

TESTER 21



---

**REFERENCE MANUAL**

**TESTER 21**



PUBLICATION NO. 44-B700-401-00

AUGUST 1973

**SINGER**  
**BUSINESS MACHINES**

2350 WASHINGTON AVE.  
SAN LEANDRO, CALIF. 94577

\*A trademark of the Singer Company.

CONTENTS

---

Section 1	INTRODUCTION . . . . .	1-1
	HARDWARE . . . . .	1-2
	Required . . . . .	1-2
	Optional . . . . .	1-2
	SOFTWARE . . . . .	1-2
	DEVICE ASSIGNMENTS . . . . .	1-2
	TESTER VERSIONS . . . . .	1-3
	Restricted TESTER . . . . .	1-3
	Relocatable TESTER . . . . .	1-3
	TESTER OPERATING PROCEDURES . . . . .	1-4
	TESTER PROGRAM RE-ENTRY. . . . .	1-4
Section 2	TESTER COMMAND SEQUENCES . . . . .	2-1
	TESTER SYNTAX. . . . .	2-1
	Disc Addresses . . . . .	2-1
	Core Addresses . . . . .	2-1
	STANDARD COMMAND SEQUENCES . . . . .	2-1
	Target Program Loading . . . . .	2-1
	Tracing the Target Program . . . . .	2-1
	Target Program Loading using Overlay Trap. . . . .	2-2
	Free Execution of the Target Program . . . . .	2-2
	Use of Utility Commands. . . . .	2-3
Section 3	TESTER 21 COMMANDS . . . . .	3-1
	INITIALIZATION COMMANDS. . . . .	3-1
	Trace. . . . .	3-1
	Trace Kill . . . . .	3-5
	Stop . . . . .	3-5
	Overlay Trap . . . . .	3-6
	Overlay Trap Kill. . . . .	3-8
	EXECUTION COMMANDS . . . . .	3-9
	Goto . . . . .	3-9
	Return . . . . .	3-9
	Load From Disc . . . . .	3-9
	Load From Cards. . . . .	3-10
	Load Target. . . . .	3-10

CONTENTS

---

UTILITY COMMANDS . . . . .	3-10
Dump Core. . . . .	3-10
Display Disc . . . . .	3-11
Disc Patch . . . . .	3-12
Core Correctors. . . . .	3-14
Object Patch . . . . .	3-18
CONTROL COMMANDS . . . . .	3-23
Quit . . . . .	3-23
Print. . . . .	3-23
Eject. . . . .	3-23
Up Paper . . . . .	3-24
Line Printer Substitution. . . . .	3-24
Simulator Writer . . . . .	3-24
Set FAC Protection . . . . .	3-25
Card . . . . .	3-25
Workstation. . . . .	3-26
 Section 4 TESTER 21 CONSOLE MESSAGES . . . . .	 4-1

Section 1  
INTRODUCTION

---

HARDWARE

SOFTWARE

DEVICE ASSIGNMENTS

TESTER VERSION

TESTER OPERATING PROCEDURES

TESTER PROGRAM RE-ENTRY

## INTRODUCTION

---

The TESTER 21 program is a conversational debugging aid that can be used to monitor, and optionally, to modify user programs. The TESTER program is usually operated through an interactive console, such as a Workstation or a Video Display Unit. The TESTER program performs the following functions:

- Simulates and traces user target program execution
- Displays disc pools, files and sectors
- Patches disc sectors
- Inserts text patch cards into an object file
- Dumps and patches core instructions and data

The TESTER program can interface with a user target program which is being debugged in one of the following modes:

- TRACE Simulated Execution. In this mode, TESTER retains control of the user program, simulating the execution of each instruction, and at the user's option, printing the results of selected simulations. Depending on the parameters stated in the Trace command, every instruction can be traced, or only selected instructions (such as branch or input/output instructions) can be traced. The Trace command can also be used with no print-out regarding instructions, in which case, execution of a complete user program is simulated without any physical output from the Trace operation. Simulated execution is substantially slower than free execution.
- Overlay Trap. In this mode the user is notified each time a new segment is loaded into core. Notification can be made before, after or both before and after loading, with the user having the option of loading or not loading or starting or not starting execution of a specific segment. TESTER relinquishes control to the segment that has been trapped only when instructed to do so by the user. The segment then executes freely, returning control to TESTER before or after loading a new segment, to repeat the process of loading or not loading, starting or not starting. In this mode, only the loading of program segments is traced. Detailed tracing within segments is not possible; the Overlay Trap and Trace commands are mutually exclusive.

## INTRODUCTION

---

- Free Execution. In this mode, the TESTER program is used to load a user program, then relinquishes control, permitting the program to run normally. In this mode, control returns to TESTER after the user program terminates. TESTER could then be used to examine selected instructions residing either in core or on disc and to modify either core or disc, if desired.

## HARDWARE

### Required

Model 20/21 Central Processor Unit

Model 40/42/44 Disc Drive

Model 70 Workstation or Model 80 Video Display Unit

### Optional

Model 50 Line Printer (Although not required, the Model 50 Line Printer is recommended when a high volume output is desired.)

Model 40 Disc Drives

Model 30 Card Reader

Model 35 Card Punch

## SOFTWARE

DMF structured disc pack

## DEVICE ASSIGNMENTS

The following devices must be assigned the codes indicated when these devices are used:

- 0 = Conversational Console
- 1 = Model 30 Card Reader
- 2 = Model 50 Line Printer
- 4 = Model 35 Card Punch

---

## TESTER VERSIONS

There are two versions for TESTER. One version loads into locations 6900 to 9999 of common core and is restricted in the type of programs it can simulate or trace. The other version can be relocated to any place in common core and has no restrictions on the type of programs it can simulate or trace.

### Restricted TESTER

This version prints a program identification of TESTER. The only function that is restricted is the simulation function. This version may not be used to simulate move address instructions, instructions using indirect addressing or instructions in common core above location 9999.

### Relocatable TESTER

This version prints a program identification of TSTR21. It is released as a job stream file which controls a Linkage Editor execution; the result of the Linkage Editor execution is a copy of TSTR21.

The user may specify the origin address for TSTR21 by altering the first PHASE statement in the job stream file. As released, this PHASE statement is as follows:

```
PHASE TSTR21(0300,16300),SAVE;
```

This statement causes TSTR21 to load between 16300 and 19999 in common core. To change this load address, change the 16300 to any desired address. For example, to create a copy of TSTR21 which loads between 26300 and 29999 in common core, change the PHASE statement to:

```
PHASE TSTR21(0300,26300),SAVE;
```

No other entry in the job stream file should be changed.

