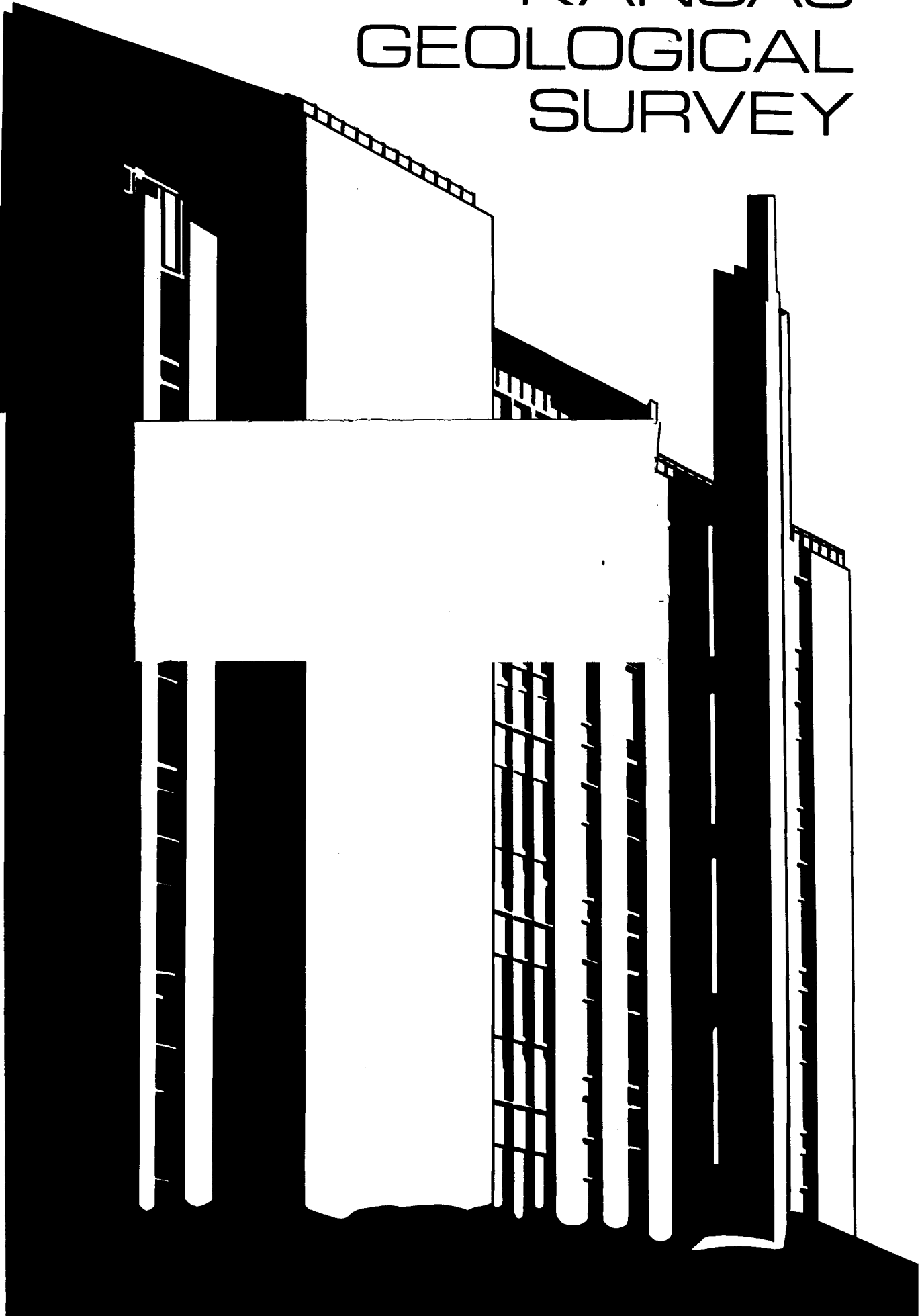


KANSAS GEOLOGICAL SURVEY

KGS OF 86-2 Shaukat, Nadeem

Easy to follow manual for working on MV8000



EASY TO FOLLOW MANUAL^{*}
FOR
WORKING ON MV8000

by

NADEEM SHAUKAT
Kansas Geological Survey
Lawrence, KANSAS

[1986]

* Extracted from Data General's manuals and explained with examples for using at Kansas Geological Survey.

