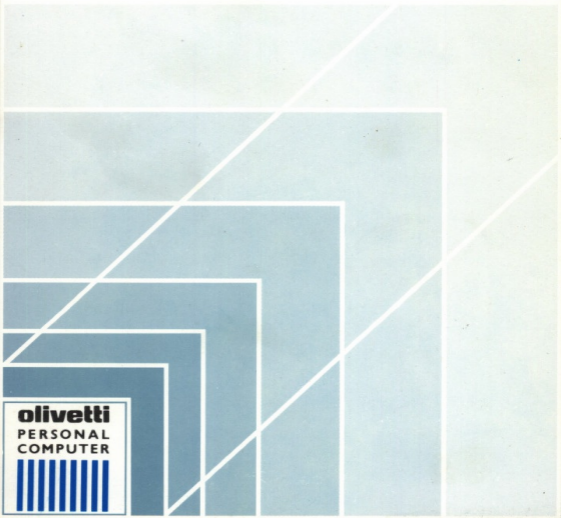




Operating Systems and Languages Library

# MS-DOS

User Guide



**olivetti**  
PERSONAL  
COMPUTER



**olivetti**



Operating Systems and Languages Library

# MS-DOS

User Guide

**olivetti**  
PERSONAL  
COMPUTER



**olivetti**

# Olivetti MS DOS User Guide

1. [Olivetti MS DOS User Guide](#)
2. [PREFACE](#)
3. [SUMMARY](#)
4. [RELATED PUBLICATIONS](#)
5. [CONTENTS](#)
6. [ABOUT THIS CHAPTER](#)

7. [CONTENTS](#)
8. [WHAT IS MS-DOS?](#)
9. [COMMAND LIBRARY](#)
10. [BATCH PROCESSING](#)
11. [REAL-TIME CLOCK](#)  
[FUNCTIONS](#)
12. [INTERFACE](#)  
[HANDLING](#)
13. [DISKS](#)
14. [VIRTUAL DISK](#)
15. [NOTATION](#)  
[CONVENTIONS](#)
16. [INTRODUCTION](#)
17. [ABOUT THIS](#)

# CHAPTER

18. CONTENTS

19. INTRODUCTION

20. MS-DOS CONTROL  
KEYS

21. EDITING FUNCTION  
KEYS

22. MS-DOS CONTROL  
KEYS AND EDITING  
FUNCTION KEYS

23. ABOUT THIS  
CHAPTER

24. HOW MS-DOS KEEPS  
TRACK OF YOUR

## FILES

25. HOW TO NAME YOUR FILES
26. WILD CARDS
27. RESERVED DEVICE NAMES
28. HOW TO COPY YOUR FILES
29. HOW TO PROTECT YOUR FILES
30. HOW TO EXAMINE THE CONTENTS OF A FILE
31. DIRECTORIES

32. FILE NAMES AND PATHS
33. HOW TO CHECK THE VALIDITY OF YOUR FILES
34. FILES AND DIRECTORIES
35. ABOUT THIS CHAPTER
36. COMMAND SYNTAX
37. INTERNAL AND EXTERNAL COMMANDS
38. INFORMATION

COMMON TO ALL  
MS-DOS COMMANDS

39. BATCH PROCESSING

40. INPUT AND OUTPUT

41. FILTERS

42. ENTERING AND

USING MS-DOS  
COMMANDS

43. COMMAND PIPING

44. ABOUT THIS

CHAPTER

45. INTRODUCTION

46. MS-DOS 3.20 AND

NETWORKING



47. ASSIGN
48. ATTRIB
49. BACKUP
50. BREAK
51. CHDIR
52. CHKDSK
53. CLS
54. COMMAND
55. COMP
56. COPY
57. CTTY
58. DATE
59. DEL
60. DIR

61. [DISKCOPY](#)
62. [COMMANDS](#)
63. [9 ECHO](#)
64. [Characteristics](#)
65. [Remarks](#)
66. [FORMAT](#)
67. [COMMANDS](#)
68. [GOTO](#)
69. [COMMANDS](#)
70. [GRAFTABL](#)
71. [GRAPHICS](#)
72. [GWBASIC](#)
73. [COMMANDS](#)
74. [HEXDUMP](#)

75. JOIN
76. LABEL
77. COMMANDS
78. MKDIR
79. MODE
80. MODE COM:
81. MODE n
82. COMMANDS
83. MODE LPT
84. COMMANDS
85. MORE
86. PATH
87. PAUSE
88. PRINT

- 89. [COMMANDS](#)
- 90. [PROMPT](#)
- 91. [RECOVER](#)
- 92. [REM](#)
- 93. [REN](#)
- 94. [COMMANDS](#)
- 95. [REPLACE](#)
- 96. [RESTORE](#)
- 97. [RMDIR](#)
- 98. [SET](#)
- 99. [SHARE](#)
- 00. [SHIFT](#)
- 01. [SORT](#)
- 02. [SUBST](#)

03. [SYS](#)
04. [TIME](#)
05. [COMMANDS](#)
06. [TREE](#)
07. [TYPE](#)
08. [VER](#)
09. [VERIFY](#)
10. [mami/smm](#)
11. [WINDOW MOVING](#)  
[FUNCTION KEYS](#)
12. [EXITING AND](#)  
[SAVING FUNCTION](#)  
[KEYS](#)
13. [COMMANDS AND](#)

# SEARCHING

14. GOTO
15. DELETE
16. FILE
17. VIDEO FILE EDITOR
18. ABOUT THIS  
CHAPTER
19. INTRODUCTION
20. HOW TO INVOKE THE  
EDLIN PROGRAM
21. EDLIN
22. INTER-LINE  
COMMANDS
23. A (APPEND LINES)

24. C (COPY LINES)
25. D (DELETE LINES)
26. E (END EDITING)
27. I (INSERT LINES)
28. L (LIST TEXT)
29. LINE EDITOR (EDLIN)
30. LINE EDITOR (EDLIN)
31. M (MOVE LINES)
32. P (PAGE)
33. Q (QUIT EDITING)
34. R (REPLACE TEXT)
35. S (SEARCH TEXT)
36. T (TRANSFER LINES)
37. W (WRITE LINES)

- 38. [INTRA-LINE  
COMMANDS](#)
- 39. [COPY1](#)
- 40. [COPYTO](#)
- 41. [COPYLINE](#)
- 42. [SKIP1](#)
- 43. [SKIPTO](#)
- 44. [KILL](#)
- 45. [INS](#)
- 46. [LINE EDITOR \(EDLIN\)](#)
- 47. [NEWTEMP](#)
- 48. [ABOUT THIS  
CHAPTER](#)
- 49. [THE LINKER](#)



- 50. [INTRODUCTION](#)
- 51. [INTERACTIVE ENTRY](#)
- 52. [COMMAND LINE  
ENTRY](#)
- 53. [THE LINKER](#)
- 54. [AUTOMATIC  
RESPONSE FILE  
ENTRY](#)
- 55. [ABOUT THIS  
CHAPTER](#)
- 56. [CONTENTS](#)
- 57. [INTRODUCTION](#)
- 58. [HOW TO INVOKE THE  
DEBUG PROGRAM](#)

- 59. DEBUG
- 60. DEBUGGING  
COMMANDS
- 61. A (ASSEMBLE)
- 62. C (COMPARE)
- 63. D (DUMP)
- 64. E (ENTER)
- 65. F (FILL)
- 66. G (GO)
- 67. H (HEX)
- 68. L (LOAD)
- 69. M (MOVE)
- 70. N (NAME)
- 71. O(OUTPUT)