TSTR21 takes 3700 locations of common core. It may be relocated to any location in common core that does not make TSTR21 cross a 10000 character page boundary. So, for example, TSTR21 may not be relocated to location 19000.

In the remainder of this manual, both versions of TESTER will be referred to as TESTER unless only one of the two versions is meant.



## INTRODUCTION

---

### TESTER OPERATING PROCEDURES

1. Load TESTER from disc, using the conversational loader.
2. If TESTER is being loaded from a DMF II disc system, and if any device listed in the configuration table is not an MTIOC device, then TESTER terminates. If TESTER has been prohibited by use of the NOTEST option of SETCOM STARTUP, then TESTER terminates.
3. If a line printer (MTIOC device 2) is either not attached, powered down or in local, the operator is informed of this fact. If the printer is in local, it should be put on line.
4. A program identification message is printed, consisting of the version (either TESTER or TSTR21) followed by the release level.
5. If the system mailbox flag indicates that the mailbox is not busy, then a branch to the TESTER program entry point is placed in location 00990 common core. If the mailbox is busy, the operator is notified that the TESTER program entry point cannot be set. See TESTER Program Re-entry.
6. If either the partition does not have IOK locations or the FAC protect character indicates that tapes or disc are inhibited, the operator is informed of the condition.
7. A question mark is displayed on the console, indicating that a TESTER command needs to be entered. This is called the TESTER command input query.
8. The user enters one or more TESTER commands. Some commands require secondary input; these commands display an asterisk to indicate that secondary input needs to be entered.
9. At completion of the use of TESTER, either use the Quit command to return the conversational loader, or load or execute a target program in free execution (with Trace, Stop, and Overlay Trap statuses off).

### TESTER PROGRAM RE-ENTRY

There are times when the operator needs to re-enter the TESTER program without reloading it; for example, to obtain a core dump after a program has gone to load. The TESTER program main entry point is at the origin address of the TESTER version; for example, the entry point of the version that loads between 6900 and 9999 common is a 6900 common. Entry points below 10000 common can be branched to directly, however, hardware considerations prevent

branching to addresses 10000 and above in a single instruction at load condition.

All versions of TESTER place a branch to the TESTER main entry point in location 00990 common (if the system mailbox is not busy when TESTER is loaded); this is the TESTER standard entry point. If this entry point is not set, or if it is apt to be destroyed (for example, by placing records longer than 405 characters in the mailbox), the user may set up an alternate entry point by using the Core Correctors command.

Section 2

TESTER COMMAND SEQUENCES

---

TESTER SYNTAX

STANDARD COMMAND SEQUENCES

## TESTER COMMAND SEQUENCES

---

### TESTER SYNTAX

#### Disc Addresses

All disc addresses must be entered as six characters. All cylinder addresses must be entered as five characters.

#### Core Addresses

All partition core addresses are entered as one to four digits, where leading zeros may be omitted.

All common core addresses are entered as one to five digits followed by the character C, where leading zeros may be omitted.

### STANDARD COMMAND SEQUENCES

#### Target Program Loading

TESTER usually gains control of the user program for simulation through the use of the Load Target command. The following sequence of operation occurs for a Load Target command:

1. After the TESTER input query (?) the user responds with a Load Target command.
2. The first segment of the user program is loaded and the S-card address is printed.
3. The TESTER input query (?) is repeated, requesting a new command.

#### Tracing the Target Program

In the simulation of a user program, a trace output may or may not be given. In either event, the following sequence of operations occurs:

1. After the TESTER input query (?), the user responds with initialization command(s): Trace and/or Stop.
2. After the next TESTER input query (?), the user responds with an execution command: Goto or Return.

3. Target user program simulation occurs.
4. Simulation stops when a Stop condition has been satisfied (if given), or when a service request has been issued.
5. When simulation stops, return is made to the TESTER input query (?).

#### Target Program Loading using Overlay Trap

TESTER gains control of the user program for simulation through the use of the Overlay Trap command. An overlay is a program segment (a group of text cards followed by a start card in an object program). The following sequence of operations occurs for an Overlay Trap command:

1. After the TESTER input query (?), the user responds with an Overlay Trap initialization command.
2. After the next TESTER input query (?), the user responds with an execution command (Load, Goto, or Return).
3. The next time that loading and overlay are required (immediately in the case of a Load command), the user is notified of the overlay loading and is asked how to proceed.
4. The user may either permit his program to continue to execute in unsimulated mode (see Step 3 above) or he may request a return to the TESTER input query (?).

#### Free Execution of the Target Program

The user may return the user program to execute in free execution. In this mode, the following sequence of operations occurs:

1. During TESTER input query (?), the initialization status must be such that Trace, Stop, and Overlay Trap are off.
2. The user may use the Trace Kill or Overlay Trap Kill command to ensure that this situation is correct.
3. The user enters an execution command (Load, Goto, or Return) and is asked to verify initialization status.
4. The program executes in unsimulated mode.
5. There is no return to TESTER.

Use of Utility Commands

The following sequence of operations occurs for a utility command:

1. After the TESTER input query (?), the user responds with a utility command.
2. Dump Core and Display Disc commands: The function requested is performed.
3. Disc Patch, Object Patch, and Core Correctors command: A cue character (\*) is printed to request secondary input; the user provides the secondary input. When the pertinent function has been performed additional secondary input is requested until a unit separator or control key is entered.
4. After the function is performed, return is made to the TESTER input query (?).

Section 3

TESTER 21 COMMANDS

---

INITIALIZATION COMMANDS

EXECUTION COMMANDS

UTILITY COMMANDS

CONTROL COMMANDS

## TESTER 21 COMMANDS

---

The TESTER program commands are divided into four classifications: initialization, execution, utility, and control commands. The initialization commands establish the mode of operation of the TESTER program, select the parameters to be monitored and set the trace limits. Initialization commands are usually followed by at least one execution command. The execution commands load and/or execute target user programs. The utility commands display or modify the contents of core locations and disc sectors. The control commands control the peripheral devices. All commands and the mnemonics of these commands are listed in Table 3-1 and are described in the following paragraphs.