I N D E X

<u>NO.</u>	<u>TOPIC</u>	<u>PAGE</u>
1.	Lossing on/Lossing off on MV8000	1
2.	Changinsing your PASSWORD	2
3.	Use of SED commands	3
4.	CTRL key combinations and their functions	12
5.	Use of CLI commands	13
6.	CLI macros	18
7.	Working with tapes	20

HOW TO LOG-ON MV8000?

1. Turn the power on.
 - a. On Dasher Terminals, find the power on/off switch at their back.
 - b. On Hard Copy Terminals, find the power on/off switch at key boards.
2. Hit CR or NEW LINE key.
In its response, the terminal will ask the USERNAME and PASSWORD.
3. Type in your USERNAME.
It will appear on the screen or paper as you type.
4. Type in your PASSWORD.
Your password will not appear on the screen or paper as you type.
5. If you have typed in the correct USERNAME and PASSWORD, the terminal will respond with the following information.

```
Last previous logon ---DATE---  ---TIME---  
AOS/vs CLI REV ----- Current Date & Time  
)
```

When you see this right paranthes on the terminal screen or paper, you are at Command Line Interpreter (CLI). It means that you have successfully logged on and are able to create files, subdirectories, write programs and edit files.

If you have typed in an incorrect USERNAME or PASSWORD, you will be asked to retype the correct information.

HOW TO LOG-OFF MV8000?

1. After you are finished working with your files, you come back at CLI level, i.e., ')'.
At this level, type BYE.
2. In this response, the terminal will indicate the connect time you worked on during this session, and will also show the current date and time.
3. Turn the power switch off.

HOW TO CHANGE THE PASSWORD?

May be you feel insecure about your files in the directory. It is easy to change the PASSWORD any time as follows.,

1. Log on as described earlier.
2. Answer the response on the terminal as follows.,
 USERNAME? <enter your username>
 PASSWORD? <enter your old password>
3. Instead of hitting the CR/NEW LINE key after you are finished typing your old password, hit CTRL and L keys simultaneously. In response to this, it will ask the new PASSWORD? Type in your new PASSWORD and hit CR-key.

This process will change your password from OLD to NEW. Next time you log on with your new password.

SED INSTRUCTIONS

Data General's SED is a versatile, screen-oriented text editor, whose commands enable you to create and modify text from your terminal.

HOW TO CREATE A FILE?

If you want to create a new file, type in X SED at CLI level. The terminal will respond as the following.,

```
NAME OF FILE TO EDIT:      <enter the file name>
```

If this file name is new, i.e., no file exists with this name in your directory at present, it will ask:

```
DO YOU WANT <filename> TO BE CREATED?  <enter YES>
```

In response to your answer, it will print * on the terminal screen. At this level you can write a program or enter a data with APPEND (or AP) command. After you typed in APPEND at *-level, the terminal will respond with the line numbers. Each hit on CR/NEW LINE key will bring you to the next line.

After you are finished typing the program or the data, hit BREAK/ESC key which will bring you at *-level again. But you are still in SED mode. Type in BYE to quit SED mode and to reach CLI-level.

N.B. You do not have to save this file. It is automatically saved under the file name you have given in the beginning.

MAIN SED COMMANDS

Before we go through the SED commands, let us find a few useful abbreviations.

C	for	Current
LA	for	Last
Li	for	List
N	for	Next
PA	for	Page
PO	for	Position
PR	for	Previous

Following are the most frequently used SED commands with simple examples. (You must be in SED mode while using these commands).

1. APPEND (or AP)

This command is used when a text is desired to be appended next to last line of a file, or to create a new file. To terminate an appending session from the terminal keyboard, press BREAK/ESC key. This returns the SED prompt *.

Examples;

1. Appending a text to last line of a file.

```
*PO LA           <will position you at last line>
*AP             <will activate APPEND command>
  THIS IS MY 11th LINE.    <appended text>
  THIS IS MY 12th LINE.    <appended text>
                        <After you are finished with all the
                        text you want to append, hit ESC key.
                        You will be again at SED prompt>
```

2. Appending a text to a file from another file.

Assume you are in FILE1's SED mode right now, and you want to append line 20 to 30 of FILE2 to FILE1.

```
*AP 20 30 FROM FILE2    <will paste lines 20 to 30 of
                        FILE2 at the end of FILE1>
```

```
<hit ESC to terminate APPEND>
```

2. BYE (or B)

This command terminates the SED session, when you are in SED mode.