- 72. [P \(PROCEED\)](#)
- 73. [Q\(QUIT\)](#)
- 74. [R\(REGISTER\)](#)
- 75. [S \(SEARCH\)](#)
- 76. [T \(TRACE\)](#)
- 77. [U \(UNASSEMBLE\)](#)
- 78. [W \(WRITE\)](#)
- 79. [THE DEBUGGER](#)
- 80. [ABOUT THIS](#)  
[APPENDIX](#)
- 81. [CONTENTS](#)
- 82. [BASIC ASCII CODE](#)  
[\(ISO 7-bit code\)](#)
- 83. [EXTENDED ASCII](#)

CODE FOR THE  
DISPLAY (ISO 8-bit  
code)

84. NATIONAL  
VARIATIONS IN  
EXTENDED ASCII  
CODE (ISO 8-bit code)
85. EXTENDED  
KEYBOARD CODES  
(FOR USA  
KEYBOARDS)
86. ASCII DISPLAY AND  
KEYBOARD CODE  
TABLES

- 87. [ABOUT THIS APPENDIX](#)
- 88. [INTRODUCTION](#)
- 89. [CURSOR MOVING FUNCTIONS](#)
- 90. [ERASING FUNCTIONS](#)
- 91. [GRAPHIC MODE FUNCTIONS](#)
- 92. [ANSI ESCAPE SEQUENCES](#)
- 93. [KEY REASSIGNMENT](#)
- 94. [ABOUT THIS APPENDIX](#)
- 95. [INTRODUCTION](#)

- 96. [CONFIGURING  
COMMANDS](#)
- 97. [BREAK](#)
- 98. [BUFFERS](#)
- 99. [COUNTRY](#)
- 200. [DRIVPARM](#)
- 201. [FILES](#)
- 202. [LASTDRIVE](#)
- 203. [SHELL](#)
- 204. [STACKS](#)
- 205. [CONFIGURING MS-  
DOS](#)
- 206. [ABOUT THIS  
APPENDIX](#)

- 207. [CONTENTS](#)
- 208. [ERROR MESSAGES](#)
- 209. [DEVICE AND DISK  
DRIVE ERRORS](#)
- 210. [ERROR MESSAGES IN  
ALPHABETICAL  
ORDER](#)
- 211. [ERROR MESSAGES](#)
- 212. [ERROR MESSAGES](#)
- 213. [ERROR MESSAGES](#)
- 214. [ERROR MESSAGES](#)
- 215. [ABOUT THIS  
APPENDIX](#)
- 216. [CONTENTS](#)

- 217. [GLOSSARY OF TERMS](#)
- 218. [GLOSSARY OF TERMS](#)
- 219. [ABOUT THIS APPENDIX](#)
- 220. [INTRODUCTION](#)
- 221. [INSTALLING VDISK.SYS](#)
- 222. [THE VIRTUAL DISK SYSTEM](#)
- 223. [ABOUT THIS APPENDIX](#)
- 224. [CONTENTS](#)



- 225. [INTRODUCTION](#)
- 226. [INSTALLING  
DRIVER.SYS](#)
- 227. [INSTALLING NEW  
BLOCK DEVICES](#)
- 228. [INDEX](#)
- 229. [A](#)
- 230. [B](#)
- 231. [oliuelli](#)

# Olivetti MS DOS User Guide

This book was  
produced in  
EPUB format by  
the Internet

Archive.

The book pages were scanned and converted to EPUB format automatically.

This process relies on optical character recognition, and is somewhat susceptible to errors. The book

may not offer  
the correct  
reading  
sequence, and  
there may be  
weird characters,  
non-words, and  
incorrect guesses  
at structure.

Some page  
numbers and  
headers or  
footers may  
remain from the

scanned page.

The process which identifies images might have found stray marks on the page which are not actually images from the book. The hidden page numbering which may be available to your

ereader  
corresponds to  
the numbered  
pages in the  
print edition, but  
is not an exact  
match; page  
numbers will  
increment at the  
same rate as the  
corresponding  
print edition, but  
we may have  
started

numbering  
before the print  
book's visible  
page numbers.  
The Internet  
Archive is  
working to  
improve the  
scanning process  
and resulting  
books, but in the  
meantime, we  
hope that this  
book will be

useful to you.

The Internet  
Archive was  
founded in 1996  
to build an  
Internet library  
and to promote  
universal access  
to all knowledge.  
The Archive's  
purposes include  
offering  
permanent

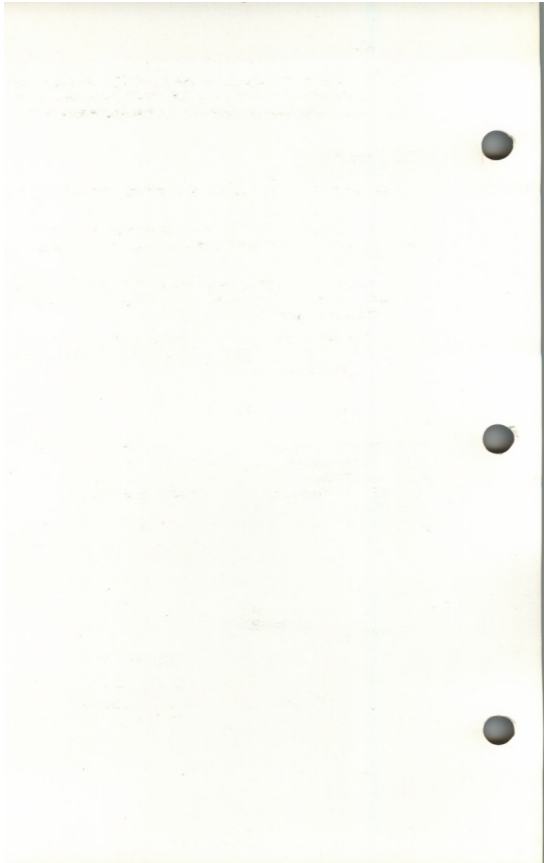


access for  
researchers,  
historians,  
scholars, people  
with disabilities,  
and the general  
public to  
historical  
collections that  
exist in digital  
format. The  
Internet Archive  
includes texts,  
audio, moving

images, and software as well as archived web pages, and provides specialized services for information access for the blind and other persons with disabilities.

Created with

abbyy2epub  
(v.1.7.2)



# PREFACE

This manual is a user guide for the MS-DOS operating system. It describes Microsoft Version 3.20 of MS-DOS, and is for anyone who wishes to use this operating system on an Olivetti Personal Computer.

# SUMMARY

The first chapter provides a general introduction to MS-DOS.

Chapters 2 and 3 describe in more detail the major functions and features of MS-DOS. These include: using control keys and function keys; files and the hierarchical directory

structure; entering and using MS-DOS commands.

Chapter 5 is a full and detailed reference to all the MS-DOS commands. It includes a section on commands to use and those not to use when the computer is connected to a network.

Chapters 6 to 10 provide detailed reference information on the Video

File Editor (EDIT), the Line Editor (EDLIN), the Linker (MS-LINK) and the Debugger (DEBUG).



