Table C-1 990 Programmer Panel Controls and Indicators (Continued)

Reference Number	Control or Indicator	Function
4	MDE	When MDE is pressed, the value displayed on the data LEDs is transferred to the memory location defined by the contents of the memory address (MA) register in the computer.
5	MAI	The memory address increment (MAI) switch increments the value stored in the memory address register of the CPU by two.
6	MDD	When MDD is pressed, the contents of the memory location defined by the contents of the memory address register is displayed on the data LEDs.
7	ENTER MA	When MA is pressed, the value displayed by the data LEDs is entered into the memory address register of the computer.
8	ENTER ST	When ENTER ST is pressed, the value displayed by the data LEDs is entered into the status register of the computer.
9	ENTER PC	When ENTER PC is pressed, the value displayed by the data LEDs is loaded into the program counter.
10	ENTER WP	When ENTER WP is pressed, the value displayed by the data LEDs is loaded into the workspace pointer register.
11	DISPLAY MA	When DISPLAY MA is pressed, the value stored in the memory address register is displayed by the data LEDs.
12	DISPLAY ST	When DISPLAY ST is pressed, the contents of the status register is displayed by the data LEDs.

Table C-1 990 Programmer Panel Controls and Indicators (Continued)

Reference Number	Control or Indicator	Function
13	DISPLAY PC	When DISPLAY PC is pressed, the contents of the program counter is displayed by the data LEDs.
14	DISPLAY WP	When DISPLAY WP is pressed, the contents of the workspace pointer register is displayed by the data LEDs.
15	LOAD	When the panel is in halt mode, pressing LOAD causes the computer to trap to the ROM loader starting address.
16	RESET	Pressing RESET results in the generation of an IORESET-, which is a pulse that resets all units in the system. The IORESET- is generated by the execution of the front panel firmware in the loader ROMs. Faults that prevent the CPU from executing instructions will also prevent the generation of an IORESET-.
17	RUN	When the computer is halted (programmer panel is active), pressing RUN returns the computer to run mode and deactivates the panel.
18	HALT/SIE	When the computer is in run mode (RUN LED is lit), pressing HALT/SIE causes the computer to halt and begin processing the programmer panel software if the key switch is turned to the UNLOCK position. Pressing HALT/SIE when the computer is not in run mode causes the computer to execute a single instruction at the present program counter (PC) address. The contents of the PC is incremented by two and displayed by the data LEDs.

Table C-1 990 Programmer Panel Control and Indicators (Continued)

Reference Number	Control or Indicator	Function
19	KEY switch	The KEY switch (OFF/LOCK/UNLOCK) prevents unauthorized computer power-up or program intervention. In order to apply AC power to the chassis, the key must be inserted into the switch and turned to the LOCK position. At this point power is applied to the computer, but the programmer panel is locked out. In the UNLOCK position, the computer can be halted by pressing HALT/SIE. You can remove the key from the OFF or LOCK position.
20	RUN	The RUN light is on when a low-active RUN signal is generated, indicating that the computer is in run mode. When lit, all switches on the panel, except HALT/SIE are disabled and the data LEDs are driven under program control. When the RUN light is off, the panel controls are active.
21	IDLE	The IDLE light is on when the computer is executing an idle instruction, indicating computer inactivity for most interrupt-driven software.
22	FAULT	The FAULT light is on when the computer detects a hardware or diagnostic test failure.
23	POWER	The POWER light is on when power is applied to the unit (that is, the key switch is turned to the LOCK or UNLOCK position).

Appendix D

DOCS Message Descriptions

The following alphabetical list describes the messages that may be output when you are running DOCS or a diagnostic test module.

ALREADY IN BUILD LIST MODE
: You have attempted to execute the build list (.BL) verb while Batch Command Stream DOCS is in build list mode.

ANSWER NOT 0 OR 1
: DOCS is requesting a YES/NO answer to a prompt. Enter either a 1 (YES) or a 0 (NO).

AT BREAKPOINT XXXX
REGISTERS 0 - 7 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
REGISTERS 8 -15 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
WORKSPACE POINTER = XXXX
: This message is displayed when a breakpoint is encountered in a test being run under Debug DOCS, and reports the values of the workspace register at the time the breakpoint was reached.

AT TIME OF ERROR WP= XXXX PC= XXXX ST= XXXX
: A test has terminated abnormally. The message states the condition of the workspace pointer, program counter, and status register at the time the test terminated.

BAD LEVEL XOP AT >XXXX.
: DOCS received a level XOP with an invalid code (0 - F). XXXX is the address of the level XOP. Retry the current activity.

CHANNEL NUMBER (0-3)? DEF = X
: If you chose a CI-403 for your printer interface, then inform DOCS of the printer channel number.

CHECKSUM ERROR!
: This message occurs during the loading of a test. A checksum error has occured. Check the object of the test being loaded.

CONDITION NOT FOUND
: This message occurs during execution of the memory search (.MS) verb, when DOCS is unable to locate the specified condition, or the condition is not satisfied within the specified number of words.

DELETE BREAKPOINT NOT FOUND IN TABLE
This Debug DOCS message occurs during execution of the .DB verb when the specified address is not listed in the breakpoint table, and therefore is not a recognized breakpoint.

DISK CONTROLLER NEVER IDLE
The disk controller is never idle and you cannot issue a command. You may need to replace the disk controller.

DISK ERROR!
This message informs you that the computer cannot read the disk because a bad hardware or disk condition exists.

DISK IS EITHER OFF LINE OR NOT READY
This message indicates that the computer cannot read from the disk because the disk drive is either off line or not ready.

DOCS ERROR MESSAGE DEVICE DEFAULT CONFIGURATION:
SPEED: 4800 BAUD
PARITY: EVEN
DUPLEX: FD - RC (ON)

CHANGE CONFIGURATION? DEF = X
This prompt appears if you have chosen a printer to receive error messages. It informs you of the DOCS defaults for that printer and then enables you to change those defaults.

DOCS TYPE?
(0= MINI, 1= MAXI, 2= BCSRUN, 3= BCSEDT, 4= DEBUG)
If you are using Dynamic DOCS, then you can choose the DOCS type. Enter the number that corresponds to the DOCS type you want.

DEVICE SPEED?
0= 160 1= 200 2= 300 3= 600
4= 1200 5= 2400 6= 3600 7= 4800
8= 7200 9= 9600 A= 19200 DEF = 7
If you wish to change the DOCS error message device defaults, then you are prompted for the printer speed.

DEVICE PARITY?

0= NONE 1= ODD 2= EVEN 3= MARK 4= SPACE DEF= 2
If you wish to change the DOCS error message device defaults, then you are prompted for the printer parity.

DEVICE DUPLEX?
0= FD 1= FD-RC (OFF) 2= FD-RC (ON) DEF = 2
If you wish to change the DOCS error message device defaults, then you are prompted for the printer duplex.

END OF TEST = XXXX
This is a test initialization message indicating the memory location of the end of the test module after it has been loaded by DOCS.

ENDING BUILD LIST MODE
This message is output when Batch Command Stream DOCS transfers from build list to normal mode. This is caused when the user enters the character @, or a verb that is illegal in build list mode.

ENDING EXECUTE LIST MODE
This message is output when Batch Command Stream DOCS transfers from Execute List Mode to Normal Mode. This is caused when you enter the @ character, or a verb that is illegal in Execute List Mode.

ENTER CONTROL OPTIONS SEPARATED BY COMMAS
(E=ERR MSG'S, H=HDR MSG'S, N=ERR #'S, P=PAUSE ON ERR'S)
DEFAULT = E,H,N
This message appears during the DOCS initialization process or during execution of the initialize control options (.IP) verb. It allows you to specify if error or header messages will be output, or if DOCS will output a pause message when it encounters an error condition during execution of a test.

ENTER NEW SUBDIRECTORY NAME
You have executed the .CN verb. This prompt requests the new subdirectory name that further loads are to be built upon. That is an .LD from this point on will give you a menu of files that are in this subdirectory.

ENTER TEST VERB
DOCS is ready to receive any legal test verb.

ENTERING BUILD LIST MODE
Batch Command Stream DOCS outputs this message when you execute the build list .BL verb.

ENTERING EXECUTE LIST MODE
Batch Command Stream DOCS outputs this message when you enter the execute list (.EL) verb.

ERROR - HIT RETURN TO CONTINUE

When you have selected the pause-on-errors option (during DOCS initialization), this message occurs if an error condition is encountered.

ERROR MESSAGE DEVICE? (0=ME, 1=PRINTER) DEF = 0

If you want the interactive device to serve also as the error message device then choose 0. Otherwise, let DOCS know that you want a hard copy of your error messages.

ERROR MESSAGE DEVICE INTERFACES?

0= 990/5	1= 990/10A	2= CI-402	3= CI-403
4= CI-421	5= CI-422	6= TTYEIA	7= MY AUX

Since a printer was chosen as an error message device, inform DOCS of the printer interface.

***ERROR XXXX

This message is output when a error occurs while a test is executing. It is often accompanied by an explanation of the error. XXXX is the test module error number. Test error messages are explained in the documentation for each test. This message can be turned off if you do not select the N option in response to the control options prompt.

EXECUTE EA VERB? (DEF=X)

This message occurs after the test loading and initialization prompts have been answered and DOCS is ready to execute the test module.

FILE # OR NAME?

This message asks you to select the test to be run by entering either its mnemonic name or the two-digit number appearing with the name on the menu.

ILLEGAL HEX INPUT

This message signifies that an input to DOCS was not in legal hexadecimal form, and that you must reenter a legal number.

** ILLEGAL INPUT

This DOCS error message occurs when you enter a verb that is illegal in the mode in which DOCS is currently operating.

ILLEGAL VERB

This message reports that the verb entered does not exist, and that you must enter the correct verb.

ILLEGAL XOP OCCURRED

DOCS has received an XOP on a level other than level 15 (or level 14 when using Debug DOCS).

INTERRUPT TRAPS HAVE BEEN CHANGED RESET TRAPS TO DOCS

This message will occur if the interrupted traps in lower memory have been changed either by a test or by a software error. You can reset these traps back to the original traps by answering 1 to the prompt.

LABEL NOT FOUND

This error message occurs in Batch Command Stream DOCS when DOCS receives an instruction to pass execution to a nonexistent label. DOCS will return to the VERB? prompt.

LEVEL 2 INTERRUPT

DOCS received a level 2 interrupt that does not correspond to a valid interrupt status. Check the CPU hardware.

LEVEL 2 INTERRUPT--STATUS= XXXX

CRU	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TO	PO	IO	ME	MP	NP	EV	WV	SO	BP	CK	AO					
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

A level 2 interrupt has occurred. This message indicates the cause of that interrupt. (Refer to paragraph 5.4.3 for CRU bit definitions.)

** LINE ALREADY DELETED

While running Batch Command Stream DOCS, you have attempted to delete a line whose number has already been removed from the batch stream. The compress list (.CL) verb will assign new line numbers to the list so the desired line can be deleted.

** LINE TRUNCATED TO 50 CHARACTERS

This message indicates that the line succeeding the message has been shortened to the maximum DOCS length of 50 characters.

** LINE TRUNCATED TO 50 CHARS

This message occurs while running Batch Command Stream DOCS and in the Build List mode. You have entered over 50 characters of text and DOCS has truncated your response to 50 characters.

LINE NO (DEF=XX)
This message appears when you execute the print line (.PL) verb under Batch Command Stream DOCS. You should specify the line number of the line to be printed.

LOAD BIAS? DEF= XXXX
DOCS is asking you to specify the load point for a test module. The default value is the location immediately following the end of DOCS.

LOAD BIAS? DEF= 00A0
This message appears when DOCS is loading S$LOADER.

LOAD TEST? (DEF=1)
This message is output during test module loading after DOCS has been initialized. If you respond with a 1 (yes), DOCS requests loading instructions. A 0 (no) response causes DOCS to return to the question VERB?.

LOOP ON WRITE ?
When the read/write CRU Word (.CW) verb is executed, this message asks you if DOCS should repeat the writing of data to the specified CRU base.

NEED MORE MEMORY FOR LOAD
DOCS needs more memory to complete the loading of a test. Reexecute the load diagnostic (.LD) verb with a lower load bias or add more memory (if less than 64 K-bytes).

NO TABLE ENTRY AT XX
This error message occurs when you are executing one of the text editing verbs under Batch Command Stream DOCS. You have specified a nonexistent line number.

NO VERB TABLE EXISTS FOR TEST
This message occurs when DOCS is unable to locate the test verb table.

NONEXISTENT FLAG
You have attempted to access beyond the maximum number of >7F flags.

NUMBER OF LOOPS ?

When executing the execute stand-alone (.ES) verb, DOCS asks for the number of times the test should be executed.

NUMBER OF WORDS IN INSTRUCTION - X

This message occurs during execution of the assign breakpoint (.AB) verb under Debug DOCS. You must specify the number of words contained by the instruction where a breakpoint should be established.

NUMBER OF WORDS MUST BE 1, 2, OR 3

This Debug DOCS message occurs when you attempt to assign a breakpoint at an instruction that contains more than three words.

OUTPUT DEV (0=ERR, 1=I/O)

DOCS asks this question when you execute the print list (.PL) verb. The default is the interactive terminal.

PHYS RECORD SIZE EXCEEDS 7120

DOCS is trying to load a file that has been initialized with a physical record size greater than 120 bytes. Rebuild the file with the correct record size.

PLEASE: PLACE CASSETTE IN TRANSPORT (MDU)
POSITION AT LOAD POINT

PRESS "RETURN" WHEN UNIT IS READY

This message occurs during the load of a diagnostic from the 733 ASR. Follow the instructions and hit <RETURN>.

POWER ON

This message indicates that a level 0 (power up) interrupt occurred during the operation of DOCS or during a test.

PRESS @ TO EXIT FROM VERB

This message reminds you that execution of a verb may be aborted by pressing the @ character on the interactive terminal.

PRESS -@- TO STOP
ADDR 0 2 4 6 8 A C E
TO CONTINUE ENTER (F=FORWARD, B=BACKWARD, DEFAULT=F) -

This message is output when the dump memory (.DM) verb is executed, and includes the words of memory that are dumped to the message terminal.