### ① INITIALIZATION COMMANDS

#### Trace Command

The Trace (T) command sets Trace status on. This status initializes the TESTER program to simulate target user program execution and to display a trace of the execution. Since the Trace and Overlay Trap commands are mutually exclusive the Trace command, when used, sets Overlay Trap status off. Once the Trace command is entered, the Trace parameters remain valid until a Trace Kill, an Overlay Trap, and another Trace command is entered; tracing a target program does not affect the Trace status.

Command Format:

T [P1][P2][addr1[addr2]][P3]

where

P1 = the trace type; (the type of trace to be performed)

values for P1 are:

blank = same as full trace; default

F = full trace; all instructions are displayed

I = input/output trace; only input/output instructions are displayed



TABLE 3-1. TESTER PROGRAM COMMANDS

COMMAND CLASSIFICATION	COMMAND NAME	MNEMONIC OF COMMAND NAME
Initialization	Trace	T
	Stop	S
	Trace Kill	TK
	Overlay Trap	OT
	Overlay Trap Kill	OTK
Execution	Goto	G
	Return	ENTER key
	Load Target	LT
	Load from Disc	L
	Load from Cards	LC
Utility	Dump Core	D
	Display Disc	DD
	Disc Patch	DP
	Core Correctors	C
	Object Patch	OP
Control	Quit	QU
	Print	P
	Eject	E
	Up Paper	U
	Line Printer Substitution	LP
	Simulator Write	SW
	Set FAC Protection	FAC
	Card	CD
	Workstation	WS

- BI = branch and input/output trace; all input/output and branch instructions are displayed
- W = write trace; only write instructions are displayed
- B = branch trace; only successful branch trace instructions are displayed
- M = modification trace; the instructions displayed are those that modify a specified field; see the trace limits

P2 = the output mode

values of P2 are:

blank = the normal output mode; each instruction traced produces one line on the printer that describes the instruction: the address, instruction mnemonic, operands, condition code, and current register settings

All information for display is collected after execution of the instruction. Input/output instructions with more than 53 characters transmitted are displayed as two lines: the first line contains the first 132 characters of the input/output buffer and the second line describes the instruction. Input/output instructions with 53 or fewer characters have the buffer contents in a single output line.

A = the address-only output mode; this mode displays the addresses of all selected instructions

addr1 addr2 = the trace limits; addr1 specifies the low limit; addr2 specifies the high limit; if only one address is specified the range is one location long; the default values are from 1000 to 9999

partition; only the TSTR21 version may use addresses over 9999

When used with trace type F, B, I, BI, and W, the trace limits restrict the selected instructions to those located within the given limits. When used with trace type M, the trace limits specify a field in core. Any instruction that modifies this field is selected for trace output. This field can be no longer than nine locations.

P3 = the trace limits use flag

values for P3 are:

blank = normal use; only addresses within the trace limits are selected

B = extended use; addresses within the trace limits and all addresses in the other core area are selected  
(Core)

If the trace limits are in partition, the addresses within the limits in partition and all the addresses in common are selected. If the trace limits are in common, the addresses within the limits in common and all the addresses in partition are selected.

Explicit specification of trace limits is required. Trace type M may not be used.

Note: In the normal output mode, the contents of the operands are displayed. The display is made after execution of the instruction, sometimes causing erroneous information to be displayed. For example, the length of the result field of an edit instruction may be wrong; or an instruction which alters an index register and uses the same index register as a modifier, the display of the operand which used the register as a modifier may be wrong. Also, the user input/output residual count is modified as a result of TESTER display output, sometimes causing errors when this value is examined by the target program.

---

Trace Kill Command

The Trace Kill (TK) command sets Trace status off.

Command Format.

TK

There are no parameters.

Stop Command

The Stop (S) command sets Stop status on. This status initializes the TESTER to simulate target program execution, and then to stop simulation based upon certain Stop parameters. If trace output is desired while simulation is taking place, the Trace command should also be used. Since the Stop and Overlay commands are mutually exclusive, the Stop command sets Overlay Trap status off. Once the Stop command is entered, the Stop parameters remain valid until any execution or utility command is executed; thus, simulating a target program sets the Stop status off.

Command Format:

S [P1]addr1[addr2]['data'] [N=nnnn]

where

P1 = the service request flag

values for P1 are:

blank = normal setting; the TESTER program stops simulation when a service request is encountered (default)

R = trace service request setting; the TESTER program does not stop simulation when a service request is encountered; branch-on-service-request instructions in the target user program are simulated

addr1 addr2 = the stop limits; addr1 specifies the low limit; addr2 specifies the high limit; if only one address is specified the range is one location long; only the TSTR21 version may use addresses over 9999

When used without a data specification, the stop limits specify that simulation is to be halted whenever an instruction within the Stop limits is about to be simulated.

When used with a data specification, the first address specifies the first address for the data field; the second address is not used.

'data' = the data specification; the data specification is optional. It is coded as up to nine characters placed between quote marks

This parameter causes simulation to stop when the data in core at the location specified by the stop limit either first becomes equal to the data specification (after initially being unequal) or first becomes unequal to the data specification (after initially being equal).

N=nnnn = the stop condition repeat count (the default is assumed to be 1)

When used without a data specification, this parameter specifies the number of instructions to be simulated within the stop limits before simulation stops. When used with a data specification, this parameter specifies the number of changes of state that are to occur before simulation stops.

### Overlay Trap Command

The Overlay Trap (OT) command sets Overlay Trap status on. This status initializes the TESTER program to notify the user before and/or after each program segment is loaded. The Overlay Trap status remains on until the user enters any initialization or execution command. Since the Overlay Trap, Trace, and Stop commands are mutually exclusive, the Overlay Trap command sets Trace and Stop status off.

Command Format:

OT [P1][P2][P3]

where

P1 = the target user program load message print control

values for P1 are:

blank = same as A (default)

A = loading message requested after loading

B = loading message requested before loading

BA = loading message requested before and after loading

P2 = the program source

values for P2 are:

blank = target user program is loaded from disc drive (default)

C = target user program is loaded from card reader

P3 = the number of loading messages that are suppressed

values for P3 are:

blank = the load message is printed for the first overlay loaded as specified by P1; (N=1) is the default

N=dd = the message suppression number; dd is a one- or two-digit number that indicates the first overlay to have a loading message printed

During loading, dd-1 loading messages are suppressed; the overlays are loaded and executed automatically. For example, if OT N=12 (followed by a Load command) is specified, 11 overlays are loaded without notification messages. When

the 12th overlay is loaded, the following message is printed.