# RELATED PUBLICATIONS

Installation and Operations  
Guide for your Personal  
Computer Getting Started  
With MS-DOS: Software  
Installation Guide (Code  
4040360 J)

MS GW-BASIC Interpreter

under MS-DOS User Guide  
(Code 4034490 C)

MS-DOS System

Programmer Guide, Vol I  
(Code 4024270 M) MS-DOS

System Programmer Guide,  
Vol II (Code 4033300 G)

MS-DOS Quick Reference  
Guide (Code 4034470 S)

DISTRIBUTION: General  
(G)

FIRST EDITION: January  
1987

Copyright © Microsoft  
Corporation 1980 - 1986

Copyright © 1987 by Olivetti  
All rights reserved

PUBLICATION ISSUED  
BY:

Ing. C. Olivetti & C., S.p.A.  
Direzione Documentazione

77, Via Jervis - 10015 Ivrea  
— Italy

- OLIVETTI is a trademark of Ing. C. Olivetti & C., S.p.A.
- OLITALK is a trademark of Ing, C. Olivetti & C., S.p.A.
- OLITERM IS a trademark of Ing. C. Olivetti & C., S.p.A.

- SORTP is a trademark of Ing. C. Olivetti & C., S.p.A.
- ASM-86 is a trademark of Digital Research
- CB-86 is a trademark of Digital Research
- CBASIC-86 is a trademark of Digital Research
- CLEO is a trademark of Phone 1 Inc.

- Concurrent CP/M-86 is a trademark of Digital Research
- Concurrent DOS is a trademark of Digital Research
- CP/M-86 is a trademark of Digital Research
- DDT-86 is a trademark of Digital Research

- Dr. Draw is a trademark of Digital Research
- Dr. Graph is a trademark of Digital Research
- Dr. Logo is a trademark of Digital Research
- ETHERNET is a trademark of Xerox Corp.
- GEM COLLECTION is a trademark of Digital

# Research

- GEM DESKTOP is a trademark of Digital Research
- GEM DRAW is a trademark of Digital Research
- GEM GRAPH is a trademark of Digital Research
- GEM PAINT is a trademark of Digital Research



- GEM WORDCHART is a trademark of Digital Research

- GEM WRITE is a trademark of Digital Research

- GSS is a registered trademark of Graphic Software Systems,

Inc.

- GSS-CHART is a trademark of Graphic Software Systems, Inc.
- GSS-DRIVERS is a trademark of Graphic Software Systems, Inc.
- GSS-KERNEL SYSTEM is a trademark of Graphic Software Systems, Inc.
- GSS-TOOLKIT is a trademark of Graphic

# Software Systems, Inc.

- GSX-86 is a trademark of Digital Research
- GW is a trademark of Microsoft Corp.
- IBM is a registered trademark of International Business Machines Corp.
- MICROSOFT is a registered trademark of

Microsoft Corp.

- MS is a trademark of Microsoft Corp.
- OMNINET is a trademark of Corvus Systems Inc,
- p-System is a trademark of Softech Microsystem, Inc
- PC-DOS is a trademark of International Business Machines Corp.

- PEACHPAK is a trademark of Peachtree Software International Ltd.

- Personal Basic is a trademark of Digital Research

- SID-86 is a trademark of Digital Research

- TerminAll is a trademark of TOPIC System, Inc.

- TOPIC is a trademark of TOPIC System, Inc.
- UCSD and UCSD Pascal are registered trademarks of the Regent of the University of California
- UNIX is a trademark of Beil Laboratories
- Z80 Is a registered trademark of Zilog Inc.

- Z8000 is a registered trademark of Zilog Inc.

- XENIX is a registered trademark of Microsoft Corp.

# CONTENTS

## 1. INTRODUCTION

WHAT IS MS-DOS? 1-1

COMMAND LIBRARY 1-1

FILE AND DIRECTORY  
HANDLING 1-1

PROGRAMMING TOOLS 1-  
1



# INTERNAL AND EXTERNAL COMMANDS

1-2

# BATCH PROCESSING 1-2

# THE AUTOEXEC.BAT FILE

1-2

# REAL-TIME CLOCK

# FUNCTIONS 1-2

# INTERFACE HANDLING 1-

3

DISKS 1-3

DISKETTE HANDLING 1-5

PURCHASING DISKETTES  
1-6

LABELING DISKETTES 1-6

WRITE-PROTECTION 1-6

VIRTUAL DISK 1-8

NOTATION

# CONVENTIONS 1-9

## 2. MS-DOS CONTROL KEYS AND EDITING FUNCTION KEYS

### INTRODUCTION 2-1

### MS-DOS CONTROL KEYS 2-1

### EDITING FUNCTION KEYS 2-4

## 3. FILES AND DIRECTORIES

HOW MS-DOS KEEPS TRACK OF YOUR FILES 3-1

HOW TO NAME YOUR FILES 3-1

WILD CARDS 3-2

THE ? WILD CARD 3-2

THE \* WILD CARD 3-3

RESERVED DEVICE  
NAMES 3-4

HOW TO COPY YOUR  
FILES 3-4

HOW TO PROTECT YOUR  
FILES 3-5

HOW TO EXAMINE THE  
CONTENTS OF A FILE 3-6

DIRECTORIES 3-6

FILE NAMES AND PATHS  
3-8

PATHNAMES 3-8

PATHS AND EXTERNAL  
COMMANDS 3-10

HOW TO DISPLAY YOUR  
CURRENT DIRECTORY 3-  
11

HOW TO CREATE A  
DIRECTORY 3-13

HOW TO CHANGE YOUR  
CURRENT DIRECTORY 3-  
13

HOW TO REMOVE A  
DIRECTORY 3-14

HOW TO CHECK THE  
VALIDITY OF YOUR FILES  
3-14

# 4. ENTERING AND USING MS-DOS COMMANDS

COMMAND SYNTAX 4-1

PARAMETERS 4-1

INTERNAL AND  
EXTERNAL COMMANDS

4-6

EXTERNAL COMMAND  
SYNTAX 4-8



INFORMATION COMMON  
TO ALL MS-DOS  
COMMANDS 4-8

BATCH PROCESSING 4-10

HOW TO CREATE AND  
EXECUTE A BATCH FILE  
4-10

THE AUTOEXEC.BAT FILE  
4-13

BATCH FILES WITH

REPLACEABLE  
PARAMETERS 4-14

INPUT AND OUTPUT 4-15

REDIRECTING YOUR  
OUTPUT 4-15

REDIRECTING YOUR  
INPUT 4-16

FILTERS 4-16

COMMAND PIPING 4-17

## 5. COMMANDS

INTRODUCTION 5-1

MS-DOS 3.20 AND  
NETWORKING 5-5

COMMANDS 5-8

ASSIGN 5-8

ATTRIB 5-10

BACKUP 5-12

BREAK 5-16

CHDIR 5-17

CHKDSK 5-19

CLS 5-23

COMMAND 5-23

COMP 5-27

COPY 5-30

CITY 5-36

DATE 5-38

DEL 5-41

DIR 5-43

DISKCOMP 5-45

DISKCOPY 5-48

ECHO 5-51

ERASE 5-52

EXE2BIN 5-52

EXIT 5-55

FC 5-56

FDISK 5-66

FIND 5-66

FOR 5-70

FORMAT 5-72

GOTO 5-76

GRAFTABL 5-78

GRAPHICS 5-79



GWBASIC 5-82

HEXDUMP 5-83

IF 5-85

JOIN 5-87

LABEL 5-92

MKDIR 5-94

MODE 5-95

MODE COM: 5-96

MODE n 5-98

MODE LPT 5-100

MORE 5-103

PATH 5-104

PAUSE 5-105

PRINT 5-107

PROMPT 5-111

RECOVER 5-114

REM 5-115

REN 5-116

REPLACE 5-118

RESTORE 5-122

RMDIR 5-126

SELECT 5-127

SET 5-  
129

SHARE

5-  
131

SHIFT

5-  
133

SORT

5-  
135

SUBST

5-  
139

SYS

5-  
143

TIME

5-  
144

TREE

5-  
146

TYPE

5-  
147

VER

5-  
149

VERIFY

5-  
149

VOL

5-  
150



XCOPY

5-  
151

VIDEO FILE EDITOR  
(EDIT)