PRINTER BASE ADDRESS? DEF = XXXX

You have chosen a printer to receive error messages. You must now inform DOCS of the printer base address.

READY TAPE AT UNIT X AND HIT RETURN

When you are using a tape to load files, this prompt reminds you that the unit must be ready before you can load a test.

REQUESTED HEX BYTE COUNT TOO LARGE

This message reports that the program has requested input of a number that exceeds >20 bytes in length.

"RETURN" SHOWS MORE FILES

There may be more diagnostics on the medium than can be displayed on the VDT screen at one time. When you press RETURN, DOCS displays another screen of menu.

TABLE CONTAINS MAXIMUM NUMBER OF BREAKPOINTS

During implementation of Debug DOCS, a maximum of 20 breakpoints may be set in a test at any given time. This message occurs when the maximum limit is reached.

** TABLE LENGTH EXCEEDED

This Batch Command Stream DOCS message occurs when the batch stream exceeds the maximum table length of >1000 bytes.

TERMINAL BASE ADDRESS -

This message is sent when DOCS is first initialized, informing you of your I/O terminal base address.

TERMINAL INTERRUPT LEVEL = XX

This is either a message or a prompt. If your I/O device is at a fixed interrupt level then this is a message. Otherwise, you need to enter the level of your interactive terminal.

TOO MANY CHARACTERS

This message occurs when you enter a string of data containing more than the amount requested.

UNEXPECTED INTERRUPT AT LEVEL = X

This message appears when DOCS has received an interrupt at a level other than 0, 1, 2, the real-time clock, or the interactive or message terminal. This is commonly caused by the incorrect entry of an interrupt.

**** WARNING ****
INITIALIZATION NOT COMPLETE -- ENTER VERB XX FOR RESTART

This message warns you that the initialization procedure is incomplete, and that the XX verb must be executed in order to continue the procedure.

**** WARNING ****
S$LOADER MEANT TO BE LOADED AT 00A0
ARE YOU SURE ? DEF=0

This message indicates that while using DOCS to load DX10, you have attempted to load S$LOADER in a location other than 00A0. S$LOADER will load the DX10 system correctly only when the selected load bias is 00A0.

WARNING - THIS OVERLAYS DOCS
ARE YOU SURE ? DEF=0

This message indicates that the load point of a module is below the end of DOCS and will destroy DOCS if loaded at that location. DOCS may be loaded over itself and then reinitialized by entering a Y.

WRITE PROTECT DISK, HIT RETURN

This message appears during the loading of a test as a reminder to write-protect the DOCS disk.

WRITE PROTECT TAPE, HIT RETURN

When you are using a tape to load files, this prompt tells you that your tape is not write protected. Write protect the tape and hit "RETURN".

XOP AT >XXXX IS BAD

This message indicates that the XOP at the specified hexadecimal location is invalid.

XOP OCCURRED BUT BREAKPOINT NOT FOUND IN TABLE

This Debug DOCS message occurs when there is no breakpoint in the table that corresponds to the XOP.

Appendix E

Assembly/Diagnostic Cross-Reference

This appendix cross references the diagnostic tests with the part numbers of the target assemblies each diagnostic is designed to test. Volume numbers refer to the volumes of the Unit Diagnostics Handbook.

Cross-Reference of Target Assemblies with Diagnostic Tests

Assembly Part No.	Target Assembly Description	Diagnostic Mnemonic Name	Volume
0841039	2230/2260 Line Printer	FLPTST	4
0937361	FPLA Personality Card	PROMPG	5
0937362	EPROM II Personality Card	PROMPG	5
0937363	PROM II Personality Card	PROMPG	5
0937505	DS10 Interface Module	DS10PD	3
0937513	DS10 Disk Drive	DS10PD	3
0938152	810 Dot Matrix Printer	LP810	4
0940050	5MT/6MT Serial Interface Module	FIVMOD	4
0940065	DS200 Interface Module	DSKTRI	3
0941555	9900/9980 Emulator Module	EMU900	7
0941570	TMS9940 Emulator Module	EMU940	7
0944905	CRU Buffer Module	CRUEXP	4
0944910	990/4 AU (8K-Byte Version)	AU04	2
0944910	990/4 AU (Extended Test Version)	AU04TST	2
0944910	990/4 AU (Memory Protect Logic)	MEMPRT	2
0944925	PROM Programmer Interface Module	PROMPG	5
0944930	990/10 AU (Processing Circuits)	CPU10	2
0944935	990/4 RAM (40K-Byte Circuits)	RAM04	2
0944950	990/10 SMI (System Memory Interface)	MAPTST	2
0944996	PROM Programmer	PROMPG	5
0945005	CRU Expansion Module	CRUEXP	4
0945075	TTYEIA Interface Module	TTYEIA	4
0945083	803 Card Reader	CRDRDR	4
0945085	TILINE Coupler	TILCOU, TLCPLR	2
0945093	990/10 RAM (48K-Byte Circuits)	MEMTST	2
0945135	PROM Personality Card	PROMPG	5
0945145	16 I/O Interface Module	IO16	4
0945150	EPROM Memory Board	EROMBT	5
0945165	EPROM Personality Card	PROMPG	5
0945185	803 Interface Module	CRDRDR	4
0945925	9900/9980 Emulator Module	EMUTST	7

Cross-Reference of Target Assemblies with Diagnostic Tests (Continued)

Assembly Part No.	Target Assembly Description	Diagnostic Mnemonic Name	Volume
0945940	FD800 Interface Module	FLPDSK	3
0945940	607 Interface Module (Computer)	RMTFLP	3
0945989	FD800 SSSD Diskette Drive	FLPDSK	3
0946075	911 Keyboard	CRT911	4
0946076	911 Interface Module	CRT911	4
0946105	Communications Interface Module	CRCOMM	6
0946110	Automatic Call Unit Module	ACUTST	6
0946655	990/10 RAM (16K-Byte Circuits)	MEMTST	2
0946695	913 Interface Module	CRT913	4
0947524	DS25/DS50 Interface Module	DSKTRI	3
0947524	DS25/DS50 Disk Drive	DSKTRI	3
0947525	DS200 Interface Module	DSKTRI	3
0947525	DS200 Disk Drive	DSKTRI	3
0947555	979A Interface Module (800 BPI)	TAPTST	3
0947853	Test Adapters	PROMPG	5
0948209	979A Tape Transport	TAPTST	3
0948560	911 Video Display Terminal	CRT911	4
0948990	979A Interface Module (800/1600 BPI)	TAPTST	3
0949024	Analog-to-Digital Converter	ADCHK	5
0949029	Digital-to-Analog Converter	DACHK	5
0949087	Local Multi-Drop Module	LOCLIN	4
0949165	607 Remote SSSD Drive Assembly	RMTFLP	3
0949320	607 Interface Module (Drive Unit)	RMTFLP	3
0949341	607 SSSD Diskette Drive Unit	RMTFLP	3
0949807	32-Bit Output Data Module	OUTMOD	4
0949807	32 Input/Transition Det. Module	INPMOD	4
0949820	32-Bit Output Data Module	INPMOD	4
0949910	Trace Module	TRACE	7
0949971	TMS9940 Buffer Module Assembly	EMU940	7
0949972	SBP9900A Buffer Module Assembly	EMU900	7
0949973	TMS9900-40 Buffer Module Assembly	EMU900	7
0949974	TMS9980A Buffer Module Assembly	EMU900	7
0949995	9900 Buffer Module Assembly	EMUTST	7
0960294	DS31 Disk Drive	DSKM3X	3
0973953	733 KSR Terminal	TST733	4
0973955	733 ASR Terminal	TST733	4
0974783	913 Keyboard	CRT913	4
0974906	DS31 Interface Module	DSKM3X	3
0974989	588 Line Printer	LPTEST	4
0974993	306 Line Printer	LPTEST	4
0975243	913 Video Display Terminal	CRT913	4

Cross-Reference of Target Assemblies with Diagnostic Tests (Continued)

Assembly Part No.	Target Assembly Description	Diagnostic Mnemonic Name	Volume
2215520	Model 840 RO Terminal	TST8XO	4
2239135	EI990 Interface Board	ENETST	6
2239155	EI300 Interface Board	ENETST	6
	TEN-X COBOL Accelerator Board	TNXTST	2
2250586	990/5,10,12 (RAM Circuits)	RAMTST	2
2261630	979A Interface Module (800 BPI)	TAPTST	3
2261635	979A Interface Module (800/1600 BPI)	TAPTST	3
2261686	FD1000 DSDD Disk Drive	DDFLOP	3
2261690	FD1000 Interface Module	DDFLOP	3
2261930	990/5 AU (Processing Circuits)	AU05	2
2261930	990/5 AU (Host-Slave Circuits)	HSTSLV	2
2261930	990/5 AU (RAM Circuits)	MEMTST	2
2261930	990/5 AU (Communications Ports)	COM90X	6
2261955	Four-Channel Communications Controller	FC3TST	6
2261964	BCAIM Communications Controller	BCAIMT	6
2261975	990/12 SMI (System Memory Interface)	MAP12	2
2262091	820 KSR Terminal	TST8XO	4
2263480	External Autocall Unit	EXTACU	6
2267730	Local Line Module	LLMTST	6
2267785	Time Function Module	TFMTST	5
2268483	990/12 AU (Processing Circuits)	CPU12	2
2269405	CD1400 Disk Drive	DSKCDD,DSKSA,	3
		DSKCOM	3
2269906	CD1400 Disk Controller	DSKCDD,DSKSA,	3
		DSKCOM	3
2270820	Tiline Peripheral Bus	TPBITS	3
2275975	Model 990/10A Minicomputer	CPU10,MAPTIC,	1,2
		CRUTIC,RAMCC	1,2
2276290	940 Electronic Video Terminal	TST940	4
2303085	X.21 BCAIM	BCAIMT	6
2308166	Model MT1600 Tape Transport	TAPTST	3
2308469	DS80 Disk Drive	DSKCDD,DSKSA,	3
		DSKCOM	3
2308500	DS300 Disk Drive	DSKCDD,DSKSA,	3
		DSKCOM	3
2310945	LP300/LP600 Printers Interface Board	PRTIF	4

Appendix F

Reference Documents

The following documents contain operation, maintenance, and troubleshooting information for 990 computers and peripheral devices. The documents are arranged under the following headings:

* System level (includes references for processors and memories)
* Mass storage devices
* Printers, terminals, and interface modules
* Industrial systems
* Communication interfaces
* AMPL systems

System Level Reference Documents

Title	Part Number
990/99000 Assembly Language Reference Manual	2270509-9701
Model 990/12 Computer Assembly Language Programmer's Guide	2250077-9701
Model 990/4 Computer System Hardware Reference Manual	945251-9701
Model 990/4 Computer System Field Maintenance Manual	945401-9701
Model 990/4 Computer System Depot Maintenance Manual	945403-9701
Model 990/5 Computer Hardware User's Manual	946294-9701
Model 990/5 Microcomputer Field Maintenance Manual	946295-9701
Model 990/5 Microcomputer Depot Maintenance Manual	946296-9701

System Level Reference Documents (Continued)

Title	Part Number
Model 990/10 Computer System Hardware Reference Manual	945417-9701
Model 990/10 Computer System Field Maintenance Manual	945402-9701
Model 990/10 Computer System Depot Maintenance Manual	945404-9701
Model 990/12 Computer Hardware User's Manual	2264446-9701
Model 990/12 Computer System Field Maintenance Manual	2264447-9701
Model 990 Computer Family Maintenance Drawings, Computer Chassis and Enclosures, Volume I	945421-9701
Model 990 Computer Family Maintenance Drawings, Processors and Memories, Volume II	945421-9702
Model 990 Computer Peripheral Equipment Field Maintenance Manual	945419-9701
DS990 Models 2, 3, and 5, Systems Installation and Operation	2250357-9701
DS990 Model 20/30 System Installation and Operation Manual	2250693-9701
DS990 Models 4, 6, and 8 Systems Installation and Operation Manual	946284-9701
Model 990 Computer FS990 System Installation and Operation Manual	946254-9701
Model 990/10A Computer	2302633-9701
Model 990/10A Computer Maintenance Manual -- Field Theory and Maintenance	2302634-9701

System Level Reference Documents (Continued)

Title	Part Number
Model 990/10A Computer Maintenance Manual - Depot Theory and Maintenance	2302635-9701
Business System 300 System Description Manual	2533308-9701
Business System 300 Field Theory and Maintenance Manual	2533309-9701
Business System 300 Depot Theory and Maintenance Manual	2533310-9701
Business System 300 System Drawings	2533311-9701
Business System 200 System Description Manual	2533251-9701
Business System 200 Field Theory and Maintenance Data	2533252-9701
Business System 200 Depot Theory and Maintenance Data	2533253-9701
Business System 200 System Drawings	2533254-9701
DX10 Micro Operating System Handbook	2533262-9701
DX10 Micro Operating System Installation Guide	2533260-9701
DX10 Micro COBOL Installation Guide	2533273-9701
DX10 Micro Sort/Merge Installation Guide	2533259-9701
Link Editor Reference Manual	949617-9701
DX10 Micro Message and Codes	2533322-9701
Development Guide for DX10 Micro on DX10	2533275-9701

Mass Storage Devices Reference Documents

Title	Part Number
Model FD1000 Flexible Disk System Installation and Operation Manual	2261886-9701
Model FD1000 Flexible Disk System with International Chassis Installation and Operation	2250698-9701
Model CD1400 Disk System Installation and Operation Manual	2272081-9701
Model FD1000 Flexible Disk System Field Maintenance Manual	945419-9703
Model FD1000 Flexible Disk Controller and International Chassis Power Supply Depot Maintenance Manual	2261885-9701
Models DS31/32 Disk System Installation and Operation	945260-9701
Models DS31/32 Disk Controller Depot Maintenance Manual	945414-9701
Maintenance Manual for Series 30 Disk Drive	961684-9701
Product Description Manual for Model 029 Power Supply (DS31/DS32 Disk Drive Power Supply	961684-9702
Models DS25/DS50 Disk Systems Installation and Operation	946231-9701
Model DS200 Disk System Installation and Operation	949615-9701
Model DS10 Cartridge Disk System Installation and Operation	946261-9701