START=addr. CONTINUE? (N/Y)

### Overlay Trap Operation

The Overlay Trap status is used only when a program segment is loaded by the TESTER loader. To ensure that the TESTER loader is used whenever the target program (the user's program) is running, the partition's loader entry point is changed to a branch to TESTER.

If a Goto or Return command is used with Overlay Trap status on, the target program executes without simulation. When the program goes to the loader to load another program segment, the TESTER loader gains control and action is taken based on the Overlay Trap parameters. One or more of the following actions is performed:

1. The operator is asked whether the next program segment is to be loaded.
2. The program segment is loaded and the operator is asked whether the segment just loaded should begin execution.
3. The program segment is loaded and executed. This loading and execution is continued through a specified number of program segments.

If the operator responds to a query by specifying that the normal sequence should not continue, TESTER returns to the command input query (?).

If a Load command is issued with Overlay Trap status on, the specified program segment is loaded, and then action is taken based on the Overlay Trap parameters. If the program segment begins execution after the load, TESTER performs as for the Goto and Return commands.

### Overlay Trap Kill Command

The Overlay Trap Kill (OTK) command set the Overlay Trap status off.

Command Format:

OTK

There are no parameters.

## ② EXECUTION COMMANDS

### Goto Command

The Goto (G) command causes the target program either to begin execution or to begin simulation starting at a specific address.

Command Format:

G addr

where

addr = the execution address

This parameter is required. Only the TSTR21 version may use addresses over 9999.

### Return Command

The Return command causes the target program either to begin execution or to begin simulation starting at the "return address". This return address is either the start address from a program load, or the next address to be executed after a prior target program simulation. The Return command has no format or parameters. It is initiated by pressing the ENTER key on the console. The Return command is valid only if a return address exists.

### Load From Disc Command

The Load From Disc (L) command loads one segment of a specified program from disc.

Command Format:

L [[pool.]file]

where

blank = loading begins at the disc address in partition location 0015 to 0020

pool.file = pool and file names where the program is located on the disc; SYSPOL is assumed if the pool name is omitted

If Overlay Trap has previously been specified with card loading (OT C), all parameters are ignored and the Load From Disc command is treated as if it were a Load From Cards command. If the Overlay



Trap status is off, after the program segment is loaded, the TESTER program loses control and the program segment begins execution.

#### Load From Cards Command

The Load From Cards (LC) command loads one segment of an object program card deck via an internal TESTER program card loader. If Overlay Trap status is off, after the program segment is loaded, the TESTER program loses control and the program segment begins execution.

Command Format:

LC

There are no parameters.

#### Load Target Command

The Load Target (LT) command loads the first segment of the user target program from a disc file and returns control to the TESTER command input query (?).

Command Format:

LT [pool.]file

where

pool = an optional pool name; SYSPOL is the default

### ③ UTILITY COMMANDS

#### Dump Core Command

The Dump Core (D) command causes formatted dumps of selected core areas to be printed. A service request terminates the dump and returns to the TESTER command input query (?).

Command Format:

D [addr1[addr2]]

where

addr1[addr2] = the dump limits; addr1 specifies the low limit; addr2 specifies the high limit; if addr2 is omitted, the dump contains one line which includes the address specified; if

both parameters are omitted, the assumed limits are 0000 to the end of the partition

### Display Disc Command

The Display Disc (DD) command causes the contents of disc sections to be displayed. The sections can be contiguous sectors specified by surface address, cylinder address or pool name; the sections can be linked-sequential sectors pointed to by initial sector or file name; or the sectors can be doubly linked sectors printed in reverse order. A service request terminates the display and returns to the TESTER command input query (?).

DD P1

where

P1 = the disc area to be displayed

values of P1 are:

pool. = causes display of a pool label followed by all sectors in the pool; if the pool label specifies that the pool is cylinderized, the pool limits are the track limits and a cylinderized display occurs;

This parameter is coded as a pool name followed by a period.

pool.file = causes display of a pool and file label followed by the sectors of the file in linked sequential order

This parameter is coded as an optional pool name followed by a period, followed by a required file name. If pool name and period are omitted, SYSPOL is used as the pool name.

daddr1 = causes display of <sup>(a)</sup>one disc sector

This parameter is coded as a six character disc address.

daddr1 daddr2 = causes display of a group of physically contiguous sectors, from

the sector specified by daddr1 to the one specified by daddr2

This parameter is coded as two six character disc addresses.

CYL lim1[lim2] = causes display of a group of sectors which are contiguous in cylinder addressing; lim1 specifies the low track address; lim2 specifies the high track address; if lim2 is omitted, the display consists of the 50 sectors in the specified disc track

This parameter is coded as the characters CYL followed by a track address in the form dccch (d is the drive; ccc is the cylinder, 000 to 199; h is the head, 1 to 9), optionally followed by another track address in the form dccch.

daddr1 LS = causes display of a group of disc sectors in linked sequential order

This parameter is coded as a six character disc address (the starting sector) followed by the character LS.

daddr1 ULS = causes display of a group of doubly linked sectors in upward linked order

This parameter is coded as a six-character disc address (the starting sector) followed by the characters ULS.

### Disc Patch Command

The Disc Patch (DP) command allows the disc contents to be modified. The modifications can be entered through the console or card reader.

Command Format:

DP [P1]

where

P1 = the input device for secondary control input  
 values for P1 are:

blank = indicates input from the console  
 (default)

W = indicates input from the console

C = indicates input from the card reader

After entering the DP command, additional information (secondary control input) is required. If this input is from the console, an asterisk (\*) cue character is printed on the console to inform the user that secondary input is needed. The cue character (\*) is printed after each input is processed, allowing several disc patches to be made. To terminate entries from the console, depress a control key. To terminate entries from a card reader, use a unit separator.

Three secondary input formats are available that allow the user to modify, print, or copy data.

To modify the data, use the following format:

[daddr][nn]'data'[N]

where

daddr = a six-digit sector address; default is the sector address used in the previous secondary input for this Disc Patch command

nn = a displacement within the sector which specifies where the modification should begin; if omitted, the default is displacement zero (equivalent to data character one)

'data' = the new data to be entered (up to 100 characters); two consecutive quote marks are interpreted as a single quote character

N = an optional directive not to print out the modified data

To print the contents of the disc sector, use the following format:

[daddr]P[S]

where

daddr = a six-digit sector address; default is the sector address used in the previous secondary input for this Disc Patch command

P = the print operator

S = the optional directive to print a scale line below the data line

To copy the contents of one disc sector to another disc sector, use the following format:

daddr1 C daddr2

where

daddr1 = a six-digit sector address of the sector from which the copy is to be made; this sector remains unchanged; default is the sector address used in the previous secondary input for this Disc Patch command

C = the copy operator

daddr2 = a six-digit sector of the address of the sector to which the data will be copied; after the copy process is complete, daddr2 is the established disc address default for subsequent entries

### Core Correctors Command

The Core Correctors (C) command allows the user to alter the contents of core locations. These corrections can be entered through the console or the card reader.