INTRODUCTION

6-1

HOW TO INVOKE THE  
VIDEO FILE EDITOR

6-1

THE DISPLAY 6-3

THE KEYBOARD 6-4

EDIT 6-7

GENERAL EDITING  
FUNCTION KEYS 6-8

WINDOW MOVING	6-
FUNCTION KEYS	16

## Table captionEXITING AND SAVING FUNCTION KEYS

### Table caption6-18

## Table captionCOMMANDS AND SEARCHING

### Table caption6-19

STRING SEARCHES	6- 19
COMMANDS	6- 21
GOTO	6- 21
DELETE	6- 22

FILE

6-  
22

LINE EDITOR (EDLIN)

INTRODUCTION

7-  
1

HOW TO INVOKE THE

7-

EDLIN PROGRAM 2

EDLIN 7-  
2

INTER-LINE 7-  
COMMANDS 5

line (EDIT LINE) 7-  
6

A (APPEND LINES)

7-  
8

C (COPY LINES)

7-  
9

D (DELETE LINES)

7-  
12

E (END EDITING)

7-

15

I (INSERT LINES)

7-

16

L (LIST TEXT)

7-

20

M (MOVE LINES)

7-

23



P (PAGE)

7-  
25

Q (QUIT EDITING)

7-  
26

R (REPLACE TEXT)

7-  
27

7-30

S (SEARCH TEXT)

T (TRANSFER LINES) 7-34

W (WRITE LINES) 7-35

INTRA-LINE COMMANDS  
7-36

COPY1 7-37

COPYTO 7-38

COPYLINE 7-40

SKIP1 7-41

SKIPTO 7-42

KILL 7-44

INS 7-46

NEWTEMP 7-47

8. THE LINKER

INTRODUCTION 8-1

TEMPORARY FILES 8-1

CHANGING DISKETTES 8-  
2

SEGMENTS, GROUPS AND  
CLASSES 8-2

INTERACTIVE ENTRY 8-3

COMMAND LINE ENTRY  
8-10

AUTOMATIC RESPONSE  
FILE ENTRY 8-13

# 9. THE DEBUGGER

INTRODUCTION

HOW TO INVOKE THE  
DEBUG

DEBUG

DEBUGGING COMMANDS

COMMAND PARAMETERS  
A (ASSEMBLE)

C (COMPARE)

D (DUMP)

E (ENTER)

F (FILL)

G (GO)

H (HEX)

I (INPUT)

L (LOAD)

M (MOVE)

N (NAME)

O (OUTPUT)

P (PROCEED)

Q (QUIT)

R (REGISTER)

S (SEARCH)

9-1

## PROGRAM 9-1

9-1

9-4

9-4

9-7

9-10

9-11



9-14

9-17

9-18

9-21

9-23

9-24

9-27

9-29

9-31

9-32

9-33

9-34

9-37

T (TRACE) 9-39

U (UNASSEMBLE) 9-40

W (WRITE) 9-42

## A. ASCII DISPLAY AND KEYBOARD CODE TABLES

BASIC ASCII CODE (ISO 7-  
bit code) A-1

EXTENDED ASCII CODE  
FOR THE DISPLAY (ISO 8-  
bit

code) A-5

NATIONAL VARIATIONS  
IN EXTENDED ASCII  
CODE

(ISO 8-bit code) A-8

EXTENDED KEYBOARD  
A-14

B. ANSI ESCAPE  
SEQUENCES

INTRODUCTION B-1

CURSOR MOVING  
FUNCTIONS B-2

ERASING FUNCTIONS B-4

GRAPHIC MODE  
FUNCTIONS B-5

KEY REASSIGNMENT B-7

C. CONFIGURING MS-DOS

INTRODUCTION C-1

CONFIGURING  
COMMANDS C-1

BREAK C-1

BUFFERS C-2

COUNTRY C-3

DEVICE C-5

DRIVPARM C-6

FCBS C-8

FILES C-9

LASTDRIVE C-10

SHELL C-11

STACKS C-12

D. ERROR MESSAGES

DEVICE AND DISK DRIVE  
ERRORS D-1

ERROR MESSAGES IN  
ALPHABETICAL ORDER  
D-3

E. GLOSSARY OF TERMS

GLOSSARY OF TERMS E-1

F. THE VIRTUAL DISK  
SYSTEM

INTRODUCTION F-1

INSTALLING VDISK.SYS



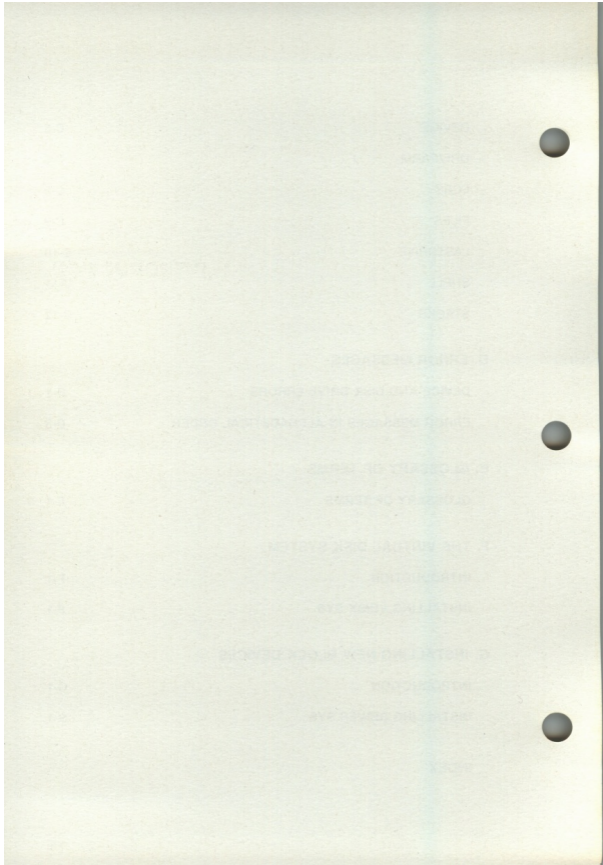
F-1

G. INSTALLING NEW  
BLOCK DEVICES

INTRODUCTION G-1

INSTALLING DRIVER.SYS  
G-1

INDEX



# ABOUT THIS CHAPTER

This chapter introduces some of the more commonly used features of MS-DOS, provides some information on disk handling and defines the notation convention used

throughout this book.