When you type in BYE out of SED mode, it logs off the terminal.

Example;

```
*BYE             <will terminate the SED session>
```

```
)BYE            <this will log off the terminal,
                 because BYE is typed at CLI level>
```

3. CUT_LINE (or CU)

This command splits the line at the desired column and forms two lines.

Example;

If you have a 9th line of the text as follows.,
ABCD123456
*CU 9 4
ABCD <newly formed 9th line after splitting>
123456 <newly formed 10th line after splitting>

N.B. The DASHER terminals have a CUT function key (11th from left in the upper most row of key board), which performs the same function but you have to move to the desired column using the arrow keys.

4. DELETE (or DE)

This command removes text from the file.

Examples;

*DE <will delete the current line>
*DE 4 10 <will delete lines 4 through 10>
*DE PA 2 <will delete the whole page 2>
*DE 1 LA <will delete all lines on the current page>

5. DESTINATION

Formats

ONTO	pathname
BEFORE	address
AFTER	address

The destination will be used with MOVE and DPLICATE later.

6. DISPLAY (or DIS)

This command displays statistics on the current file and mode.

Example;

*DIS <will print the statistics about your current file>

7. DO

This command temporarily suspends the editing session to execute CLI commands. When execution of CLI command is complete, the SED prompt * comes back after you hit CR/NEW LINE key.

This command is very helpful when you want to see something in another file without terminating the current SED mode.

Example;

If you are currently in SED mode of FILE1 and you want to access another file FILE2 without terminating the SED mode of FILE1, follow as below.,

```
*DO X SED/NO_ED FILE2      <This will bring you into SED
                             mode of FILE2 temporarily. Do
                             whatever you want to do with
                             file and hit CR/NEW LINE key,
                             which will bring you back to
                             the SED mode of FILE1>
```

8. DUPLICATE (or DUF)

This command copies text on anywhere on the current page or on another file.

Examples;

```
*DUP 1 20 AFTER LAST      <will duplicate lines from 1 to 20
                             after the last line of current page>

*DUP 1 10 BEFORE 35      <will duplicate lines from 1 to 10
                             before 35th line of current page>

*DUP 1 50 ONTO FILE6     <will duplicate lines from 1 to 50
                             onto another file named FILE6>
```

9. FIND (or F)

This command finds a text string. It is very useful in searching the string desired to make any changes.

12. JOIN (or J)

This command removes a page break in the file. Each page in the SED has maximum capacity of 1023 lines. But sometimes if you have broken your page with SPLIT command to accommodate your extra text and then want to rearrange the pages, you go to next page and join it to the previous page.

Example#

Assume your first page has 920 lines, and you want to complete it by joining the rest of lines from the next page, i.e., page 2. Follow these instructions.,

```
*PO PA 2          <will position page 2>
*JOIN            <will join page 2 at the end of page 1
                  and rearrange the line numbers>
```

13. LIST (or L)

This command lists the required number of lines on the terminal screen.

Examples#

```
*L 1 10          <will list lines from 1 to 10>
*L ALL          <will list the whole current page>
*L LA          <will list the last line on current page>
```

14. MODIFY (or MOD)

This command is very useful in making changes in your text with moving the cursor to the desired location. After each change, press CR/NEW LINE key to have a line with modifications.

Movement of cursor can be controlled by the following combination of keys.,

```
CTRL A          : <move to end of the string>
CTRL B          : <move back to the end of previous word>
CTRL E          : <enter/exit insert character mode>
CTRL F          : <move forward to the start of next word>
CTRL H          : <move to beginning of the string>
```

CTRL I : <insert a tab>
CTRL K : <erase all to the right of the cursor>

15. MOVE (or MOV)

This command moves a text from one place to the other within the same file or onto another file. It deletes the text from its original position.

Examples#

*MOVE 1 10 AFTER LA <will move first 10 lines after the last line>
*MOVE 1 10 ONTO FILE4 <will move first 10 lines of the current file to the end of FILE4 if it already exists, otherwise will create a new file with this name>

16. PASTE_LINES (or PAS)

This command merges a number of lines into one line, or the line addressed with the following line.