Mass Storage Devices Reference Documents (Continued)

Title	Part Number
Model DS10 Cartridge Disk System Field Maintenance Manual	945419-9702
Model DS10 Cartridge Disk System Depot Maintenance Manual	946262-9701
Model FD800 Floppy Disk System Installation and Operation Manual	945253-9701
Model FD800 Floppy Disk Controller and International Chassis Power Supply Depot Maintenance Manual	945418-9701
Model FD1000 Flexible Disk System Installation and Operation	2261886-9701
Model 607 Diskette Unit Remote Floppy Disk Buffer Board and Interface/Power Supply Board Depot Maintenance Manual	944768-9701
Model 979A Magnetic Tape System Installation and Operation	946229-9701
Model 979A Tape Transport Installation and Operation	949612-9701
Model 979A Magnetic Tape Controller Depot Maintenance Manual - Volume I	946237-9701
Model 979A Tape Transport Subsystem Maintenance Manual	949613-9701
Model 979A Tape Transport Subsystem Drawings	949613-9702
WD500 Mass Storage System Installation and Operation Manual	2302688-9701
WD800 Mass Storage System Installation and and Operation Manual	2302629-9701
Model MT1600 Magnetic Tape System Field Maintenance Manual	2302649-9701
Model MT1600 Magnetic Tape Subsystem Installation and Operation Manual	2302642-9701

Mass Storage Devices Reference Documents (Continued)

Title	Part Number
Model MT1600 Magnetic Tape Subsystem Maintenance Drawings Manual	2276559-9701
Business System 200, WD500 Disk Unit Operator's Guide	2533269-9701

Printers, Terminals, and Interface Modules Reference Documents

Title	Part Number
Model 911 Video Display Terminal Installation and Operation	945423-9701
Model 911 Video Display Terminal Depot Maintenance Manual	945424-9701
Model 913 Video Display Terminal Installation and Operation	943457-9701
Model 913A Video Display Terminal Depot Maintenance Manual	945406-9701
Models 2230 and 2260 Line Printers Installation and Operation	946256-9701
Models 306 and 588 Line Printers Installation and Operation	945261-9701
Model 306 Printer Technical Manual (Plus Model 306C Addendum)	974993-9701

Printers, Terminals, and Interface Modules Reference Documents (Continued)

Title	Part Number
Model 500 Printer Technical Manual (Plus Model 588 Addendum)	974998-9701
Model 810 Printer Installation and Operation	939460-9701
Model 733ASR/KSR Data Terminal Installation and Operation	945259-9701
Silent 700* Electronic Data Terminal Models 732/733 ASR/KSR Maintenance Manual	960129-9701
Model 743 KSR Data Terminal Installation and Operation	943462-9701
Model 820 KSR Terminal Installation and Operation Manual	2250445-9701
Model 820 KSR Terminal Operating Instructions	949854-9701
Model 820 KSR Terminal Maintenance Manual	999853-9701
Model 990 Computer 16 Input/16 Output EIA Data Module Depot Maintenance Manual	945415-9701
Model 990 Computer 16 Input/16 Output TTL Data Module Depot Maintenance Manual	945407-9701
Model 990 Computer TTY/EIA Terminal Interface Module Depot Maintenance Manual	945408-9701
Model 940 Systems Manual	2213826-9701
Model LP300/LP600 Line Printers Field Maintenance Manual	2302644-9701
Model LP300/LP600 Line Printers Installation and Operation Manual	2302643-9701
Business System 200, Model 840 RO Printer Operator's Guide	2533270-9701
TILINE* Peripheral Bus Interface Controller Depot Maintenance Manual	2306141-9701

* Trademark of Texas Instruments

Industrial Systems Reference Documents

Title	Part Number
Model 990 Computer Analog/Digital Converter Module, Digital/Analog Converter Module Installation and Operation Manual	944774-9701
Model 990 Computer Analog/Digital Converter Module, Digital/Analog Converter Module Depot Maintenance Manual	944775-9701
Model 990 Computer 5MT/6MT Serial Interface Module Installation and Operation	946267-9701
Model 990 Computer 32-Input/Transition Detection Module, 32-Bit Output Data Module, and Digital Input/Output Termination Panel Installation and Operation	946269-9701
Model 990 Computer PROM Programming Module Depot Maintenance Manual	945405-9701

Communication Interfaces Reference Documents

Title	Part Number
Model 990 Computer Bit-Oriented/Character-Oriented/Asynchronous Interface Module (BCAIM) Installation and Operation Manual	2263886-9701
Model 990 Computer X.21 Bit-Oriented/Character-Oriented/Asynchronous Interface Module (X.21 BCAIM) Installation and Operation Manual	2263883-9701
Model 990 Computer X.21 Bit-Oriented/Character-Oriented/Asynchronous Interface Module (X.21 BCAIM) Maintenance Manual	2263884-9701
Model 990 Computer Pulse/Tone Automatic Call Unit and External Automatic Call Unit Interface Installation and Operation	945425-9701
Model 990 Computer Automatic Calling Unit Depot Maintenance Manual	946225-9701

Communication Interfaces Reference Documents (Continued)

Title	Part Number
Model 990 Computer Communications System Installation and Operation	945409-9701
Model 990 Computer Communications Interface Module Depot Maintenance Manual	945410-9701
Model 990 Computer Four Channel Communications Controller Maintenance Manual	2263879-9701
Model 990 Computer Local Line Module Operation and Maintenance Manual	2250676-9701

AMPL Systems Reference Documents

Title	Part Number
Model 990 Computer AMPL TMS9900/9900-40, SBP990A, and TMS9980A/9981 Emulator and Buffer Modules Installation and Operation Manual	946278-9701
Model 990 Computer AMPL TMS 9940 Emulator and Buffer Modules Installation and Operation Manual	946279-9701
Model 990 Computer TMS 9900/9980A Emulator and Buffer Modules Depot Maintenance Manual	946239-9701
Model 990 Computer AMPL Logic State Trace Data Module Depot Maintenance Manual	946242-9701
Model 990 Logic State Trace Data Module Installation and Operation	946241-9701
Model 990 Computer AMPL TMS 9900/9900-40, SBP990A, and TMS 9980A/9981 Emulator and Buffer Modules Depot Maintenance Manual	946280-9701
Model 990 Computer AMPL TMS9940 Emulator and Buffer Modules Depot Maintenance Manual	946281-9701

Appendix G

CONVER Diagnostic Utility

G.1 INTRODUCTION

CONVER is a diagnostic utility that identifies the hardware configuration of the 990 system on which it is executed. You can use CONVER to perform the following tasks:

* To verify that a system was configured properly during initial installation
* To check that a system was reconfigured properly
* To identify the current system configuration

After being loaded into a 990 central processing unit (CPU), CONVER identifies the following types of system components:

* CPU environment
* Communications register unit (CRU) controller(s) and device(s)
* TILINE controller(s) and device(s)
* Chassis interrupt connections

For the CPU environment, CONVER gives you the following information:

* CPU type
* Interrupt level of the real-time clock
* CRU expansion chassis presence
* Memory size

For each CRU and TILINE device present, CONVER gives you the following information:

* Device type
* Device I/O base address (CRU or TILINE)
* Device interrupt level

CONVER accomplishes its objectives by reading the location of each I/O base address in the system. If it determines that a device is present at an address, CONVER attempts to identify the type of device by its response to handshaking operations. Four types of findings are possible when CONVER polls a specific address:

* No device is present.
* A device is suspected to be present but cannot be confirmed.
* An unidentified device is present.
* An identified device is present.

G.2 TEST REQUIREMENTS

CONVER requires the following equipment for proper operation:

* A Model 990 Computer with a minimum of 64K (where K equals 1,024) bytes of memory
* An interactive terminal
* A device to load the test module

G.3 TEST CHARACTERISTICS

Familiarize yourself with the following characteristics of CONVER before executing the utility:

* CONVER operates under the control of the Diagnostics Operational Control System (DOCS). This volume contains information about DOCS operation.

* CONVER can operate in automatic mode. In this mode, the configuration of the unit under test is compared with a configuration file in the test system. The test system file lists the expected configuration for the test system. In this way, the actual and expected configurations are compared. Running CONVER in automatic mode is required for automatic system test at the factory.

* In general, CONVER only reports and identifies devices that are properly powered up. A device that is not powered up is usually reported as no device.

* CONVER handles offline devices differently, according to type. With the exception of Winchester disk drives, CONVER can usually identify the disk type of offline disk drives. It identifies offline tape drives as tape drives, but it usually cannot report the type.

G.4 DEVICES IDENTIFIED

CONVER has internal profile information to concretely identify the following devices:

* Central processing units for the following models of the 990 computer: 990/4, /5, /10, 10A, /12
* CRU controllers and devices:
  - FD800 -- Single-sided, single density diskette
  - 911 VDT -- 911 video display terminal
  - 913 VDT -- 913 video display terminal
  - COMM/IF -- Communications interface module
  - 16 I/O interface with an attached high-speed printer
  - TTY/EIA Interface with an attached printer (for example, an 810 or a 743)

* TILINE controllers and devices:
  - FD1000 -- Double-sided, double-density diskette
  - DS10 -- Disk drive
  - DS31/DS32 -- Diablo disk drive
  - DS25 -- 25M byte disk drive
  - DS50 -- 50M byte disk drive
  - DS80 -- 80M byte disk drive
  - DS200 -- 200M byte disk drive
  - DS300 -- 300M byte disk drive
  - CD1400 -- 16, 48, or 80 megabyte cartridge disk
  - WD500 -- 5 1/4 inch Winchester disk
  - WD800 -- 8 inch Winchester disk, 3 or 7 tracks/cylinder
  - 979A -- Magnetic tape transport
  - MT1600 -- Magnetic tape transport
  - MCTU -- Magnetic cartridge tape unit

G.5 INITIALIZATION

DOCS must load and initialize CONVER before execution can begin. This volume explains the loading procedure for diagnostics.

After loading CONVER, DOCS writes the name and version of the test as follows:

CONVER - CONFIGURATION VERIFIER. VERSION = JJJ/YY *X

During initialization, CONVER tests the memory area designated for storing device information. A memory error causes an error message to appear but does not cause initialization to abort. The error message tells you the faulty memory location.

No prompts appear during initialization. The following message appears when the procedure is complete:

CONVER INITIALIZATION COMPLETE

The following prompt appears next:

EXECUTE EA VERB? DEF= 1

Accept the default if you want to execute CONVER immediately. Otherwise, enter 0. You may execute any DOCS or CONVER verb when the VERB? prompt appears.

G.6 LOADING

The loading procedure on a Model 990 Computer is explained in another section of this volume.

G.7 VERBS

The EA verb executes the CONVER diagnostic utility and the IT verb allows you to reinitialize the diagnostic. All other verbs are utility verbs that provide additional information. CONVER provides the following verbs:

Verb	Function
AR	Write Automatic Report
DM	Show Device Menu
EA	Identify Devices and Write Report
IT	Initialize Test
PV	Print Available Verbs
RP	Write Report, After Configuration

G.7.1 Execute All (EA) Verb

The EA verb executes the CONVER diagnostic utility, which performs the following:

1. Accumulates information about the CPU environment.
2. Accumulates information about the presence of devices at each CRU and TILINE address.
3. Automatically executes the RP verb, which produces the following:
   * A CPU environment report
   * A CRU map report
   * A device identification report

If the error message device that you selected during initialization is different from the interactive device, the reports appear at both devices. See the RP verb for an explanation of how to interpret the configuration reports.

The time required for the identification process depends on the CPU type of the host computer and the device types configured to it. CONVER requires approximately five seconds for each MT1600 tape drive on the system and three seconds for each DS10 disk drive. On a 990/12 system that is free of MT1600 or DS10 drives, CONVER should complete its identification phase in approximately one second.

NOTE

The EA verb takes over complete control of the interrupt vectors during the identification process, with the exception of levels 0 through 2 and the real-time clock. Thus, an operator-generated interrupt from the interactive device keyboard confuses the identification process by producing unexpected interrupts. After the CPU environment report completes, you may interrupt the output of the other configuration reports by pressing the pause key on the interactive device.

G.7.2 Display Device Menu (DM) Verb

The DM verb writes a list of the devices that CONVER is able to identify. The following is a list of supported devices:

```
NO.      DEVICE

1104 -- EIA/TTY DEVICE
1108 -- 911 VDT
110C -- 913 VDT
1204 -- COM-IF
1304 -- FD800 SSSD
1404 -- 16 I/O 2230,2260,PRINTRONIX
2304 -- TILINE MAG-TAPE,979A
235C -- TILINE MAG-TAPE,MT1600
2308 -- FD1000 DSDD
230C -- DS31/32 DIABLO DISK
2310 -- DS10 CARTRIDGE DISK
2314 -- DS25 TRIDENT DISK
2318 -- DS50 TRIDENT DISK
231C -- DS200 TRIDENT DISK
```

```
2320 -- CD1400 - FIX OR REM - 1 TRK/CYL
2324 -- CD1400 - FIX - 3 TRK/CYL
2328 -- CD1400 - FIX - 5 TRK/CYL
232C -- DS80 80 MB DISK
2330 -- DS300 300MB DISK
2360 -- WD500 DISK
233C -- WD800 - 3 TRK/CYL
2340 -- WD800 - 7 TRK/CYL
2344 -- DSDD DISKETTE FOR WD BACKUP
2348 -- TILINE CARTRIDGE TAPE UNIT
234C -- MEMORY CONTROLLER, 16K W/ECC
2350 -- MEMORY CONTROLLER, 16K CACHE
2354 -- MEMORY CONTROLLER, 64K CACHE
2358 -- MEMORY CONTROLLER, CCC
```

Each number under the NO. column is the AST (automatic system test) part number associated with the entry in the DEVICE column. You can use these numbers to interpret the BP entries produced by the AR verb.