# CONTENTS

WHAT IS MS-DOS? 1- WRITE  
1 PROTEC

COMMAND LIBRARY 1- VIRTUA  
1

FILE AND DIRECTORY 1- NOTAT

HANDLING 1 CONVE

PROGRAMMING 1-

TOOLS 1

INTERNAL AND  
EXTERNAL 1-  
COMMANDS 2

BATCH 1-

PROCESSING 2

THE  
AUTOEXEC.BAT 1-  
FILE 2

REAL-TIME 1-  
CLOCK 2  
FUNCTIONS

INTERFACE 1-

HANDLING 3

DISKS 1-  
3

DISKETTE 1-  
HANDLING 5

PURCHASING 1-  
DISKETTES 6



Table caption LABELING  
DISKETTES 1-6

# WHAT IS MS-DOS?

The Microsoft Disk Operating System (MS-DOS) is a group of programs that controls the running and operation of your computer. It provides an interface between you and your computer.

Through MS-DOS you communicate with the Central Processing Unit (CPU), monitor, disk drives, printer, and other peripherals. The Disk Operating System enables you to manipulate program and data files stored on diskette or hard disk.

MS-DOS is supplied on your MS-DOS diskette.

The major features of MS-

DOS are outlined in the sections that follow.

# COMMAND LIBRARY

MS-DOS has a command library of over 40 commands that provide you with an environment suitable for handling files of information, developing programs, and running applications.

# FILE AND DIRECTORY HANDLING

File handling commands not only allow you to copy and delete files, copy entire diskettes, display the contents of files, etc., but also to group files into directories at your convenience. Moreover, MS-DOS enables you to create directories within a directory, thereby creating a

hierarchical directory structure. Refer to Chapter 3 for details.

## PROGRAMMING TOOLS

A set of programming tools which enables you to write and develop programs. You can edit program files using the Video File Editor (see Chapter 6) or Line Editor EDLIN (see Chapter 7), link object files using the LINK

utility (see Chapter 8), and debug executable files using the DEBUG utility (see Chapter 9). Macro Assembler, PASCAL, FORTRAN and other high level languages are separately available, to produce executable files. The GW-BASIC interpreter is available on your system disk for interpretive programming.



# INTERNAL AND EXTERNAL COMMANDS

When MS-DOS is initialized some commands are loaded into memory and reside there. Other commands remain on disk. The former are known as internal commands, the latter are external commands and are loaded into memory and executed when required. Most of these external

commands, after they are executed, are removed from memory, thereby optimizing the use of memory. However a few of these external commands, (GRAFTABL, GRAPHICS, PRINT, and SHARE), remain resident in memory after they have executed (see Chapter 5 for details of these commands).

# BATCH PROCESSING

MS-DOS enables a commonly executed series of commands to be grouped into one file -a batch file -that can be executed simply by entering the file name. Refer to Chapter 4 for details.

# THE AUTOEXEC.BAT FILE

The AUTOEXEC.BAT file is a special batch file which, if present, is executed automatically at system initialization. This is useful if your application requires a certain sequence of commands to be executed every time the system is initialized (see Chapter 4).

# REAL-TIME CLOCK FUNCTIONS

When no AUTOEXEC.BAT file is present MS-DOS asks you the DATE and TIME.

MS-DOS has two commands that utilize the Real-Time

Clock. These are:

- DATE which enables you to set the date
- TIME which enables you to set the time

These are important not only for programs that use time-dependent functions, but also because MS-DOS provides you with information about the time and date of creation

or modification of your files.

1-2

**MS-DOS USER GUIDE**

# INTERFACE HANDLING

MS-DOS allows you to communicate with compatible devices (plotters, printers, modems, etc.) via the standard RS-232-C serial interface. You will need to set the protocol for the interface using the MODE



command (see Chapter 5).

# DISKS

Information is stored either on 3 1/2 in. or on 5 1/4 in. floppy disks or, if your system has, on hard disk.

This manual will refer to the former as diskettes and the latter as the hard or fixed disk. The term "disk" will be used to mean either diskette or hard disk.

Drive letters (A,B,C through Z) are the means by which commands can identify a particular drive.

The drive letter of the first diskette drive in any system is A. The drive letter of any second diskette drive is B. The drive letter of the first hard disk is C. Drive letters D through Z are used for additional hard disks, disk partitions, virtual disks and

dummy drives.

The capacity, of a disk drive determines the type of diskettes that can be used in it.

Diskettes can have a variety of capacities to hold data, as illustrated in the following table. See your Installation and Operations Guide to check what types of Diskette capacities your disk drive(s)

can read and write.

## Table caption Diskette Capacities

	High Density
Double Density	80 track
40 track (48	(96 t.p.i. or

t.p.i.) 135  
t.p.i.)

8 9 8 9  
sector sector sector secto

Single 160 KB 180 KB 320  
Sided KB

Double 320 KB 360 KB 640  
Sided KB

## 3 1/2 Inch Diskette Compatibility

These diskettes are 135 tracks per inch (t.p.i.) and can be formatted single or double sided with 80 tracks. Normally each track will be formatted with nine sectors, giving single sided disks a capacity of 360 KB and double sided disks a capacity

of 720KB. A 3 1/2 inch disk drive with two heads can read and write single sided and double sided diskettes. A 3 1/2 inch disk drive with one head can only read and write single sided diskettes.

Obviously you should not place 3 1/2 inch diskettes in a 5 1/4 inch drive, nor can you place 5 1/4 inch diskettes into a 3 1/2 inch drive.

## 5 1/4 Inch Diskette



# Compatibility

Standard formatting in Normal-Capacity drives is 40 tracks, 9 sectors per track.

Formatting in High-Capacity drives is 80 tracks, 15 sectors per track. To format diskettes as 40 tracks, 9 sectors per track in HighCapacity drives use the /4 switch. However note that NormalDensity diskettes written to in High-Capacity drives cannot be

reliably read in Normal-Capacity drives. To prevent accidental writing to Normal-Density diskettes in a High-Capacity drive, write protect the diskette.

The following figure shows 5 1/4 inch diskette compatibility in different drives:

MS-DOS USER GUIDE

1-4



## DRIVE

Normal	High
Capacity	Capacity

Single	Double
--------	--------

sided	sided
-------	-------

(160/180 (320/360  
KB) KB)

D 48

1 tpi

S

K

E

Single

Read/Write

sided

T

T 96

E tpi

S

Double

—

Read/Write

sided

Double — —

sided

\* Once written the diskette cannot be reliably read in Normal Capacity Double Sided Drives.



Fig. 1-1 Diskette Type  
Compatibility in Different  
Capacity Drives

# DISKETTE HANDLING

Although diskettes are generally durable, damage to diskettes will be minimized if you take the following precautions:

- Never bend 5 1/4 inch diskettes.
- Do not touch the exposed surface of the diskette or allow liquids, dust or

cigarette ash to come into contact with it.



Never expose your diskettes to strong magnetic fields, for example keep them away from telephones and tape recorders.

- Keep your diskettes away from direct sunshine, and store them in a cool place.



1-5

- Always keep a 5 1/4 inch diskette in its cardboard envelope when not in use
- Always file them in the diskette carton.
- Keep dust out of the diskette drives by keeping the drive covers closed when not in use.

- Do not attach anything to diskettes with paper clips or rubber bands.

## PURCHASING DISKETTES

When purchasing your media make sure that the diskettes are Double Sided for Double Sided Disk Drives. High-Capacity 5 1/4 inch disk drives require 96 t.p.i. High-Density Diskettes. 3 1/2 inch disk drives require 135 t.p.i.

# Micro Floppy Diskettes.

## LABELING DISKETTES

Every carton of diskettes contains a supply of self-adhesive labels for identifying diskettes. It is good practice to write all relevant details on the label before attaching it to the diskette. But if you do find it necessary to write on the label after sticking it to a 5

1/4 inch diskette, you should avoid using sharp pencils or ball-point pens as these may damage the surface of the diskette. In this case a felt-tipped pen is recommended.