Example#

*LI 1 3 <will list lines from 1 to 3>
1 THIS IS A
2 N EXAMPLE OF PAS <listing of lines from 1 to 3>
3 TING.
*PAS 1 3 <will paste all 3 lines>
1 THIS IS AN EXAMPLE OF PASTING. <pasted line>

N.B. The function key F11 performs the same process.

17. POSITION (or PO)

This command moves the current line position to a different address in the file.

Examples#

*PO PA 2 <will position page 2>
*PO 25 <will position 25th line of current page>

N.B. Upward and downward arrow key can also be used for this purpose.

18. REPLACE (or R)

This command deletes and replaces it with a new text entered from the terminal keyboard.

Example#

```
*LI 1 2          <will list first 2 lines>
1   THIS IS LINE 1.
2   THIS IS LINE 2.
*REPLACE
  THIS IS A NEW LINE.    <new line typed in>
                          <hit ESC key>

*LI 1 2
1   THIS IS LINE 1.    <first line is the same as before>
2   THIS IS A NEW LINE. <second line is replaced with the
                          new one>
```

19. SAVE

This command writes a copy of file including all changes made in the current editing session to a SAVE file. The name of the file will be the current file name with .SV added to to the file name.

Example#

```
*SAVE          <will save text into <filename>.sv>
```

20. SPLIT (or SP)

This command makes a page break in the file.

Sometimes you have to insert a line or lines in a page which has already 1023 lines (maximum capacity of a page). Then you have to split that page into two segments to accomodate new text.

Example#

```
*PO PA 1        <will position page 1>
*PO 225         <will position page 225>
*SPLIT         <will split page 1 into two pages
                making lines 1 to 224 on first page
```

and line 225 and rest on the second
page>

21. SUBSTITUTE (or SUB)

This command substitutes a word or phrase for another word or phrase throughout the range of the text.

Examples:

*SUBS'1977'F'1976'CU	<will change 1976 to 1977 on the current line>
*SUBS'1977'F'1976'ALL	<will change 1976 to 1977 in all places on the current page>
*SUBS'1977'F'1976'IN PA ALL	<will change 1976 to 1977 on all pages>

22. UNDO (or U)

This command restores the most recently deleted text in the file.

Example:

*LI 1 2	<will list lines 1 and 2>
1 THIS IS LINE 1.	
2 THIS IS LINE 2.	
*DEL 1 2	<will delete lines 1 and 2>
*LI 1 2	<will list lines 1 and 2>
1 THIS is LINE 3.	
2 THIS IS LINE 4.	
*UNDO	<will restore the deleted text>
*LI 1 2	
1 THIS IS LINE 1.	
2 THIS IS LINE 2.	

CTRL-KEY COMBINATIONS AND THEIR FUNCTIONS

Following combinations of various keys with CTRL-key are very helpful and efficient in editing.

CTRL A : move to end of string.
CTRL B : move back to end of previous word.
CTRL E : enter/exit insert character mode.
CTRL F : move forward to the start of next word.
CTRL H : move to the beginning of string.
CTRL I : insert a tab.
CTRL K : erase anything on the right of the cursor.
CTRL L : clear screen. (CLI level only)
CTRL S : stop displaying.
CTRL Q : resume displaying.
CTRL U : erase the current line.
CTRL C - CTRL A : interrupt the execution of current command.
CTRL C - CTRL B : abort the current program and comes back to present level. (if on CLI, will log off)

SOME USEFUL CLI COMMANDS

Following are the most frequently used CLI commands with examples.

1. ACL

Set or display the access control list for a file.

OPTIONS:

O --- Owner's access
W --- Write access
A --- Append access
R --- Read access
E --- Execute access

Example:

)ACL MYFILE OWARE <will give the all access permissions
to the file named MYFILE>

2. BYE

Terminate the CLI at your console. Typing BYE at CLI level at any stage will log you off on that console.

3. COPY

Copy one or more files to a destination file.

Format:)COPY destination:filename sourcefile

Examples:

1.)COPY FILE2 FILE1 <will copy FILE1 onto another file
FILE2 in the same directory>
2.)COPY :UDD:GH.SCRATCH:FILE2 FILE1
<will copy FILE1 onto another file
FILE2 in directory SCRATCH>
3.)COPY/A FILEALL FILE1 FILE2 FILE3
<will append all three files FILE1
FILE2 and FILE3 in the same order
and will copy onto another file
named FILEALL>

4. CREATE

Create a file or directory.