G.7.3 Report Current Information (RP) Verb

You must execute the EA verb at least once before executing the RP verb separately. The RP verb takes the information accumulated by the last execution of the EA verb and produces a CPU environment report. An example of the report format is as follows:

```
CPU IS (type)
CLOCK INTERRUPT LEVEL = XX
```

One of the following messages is displayed next:

```
CRU EXPANSION CHASSIS PRESENT-(lists chassis present)
          or
NO CRU EXPANSION CHASSIS PRESENT-
```

The following message is displayed next:

```
MEMORY: FROM >XXXXXX TO >XXXXXX
```

It then gives you the option to produce the following reports:

1. A CRU map report
2. A device identification report

If the error message device that you selected during initialization is different from the interactive device, the reports appear at both devices.

The prompt for the first optional report appears as follows:

```
OUTPUT CRU MAP ??  DEF=1 -
```

Accept the default if you want to produce the CRU map. Otherwise, enter 0. An example of the output for this report appears as follows:

```
                        C R U   M A P
        00   20   40   60   80   A0   C0   E0  100 ...  1E0
0000     X    .    X    .    X    X    X    X    X ...   .
0200     .    .    .    .    .    .    .    .    . ...   .
0400     .    .    X    .    .    .    .    .    . ...   .
0600     .    .    .    .    .    .    .    .    . ...   .
```

In this map, the low-order addresses move from left to right; the high-order addresses from top to bottom. For example, the first entry in the table is for address 0000. The next address to the right is 0020. The next address down is 0200. Each entry in the map represents 16 bits of the CRU. The CRU sets the bits of an address without a device present to all' 1's. The map places a period (.) for each address without a device present (all '1's) and an "X" for each address with a device present (at least one of the bits was a 0). Note that CONVER ignores on-board ports of the /5 and the /10A CPUs. The prompt for the second optional report appears as follows:

"OUTPUT DEVICE INFORMATION ?? DEF=1 -".

Accept the default if you want a table of the devices configured on your system. An example of a device information table appears as follows:

```
 ADDR   INT    CH   PO    TYPE-ID      DEVICE-NAME
0440     9      1    3    CONFIRMED    911 V-D-T
0100                      SUSPECTED DEVICE NOT IDENTIFIABLE,
                          CRU RESPONSE >608D
00E0     A                CONFIRMED    911 V-D-T
00C0     A                CONFIRMED    911 V-D-T
0080     B                CONFIRMED    FD800 SSSD
0040     4                CONFIRMED    COM-IF ID = 7F
0000     6                CONFIRMED    EIA/TTY DEVICE
F800u0   D                CONFIRMED    DS200 TRIDENT DISK
F810u0   C                >OFFLINE<    DS31/32 DIABLO DISK
F820u0   7                CONFIRMED    FD1000 DSDD
F820u1   7                CONFIRMED    FD1000 DSDD
F880     8                CONFIRMED    TILINE MAG-TAPE, MT1600
F900                      SUSPECTED    TILINE DEVICE NOT IDENTIFIABLE
```

Interpret the table headings as follows:

ADDR	CRU OR TILINE base address (in hexadecimal). The address may be followed by a unit number (uX).
INT	Interrupt level associated with the device.
CH	Expansion chassis number (appears only if expansion chassis is present).
PO	Position number (appears if device is in an expansion chassis).
TYPE-ID	Status of the devices at the address.
DEVICE NAME	Name of device (if identifiable).

Keep the following in mind when interpreting the data:

* Entries under TYPE-ID can be listed as CONFIRMED, SUSPECTED, or >OFFLINE<. CONFIRMED means CONVER made a positive identification. SUSPECTED means CONVER made a limited identification. If a device is suspected, CONVER either gives you its best guess at a specific device, tells you the class of the device, or says only that it cannot identify the device. >OFFLINE< means CONVER identified the device but found the device offline.

* The on-board ports of /5 and /10A CPU types (starting at CRU address >1700) are not reported.

G.7.4 AR -- Automatic Report (AR) Verb

You must execute the EA verb at least once before executing the AR verb. The AR verb transfers a configuration file to a remote system. The file is formatted in ASCII and contains the CPU and device information. The file record is printable on a standard terminal. A pair of @ symbols terminate a record.

The report is divided into blocks. One @ symbol terminates each block. The first block gives you information on the CPU environment. It appears in the following format:

```
CPUxxx
BAxxxxxx
EAxxxxxx
ENxx  (appearance is variable)
```

Interpret the entries as follows:

* CPUxxx gives you the 990 model number (xxx is equal to either 004, 005, 00A, 10A, or 00C).

* BAxxxxxx gives you the beginning address of the memory area.

* EAxxxxxx gives you the ending address of the memory area.

* ENxx gives you the CRU expansion chassis number (if present). One entry appears for each expansion chassis present.

Each block that follows the CPU environment block describes one of the devices in the system. Only devices that the RP or EA verbs list as CONFIRMED are described by the AR verb. A TILINE DISK may be confirmed by the RP but not recognized by the AR verb because its store register results do not match any known device. In this case, the AR verb reports the reference part number as >23FF.

Each device block contains one or more lines. Each line begins with a two- or three-character identifier. Interpret entries according to their indentifiers, which are as follows:

* BPxxxx gives you the reference part number for the device controller. To translate the meaning of xxxx, execute the DM verb and look through the menu until you find the device referenced by xxxx. However, you can come to a general understanding of xxxx by analyzing the meaning of the nibbles, as follows:

Nibble	Meaning
1xxx	CRU devices
2xxx	TILINE devices
x1xx	I/O devices (such as VDT911)
x2xx	Communications devices (such as COM-IF)
x3xx	Storage devices (such as FD1000)
x4xx	Output-only devices (such as printers)
xxFF	Incompletely determined devices (xx describes known information)
xFFF	Unknown devices (not identified)

As you can see, the two high-order nibbles describe the device type, and the two low-order nibbles describe the unique identifiable device:

* TLxxxx gives you the TILINE base address of a device. For disk and tape devices, the last two bits of this number give the unit number.

* CRxxxx gives you the CRU base address of a device.

* INxx gives you the interrupt level of the device.

* CExx gives you the expansion chassis number (appears only if an expansion chassis is present).

* CIxx gives you the chassis position number (appears if device is an expansion chassis).

G.8 INFORMATIVE MESSAGES

The CONVER diagnostic utility may produce the following informative message during execution:

*** CONVER MEMORY HAS ERRORS
MEM ERROR AT >XXXX, IS >XXXX, SHOULD BE >XXXX, BAD BITS ARE >XXXX

CONVER writes this message at the end of initialization if it detects memory errors in the area where CONVER is going to build its device tables.

FD800 SELFTEST FAIL #X <variable part failure message>

CONVER writes this message for each FD800 self-test that fails. The message tells you the self-test that failed and the hardware component that is malfunctioning.

UNRECOGNIZED TILINE DISK CONTROLLER
STORE REGISTER RESULTS ARE:
WORDS/TRACK . . .>XXXX
SECTORS/TRACK . .>00XX
OVERHEAD/RECORD . 00XX
TRACKS/CYLINDER .>XXXX
CYLINDERS/UNIT .>XXXX

CONVER writes this message if it identifies a device as a TILINE disk controller but finds that the controller does not return recognizable store registers values. In other words, CONVER cannot identify the type of disk drive connected to the controller.

*** ERROR: NO INTERRUPT RECEIVED FROM ABOVE DEVICE

CONVER writes this message if it can positively identify a device's type but cannot generate an interrupt from the device.

*** ERROR: UNEXPECTED INTERRUPT RECEIVED, LEVEL X, CHASSIS X, POSITION X

CONVER writes out this message in the following two cases:

* When CONVER receives an unexpected interrupt
* When a device generates multiple interrupts when CONVER only asked for one

In both cases, CONVER performs an I/O reset, since it has no other way to clear the interrupt.

UNCLEARABLE INTERRUPT-RETURN TO DOCS

CONVER writes this message if its attempt to clear an interrupt fails. It aborts the identification process and returns control to DOCS.

EXECUTE EA VERB FIRST

CONVER writes this message if you attempt to execute one of the report verbs (AR or RP) without first executing the EA verb. It also appears if one of the report verbs is requested and the IT verb was executed since the last execution of the EA verb.

Appendix H

Cross-Reference of ASCII Characters With Hexadecimal Codes

ASCII Character	Hex. Code	ASCII Character	Hex. Code
NUL	00	0	30
SOH	01	1	31
STX	02	2	32
ETX	03	3	33
EOT	04	4	34
ENQ	05	5	35
ACK	06	6	36
BEL	07	7	37
BS	08	8	38
HT	09	9	39
LF	0A	A	41
VT	0B	B	42
FF	0C	C	43
CR	0D	D	44
SO	0E	E	45
S1	0F	F	46
DLE	10	G	47
DC1	11	H	48
DC2	12	I	49
DC3	13	J	4A
DC4	14	K	4B
NAK	15	L	4C
SYN	16	M	4D
ETB	17	N	4E
CAN	18	O	4F
EM	19	P	50
SUB	1A	Q	51
ESC	1B	R	52
FS	1C	S	53
GS	1D	T	54
RS	1E	U	55
US	1F	V	56
SPACE	20	W	57
!	21	X	58
.	2E	Y	59
/	2F	Z	5A
?	3F	DEL	7F

Appendix I

S200 Self-Tests

I.1 INTRODUCTION

This section describes the self-test and extended-test diagnostics routines developed for the Business System 200. The section also covers self-test operations, self-test error reports, extended-test operations, and extended-test error reports.

I.2 TEST REQUIREMENTS

The Business System 200 with its associated peripherals is the only hardware required to execute these diagnostics.

I.3 TEST CHARACTERISTICS

The S200 microprocessor Self-Test begins operation when you power-up the system. The self-test checks the logic and memory on the S200 board and begins the initial program load (IPL) operation upon successful completion. The self-test takes approximately ten seconds to execute. The S200 Self-Test completes before the display and keyboard tests; thus when the display and keyboard tests complete, the initialization process begins immediately.

I.4 DESCRIPTIONS OF SELF-TESTS

The following paragraphs describe the self-tests for each of the S200 components.

I.4.1 9900 CPU Test

This test checks the central processing unit (CPU). The CPU test has two parts. The first part checks all the 9900 instructions. The second part checks the addressing ability. The test uses one instruction from each format group to check all applicable addressing modes. If the test succeeds, execution proceeds to check the read-only memory (ROM). If the test fails, the associated error light-emitting diode (LED) remains on, and no further operations occur. The CPU executes an IDLE instruction.

I.4.2 ROM Validity Test

This test checks the S200 ROM for validity by using a cyclic redundancy check (CRC), specifically, a CRC-16 polynomial checksum. This test detects most multibit errors. If the test fails, the associated error LED remains on, and testing halts (IDLE). If the test succeeds, the PBI logic test executes next.

I.4.3 PBI Logic Test

This test checks the Peripheral Bus Interface (PBI) logic for proper operation. If the test fails, the system records the error status and testing continues. If the test succeeds, testing also continues. The PBI remains in a ready state.

I.4.4 Memory Test

The first part of this test checks the memory parity logic, including parity generation and error detection. If the test detects an error, no further memory testing occurs and the self-test proceeds with the next test. The system records the error for reporting at the end of the S200 Self-Test.

If the first part of the test detects no error, the second part begins, checking all memory with four unique patterns. The four patterns cycle to check for addressing errors and bits that fail to switch condition properly. If the test detects an error, the system records it for reporting at the end of the S200 Self-Test. It does not report which address or pattern failed.

I.4.5 TMS9901 Interval Timer Test

This test verifies that the 9901 interval timer is operational by testing with the maximum count timed by an instruction loop. The test checks both the time and interrupt generation. If the test detects an error, the system records it for reporting at the end of the S200 Self-Test.

I.4.6 TMS9901 Interrupt Function Test

This test checks the 9901 interrupt function on interrupt 1 (diagnostic interrupt), interrupt 2 (error interrupt), and interrupt 3 (9901 interval timer). If the test detects an error, the system records it for reporting at the end of the S200 Self-Test.

I.4.7 Communications Option Test

This test checks for the presence of the communications option (COMM-OPTION) and executes further tests if appropriate. The test checks only the functions of the 9902/9903 integrated circuits (ICs). The test executes in the onboard loopback mode. No special loopback connector is needed. If the test detects an error, the system records it for reporting at the end of the S200 Self-Test.

I.5 SELF-TEST COMPLETION

The S200 Self-Test takes approximately ten seconds to run. When the test completes, the error LEDs set by the test do not further change. The S200 now waits until the display and keyboard components are online.

I.5.1 System Initialization

After the S200 Self-Test completes and the display and keyboard components are online, the self-test initializes system parameters to the following values:

* Communication parameters
  - 9600 baud
  - No parity, zero fill-in parity bit
  - Full duplex (DC1/DC3 protocol)
  - No fail-safe disconnect
  - No disconnect
  - Receive parity not checked
  - DEL character for a time-fill character

* Transmission parameters
  - Character mode--no self echo
  - No line turn-around (LTA) transmitted on carriage return
  - TTY mode--echo
* Operational parameters
  - Cursor position--buffer address
  - Cursor wrap on
  - Cursor blink off
  - Attribute select--high intensity
  - Dark background
  - No keyboard lock after an AID byte
  - Short beep for a bell character
  - Attention interrupt off
  - No select--alternate keypad
  - Function keys F1 through F12--nonprogrammable
  - New line = carriage return (CR) key
  - Stop at first end-of-line (EOL) or end-of-file (EOF)
  - Print output--convert nulls to spaces
  - Transmit CR line feed (LF) on margin reads
  - Attributes not transmitted to host
  - Normal row column addressing
  - Erase with nulls
  - Printer - CR LF on EOL
  - Print to AUX-1

I.5.2 Load Sequence

When the system initialization completes, the self-test writes the message BEGIN SYSTEM LOAD to the center of the screen and begins a 2.5-second wait. This wait allows time for entry into the extended-test mode. See paragraph I-10 for more information on entry into extended-test mode.

Upon expiration of the wait time, the normal load sequence begins. The system finds the highest priority write-enabled disk and reads the TRACK1 loader. Execution continues in the TRACK1 loader and then goes on to the operating system itself.

If the self-test completes successfully, you can shorten the 2.5-second wait time by striking any key other than N or ESC (escape).