Command Format:

C [P1][P2]

where

P1 = the input device designation for secondary control input

values for P1 are:

blank = input is from the console  
(default)

W = input is from the console

C = input is from the card reader

P2 = the T-card punch specification; specifies whether object text cards (T-cards) are to be punched

values for P2 are:

blank = T-cards are not to be punched

P = T-cards are to be punched

After entering the Core Correctors command, additional information (secondary control input) is required. If this input is from the console, an asterisk (\*) cue character is printed on the console to inform the user that secondary input is needed; the asterisk is printed after each input is processed, allowing several core corrections to be made. To terminate entries from the console, press a control key. To terminate entries from a card reader, use a unit separator card.

Four formats allow the user to modify core with data, modify core with instructions, print, or punch the contents of core locations:

To modify core with data, use the following format:

[caddr]'data'[N][comment]

where

caddr = the address of the core location where the data is to be stored; if omitted on the first entry of a Core Correctors command, the default is 0000; if omitted on a subsequent entry, the default is immediately after the previous entry

'data' = the new data to be entered (up to 63 characters) enclosed between quote marks: two consecutive quote marks are interpreted as a single data quote character

N = the console response indicator; if present, a message verifying the correction will be printed on the line printer only; if

omitted, such messages are printed on the console and line printer

comment = the comment field to be inserted after the text field in the punched text card; if omitted, no comment is inserted; coded as an asterisk (\*) followed by an arbitrary number of characters

To modify core with an instruction, use the following format:

[caddr]op operand [N][comment]

where

caddr = the address of the core location where the data is to be stored; if omitted on the first entry of a Core Correctors command, the default is 0000; if omitted on a subsequent entry, the default is immediately after the /previous entry incremented, if necessary, to the next ten's boundary.

op = the mnemonic operation code for the instruction; acceptable values are: A, BC, C, D, E, FN, M, MA, MC, MN, R, S, W and X; if the op code is C (compare), caddr must be followed by a space

operand = the operand; for operation codes A, D, FN, M, R, S and W the format is

addr(len),addr(len) *2 lengths*

where

addr = a core address

len = a number from one to ten specifying the length, optionally followed by a comma and an index register value from one to three

for operation codes C, E, MA, MC, MN and X, the format is

addr(len),addr

*One length*

where

addr = a core address

len = a number from one to 100, specifying the length, optionally followed by a comma and an index register value from one to three

for operation code BC, the format is

addr (code), addr (code) *Branch Conditional*

or

addr (code)

or

addr

where

addr = a core address

code = a single digit specifying the condition code; if omitted, it defaults to five

N = the console response indicator; if present, a message verifying the correction will be printed on the line printer only; if omitted, such messages are printed on the console and line printer

comment = the comment field to be inserted after the text field in the punched text card; if omitted, no comment is inserted; coded as an asterisk (\*) followed by an arbitrary number of characters

To punch the contents of core onto object text cards (T-cards) (and not modify core), use the following format:

[caddr]=len[N][comment]

where



- caddr = the address of the core location where the data is stored; if omitted on the first entry of a Core Correctors command, the default is 0000; if omitted on a subsequent entry, the default is immediately after the previous entry
- =len = the length specification (1 to 4 digits)
- N = the console response indicator; if present, a message verifying the correction will be printed on the line printer only; if omitted, such messages are printed on the console and line printer
- comment = the comment field to be inserted after the text field in the punched text card; if omitted, no comment is inserted; coded as an asterisk (\*) followed by an arbitrary number of characters

To print the contents of core locations (20 characters), use the following format:

- caddr  
where  
caddr = the address of the core location where the data is stored

### Object Patch Command

The Object Patch (OP) command allows the user to modify an object file by inserting object text cards (T-cards) into the file. These text cards are placed immediately before the start card (S-card) in a specified overlay. No facility is provided to alter an existing text card nor to insert a text card at any location in the file other than before a start card. The Object Patch input can be entered through the console or the card reader.

#### Command Format

- OP [pool.]file[P1][P2]
- where
- [pool.]file = the name of the file in which the object patches are to be made

This parameter is coded as an optional pool name and period, followed by a mandatory file name. If the pool name is omitted, SYSPOL is the default.

P1 = the input device designation for secondary control input

values for P1 are:

blank = input is from the console; default

W = input is from the console

C = input is from the card reader

P2 = the T-card punch specification; specifies whether object text cards (T-cards) are also to be punched

values for P2 are:

blank = T-cards are not to be punched

P = T-cards are to be punched

After entering the Object Patch command, additional information (secondary control input) is required. If this input is from the console, an asterisk (\*) cue character is printed on the console to inform the user that secondary input is needed; the asterisk is printed after each input is processed, allowing several object patches to be made. To terminate entries from the console, press a control key. To terminate entries from a card reader, use a unit separator card.

Four secondary input formats allow the user to generate an object patch with data, generate an object patch with instructions, generate an object patch with data taken from core or print the contents of core with no patch made.