Examples:

1.)CREATE/I MYFILE <will create a file with name
))INTEGER A1, A2 MYFILE with the contents typed
))..... in>
)).....TEXT.....
)).....
))END
))
)

2.)CREATE/DIR PROJ1 <will create a new subdirectory
) PROJ1>

5. DELETE (or DEL)

Delete one or more files or subdirectories.

Examples:

1.)DEL MYFILE <will delete a file MYFILE>

2.)DEL FILE1 FILE2 FILE3 <will delete all three files>

3.)DEL/V FILE+ <will delete all files with name
 deleted FILE1 starting FILE and will verify
 deleted File2 (/V) that all the files with these
 names have been deleted>

4. DEL/C/V FILE <will confirm and verify deletion
 =FILE1 ? Y of each file one by one>
 deleted FILE1 <type Y if you want to delete,
 otherwise CR/NEW LINE>

6. DIRECTORY (or DIR)

Set or display the current directory settings.

Examples:

1.)DIR <will display your current directory
 name>

-) :UDD:GH.NAME <your current directory name>
- 2.)DIR/I <will take you to your parent directory>
- 3.)DIR :UDD:GH.SCRATCH <will take you to directory SCRATCH>
- 4.)DIR SUBDIR <will take you to subdirectory, where SUBDIR is the name of your subdirectory>

N.B. You cannot transfer from one subdirectory to another subdirectory. You must have to go to your parent directory first, and then to another subdirectory.

7. DISMOUNT

Request operator to dismount the tape you have mounted to work with.

Example:

)DISMOUNT TAPE <operator will dismount the tape from the drive>

8. EXECUTE (or X)

Execute a program.

Example:

)X MYPROGRAM <will execute the program MYPROGRAM>

9. FILESTATUS (or F)

Displays status information for one or more files.

Examples:

- 1.)F/S <will list all filenames in your directory in alphabetical order>
- 2.)F/AS/S <will display the assortment (AS) of file information, file type, date and time of creation, and file length>

10. HELP

Explain a CLI command, pseudo-macro or general topic.

Examples:

1.)HELP <will list all CLI commands available>
2.)HELP *NAME <will list the information about the use of the command, where NAME is the command name>

11. MOUNT

Ask the operator to mount the tape.

Example:

)MOUNT TAPE ##### (with or without) RING

12. MOVE

Move copies of one or more files.

Examples:

1.)MOVE :UDD:GH.SCRATCH FILE1 <will move a copy of FILE1 from the current directory to directory SCRATCH>
2.)MOVE/V :UDD:GH.SCRATCH FILE+ <will move all files with names starting with FILE to directory SCRATCH, with verification (/V) list>

N.B. This command does not delete the original file.

13. PERMANENCE (or PERM)

Set or display a files permanence attributes.

Examples:

1.)PERM MYFILE ON <will make MYFILE permanent and undeleteable>

2.)PERM MYFILE OFF <will disable permanence of MYFILE>

14. QPRINT

Place an entry on the PRINT queue.

Example#

)QPRINT MYFILE <will print MYFILE on the printer, with your filename and username on the printout>

15. RENAME

Change a file's name.

Example#

)RENAME MYFILE FILE1 <will change the filename MYFILE to FILE1>

16. SPACE

Display the amount of disk space in the control-point directory or logical disk.

17. TYPE (or TY)

Type the contents of a file.

Example#

)TY MYFILE <will display the contents of MYFILE on the console>

CLI MACROS

You can save time and effort by writing a sequence of commands into macro files. When you type the macro name, the CLI executes all the commands in the macro.

HOW TO CREATE CLI MACROS?

CLI macros can be created in SED in the same way we create program files or data files. But you must name the file ending with .CLI .

Another way to create the CLI macro file is as following.,

```
)CREATE/I macroname.CLI
))
```

where

CREATE	----	is the command name.
/I	----	indicates that the input for the macro is from @INPUT, in this case from the USER's console.
macroname	----	is the name you supply for the macro file.
.CLI	----	indicates that this file is a macro.

When you receive the double prompt))), enter the macro text. Terminate each input line with a CR/NEW LINE key, and end the macro with a right paranthes and CR/NEW LINE.