If the self-test fails, you can attempt an initial program load (IPL) by striking any key other than N or ESC.

I.5.3 Elapsed Time Since Power-Up

Approximately 23 seconds elapse between power-up and the start of reading the TRACK1 loader. The time may be longer if the display component contains optional character ROMs.

I.5.4 Default Load Device Selection

The IPL sequence begins the load on the lowest numbered disk that is write enabled. See paragraph I.12.1 for disk unit and select bit assignment.

I.6 SELF-TEST ERROR REPORTS

The S200 fault LEDs are fail-safe error indicators. Visible through the ventilation grill on the top right-hand side of the display cabinet, these LEDs provide error indication even if the display and keyboard components fail their self-test.

I.7 FAULT LED ASSIGNMENT

Each lit LED illuminates a number on a clear plastic insert above the LEDs. The illuminated number indicates a specific test condition or failure, as shown in the following chart:

The green LED (#1) is the LED closest to the front of the display cabinet. The red LED (#8) is the LED farthest away from the front of the cabinet.

LED Number	Self-Test Indication (if on)	LED Color	
8	COMM Option failure	Red	(back)
7	PBI failure	Red	
6	9902 failure	Red	
5	9901 failure	Red	
4	RAM failure	Red	
3	ROMCRC failure	Red	
2	CPU failure	Red	
1	Self-Test success	Green	(front)

I.7.1 LED Sequence During Self-Test

The LEDs appear in the following sequence:

1. All LEDs light when you power-up the system.

2. LED-1, the green LED, turns off immediately. If the CPU fails to function, all the LEDs remain lit.

3. LED-2, the first red LED, turns off after the CPU passes the self-test. If the CPU fails the self-test, this LED remains on and no further testing occurs. Thus, all the red LEDs remain lit.

4. LED-3 turns off after the ROM passes the CRC validity test. If the ROM test fails, this LED remains on and no further testing occurs. Thus, only LED-1 and LED-2 are off. This test takes approximately one second to complete.

5. LED-7 turns off after the PBI onboard logic passes the self-test. If the PBI fails, this LED remains on, but pass or fail, testing continues. Note that this test is out of LED numerical sequence.

6. LED-4 turns off after the random-access memory (RAM) passes the self-test. If the RAM fails, the LED remains on and testing continues. RAM failure could be serious enough to cause S200 Self-Test failure, disallowing any other tests to proceed. This test takes approximately 8.5 seconds to complete.

7. LED-5 turns off after the 9901 passes the self-test. If the 9901 fails, this LED remains on, but pass or fail, testing continues.

8. LED-6 turns off after the 9902 passes the self-test. If the 9902 fails, this LED remains on, but pass or fail, testing continues.

9. LED-8 turns off after the COMM-OPTION board either is found to be absent or passes the self-test. If the COMM-OPTION fails, this LED remains on, but pass or fail, testing continues.

I.8 SELF-TEST ERROR MESSAGES TO DISPLAY

As the self-test proceeds with the minor fault tests (LEDs 3 through 8), the system records descriptive messages for presentation on the display screen.

The system presents these self-test failure messages on the display screen after it is initialized. The test-specific error messages appear in the center of the display screen. The message SELF TEST FAILURE appears in high-intensity characters at the right end of line 24 (bottom right corner of your screen). The terminal sounds a short tone (a beep) when presenting the message.

The self-test error messages are listed in the following paragraphs in the order of possible occurrence. If you receive a message occurring later in the list, the earlier tests have all passed successfully.

I.8.1 PBI Test Failure Message

PBI FAILURE

This message indicates that the check of the PBI logic failed.

Possible causes of failure include a PBI cable plugged in improperly, a bad PBI cable, a malfunctioning PBI formatter board, PBI parity generation/checking failure, device attention interrupt failure, or failure of the line drivers.

I.8.2 RAM Test Failure Messages

RAM PARITY FAILURE

This message indicates that the RAM parity generation or checking logic has a fault. If this occurs, no further RAM tests execute.

Possible causes of failure include chip failure or timing failure.

RAM FAILURE

This message indicates that a memory location failed to store a test pattern properly. The location of the error is not reported.

Possible causes of failure include chip failure, bad address decode, timing failure, or stuck bit fault.

I.8.3 9901 Test Failure Messages

9901 TIMER FAILURE

This message indicates that the timer function of the 9901 has failed.

Possible causes of failure include bad 9901, faulty CRU logic, or system timing failure.

9901 INTERRUPT FAILURE

This message indicates that the interrupt function of the 9901 failed.

Possible causes of failure include bad 9901, faulty CRU logic, or system timing failure. s1

I.8.4 9902 Test Failure Message

9902 FAILURE

This message indicates that the 9902 (used to communicate

with the display and keyboard components) has failed.

Possible causes of failure include bad 9902, faulty CRU decode, or a bad 9901.

I.8.5 COMM-OPTION Test Failure Messages

COMM-OPTION TOTAL FAILURE

This message indicates that the CRU bit indicating the presence of the COMM-OPTION is true and the test found no responding device to test.

If this error occurs and the COMM-OPTION is really not present, possible causes of failure include CRU fault or incorrectly connected board.

COMM-OPTION ERROR AT >B00 FAILURE

This message indicates the CRU base address of the port failing the test. This message occurs when a device is responding but is not identifiable as a 9902 or a 9903. The same message can occur for the CRU base address of >B80.

Possible causes of failure include a bad 9902 or 9903, improper IC installation, or CRU failure.

NOTE

A value preceded by a right angle bracket (>) indicates a hexadecimal value.

COMM-OPTION 9902 AT >B00 FAILURE

The CRU base address of the 9902 is >B00. This indicates that the 9902 has failed its timer tests or data pattern tests. Note that it did respond enough to be identified as a 9902. The same message can occur for the CRU base address of >B80.

Possible causes of failure include a bad 9902, improper IC installation, or CRU failure.

COMM-OPTION 9903 AT >B00 FAILURE

The CRU base address of the 9903 is >B00. This indicates that the 9903 has failed its timer tests or data pattern tests. Note that it did respond enough to be identified as a 9903. The same message can occur for the CRU base address of >B80.

Possible causes of failure include a bad 9903, improper IC installation, or CRU failure.

I.9 EXTENDED-TEST OPERATION

The following paragraphs discuss the extended-test mode of operation.

I.10 ENTERING EXTENDED-TEST MODE

If the self-test succeeds, you can press the N or ESC key before the beginning of the IPL to place the system into the extended-test mode. Selection of the extended-test mode must take place during the 2.5-second interval between appearance of the message BEGIN SYSTEM LOAD and the actual beginning of IPL.

If the self-test fails, you can press the N or ESC to place the system into the extended-test mode. The system does not attempt to execute an IPL if the self-test detects errors.

If the system detects the PBI run-in connector, the system enters the extended-test mode automatically at the end of the self-test. The system immediately displays the extended-test menu.

If you do not want to enter the extended test mode, press any other key, and the IPL process begins immediately.

I.11 EXTENDED-TEST MENU

The extended-test menu appears at the top of the display screen when the system enters the extended-test mode, as follows:

F1 - LOAD DEVICE	F4 - 9902/940	F7 - KEY ECHO
F2 - TEST & BOOT	F5 - COMM-OPTION	F8 - IO TESTS
F3 - RAM TEST	F6 - PBI TESTS	F9 - SET TEST LOOP MODE

I.12 EXTENDED-TEST SELECTIONS

After the extended-test menu appears, press one of the special function keys to select the desired function. These functions include selection of a previously defined load device, return to self-test mode, or selection of various extended-tests.

The following paragraphs describe the results for each special function key (Fn) selection.

I.12.1 Load Device

If you press the F1 key, the select device for the IPL message SELECT - {F1,...,F8} = appears on the screen below the menu.

Press the appropriate function key (F1 through F8) to select the desired load device. The IPL begins immediately after load device selection.

The function key to disk unit standard assignment is as follows:

Disk Unit	Function Key	Select Bit Number
DS1	F5	4 (>08)
DS2	F6	5 (>04)
DS3	F7	6 (>02)
DS4	F8	7 (>01)
DS5	F1	0 (>80)
DS6	F2	1 (>40)
DS7	F3	2 (>20)
DS8	F4	3 (>10)

The default load sequence follows the order in which the disk units appear in the preceding list. In the default sequence, the system looks for the first write-enabled disk unit, beginning with DS01. It then attempts to load from that disk unit.

The system ignores magnetic tape units during the default IPL process. However, if a magnetic tape is specified by the function key, the system attempts an IPL from that device. The tape should contain the tape equivalent of a TRACK1 loader in its first record.

To return to the extended-test menu without selecting a device, press the ESC key. To begin the default load sequence, press the F2 key, as described in the following paragraph.

I.12.2 Test and Boot

If you press the F2 key, the restart IPL process message RESTART... appears on the screen below the menu.

The system immediately returns to the beginning of the self-test as if the unit was just powered-up. At the end of self-test, the system begins the default IPL process unless you return the system to the extended-test mode.

I.12.3 RAM Test

If you press the F3 key, the extended-test message RAM TEST appears on the screen below the menu.

This test takes approximately 30 seconds to complete. The RAM test reports only the first occurrence of an error in a memory test area. Testing continues with the next area. The system tests the memory parity logic first. Then it tests the RAM, using 14 different test patterns. The memory test areas are >7C00 through >7EE0, >4 through >7BFE, and >7000 through >FFF6.

I.12.4 9902/940

If you press the F4 key, the extended-test message 9902-940 TEST appears on the screen below the menu.

This test exercises the 9902/940 input/output (I/O) port. This I/O port is used for communication between the 9902 and the display and keyboard components. Test execution consists of running the following tests:

* 9902 chip test
* Loopback test
* Cursor position test
* All display and keyboard self-tests

At the end of this test, a ROM status message appears with the following format:

ROM STATUS: ssssss

The six characters give the ROM status. See paragraph I.15.2 for more information.

I.12.5 COMM-OPTION

If you press the F5 key, the extended-test message COMM-OPTION TEST appears on the screen below the menu.

This tests the 9902 and 9903 in internal loopback mode. The system displays the message COMM-OPTION NOT PRESENT on the screen below the menu if the communications option is not installed.

I.12.6 PBI Tests

If you press the F6 key, the PBI operational test message PBI TEST appears on the screen below the menu.

This test asserts reset on the PBI (to get an effective onboard loopback mode) and checks all 256 8-bit data patterns.

This test communicates with each device present. It executes the following series of PBI commands to each responding device with the exception of magnetic tape devices:

* Restore
* Diagnostic Loopback
* Return Peripheral Parameter Block
* Store Registers

The system checks responses for any completion errors. Only the diagnostic loopback data is checked for validity. The test checks all data transfers for the correct record length.

I.12.7 Key Echo

If you press the F7 key, the extended-test message KEY ECHO TEST appears on the screen below the menu.

This test reads the terminal keyboard and displays the input characters on the video screen.

If you press the ESC key twice, the key echo test terminates.

Pressing the ESC key followed by any other key or key sequence results in that sequence being sent to the display and keyboard components. This allows sending of special escape sequences without using the offline mode.

I.12.8 IO Tests

If you press the F8 key, the extended-test message IO TESTS appears on the screen below the menu.

These tests check the Portable Test System (PTS) loopback, output to the AUX port, and output to the screen.

If the PTS is present, the test sends the same escape sequence the display and keyboard components use to do a loopback test. The sequence used is: ESC G ESC (test pattern >20 through >7E ESC). The PTS replies with the test pattern.

The test then writes a rotating print pattern to the AUX port. The pattern consists of 60 lines of print. Each line consists of 78 characters followed by a CR LF sequence. The characters written are from >20 through >7E (ASCII character sequence of a blank through the tilde character).

The test then writes 20 lines of the same rotating print pattern to the display screen.

I.12.9 Set Test Loop Mode

If you press the F9 key (shift F1), the extended-test message ENTER LOOP-COUNT AND TESTS appears on the screen below the menu. Enter instructions for this test in the following format:

nnnnn Fn Fn Fn Fn Fn CR

where:

nnnnn	is a series of digits that indicates the number of times the test(s) should loop.

Fn Fn Fn Fn Fn	is a series of Fn keys, entered in the order of desired testing.
CR	is the carriage return key that terminates the loop count/test list.

The loop count you specify is displayed on the screen when the system receives the CR key indicating the end of instructions. The maximum number of test loops allowed is 65,535. (Any entry larger than this results in a loop count of 65,535.) If no loop count or a loop count of 0 is specified, the test loops indefinitely (65,535 passes if uninterrupted). Press the F1 key to terminate the loop upon completion of the current test.

Only tests F3 through F8 are allowed in the test list. The key sequence #= is also allowed. The #= sequence is the functional equivalent of the F13 key. This sequence reports the current test loop status and can be used as often as desired in the test loop sequence. The maximum number of Fn keys allowed in the list is 32. If no Fn keys are specified, tests F3 through F6 are executed in sequence for each pass through the loop.

Upon receipt of the CR, the system displays the following message: n * TESTS [test list]. The n is the specified loop count and the [test list] is a list of the requested test numbers, or 3-6 if the default test sequence is selected. The first test begins after output of the test loop information. Press the #= key sequence to see the current test loop status. Status is reported upon completion of the current test. Testing continues after the status report.

Press the ESC key before pressing the RETURN key to clear the instruction information and restart the input process. Press the ESC key twice to exit the loop mode setup sequence and return the system to the extended-test menu selection state.

After the specified number of test loops or other termination of the test sequence, the system reports the total number of errors encountered and the total number of loops executed. This report message appears as follows:

n ERRORS m PASSES

where:

n represents the number of errors.

m represents the number of passes (loops).

The system displays the extended-test menu for further selection.

When the system executes extended-tests in the loop mode, it suppresses test header messages. A summary at the completion of the loop reports any error messages. Individual non-loop tests report a test header and error messages. A test header followed by the extended-test menu indicates successful completion of the test.

I.12.10 Offline Function

If you press the F11 key (shift F3), the special message OFFLINE FUNCTION ENABLED appears on the screen below the menu.