## WRITE-PROTECTION

To protect your data from being accidentally overwritten, you can apply write-protection to your diskettes.

## 5 1/4 Inch Diskette Write-Protection

For 5 1/4 inch diskettes a sheet of aluminized write-protect tabs is provided with every carton of diskettes. To apply write-protection, simply stick a tab over the write-protect notch cut into the side of the diskette. To remove write-protection, simply remove the tab. See the following figure which

shows you the position of the  
write protect notch;

1-6

## MS-DOS USER GUIDE

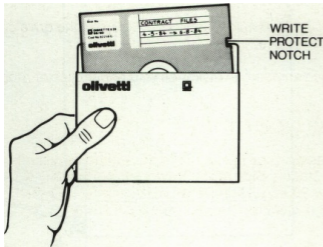


Fig. 1'2 The position of the  
Write Protect Notch on 5 1/4

# Inch Diskettes

## 3 1/2 Inch Diskette Protection

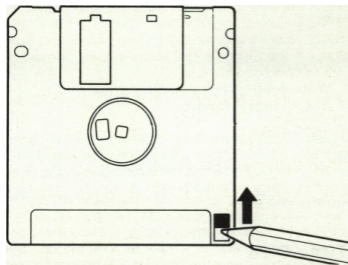
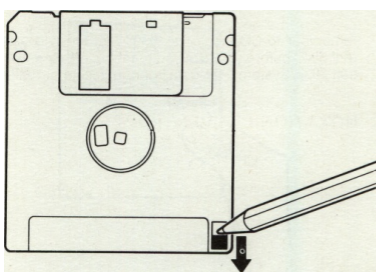


Fig. 1-3 Unprotected 3 1/2  
Inch Diskette



## Fig. 1-4 Write Protected 3 1/2 Inch Diskette

On 3 1/2 inch diskettes there is a movable tag on the reverse side in the right hand corner (see the Figure "Unprotected 3 1/2 Inch Diskette"). The first figure



shows the diskette with the tag up, this diskette is not write protected. Slide the tag down to the bottom of the slot, it clicks into place (see the Figure "Write Protected 3 1/2 inch Diskette"). Now if the computer tries to write to this diskette or to delete any files on the diskette, the result will be an error message:

Write protect error writing

drive A Abort, Retry, ignore

If you really wish to write to the diskette, remove it from the drive, slide the tag up, until it clicks into place, replace it in the drive; then press R to retry. If you had the wrong disk in the drive do not change disks at this stage, instead press A to abort the operation; then exchange the diskettes.

# VIRTUAL DISK

A virtual disk is part of main memory which emulates a backing store

1-8

MS-DOS USER GUIDE

disk. The virtual disk appears

to the operating system just like any other disk drive. For example, If a virtual disk is installed on a computer with two floppy disk drives “A:” and “B:”, the virtual disk is drive “C:”. The only difference between virtual disk and real disk is that when you turn your machine off, the information on virtual disk will be lost. So remember to COPY all files you want to keep from virtual

disk to a real disk, before you turn you machine off. See Appendix F “The Virtual Disk System” for details on installing virtual disk.

# NOTATION CONVENTIO

The following notation conventions are used throughout this book:

- Uppercase, bold letters and words within a syntax line represent keywords that must be typed exactly as shown.

Example:

In the command line:

```
DISKCOPY [sourcedrive:]  
[targetdrive:]
```

DISKCOPY should be typed as shown.

Outside syntax lines, keywords are shown in uppercase but not in bold.

Note that uppercase letters and words are used simply as visual aids in this manual. Keywords may be typed in lowercase if desired.

- Lowercase italicized characters and words represent parameter names. They indicate that variable information is to be provided by the user.

Example:



In the command line:

DISKCOPY A: B:

both sourcedrive and targetdrive have been replaced by specific values, that is A and B.

- Hyphens may join lowercase letters or words to form a single parameter name.

# Example:

In the command line:

```
R [register-name \ F]
```

register-name is a single variable item that should be replaced by a single specific value, for example AX.

- A blank, a comma, a colon, or a semicolon may be used to separate the items in a

line. In this manual the blank is usually shown in syntax lines.

- The symbols listed below are used to define the syntax of a line, but should not be typed in the actual line:

[ ] brackets

[ ] braces

| vertical stroke (“or” sign)

# ellipsis

- Items contained by brackets ([ ]) are optional and so may be selected or not.

1-10

MS-DOS USER GUIDE

# INTRODUCT

Example;

The representation:

[filespec]

indicates that a filespec may be entered or omitted.

- Items enclosed by braces ([ j) and separated by vertical

strokes (|) are alternatives.  
You should select only one  
such alternative.

Example:

The representation:

[A|B|C]

indicates that either A or B or  
C should be selected.

- Items enclosed by brackets

( [ ] ) and separated by vertical strokes ( | ) are optional alternatives. You may choose one such alternative, or none at all.

Example:

The representation:

[A|B|C]

indicates that A or B or C may, but need not, be

selected.

- An ellipsis indicates that the preceding item or group of items may be repeated more than once in succession.

Example:

The representation:

A [B]...



indicates that A can be typed alone or can be followed by one or more occurrences of:

**B**

- Letters and words in bold indicate MS-DOS messages that appear on your Personal Computer screen. For example:

Insert new diskette for drive **B:** and strike **ENTER** when

ready

- Letters and words shown in condensed bold indicate that you must press a specific key. For example the key whose inscription is CTRL is always referred to as CTRL.
- Commands need to be confirmed by pressing ^ (the ENTER key), at the end of the command line.

1-12

# MS-DOS USER GUIDE

# ABOUT THIS CHAPTER

This chapter describes the keystroke combinations that are required to execute control functions such as Enter, carriage return, line feed, break, etc. It also

indicates the keystrokes that perform special editing functions. For further details of the latter refer to Chapter 7.

# CONTENTS

INTRODUCTION 2-1

MS-DOS CONTROL KEYS  
2-1

EDITING FUNCTION KEYS  
2-4

# INTRODUCT

Other than enabling you to enter text in the manner of a typewriter, your keyboard is capable, under MS-DOS, of allowing you to invoke other functions by using certain combinations of key-strokes. Such features fall into two groups: control keys and editing function keys.





# MS-DOS CONTROL KEYS

MS-DOS control keys utilize a variety of key-stroke combinations. They are used to correct typing errors, abort command execution, etc. These functions are described

in the following table. Note that if you are using an Olivetti extended function key Keyboard, the key combinations will have minor differences from those stated here (see your Installation and Operations Guide).

FUNCTION	KEY COMBINATION
----------	--------------------

ENTER

j

or

CTRL M

Shift

1 (SHIFT)

Shift lock      CAPS LOCK

FUNCTION KEY  
COMBINATION

Keypad  
lock

NUM LOCK





Backspace or

CTRLH

CTRL 1

TAB

or

—1

CTRL ENTER

Line Feed or

CTRL J

Escape

ESC

2-2

## MS-DOS USER GUIDE

FUNCTION KEY  
COMBINATION

CTRL BREAK

Abort

or

CTRLC

CTRL NUM  
LOCK

Suspend

or

CTRL S

Echo

Output

CTRL PRT SC

or

CTRL P

Print

SHIFT PRT SC

Screen



System

CTRL ALT DEL

Reset

# EDITING FUNCTION KEYS

These commands are entered by pressing a single key.