To generate an Object Patch with data, use the following format:

[(ov)][caddr]'data'[C][N][comment]

where

(ov) = the overlay number; coded as a one- or two-digit number enclosed in parenthesis (from 1 to 99); if omitted on the first entry of an Overlay Patch command, the default is 01; if omitted on

a subsequent entry, the default is the same as the previous entry

- caddr = the address of the core location where the data is to be stored; if omitted on the first entry of an Object Patch command, the default is 0000; if omitted on a subsequent entry, the default is immediately after the previous entry
- 'data' = the new data to be entered (up to 63 characters) enclosed between quote marks; two consecutive quote marks are interpreted as a single data quote character
- C = the core correction indicator; if present, core is also corrected; if omitted, only an object patch is generated
- N = the console response indicator; if present, a message verifying the patch will be printed on the line printer only; if omitted, such messages are printed on the console and line printer
- comment = the comment field to be inserted after the text field in the punched text card; if omitted, no comment is inserted; coded as an asterisk (\*) followed by an arbitrary number of characters

To generate an Object Patch with an instruction, use the following format:

[(ov)][caddr]op operand[C][N][comment]

where

- (ov) = the overlay number; coded as a one- or two-digit number enclosed in parenthesis (from 1 to 99); if omitted on the first entry of an Overlay Patch command, the default is 01; if omitted on a subsequent entry, the default is the same as the previous entry
- caddr = the address of the core location where the data is stored; if omitted on the first entry of an Object Patch command, the default is 0000; if omitted on a subsequent entry, the default is immediately after the previous entry, incremented, if necessary, to the next ten's boundary

op = the mnemonic operation code for the instruction; acceptable values are: A, BC, C, D, E, FN, M, MA, MC, MN, R, S, W, and X; if the op code is C (compare), caddr must be followed by a space

operand = the operand, for operation codes A, D, FN, M, R, S, and W the format is

addr(len),addr(len)

*2 length*

where

addr = a core address

len = a number from one to ten specifying the length, optionally followed by a comma and an index register value from one to three

for operation codes C, E, MA, MC, MN and X, the format is

addr(len),addr

*one length*

where

addr = a core address

len = a number from 1 to 100, specifying the length, optionally followed by a comma and an index register value from one to three

for operation code BC, the format is

addr(code),addr(code)

*Branch Conditions*

or

addr(code)

or

addr

where

addr = a core address

- code = a single digit specifying the condition code; if omitted, it defaults to five
- C = the core correction indicator; if present, core is also corrected; if omitted, only an object patch is generated
- N = the console response indicator; if present, a message verifying the patch will be printed on the line printer only; if omitted, such messages are printed on the console and line printer
- comment = the comment field to be inserted after the text field in the punched text card; if omitted, no comment is inserted; coded as an asterisk (\*) followed by an arbitrary number of characters

To generate an object patch with data taken from core, use the following format:

[(ov)][caddr]=len[N][comment]

where

- (ov) = the overlay number; coded as a one- or two-digit number enclosed in parenthesis (from 1 to 99); if omitted on the first entry of an Overlay Patch command, the default is 01; if omitted on a subsequent entry, the default is the same as the previous entry
- caddr = the address of the core location where the data is stored; if omitted on the first entry of a Core Correctors command, the default is 0000; if omitted on a subsequent entry, the default is immediately after the previous entry
- =len = the length specification (1 to 4 digits)
- N = the console response indicator; if present, a message verifying the patch will be printed on the line printer only; if omitted, such messages are printed on the console and line printer
- comment = the comment field to be inserted after the text field in the punched text card; if omitted, no comment is inserted; coded as an

asterisk (\*) followed by an arbitrary number of characters

To print contents of core locations (20 characters), use the following format:

caddr

where

caddr = the address of the core location where the data is stored

#### 4 CONTROL COMMANDS

##### Quit Command

The Quit (QU) command causes an exit, loading the DMF system conversational loader.

Command Format:

QU

There are no parameters

##### Print Command

The Print (P) command causes 20 consecutive core locations to be printed on the console and the line printer.

Command Format:

P addr

where

addr = the starting address of the core locations to be printed

##### Eject Command

The Eject (E) command causes line printer paper to advance to the top of the form so that the output is ready to be torn off.

Command Format:

E [n]

where

n = the number of ejects to be issued; if omitted, the default is 4

Up Paper Command

The Up Paper (U) command causes the line printer to space forward until the preceding page is visible.

Command Format:

U

There are no parameters

Line Printer Substitution Command

The Line Printer Substitution (LP) command causes displays (print-outs) that usually go to the line printer to go to some other device (usually a conversational console device). The displays that are affected by the assignment are the data and heading displays for the Trace, Display Disc, and Dump Core functions. Some other messages that duplicate console output are still sent to the line printer, but if the line printer is not attached, these duplicated outputs will be dropped. The device assignment of an LP command remains the same until another command is initiated.

Command Format:

LP [n]

where

n = the one-digit device assignment of the device to be used to replace the line printer; if n is omitted, the default is 2

Simulator Write Command

The Simulator Write (SW) command causes the last instruction simulated to be printed on the console and line printer. The Simulator Write function is normally used when the program being traced or simulated goes into a load status. When the user returns control to the TESTER program (see TESTER program re-entry), the Simulator Write command causes the fault location and its contents to be printed. If the Simulator Write command is used when the program did not go to a load status, the address and instruction output in the SW print-out is the next instruction to be simulated.

Command Format:

SW

There are no parameters.

#### Set FAC Protection Command

The Set FAC Protection (FAC) command sets the FAC protect character for partition zero. This command can be performed only from partition zero.

Command Format:

FAC [x]

where

x = the new FAC protect character for partition zero; if omitted, blank is assumed; for information about the FAC protect character, see the A-register description in the Programmer's Reference Booklet, Singer Form 40-101R

#### Card Command

The Card (CD) command causes the TESTER program to accept all subsequent commands from the card reader until the Card command is negated by a Workstation command. Card reader input is useful when the sequence of commands is determined in advance. Some console verification messages are bypassed during card input processing.

If a unit separator card (11/7/8/9 punch) is read or if the card reader is turned off when a command is expected, a Workstation command is generated and processed.

Command Format:

CD

There are no parameters.



Workstation Command

The Workstation (WS) command causes the TESTER program to accept all subsequent commands from the console until the Workstation command is negated by a Card command. Workstation mode is the normal mode of processing.

Command Format:

WS

There are no parameters.

Section 4

TESTER 21 CONSOLE MESSAGES

---



## TESTER 21 CONSOLE MESSAGES

---

Messages ending with the query "? (N/Y)" indicate a user response of N (no) or Y (yes) is required. Any user response other than Y is interpreted by the TESTER program as an N (no) response.

BAD LINK IN SECTOR mnnnnn; CONTENTS: mmmmmmm.

Condition: The TESTER program is being loaded and the TESTER object program has a bad linkage. The loading is terminated.

User Action: Correct the problem, and reload the TESTER program.

CANCEL

Condition: A Goto, Return, or Load command which required user verification was entered. The user rejected the command.

User Action: Enter a new command.

CANNOT TRACE ABOVE 9999.

Condition: This version of TESTER cannot trace or stop at locations above 9999C.

User Action: Correct the command or use the TSTR21 version.