This test sends the proper sequences to the display and keyboard components to allow the ALT 4 key sequence to set the display and keyboard to the offline mode. The display and keyboard components retain this ability until power is cycled. This special function cannot be performed in the loop test mode.

I.12.11 Output Pattern Test

If you press the F12 key (shift F4), the special output pattern test appears on the display screen.

This test sends a continuous stream of the letter O to the display screen. If you press any key on the keyboard, this test terminates. This special test cannot be performed in the loop test mode.

I.13 Fn KEY REPRESENTATIONS

When you select the extended-test mode, the S200 sets the function keys to generate a specific string when pressed. Each function key generates a two-byte string. The first byte is always a # (number sign, >23). The second byte is the output value of >30 plus n.

The following list shows the actual string produced when you press a given function key:

F1 - #1	F4 - #4	F7 - #7	F10 - #:
F2 - #2	F5 - #5	F8 - #8	F11 - #;
F3 - #3	F6 - #6	F9 - #9	F12 - #<

Note that F9 through F12 are shifted keys. F10 is not used. In addition to the preceding sequences, the #= key sequence is used as the functional equivilent of F13. (The keyboard electronics do not allow F keys beyond F12 to be set to generate a string.) You can obtain any of the Fn key functions by pressing the equivalent key sequence as previously listed.

I.14 EXTENDED-TEST ERROR REPORTS

After completion of the individual tests, the error messages appear below the test header that appears below the extended-test menu. When operating in loop mode, the system does not write test headers, only error messages.

I.15 EXTENDED-TEST ERROR MESSAGES

The extended-test error messages are listed in the following paragraphs in the order of possible occurrence by function key sequence.

I.15.1 F3 -- RAM TEST ERROR MESSAGES

RAM PARITY FAILURE

This message indicates that the system detected a fault in the RAM parity logic. Further testing of RAM is skipped after this error.

Possible causes of the error include chip failure, CRU failure, or timing failure.

RAM ERROR: aaaa, rrrr, pppp, eeee

aaaa is the RAM address that has an error.

rrrr is the incorrect value that was read from RAM.

pppp is the correct test pattern that should have been in RAM.

eeee is the word in error. Any 1 in this word indicates a bit in error.

Further testing of the particular test area is skipped after the first error.

Possible causes of the error include bad RAM chip, bad address logic, bad data path, or bad timing.

RAM PARITY ERROR DETECTED

This error indicates that a parity error occurred in a particular test area. If no data error was reported, the data patterns have completed correctly, and the RAM chip associated with parity is suspect.

Possible causes of failure include bad RAM chip, bad address logic, bad data path, or bad timing.

I.15.2 F4 -- 9902/940 ROM Status Message

ROM STATUS: ssssss

ssssss is the display and keyboard components ROM status reply. This contains the information as to the ROM version and revision number. Refer to the OPTI 900 Model 940 Electronic Video Terminal Systems Manual, part number 2213826-9701, for more information.

This message is skipped if the test is executed in loop mode.

I.15.3 F4 -- 9902/940 Error Messages

940 LOOPBACK TEST FAILURE

The display and keyboard components failed to correctly execute the Extended Write Loopback command. This test checks character patterns from >20 through >7E.

940 CURSOR POSITION TEST FAILURE

The display and keyboard components have not correctly executed or replied to a series of commands to read and position the cursor.

940 SELF-TEST REPLY FAILURE

The display and keyboard components have not replied in 80 seconds to the Extended Write command to execute a self-test.

940 SELF-TEST ? FAILURE

The display and keyboard components have replied to a self-test command with a failure indication. The question mark (?) is the test character (C, D, E, F, G, H, or I). See the 940 EVT Systems Manual for a description of the self-test.

I.15.4 F5 -- COMM-OPTION Error Messages

COMM-OPTION TOTAL FAILURE

This message indicates that the CRU bit indicating the presence of the communications option is true and the test found no responding device to test.

If this error occurs and the COMM-OPTION is not present, possible causes of failure include CRU fault or incorrectly connected board.

COMM-OPTION ERROR AT >B00 FAILURE

This message indicates the CRU base address of the port failing the test. This message occurs when a device is responding but is not identifiable as a 9902 or a 9903. The same message can occur for the CRU base address of >B80.

Possible causes of failure include a bad 9902 or 9903, improper IC installation, or CRU failure.

COMM-OPTION 9902 AT >B00 FAILURE

The CRU base address of the 9902 is >B00. This indicates that the 9902 has failed its timer tests or data pattern tests. Note that it did respond enough to be identified as a 9902. The same message can occur for the CRU base address of >B80.

Possible causes of failure include a bad 9902, improper IC installation, or CRU failure.

COMM-OPTION 9903 AT >B00 FAILURE

The CRU base address of the 9903 is >B00. This indicates that the 9903 has failed its timer tests or data pattern tests. Note that it did respond enough to be identified as a 9903. The same message can occur for the CRU base address of >B80.

Possible causes of failure include a bad 9903, improper IC installation, or CRU failure.

I.15.5 F6 - PBI TESTS Error Messages

PBI FAILURE

This message indicates that the check of the PBI logic failed.

Possible causes of failure include PBI cable plugged in improperly, bad PBI cable, malfunctioning PBI formatter board, PBI parity generation/checking failure, device attention interrupt failure, or failure of the line drivers.

NO PBI DEVICE PRESENT

There was no reply to a PBI reset command.

Possible causes of failure include PBI cable not plugged in (or improperly plugged in), the disk units are not powered-up, or the PBI device has failed.

PBI UNIT u STATUS ss COMMAND >cccc TIMED OUT

u is the unit number under test.

ss is the most recent status byte from that unit.

cccc is the command code to the formatter, and it has not responded in five seconds. See PBI documentation for a description of the code.

PBI UNIT u STATUS ss WRONG NUMBER OF DATA BYTES

u is the unit number under test.

ss is the most recent status byte from that unit.

The formatter/device has replied with an incorrect number of data bytes.

PBI UNIT u STATUS ss ABNORMAL COMPLETION ddee

u is the unit number under test.

ss is the most recent status byte from that unit.

ddee is the abnormal completion code. The high byte dd is the device selection bit. The low byte ee is the error code described in paragraph I.5.3.

PBI UNIT u STATUS ss DEVICE SELF-TEST ERROR >eeee FAILURE

u is the unit number under test.

ss is the most recent status byte from that unit.

eeee is the self-test error code. See the mass storage device documentation.

PBI UNIT u STATUS ss LOOPBACK TEST FAILURE
u is the unit number under test.

ss is the most recent status byte from that unit.

The Diagnostic Loopback command failed to verify.

I.16 PBI STATUS ERROR CODES

PBI status error codes that can occur in the status byte are listed along with the error code meanings as follows:

Code	Meaning
>01	Command validity error
>02	Receive data parity error
>03	Time-out--receive
>04	Time-out--send
>05	Status byte count error
>06	Status received while command active
>07	Loopback in byte count error
>08	Loopback test failure
>09	Formatter detected bus error
>0A	Data byte count error
>0B	CSB byte count error
>0C	Invalid unit responding
>0D	Action byte parity error
>0E	Time-out busy and ready
>80	DAB present and no action byte

Appendix J

Loopback Modes

J.1 INTRODUCTION

The communications diagnostic tests can take various paths through the communications system under test. The path selected depends on the loopback mode that you choose during test initialization. This appendix explains which system components are tested in each mode and the prerequisites to the use of each mode.

The communications tests run in either internal (IN) mode or one of three external modes:

* digital loopback (DI) mode
* analog loopback (AL) mode
* far-end loopback (FE) mode

When a loopback connector, modem, or transmission medium is installed, the test may be placed in an external mode; the test must be run in internal mode when none of these devices is present. Most tests require you to use the loopback connector that has LEDs (part number 948550-0001, revision *C or later). The LEDs on the connector indicate the condition of bit signals that are not looped back. The LED connector also loops back an internal test clock to the transmit and receive interfaces. It allows full external loopback testing in synchronous mode. Figure I-1 illustrates the signals that are looped back and the pin assignments in the loopback connector.

It is recommended that you run the test first in internal mode and then progress through digital, analog, and then far-end mode if the system configuration permits and you have the necessary test equipment. Table I-1 lists the parts of the system that are tested in each test mode. Figure J-2 illustrates the testing path each mode takes.

The system under test may be entirely local or it may be a local system connected by transmission media to a remote CPU and communications interface in a different area. The transmission medium may be a telephone line to another city or a twisted cable to the next room.

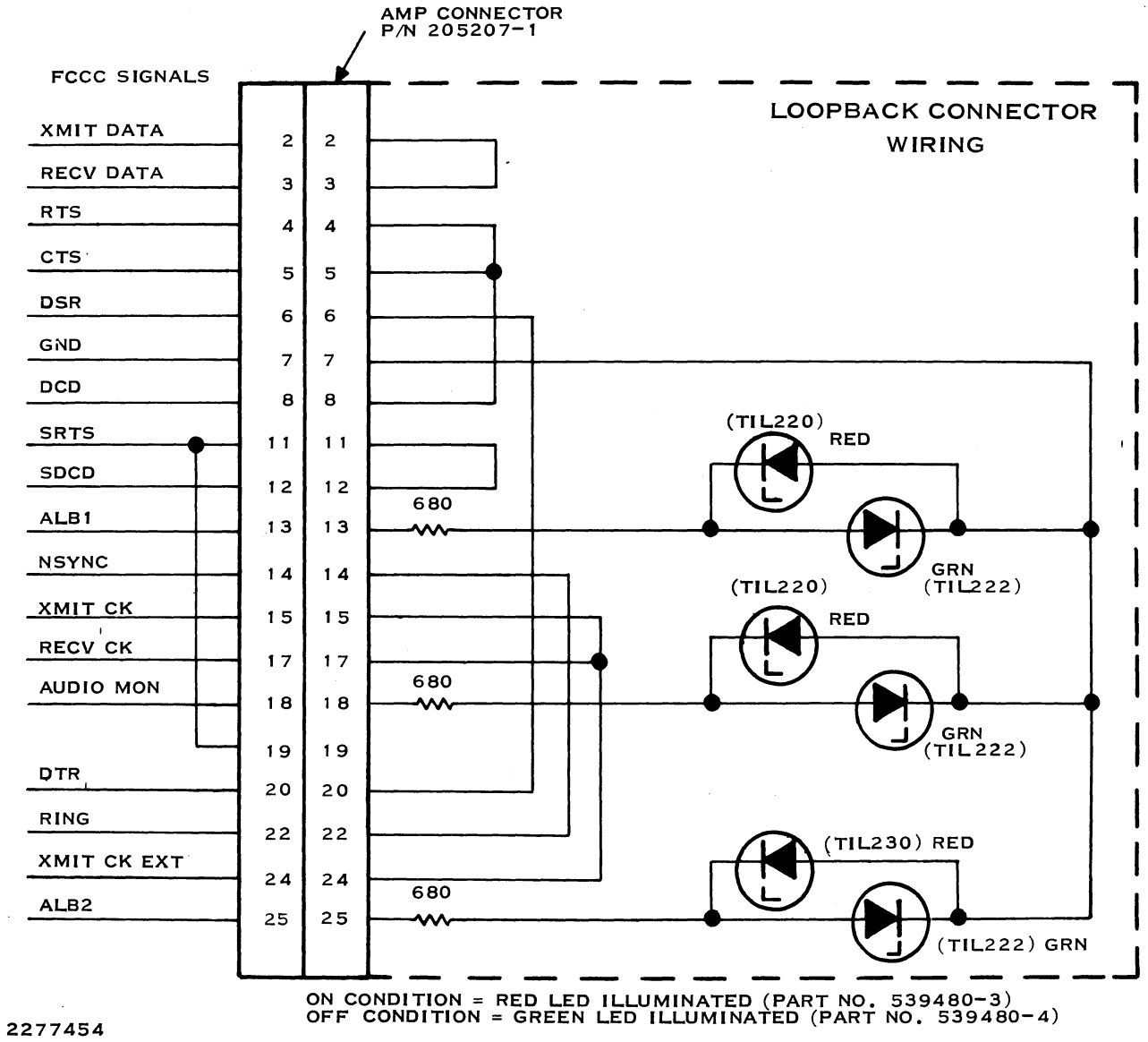


Figure J-1 Loopback Connector Wiring

Table J-1 System Components Tested in Test Modes

Test Mode	Component Tested				
	Local Interface	Local Modem Cable	Local Modem	Transmission Medium	Remote Modem
Internal	yes	no	no	no	no
External:					
Digital	yes	yes*	no	no	no
Analog	yes	yes	most	no	no
Far-End:					
A-local line loopback	yes	yes	yes	no	no
B-remote line loopback	yes	yes	yes	yes	no
C-remote modem analog loopback switch	yes	yes	yes	yes	partial
D-remote CIM with RT verb	yes	yes	yes	yes	yes

Note:

Only applicable if the loopback connector is used on the modem end of the cable

J.1.1 Internal Loopback Mode

Internal mode loops back digital signals on the local communications interface, testing most of the on-board circuitry. No additional hardware or special test equipment is required. This mode can be used when you want to eliminate the CPU and communications interface as the source of errors found when a test is executed under another test mode.

J.1.2 Digital Loopback Mode (External)

Digital mode tests the local communications interface and, optionally, the interface-to-modem cable. Install a loopback connector at the modem end of the cable to test both the cable and the interface or at the interface connector to test only the interface.

J.1.3 Analog Loopback Mode (External)

Analog mode loops back signals at the output of the local modem and tests the communications interface circuitry, the interface-to-modem cable, and most of the local modem. The transmission medium and any hardware beyond the local modem are not tested in this mode.

Analog mode can be used only with TI modems or with vendor modems that have a manual analog loopback switch. Vendor modems do not normally respond to the analog loopback signal.

J.1.4 Far-End Mode (External)

Far-end mode tests the local modem and the components beyond this device. This mode is used when the modems on both ends of the system are capable of full-duplex operation and a four-wire line is used between them. In this case, the far-end modem can be placed in digital loopback mode to allow signals to be transmitted to the far-end modem and looped back to the local system. This provides a quick check of the local system, the transmission medium, and the far-end modem.