Most of these commands use the function (F) keys.

When you press ENTER to

execute a command, a copy of the command is kept in a special buffer called the source line. MS-DOS has a range of commands that you can use to enter a command line by copying all or part of the source line. For example if you wish to copy several files from drive B to drive A you might copy the first of those files by typing:

**COPY B:MYFILE A:** then

press ENTER

When you press ENTER this command would be written to the source line. If the next file you want to copy is called YOURFILE, then the character strings "COPY B:" and "FILE A:" can be copied from the source line using the function keys. The only part you need type in yourself is the string "YOUR".

The most common use of editing function keys, however, is in performing edit operations within a line of text when using the line editor EDLIN. A detailed description is therefore given in Chapter 7, but a brief description of each of the editing function keys is given in the following table:

FUNCTION	KEY
----------	-----

COMBINATION

BACKUP

COPY1

F1 or

COPYTO

F2 then type a  
character

# MS-DOS USER GUIDE

2-4



# **MS-DOS CONTROL KEYS AND EDITING FUNCTION KEYS**

FUNCTION KEY

COMBINATION

COPYLINE F3

SKIP1

DEL

SKIPTO

F4 then type a  
character

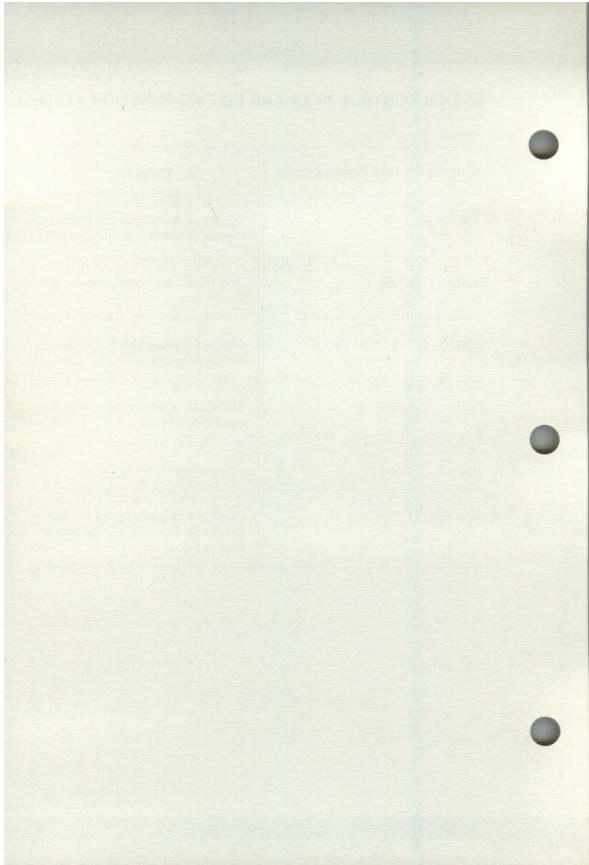
KILL

ESC

INSERT

INS

NEWTEMP F5



# ABOUT THIS CHAPTER

This chapter describes how to manipulate files and directories, how you can access files in directories by means of paths and how to create and delete directories.

For further details of commands mentioned in this chapter refer to Chapter 5.

## CONTENTS

HOW MS-DOS KEEPS TRACK OF YOUR FILES	3- 1
--	---------



HOW TO NAME YOUR FILES 3-1

WILD CARDS 3-2

THE ? WILD CARD 3-2

THE \* WILD CARD 3-3

RESERVED DEVICE NAMES 3-4

HOW TO COPY YOUR FILES 3-4

HOW TO PROTECT YOUR FILES 3-5

HOW TO EXAMINE 3-  
6

THE CONTENTS OF A  
FILE

DIRECTORIES 3-  
6

FILE NAMES AND 3-  
PATHS 8

PATHS AND

EXTERNAL COMMANDS	3- 10
HOW TO DISPLAY YOUR CURRENT DIRECTORY	3- 11
HOW TO CREATE A DIRECTORY	3- 13

HOW TO CHANGE  
YOUR CURRENT  
DIRECTORY 3-  
13

HOW TO REMOVE A  
DIRECTORY 3-  
14

HOW TO CHECK THE  
VALIDITY OF YOUR  
FILES 3-  
14

# HOW MS- DOS KEEPS TRACK OF YOUR FILES

The names of files are kept in directories on disk. These

directories also contain information on the size of the files, their location on the disk, and the dates that they were created or updated. The directory you are working in is called your current directory.

An additional system area is called the File Allocation Table. It keeps track of the location of your files on the disk. It also allocates the free

space on your disks so that you can create new files.

These two system areas, the directories and the File Allocation Table, enable MS-DOS to recognize and organize the files on your disks. The File Allocation Table is created on a disk when you format it with the MS-DOS FORMAT command. One empty directory is also created on it,



known as the root directory.

# HOW TO NAME YOUR FILES

A file name can comprise;

- One to eight characters (for legal characters see below).

For example: NEWFILE.

- One to eight characters, followed by a period (.) and a one to three character file name extension. For example NEWFILE.EXE.

A file name may be made up of any of the following characters;

A-Z 0-9 \$ & #

%'()-\_

@ ' [ j ! a-z

Alphabetic characters within the file name can be entered in upper or lower case, but MS-DOS will translate lower case letters into upper case.

How you specify a file depends on which directory of which disk drive it resides;

- If the file is in the current directory on the default drive you need only specify the file name, for example:

MYFILE

- If you are using tree-structured directories (see the section entitled "Directories" later in this chapter) and the file is on the default drive but not in the current directory, then you

need to specify a path; for example:

```
\MIDIR\MYFILE
```

Path names are described later in this chapter in the section entitled "File Names and Paths".

- If the file is not on the default drive, you will need to specify the drive; for example;

**B:MYFILE**

# WILD CARDS

Two special characters (called wild cards) can be used in file names and extensions: the asterisk (\*) and the question mark (?). These special characters give you greater flexibility when using file names in MS-DOS



commands.

## THE ? WILD CARD

A question mark (?) in a file name or file name extension indicates that any character can occupy that position. A question mark(?) as the final character in a file name or file name extension indicates a character or no character can occupy that position. For example, the MS-DOS

command:

DIR TEST7RUN.EX?

will list all entries in the current directory on the default drive that begin with TEST, have any next character, end with the letters RUN, and

3-2

MS-DOS USER GUIDE

have a file name extension of EX followed by any character or no character. Here are some examples of files that might be listed by the above DIR command:

TESTORUN.EX

TEST1RUN.EXE

TEST2RUN.EXE

TEST6RUN.EXE

## THE \* WILD CARD

An asterisk (\*) in a file name indicates that any valid character or sequence of valid characters can occupy that position in the file name. Any characters after the \* are ignored by the system. For example:

```
DIR TEST*.EXE
```

will list all entries in the

current directory on the default drive with file names that begin with the characters TEST and have an extension of EXE. Here are some examples of files that might be listed by the above DIR command:

TEST1RUN.EXE

TEST2RUN.EXE

TEST6RUN.EXE

# TESTALL.EXE

The wild card designation \*.  
\* refers to all files in the current directory. Note that this can be very powerful when used in MS-DOS commands. For example, the command:

```
COPY A:*. * C:
```

copies all files on the current directory of the A: drive,

regardless of file name, to the current directory of the C: drive.