COMMON AREA UNAVAILABLE. OVERRIDE AND CONTINUE? (N/Y)

Condition: TESTER is being loaded and DMF II standard common is initialized. The area in which TESTER is to be loaded is neither free nor unreserved.

User Action: Respond Y to continue loading without further common core allocation. Programs, data or allocation control blocks may be destroyed.

DATA EQUAL or DATA UNEQUAL

Condition: One of these messages is printed when simulation stops as a result of a Stop command with data specified.

User Action: Enter a new command.

DISC I/O FROM COMMON or DISC I/O -- READ ONLY

Condition: At the time the TESTER program is loaded, the FAC protect character does not permit full read and write operations from the disc. Processing continues.

User Action: If this condition occurs in partition zero, the user has the option of changing the condition, using the FAC command. In other partitions, the user cannot change the condition.

DRIVE d LOCKED -- PTN nn.

where

d = the number of the locked drive

nn = the locking partition number

Condition: The drive lock in DMF II standard common for the specified drive is locked by the specified partition.

User Action: Press the enter key to retry the drive lock 1000 times. Press any control key to cancel the current step.

ENTER DRIVE NUMBER OF VOLUME vvv FROM SYSTEM ssssss -- ddddddddddddddd.

where

vvv = the volume ID

ssssss = the system name

dddddddddddddd = the description of the volume to be mounted

Condition: The specified volume (with the specified system name, identification and description) was not found during a search of mounted volumes.

Mounted volumes are not examined if they are listed as unavailable in the drive status table (and if DMF II standard common is initialized).

User Action: Mount the correct volume, or determine that it is already mounted. Enter a one-character response which specifies the drive (from 0 to ?) where the required volume is located. To cancel access to this volume, press any control key.

EOF DURING LOADING

Condition: The user specified the loading of a program file, and an end-of-file (EOF) condition was detected before an object start card (S-card) was encountered.

User Action: Display the contents of the specified program file and correct the condition. Then, re-enter the command.

ERROR

Condition: A command has been entered but there is a syntax error in the command.

User Action: Enter the correct command syntax. (See the TESTER commands section of the manual for the correct syntax.)

or

Condition: A utility command has been executed and an unusual condition has occurred during command execution; e.g., a read operation has occurred during execution of a Display Disc command.

User Action: Examine the printer listing to locate the errors encountered.

FAC CHARACTER SET TO 'x'.

Condition: The FAC command has set the FAC character to the specified value.

User Action: None.

FILE IS LOCKED -- PTN nn.

Condition: The Overlay Patch command cannot be performed because the file is locked. This can occur on a DMF II system only.

User Action: Use MAINT UNLOCK to unlock the file.

FILE IS NULL

Condition: The Overlay Patch command cannot be performed because the file is null.

User Action: None.

FILE NAME ERROR

Condition: A command has been entered with a file name specified that is not the name of a file in the specified pool.

User Action: Enter a command with the correct file name.

FREE SECTORS EXHAUSTED

Condition: The Overlay Patch command cannot be performed because the pool has no free sectors.

User Action: Use MAINT DELETE to delete some existing file.

ILLEGAL ADDRESS

Condition: An address specified in a Core Correctors or Overlay Patch command secondary input was incorrectly entered. The correct format is one to four digits for partition addresses, and one to five digits followed by a C for common core addresses.

User Action: Enter the correct address in the correct format.

ILLEGAL FORMAT

Condition: Incorrectly formatted Disc Patch, Core Correctors, or Overlay Patch secondary input was entered. Refer to the description of the specific command for the correct format.

User Action: Enter the secondary input in the correct format.

ILLEGAL FUNCTION CODE

Condition: An instruction function code in a Core Correctors or Overlay Patch command secondary input was expected and was incorrect.

User Action: Enter a valid secondary input or terminate secondary input with a control key.

ILLEGAL OVERLAY NUMBER

Condition: The Overlay Patch secondary input has an invalid overlay number.

User Action: Re-enter the entry with a valid overlay number.

LOAD NEXT OVERLAY? (N/Y)

Condition: An overlay is about to be loaded. This is the before-loading option of the Overlay Trap command.

User Action: Enter Y to load the overlay. Enter any other character to return the TESTER program input query (?).

LENGTH FORMAT INCOMPATIBLE WITH CORRECTIONS

Condition: An Overlay Patch secondary input specifies both a length format and core correction, or a Core Correctors secondary input specifies length format.

User Action: Correct the entry and re-enter.

MAILBOX IS BUSY, ENTRY POINT NOT SET.

Condition: The common core mailbox flag at location 00580C indicates busy status (B). The TESTER program entry point at 00990C is not set.

User Action: Do not use the TESTER entry point. Either branch directly to the TESTER origin point (if less than 09999C), use the Core Correctors command to set up an entry point branch to TESTER, or make the mailbox available and reload TESTER program.

NON-NUMERIC LINK

Condition: A non-numeric linkage other than end-of-file has been encountered in a linked sequential file dump. The dump terminates.

User Action: None.

NON-NUMERIC LINK IN LOADING

Condition: The user specified the loading of a program file, and a non-numeric linkage, other than an end-of-file, has been encountered before an object start card (S-card).

User Action: Display the contents of the specified program file and correct the condition. Then, re-enter the command.

NOT PARTITION ZERO.

Condition: The FAC command has been specified from a partition other than zero.

User Action: Perform the FAC command from partition zero.

'OT' TURNED OFF

Condition: A Trace or Stop command was specified, and the existing Overlay Trap command was cancelled.

User Action: None.

OVER 63 CHARACTERS.

Condition: A data entry on an Overlay Patch or Core Correctors secondary input contains more than 63 data characters.

User Action: Re-enter the entry with a smaller data parameter.

OVERRIDE RECORD PROTECT TABLE LOCK -- PTN nn. (N/Y)

where

nn = the number of the locking partition

Condition: The record protect table is locked by the specified partition after checking through 1000



switches. Common core allocation processing cannot take place.

User Action: Respond Y to clear the record protect table lock and to continue loading TESTER. Respond N to terminate.

(OVL TRAP)

Condition: A Goto, Return, or Load command was specified with Overlay Trap on.

User Action: None.

PARTITION SIZE: nK

where

n = a digit from 1 to 9

Condition: Partition size is less than 10K.

User Action: None.

POOL NAME ERROR

Condition: A condition has been entered with a pool name specified that is not the name of a pool on this disc system.

User Action: Enter a command with the correct pool name.