A few modems (such as the Bell 212A) are capable of full-duplex operation on two-wire systems, including the switched telephone network. If this type modem is used at both the local and remote sites in a system, digital loopback of the far-end modem is possible on a two-wire system.

Far-end loopback mode has a loopback path at the remote modem where the analog signals transmitted to the remote modem are converted to digital signals, looped back, converted back to analog, and then transmitted back to the local system.

There are four loopback points available when using far-end mode. The points that can be used depend on how the system is configured. These points are discussed in the following paragraphs.

J.1.5 Point A -- Local Line Loopback

When you make a loopback connection at this point, signals are looped back at the local modem and that unit is included in the testing. (See item A in figure J-2.)

J.1.6 Point B -- Remote Line Loopback

When you make a loopback connection at this point, the test transmits signals through the transmission medium up to but not including the remote modem and then loops these signals back. (See item B in figure J-2.)

J.1.7 Point C -- Remote Modem Analog Loopback Switch

A loopback connection at this point loops back signals at the remote modem. (See item C in figure J 2.)

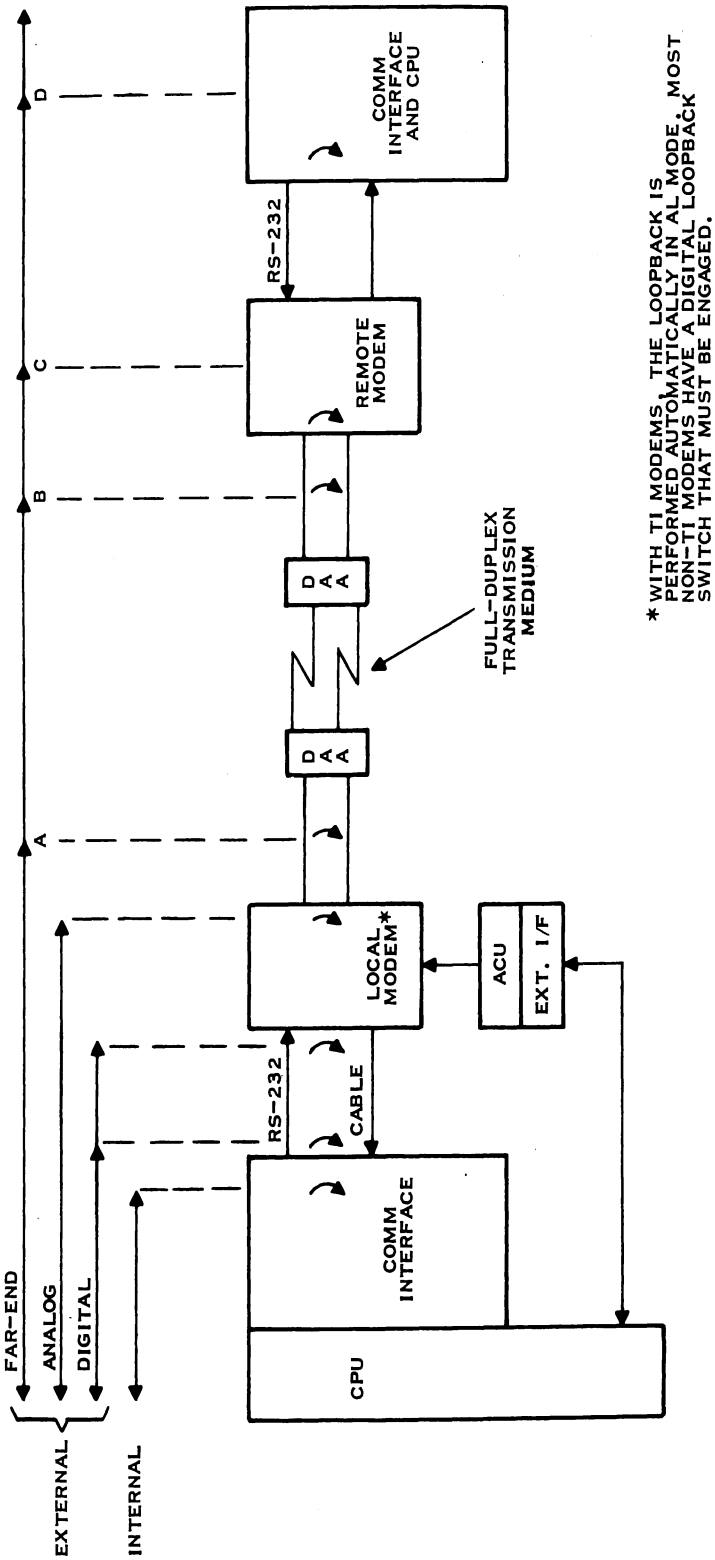


Figure J-2 Additional Loopback Connector Wiring

Appendix K

Operation of the Universal Systems Terminal (UST)

K.1 INTRODUCTION

This appendix describes the operation of the Universal Systems Terminal (UST), which replaces the loading capabilities of the Maintenance Diagnostic Unit (MDU), and serves as an interactive device for the actual running of DOCS and all diagnostic tests.

K.1.1 Description

The UST is a DSG Field Service tool designed to increase the productivity of field personnel. It consists of a TI PRO-LITE(TM) computer with at least 256K of memory, one 720K microfloppy disk drive, and special hardware and software.

K.1.2 The Front Panel Monitor (FPM)

The Front Panel Monitor is a special UST software package that allows you to manipulate any 990 system through an emulated programmer's panel and to run diagnostics on this system. The FPM replaces the MDU, which contains an actual 990 programmer's panel and tape loading system. The FPM is used in instances where the existing 990 front panel is not working or is not there, and when other load devices, such as the T-50, CD1400, or FD1000, cannot be used.

The FPM replaces the standard 990 programmer's panel with a video display that shows a panel at the top of the screen. It contains the four main indicator lights (POWER, RUN, IDLE, and FAULT), the 16-bit data display, and the hexadecimal data display that is found on the newer Business System front panels. The command keys and switches are replaced by program commands that can be typed in or activated by pressing function keys on the UST.

In addition to reduced size and quantity of media, the diagnostics downloading capability of the MDU is duplicated in the FPM package of the UST, with significant improvements in speed and reliability. The loading process is also made simpler to save time and trouble.

PRO-LITE is a trademark of Texas Instruments Incorporated.

The FPM also includes advanced features which vastly increase its usefulness. For front panel work, long sequences of commands called "fat finger" routines can be programmed to execute automatically with a single command. For diagnostics, the UST can be used as an I/O and error message device, allowing you to run diagnostics on a system with no functional load or I/O devices.

K.1.3 Hardware

Other than the PRO-LITE computer, hardware for the UST consists of the Front Panel Interface (FPI) module and the FPI cable.

CAUTION

The PRO-LITE must be powered down before you plug in the FPI module. Inserting or removing any option module while the power is on can seriously damage your computer.

The FPI cable can be attached or disconnected at any time, allowing you to install the module without connecting the cable, and close the module up for later use.

NOTE

The FPI cable can be connected to the system being tested while that system's power is on.

The FPI module is a typical PRO-LITE option card with a DB-25 socket. This module looks just like the Asynchronous Communications (AC) module, so be sure to check the label on any module before using it. The FPI can be plugged into either of the option slots on the left side of the PRO-LITE. It is preconfigured to use I/O address block >0200->02FF. When using two option modules, make sure that the addresses do not conflict. The reference guides for the UST or PRO-LITE modules list the addresses used by each module. Communications and Modem modules do not cause conflicts with the FPI module since they use a different address scheme.

Should a conflict occur, you can change the I/O address block of the FPI by changing a jumper on the board. Remove the four hexagonal-head screws in the module casing and lift off the cover. Then remove the board carefully and orient it so the component side faces up and the DB-25 external connector faces away from you. The jumper to be moved is directly above the IC U4 and has pins labeled E1-E5 (left to right). The default setting connects E1 and E2. Table K-1 lists the correct jumper

settings for addresses >0100, >0200, and >0300. After resetting the jumper, carefully turn the board over and fit it into the case. Then, replace the bottom of the case and refasten it.

If you set the address to something other than >0200, you should mark the address on the outside of the module to avoid confusion at a later date.

Table K-1 FPI Module Jumper Settings

I/O Address	Connect Pins
>0100	E1 and E2
>0200	E2 and E3 (or E3 and E4)
>0300	E4 and E5

The FPI cable is specially built for the Front Panel Monitor. The DB-25 plug connects to the FPI module, and the 20-pin receptacle is identical to the standard front panel connector. The red or blue stripe on one side of the ribbon cable should be on the right or the top when it is plugged into the test system CPU board, depending on the orientation of the board in the particular chassis. For information on chassis types, consult Appendix C of the UST Front Panel Monitor Reference Manual, Version 1.0, TI part number 2533996-0001. In all cases, the stripe should be oriented in the same direction as the front panel cable it is replacing.

K.1.4 Software

The FPM diskette contains the file FPM.EXE, which is the actual Front Panel Monitor software and some useful macros. Make backup copies of each microfloppy diskette in the package using either of the standard MS-DOS commands, COPY or DISKCOPY. The contents of each diskette are indexed on page 7-6 of this manual.

If you have installed the FPI module, with or without attaching the cable, you can now power on the UST. The FPM diskette is bootable, so the standard MS-DOS messages are displayed after a few seconds.

Answer the date and time prompts. The computer will wait for your command after displaying an "A>" prompt. To start the Front Panel Monitor, answer as follows:

```
A> FPM m
```

where:

m is a mode parameter from the following list.

F .... Front Panel Emulation (FPE)
R .... FPE; Reboot the System
L .... File Loader
C .... DOCS Conversation

Either F or R will start the Front Panel Emulation mode, the only difference being that R causes the system to reboot, while F does not. L selects the download (loader) mode. C starts the DOCS conversation mode in which actual tests are run. C is seldom if ever used in the initial stages of UST sessions, since DOCS must already be loaded before C can be used.

If you are starting with either the L or C mode, the system being tested must be powered up. Even with R mode, you will have problems if the 990 power is off. FPM will return an error message stating that you cannot reboot a powered-down computer. You do not need to power up the test system if you are using the F mode, but you will still get an error message.

To leave the program at any time, press the ESC key once in FPE mode, or twice if downloading a test or conversing with DOCS.

K.2 FILE LOADER

The FPM File Loader replaces the load capabilities of the MDU by loading diagnostics control files, diagnostic tests, and other object code files from a UST microfloppy diskette. It provides menu selection of files, easy changing of diskettes, and a fast download process. The loader mode is activated either by using the LOADER command while in Front Panel mode or by specifying the L mode when starting the Front Panel Monitor program.

K.2.1 MDU Comparisons

As mentioned, the MDU uses ordinary cassette tapes to record diagnostic tests for 990 systems. Since cassette recording or playing is an analog process, this method tends to be slow, inefficient, and unreliable. The diagnostics are stored one to a side on each tape because the tape must be set manually to the beginning of the file to be loaded. This is simple if the file is at the start of the tape, but extremely awkward otherwise. For this reason, the cassette-based unit diagnostics library is stored on approximately 50 cassettes.

The UST microfloppies are high-capacity digital recording media. They hold 720K each and are extremely fast and reliable. The entire set of Diagnostics Operational Control System (DOCS) files and diagnostic tests fit on three of these diskettes, which together take up slightly more space than a single cassette tape. The time savings in loading from a microfloppy over loading from the MDU is about 90%. Unlike cassettes, diskettes rarely suffer

from reliability problems like parity errors. Reliability is generally of little concern with diskettes. Damaged or destroyed cassettes must be reordered, whereas damaged or destroyed microfloppies can be copied from backup files using the UST.

K.2.2 Types of Load Files

The file loader works with three types of files:

* DOCS files
* Diagnostic tests
* Other programs in standard 990 object code format

The LOADER command has a parameter, LOADCODE, that represents each of these file types. Each type requires slightly different load circumstances.

K.2.2.1 DOCS Files. A DOCS file is usually the first file type to be downloaded. DOCS files are specialized versions of the operating system required to run diagnostics on any 990 computer or Business System. Each version is tailored to a specific need. Table K-2 shows a list of the names and functions of the DOCS files to be found on UST microfloppy disk #1. All DOCS files have the file extension "DCS".

Table K-2 The DOCS Types

File Name	Function
UMXDOCS.DCS	Main (Maxi-) DOCS file
UMNDOCS.DCS	Mini-DOCS (stripped-down version of UMXDOCS)
UBCDOCS.DCS	Batch-stream DOCS (allows creation and use of batch command streams, much like macros in FPM and batch commands in SCI)
UDBDOCS.DCS	Debug DOCS (allows tracing and debugging of loaded software)

Since DOCS files are not in standard format but rather in a special compressed format, an up-front loader that can translate this format must first be loaded. The program that performs that translation is UNITLDR.LDR. Whenever a DOCS file is to be downloaded, UNITLDR is automatically sent first, and the appropriate DOCS file is loaded. The transfer of UNITLDR takes from 1 to 25 seconds, depending on the system. This compression method greatly reduces the size of the diagnostic files, both DOCS and tests alike. Once DOCS has been loaded, it need not be loaded again until you want either to change message devices or

to use a different DOCS version. Although the front panel display is not updated while UNITLDR is being sent, it does reflect the current address during the download of any standard or specially compressed file, such as the DOCS files.

K.2.2.2 Diagnostic Tests. Since tests are designed to be loaded while under the control of DOCS, a DOCS file must be loaded first. After that, any number of tests may be loaded, one after the other. The microfloppy versions of the tests are also in the special compressed format mentioned in the previous paragraph. The display indicates the progress of the download by showing the current value of the Program Counter (PC), which is updated for each byte sent. All diagnostic tests have the file extension "DIA". The various diagnostic tests are listed and indexed as to microfloppy disk in paragraph 7.2.2 of this manual.

K.2.2.3 Object Code Files. The file loader can also download standard 990 object code in the form of stand-alone programs, small versions of operating systems, and other applications. This is a simplified version of file downloading that makes no assumptions about file use, (as is done for DOCS and test files). All files loaded by this method have the file extension "OS".