Examples;

```
1. )CREATE/I SCRATCH.CLI
   ))DIR :UDD:GH.SCRATCH
   ))WRITE YOU ARE NOW IN DIR :UDD:GH.SCRATCH
   ))
   )
```

Here you have created a small macro with name SCRATCH.CLI which on typing only SCRATCH at CLI level will take you to directory SCRATCH. The 3rd line in the example is optional. It appears on the console whatever you write after WRITE in a macro file. In this case, the third line lets you know on execution of the macro that you have reached the directory SCRATCH.

No matter in which directory you are, you can activate the macro file by typing just macro filename. In this case, typing only SCRATCH at CLI level will take you to directory SCRATCH.

This macro will save your time because whenever you have to reach directory SCRATCH, you will have to type only SCRATCH instead of DIR :UDD:GH.SCRATCH.

```
2.  )CREATE/I SED.CLI
    ))X SED/NO_ED/NO_FORM %1%
    ))
    )
```

This macro activates the SED with options NO_ED and NO_FORM. %1% is a dummy argument. It replaces the first argument following the macro name.

With this macro in your main directory, you just type SED at CLI level, and it will ask you to enter the filename to be edited.

N.B. For more details, see CHAPTER 5 of AOS CLI USER's manual.

WORKING WITH TAPES

Following are different situations with examples which explain reading from and writing onto tapes on MV8000.

1. HOW TO COPY (WRITE) A FILE ONTO A TAPE FROM YOUR DIRECTORY?

Assume you have a file named MYPROGRAM in your working directory and you want to write it on a tape for later use. Follow these instructions at CLI level.

```
)MOUNT TAPE ##### WITH RING    <will ask operator to mount
                                the tape on drive; ##### is
                                the name or number on the
                                tape>
)DUMP/V/L=FILENAME TAPE:0 MYPROGRAM
                                <this will write MYPROGRAM on
                                tape location 0; FILENAME
                                can be any name>
)TY FILENAME                    <will verify name of file
                                dumped with date and time>
)DISMOUNT TAPE                  <will ask operator to dismount
                                tape>
```

N.B. (1) Repeat the DUMP command for any number of files you want to write on the tape changing the TAPE:# each time successively (i.e., TAPE:0 for first file, TAPE:1 for second file and so on).

(2) Sometimes we need to write a number of files with names starting or ending with a common name i.e., FILE1, FILE2, FILE3, _____. Then type

```
)DUMP/V/L=FILENAME TAPE:0 FILE+
```

This will write all files with name starting with FILE on TAPE:0.

2. HOW TO READ A FILE FROM TAPE (WRITTEN ON MV8000) ONTO YOUR DIRECTORY?

Assume you have a file MYFILE saved on tape in TAPE:0 (the first file on the tape), and you want to read and save it in your directory. Follow these instructions at CLI level.

```
)MOUNT TAPE ##### NO RING      <will ask the operator to
                                mount your tape>
)LOAD/V TAPE:0 MYFILE          <will bring the first file
                                from the tape and store it
                                in your directory under
                                filename MYFILE>
```

After you are finished reading all the required files from the tape, dismount the tape by typing the followings.,

```
)DISMOUNT TAPE
```

N.B. If you do not know the name of file on the tape, you can know it by typing the followings.,

```
)LOAD/N TAPE:0 +
```

This will list all file names stored in TAPE:0.

3. HOW TO WRITE A FILE ONTO A TAPE ON MV8000 WHEN THE TAPE HAS TO BE READ ON SOME OTHER SYSTEM?

Sometimes you are requested to send a program or data on a tape to some outside agency. Follow these instructions to copy a program or data onto a tape.

```
)MOUNT TAPE ##### WITH RING
)X D2F/B=10 MYFILE TAPE:0
)DISMOUNT TAPE
This process will copy MYFILE onto TAPE:0.
```

4. HOW TO READ A FILE FROM A TAPE RECEIVED FROM OUTSIDE (WRITTEN ON A SYSTEM OTHER THAN MV8000)?

If you received a tape from outside and you want to read a file or files from it, follow these instructions.,

IF WRITTEN IN ASCII

```
)MOUNT TAPE ##### NO RING
)X F2D/R=80/B=10 TAPE:0 FILENAME(any)
)DISMOUNT TAPE
If record length is 132, replace R=80 with R=132.
```

IF WRITTEN IN EBSDIC

```
)MOUNT TAPE ##### NO RING
)X EBCASC/R=132/B=13200 TAPE:0 FILENAME(any)
)DISMOUNT TAPE
```