POOL t LOCKED -- PTN nn.

where

t = L or S (L means long term lock, S means short term lock)

nn = the locking partition number

Condition: The pool specified in the command was locked by the specified partition.

User Action: Press the enter key to retry the pool lock for 100 times. Press any control key to cancel the current step.

READ ERROR -- CC n

where

n = either 1 or 3, specifying the condition code encountered

Condition: An error has occurred in executing a Disc Patch command; this error may be combined with a write error.

User Action: None.

READY DEVICE M1

Condition: A program is being loaded from cards and condition code other than 2 was encountered.

User Action: Either ready the card reader and press the enter key to continue, or press any control key to abort the loading.

RE-ENTER

Condition: The drive number entered was invalid or the volume was not found on that drive.

User Action: Mount the volume and enter the correct drive number.

RETURN ADDRESS NOT DEFINED -- GOTO REQUIRED

Condition: A Return command was specified, but no return address is defined.

User Action: Enter a Goto command.

SECTOR nnnnnn

Condition: The specified sector was unable to be read while loading TESTER overlays.

User Action: If drive is not ready, ready the drive and branch to 9990C, or use a disc analysis program.

SERVICE REQUEST

Condition: The service request condition was encountered. The operation in progress is terminated.

User Action: None.

SIMULATOR NOT USED: USER PROGRAM HAD CONTROL

Condition: The SW command is specified, but no return address is defined.

User Action: None.

SIMULATOR STOPPED AT aaaaa

where

aaaaa = a core address

Condition: TESTER had just completed a Trace simulation; the next address to be executed is aaaaa, or the program being traced went into a load, and the user branched to 0990C; the instruction being traced was at aaaaa.

User Action: None.

SIMULATOR STOPPED AT aaaaa, CONTAINING xxxxxxxxxxxx

where

aaaaa = a core address

xxxxxxxxxxx = the ten characters starting in location aaaaa

Condition: The SW command was specified. aaaaa is either the address being simulated when the program went into a load, or the next address to be executed.

User Action: None.

SKIPPED nnnn NON-TEXT CARDS

where

nnnn = the number of non-object cards

Condition: A program file has been loaded. In that file, the specified number of non-object cards were ignored. These are cards whose first character is less than S. Program files containing such cards will not load using the standard linked sector disc loader.

User Action: Remove these cards before using the linked sector disc loader.

SOFT END-OF-FILE. CONTINUE? (N/Y)

Condition: A soft end-of-file has been detected during the display of a file.

User Action: Respond Y to proceed with the display. Respond N to cancel the display.

START=aaaaa. CONTINUE? (Y/N) or START=aaaaa.-cccccc-CONTINUE? (Y/N)

where

aaaaa = the object start card (S-card) address

Condition: A program segment has been loaded using the Load Target command.

User Action: Enter a new command.

START = aaaaa - CONTINUE? (N/Y)

START = aaaaa - ccccc - CONTINUE? (N/Y)

where

aaaaa = address from the object start card (S-card)

cccccc = comments in the object start card

Condition: An overlay has just been loaded. This is the after-loading option of the Overlay Trap command.

User Action: Enter Y to execute the program in an unsimulated mode starting at location aaaa. Enter any other character to return to the TESTER program input query (?).

(STOP)

Condition: Goto or Return command was specified with Stop on and Trace off.

User Action: None.

SYNTAX ERROR

Condition: A command has been entered with an illegal pool.file syntax.

User Action: Enter the command with valid syntax.

SYSTEM LOCKED -- PTN nn.

where

nn = the locking partition number

Condition: The system lock in DMF standard common is locked by the specified partition.

User Action: Press the enter key to retry the system lock for 100 times. Press any control key to cancel the current stop.

TAPES INHIBITED: 1234

Condition: One or more of the digits may be replaced by a blank. At TESTER load time, FAC protect character specifies that the listed tape drives are inhibited.

User Action: If this is partition zero, the user has the option to change the condition using the FAC command. In partitions other than zero, the user cannot change the condition.

TESTER ACTIVE IN PARTITION xx. CONTINUE (N/Y)

Condition: TESTER is being loaded. There is a version of TESTER already loaded which was last used by another partition.

User Action: Respond Y to continue loading TESTER. Respond N to terminate TESTER.

TESTER IS PROHIBITED ON THIS SYSTEM

Condition: This is a DMF II System in which the system security flag indicates that TESTER is not permitted to run. TESTER exists to the conversational loader.

User Action: Use SETCOM STARTUP to remove the restriction on the use of TESTER.

TESTER LOSES CONTROL. CONTINUE? (N/Y)

Condition: A Goto, Return, or Load command was specified without specifying Trace, Stop, or Overlay Trap.

TESTER 21 CONSOLE MESSAGES

---

User Action: Enter Y to perform the user specified command;  
enter any other character to cancel the user  
specified command.

TESTER MUST BE LOADED ENTIRELY ON ONE PAGE.

Condition: The copy of TESTER being loaded crosses a 10000  
character page boundary.

User Action: Regenerate TESTER using the Linkage Editor,

TESTER r

where

r = the release level

Condition: TESTER has been loaded. This version does not  
permit tracing of extended common core.

User Action: None.

THIS PROGRAM MUST BE LOADED FROM DISC.

Condition: The user has attempted to load TESTER using the  
object card loader. TESTER initiates a program  
check.

User Action: File TESTER on disc and reload.

(TRACE)

Condition: Goto or Return command was specified with Trace  
on and Stop off.

User Action: None.

(TRACE/STOP)

Condition: Goto or Return command was specified with Trace  
and Stop on.

User Action: None.

TRACE TURNED OFF

Condition: An Overlay Trap command was specified and the  
existing Trace and/or Stop command was  
cancelled.

User Action: None.

TSTR21 r.

where

r = the release level

Condition: TESTER has been loaded. This version permits tracing of extended common core.

User Action: None.

TURN PRINTER ON-LINE

Condition: At TESTER load time, the line printer (device 2) is either not attached, powered down, or in local.

User Action: If the line printer is on local, put the device on-line.

UNABLE TO FIND THE SPECIFIED OVERLAY.

Condition: There are fewer overlays in the program file than the overlay specification in the Object Patch secondary input.

User Action: Display the program file and correct the entry.

WRITE ERROR -- CC n

where

n = either 1 or 3, specifying the condition code encounter

Condition: An error has occurred in executing a Disc Patch command; this error may be combined with a read error.

User Action: None.

**SINGER**  
BUSINESS MACHINES

PUB. NO. 44-B700-401-00