K.2.3 Using the Loader

The method of downloading a file to the test system is broken down into three steps:

* Menu display and selection
* Download
* Terminal prompt (optional)

K.2.3.1 Menu Selection. Using the LOADER command from FPE mode or starting FPM in L mode brings up a menu of the files of a specified type available on the diskette. The file type is determined by the parameter given in the LOADER command, or is assumed to be D (DOCS file). The remaining two possible LOADCODE values are T for test, and O for standard object format. Figure K-2 shows a typical menu display of diagnostic tests. When the menu first appears, the file name in the upper left corner is underlined. This "cursor" underlines the current selection. To choose a particular file, use the Arrow keys to move the cursor around the screen. When the cursor underlines the desired file, press the RETURN key. This activates the download process.

```
================================================================================
ADCHK     AU04      AU04TST   AU05      CPU10     CPU12A    CPU12B    CPU12C
CPU12D    CPU12E    CRDRDR    CRUEXP    CRUTIC    CTLTST    DACHK     EMU900
EMU940    EMUTST    EROMBT    FIVMOD    INPMOD    IO16      MAP12     MAPTIC
MAPTST    MEMPRT    MEMTST    OUTMOD    PROMPG    RAM04     TFMTST    TNXTST
TRACE     WATDOG
```

Figure K-1 Sample Diskette Menu

If you wish to exit the File Loader and return to the Front Panel Emulator (FPE) mode, press the ESC key. If you cannot find the desired file on the current diskette, simply remove it and insert one you suspect contains the file. Then press the RETURN key. This tells the menu routine to re-read the diskette directory. If you receive a message saying that no files of the desired type exist on the diskette, press the space bar on the UST. To change the file type, you must exit the loader and re-enter it using another file type. The various diagnostic tests are listed and indexed by microfloppy diskette in paragraph 7.2.2 of this manual.

K.2.4 Download Process

When you press the RETURN key, the menu routine first checks to ensure you have the right diskette in the drive. If not, it displays an error message and returns you to the menu display. If verified, the file name is passed to the actual load routine. If the file is a DOCS file, UNITLDR>LDR is loaded first. (If UNITLDR>LDR is not on the current diskette, an error message is displayed, and you are returned to the file menu.) Then the specified file is transferred, a byte at a time, to the 990 system. The data display indicates the current 990 memory address being loaded. The transfer rate is approximately 2K per second. You cannot abort the download process. If a problem develops, the only recourse is to reboot the UST. If the Loader detects any problems in the download process, it terminates with an error message and exits to the Front Panel mode.

K.2.5 UST Terminal Status

If all goes well with the download, one of several things will happen. If the file type was 0, the Loader will start the menu selection process again, following the above two-step loop until you press the ESC key to terminate it.

If a DOCS file was selected, the Loader asks you whether the UST should act as an I/O and error message device (besides being the loading device). This occurs by changing to DOCS Conversational mode. An N response tells the Loader that until DOCS is reloaded, any normal I/O and errors will be handled through

another terminal. Consequently, Y must be entered on the terminal you intend to use as the I/O device. The Y response works the same as with any other device in telling DOCS that it should be the I/O device. Unlike other devices, however, the UST needs the N response to allow it to continue to be a loading device. The Loader marks which (Y or N) is chosen by setting a terminal status flag that indicates whether or not the UST is to be used as the primary DOCS terminal.

If the terminal status flag is true, FPM exits the Loader and begins Conversational mode. Otherwise, the Loader returns to the menu display for more files to download.

If the file just loaded was a DOCS or test file, the test file menu is shown. Otherwise the standard object code file menu appears again. The Loader will continue to select and download files until you terminate L mode by pressing the ESC key.

K.3 INTERACTIVE DEVICE

The UST functions as an interactive device, as well as a test loader. Loading 990 diagnostics is the primary function of the MDU. It was not designed to operate as a terminal because of the unusual front panel circuitry of the 990. Therefore, you must still locate a functional terminal to use as an I/O device to interact with DOCS.

The FPM, however, makes use of special hardware and software to surmount this problem, and includes the capability to use the UST as a DOCS terminal as well as a loading device. In this way, the only part of the system being tested that must be working is the CPU board itself, along with the loader ROM and sufficient good memory to load DOCS. DOCS Conversational mode is the function of FPM that interacts with DOCS, in a manner very similar to such video devices as the 911 or 931 VDT.

K.3.1 DOCS Conversational Mode Entry

DOCS Conversational mode is entered in one of three ways. The most common is by way of the file loader. If a DOCS file has been downloaded, the Loader asks you if the UST will be the I/O device for the DOCS session. The test system, in the meanwhile, has already started scanning for a Y from all recognized interactive devices, including the UST. By responding to the prompt with a Y, you select the UST as the DOCS terminal. After a short delay while DOCS initializes itself, the screen clears and the usual DOCS header message appears.

Another way to enter Conversational mode is through the CONV command while in the Front Panel mode. This command is only used to reenter DOCS when you have exited the Conversational mode

earlier for some reason (such as to check the current PC or display a block of memory) and utilized the Front Panel mode. You can now return to the Conversational mode by entering CONV and pressing the RETURN key.

The third method of entry is used only if you have to leave FPM completely. Re-enter Conversational mode by starting the Front Panel Monitor in the C mode.

K.3.2 Conversational Mode Use

Use of the Conversational mode is very similar to the use of any other video terminal with DOCS. Figure K-3 shows a sample session with UMXDOCS, the primary unit DOCS. Uppercase represents DOCS messages and prompts. Lowercase, underlined characters are typed in by the operator. Unlike other video terminals, the UST has a type-ahead keyboard buffer that allows you to set up the answers to familiar prompts before you actually see them.

```
==========-============================================================
** MAXI DOCS                VERSION = 265/84          TI990/10  **
TERMINAL BASE ADDRESS - 1FE0
TERMINAL INTERRUPT LEVEL - FF
ERROR MESSAGE DEVICE? (0=ME, 1=PRINTER) DEF = 0 -
ARE THE SELECTED I/O & ERROR MESSAGE DEVICES OK? (DEF = 1) - 1
                                                              ___

                    ****** ATTENTION ******

     DOCS HAS BEEN REDUCED IN SIZE.  IT IS NECESSARY TO
RELOAD DOCS IN ORDER TO CHANGE I/O OR ERROR MESSAGE DEVICES.


LOAD TEST? (DEF=1) -
CASSETTE/UST LOADER
LOAD BIAS? (DEF= 3D1C) -
PLEASE: PLACE CASSETTE IN MDU
        PUSH RESET AND REWIND
        -------- O R --------
        INSERT THE UST MICROFLOPPY
        WITH THE DESIRED DIAGNOSTICS
PRESS "RETURN" WHEN DONE-
=============================================================
```

Figure K-2 Example of DOCS/Operator Interaction

After the DOCS header is printed, DOCS displays the CRU base address and interrupt level of the terminal. Although the CRU base (>1FE0) is correct, the interrupt level (>FF) is misleading. The front panel circuitry on a 990 CPU board does not have any provisions for interrupts, so any device connected to it cannot have an interrupt level. To circumvent this problem, the UST uses its front panel capabilities to HALT the computer and perform a BLWP instruction to an interrupt handler in a manner similar to an actual interrupt trap. The overall effect, from the operator's viewpoint, is indistinguishable from a standard interrupt. The UST's interrupt level is given as >FF to indicate to DOCS that it should not be considered in the normal process of setting and using interrupt vectors.

After displaying this information, DOCS prompts you for an error message device. Usually, the UST is used for error messages. You can choose any standard output device for error messages, however. In Figure K-3 the UST was chosen.

DOCS then asks you to verify your choices. If you respond negatively, you would be prompted for the information again. Once you have made your choices, DOCS proceeds to throw out any nonessential hardware service routines (HSRs) and initialization code to make room for diagnostic tests. DOCS then informs you that it has been reduced in size and cannot be reinitialized without reloading.

At this point, you now have the option to load a test immediately, or to execute any of the various verbs supplied with DOCS.

As illustrated in Figure K-3, you load a test by taking the default response to the LOAD TEST (DEF=1)? prompt. DOCS informs you that this is the Cassette/UST loader, and asks for a load bias. The load bias is the address in test system memory where the test will begin loading. The default is the first memory word following DOCS. Specifying a lower load address causes DOCS to issue a warning that DOCS itself will be overlaid and must then be reloaded. This might be necessary when running a stand-alone test (SAT). Specifying a higher address does no harm, but uses memory inefficiently, and may not leave enough room for the test to load.

The next prompt waits for you to make sure you have the correct diskette in the UST drive. If the test you wish to run is on another diskette, now is the time to exchange diskettes. You then simply wait for the test menu to appear and change test selections.

When you are ready, press the RETURN key. A message flashes briefly on the screen and is replaced by the diagnostic test menu which you will recognize from the File Loader mode. The brief message is of concern only to MDU users, and can be ignored if you are using a UST.

K.3.3 Loading Diagnostic Tests

At this point, you are once again in the File Loader mode. You now select the diagnostic test you wish to run. As mentioned earlier, you may also change diskettes and read the new diskette directory. Once you select a test and press the RETURN key, the test you selected will be downloaded to the test system.

Once the test has been loaded, DOCS regains control. DOCS asks the questions appropriate to the test chosen and then transfers control to the test. The test will interact with the UST just as DOCS does. Details of this interaction vary from test to test. Figure K-4 shows a sample interaction between the operator and the CRT911 diagnostic test. For particulars of the interaction on any given test, consult the Unit Diagnostics Handbook volume (I through VII) which deals with that particular test.

```
*** DOCS Conversational mode ***

END OF TEST = C5F2

CRT-911 - 911 CRT DIAGNOSTIC. VERSION # =  086/84 *J

CRT UNDER TEST CRU BASE ADDRESS? DEF = 0100 - 0120
                                                ____

RUN WITH INTERRUPTS? DEF = 1 -
CRT UNDER TEST INTERRUPT LEVEL? DEF = 000A - 8
                                                ___

ENTER LINE FREQ (0=50, 1=60) DEF = 1 -
ENTER CONTROLLER FREQ (0=50, 1=60) DEF = 1 -
UNIT UNDER TEST IS DOCS I/O DEVICE? DEF = 0 -
CONTROLLER CHARACTER SET OPTIONS:
 AR = ARABIC              JK = JAPANESE/KATAKANA
 DN = DANISH/NORWEGIAN    SF = SWEDISH/FINNISH
 FR = FRENCH              UK = UNITED KINGDOM
 GR = GERMAN              US = UNITED STATES
 FP = FRENCH PROCESSOR
 CONTROLLER CHARACTER SET? DEF = US -
 REPORT FAILURES BY FIELD REPLACEABLE UNIT? DEF = 0 -
 KEYBOARD PRESENT? DEF = 1 -
 RUN DIAGNOSTIC ON BOTH HALVES OF DUAL CONTROLLER? DEF = 0 -
 EXECUTE EA VERB? (DEF=1) -
```

Figure K-3 CRT911 Test Prompts

Once the test is finished, control passes back to DOCS and the VERB? prompt is displayed. You are free to execute any verb, including .LD, to load another test. You can also terminate DOCS at any time, as described in the following paragraphs.

K.3.4 Special Keys

DOCS recognizes the use of several special keys, and FPM recognizes some for its own purposes. One of these is the ESC key, which exits Conversational mode and returns you to Front Panel mode. This does not terminate the DOCS session. You can return to Conversational mode and the DOCS session by using the CONV command, unless DOCS attempts to write messages to the UST while you are in the Front Panel mode. If this is the case, then your DOCS session has timed out.

Another useful key is the "@" key, which is the usual DOCS abort key. It aborts the current test or prompt exactly as with any other I/O device.

NOTE

There is no PAUSE key currently defined for DOCS on the UST. At present, using the UST's PAUSE key causes the PRO-LITE to lock up.

The Back Space and Left Arrow keys on the PRO-LITE keyboard do not currently work with DOCS in the UST or FPM sessions.

Index

This index lists key topics of this manual and specifies where each topic appears, as follows:

* Sections -- Section references appear as Section n, where n represents the section number.

* Appendixes -- Appendix references appear as Appendix Y, where Y represents the appendix letter.

* Paragraphs -- Paragraph references appear as alphanumeric characters separated by decimal points. This first character refers to the section or appendix containing the paragraph, and any other numbers indicate the sequence of the paragraph within the section or appendix. For example:

  - 3.5.2 refers to Section 3, paragraph 5.2.

  - A.2 refers to Appendix A, paragraph 2.

* Figures -- Figure references appear as Fn-x or FY-x, where n represents the section and Y represents the appendix containing the figure; x represents the number of the figure within the section or appendix. For example, F2-7 refers to the seventh figure in Section 2.

* Tables -- Table references appear as Tn-x or TY-x, where n represents the section and Y represents the appendix containing the table; x represents the number of the table within the section or appendix. For example, TB-4 refers to the fourth table in Appendix B.

* See and See also references -- See and See also direct you to other entries in the index. For example:

  Logical Unit Number . . . . . See LUNO
  Device . . See individual device names or numbers

Page numbers that correspond to these index references appear in the Table of Contents.

CUT ALONG LINE

USER'S RESPONSE SHEET

Manual Title: Unit Diagnostics Handbook, Volume 1, General Diagnostic Information (945400-9701)

Manual Date: July 1984 **Date of This Letter:** _____

User's Name: _____ **Telephone:** _____

Company: _____ **Office/Department:** _____

Street Address: _____

City/State/Zip Code: _____

Please list any discrepancy found in this manual by page, paragraph, figure, or table number in the following space. If there are any other suggestions that you wish to make, feel free to include them. Thank you.

Location in Manual	Comment/Suggestion
_____	_____
	_____
	_____
	_____
_____	_____
	_____
	_____
	_____
_____	_____
	_____
	_____
	_____

NO POSTAGE NECESSARY IF MAILED IN U.S.A.
FOLD ON TWO LINES (LOCATED ON REVERSE SIDE), TAPE AND MAIL

FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 7284 DALLAS, TX

POSTAGE WILL BE PAID BY ADDRESSEE

TEXAS INSTRUMENTS INCORPORATED
DATA SYSTEMS GROUP

ATTN: TECHNICAL PUBLICATIONS
P.O. Box 2909 M/S 2146
Austin, Texas 78769

FOLD