## Examples

To list all files named NEWFILE with any extension in the current directory on drive A, simply enter:

```
DIR A:NEWFILE.
```

To list all file names that have less than or equal to six-characters plus an extension of .TXT in the current directory of the diskette in B, enter:

```
DIR B:??????.TXT
```



# RESERVED DEVICE NAMES

MS-DOS treats device names specially, and certain words are reserved for the names of these devices. These names cannot be used as file names or extensions. Reserved

names are as follows:

**AUX:** or **COM1:** Used when referring to input from or output to the built-in communications port.

**COM2:** Used when referring to input from or output to a se

cond communications port.

**CON:** Used when referring to

keyboard input or screen output.

PRN: or LPT1: Used when referring to the first parallel printer.

LPT2: or LPT3: Used when referring to the second and third parallel printers.

NUL: Used when you do not want to create a particular

file,

but the command requires an input or output file name.

The colon following the reserved device name is optional.

You can use a device name instead of a file name. Using "Redirection" (see Chapter 4) data can then be input from a device or output to a

device, instead of a file.

# HOW TO COPY YOUR FILES

You often need more than one copy of a file. The COPY command allows you to copy one or more files to another disk, to another file on the same disk or to another

directory on the same disk (see the section entitled "File Names and Paths").

You can also give the copy a different name if you specify the new name in the COPY command.

3-4

## MS-DOS USER GUIDE

You cannot make a copy of a file in the same directory

unless you specify a different file name for the new copy.

For example,

```
COPY A:MYFILE.TXT  
B:MYFILE.TXT
```

will copy the file  
MYFILE.TXT on the diskette  
in drive A to a file named  
MYFILE.TXT on the diskette  
in drive B.



If you want to duplicate the file named MYFILE.TXT on the same disk, enter:

```
COPY A-.MYFILE.TXT  
A:NEWNAME.TXT
```

You now have two copies of your file on drive A one named MYFILE.TXT and the other named NEWNAME.TXT.

You can also use the COPY

command along with the wild card feature, to copy groups of files from one disk to another, or to copy all files from one disk to another.

For example, the command

```
COPY A:*.EXE C:
```

copies all the .EXE files from the current directory of the A: drive, to the current directory of the C: drive.



# HOW TO PROTECT YOUR FILES

MS-DOS is a powerful and useful environment for processing your personal and business information. As

with any information system, inadvertent errors may occur and information may be misused. If you are processing information that cannot be replaced or requires a high level of security, you should take steps to ensure that your data and programs are protected from accidental or unauthorized use, modification, or destruction. Simple measures you can

take, such as removing your disks when they are not in use, keeping back-up copies of valuable information, write-protection, and installing your equipment in a safe place, can help you maintain the integrity of the information in your files.

# HOW TO EXAMINE THE CONTENTS OF A FILE

There are two commands that enable you to examine a file.

These are:

- The TYPE command that displays the contents of a text file.
- The HEXDUMP file that displays a file in hexadecimal.

If you want to print a file you can do this by means of the ">" I/O redirection symbol.

For example enter:



TYPE myfile > PRN

will send the contents of myfile to the printer instead of the screen. For details about I/O redirection refer to Chapter 4.

# DIRECTORIES

The names of your files are kept in directories on each disk. Each directory also contains information on the size of the files and the dates that they were created or last updated.

If your computer is used by several people, or if you are working on several different

projects, the number of files in the directory can become large and unwieldy.

Moreover, this will certainly be the case if you are using a hard disk as it is capable of storing a vast number of files. You may want your own files kept separate from those of a colleague: or, you may want to organize your programs into categories that are convenient to you.

MS-DOS allows you to organize the files on your disks into directories.

Directories are a way of dividing your files into convenient groups. For example, you may want all your accounting programs in one directory and text files in another. Any directory can contain another directory, this is a sub-directory. This method of organizing your files is called a hierarchical

directory structure.

A hierarchical directory structure can be thought of as a "tree" structure:

directories are branches of the tree and files are the leaves, except that the "tree" grows downward; that is, the "root" is at the top. The root is the first level in the directory structure. It is the directory

## MS-DOS USER GUIDE

that is automatically created when you format a disk and start putting files in it. Any directory can contain files as well as sub-directories.

The tree structure grows as you create new directories for groups of files. Within each new directory, files can

be added, or new subdirectories can be created.

It is possible for you to "travel" around this tree; for instance, you can find any file in the system by starting at the root and traveling down any of the branches to the desired file. Conversely, you can start where you are within the file system and travel towards the root.

The following figure illustrates a typical hierarchical directory structure:

(root)



GAMES BIN USERS  
ACCOUNTS PROGRAMS

JOE

DENISE MARY



TEXT.TXT FORMS  
TEXT.TXT

## Fig. 3-1 A Sample Hierarchical Directory Structure

The root directory is the first level in the directory structure. You can create subdirectories from the root by using the MKDIR command. The root directory may also contain files.

In this example, five subdirectories of root have been created. These include:

- A directory of games, named GAMES.
- A directory of all external commands, named BIN.
- A USERS directory containing a separate subdirectory for each user of the system.

- A directory containing accounting information, named ACCOUNTS.
- A directory of programs, named PROGRAMS.

Joe, Denise and Mary each have their own directories which are subdirectories of the USERS directory. Denise has a subdirectory under the \USERS\DENISE directory named FORMS. Denise and

Mary have files in their directories, each named TEXT.TXT. Notice that Mary's text file is unrelated to Denise's.

This organization of files and directories is not important if you only work with files in your own directory; but if you work with someone else or on several projects at one time, the hierarchical directory structure becomes

extremely useful. For example, you could get a list of the files in Denise's FORMS directory by entering:

DIR

\USERS\DENISE\FORMS

Note that the back-slash (\) is used to separate directories from other directories and files. The first back-slash in a directory sequence represents

the root directory.

To find out what files Mary has in her directory, you could enter: DIR  
\USERS\MARY

# FILE NAMES AND PATHS

When you use hierarchical directories, you must tell MS-DOS where the files are located in the directory structure. Both Mary and Denise, for example, have

files named TEXT.TXT. Each will have to tell MSDOS in which directory her file resides if she wants to access it. This is done by giving MS-DOS a pathname to the file.

## PATHNAMES

A pathname is a sequence of one or more directory names followed, optionally, by a file name, each separated from



the previous one by a backslash (\).

MS-DOS USER guide ;

3-8

The syntax of a pathname is:

[drive:] [ \ ] [ [directory] [ \  
directory] ... \ ] filename

or

[drive:]] \ ][d/recfory][ \  
directory]...

If a pathname begins with a  
backslash, MS-DOS searches  
for the file beginning at the  
root (or top) of the tree.

Otherwise, MS-DOS begins  
at the user's current directory  
and searches downward from  
there. For example, the path  
of Denise's TEXT.TXT file  
is

\USERS\DENISE\TEXT.TXT

That is, if you are in directory JOE and you want to access the file named TEXT.TXT in directory DENISE you would use the pathname:

`\USERS\DENISE\TEXT.TXT`

If you were in directory USERS, however, you could access the same file using the pathname:

DENISE\TEXT.TXT

However, if you are already in directory DENISE you can simply use the file name:

TEXT.TXT

If you are in directory MARY and you wish to make a copy of Mary's file TEXT.TXT in Denise's directory, but rename the copy COPY.TXT, you would

enter:

```
COPY TEXT.TXT
```

```
\USERS\DENISE\COPY.TXT
```

MS-DOS provides special shorthand notations for the current directory and the parent directory (one level up) of the current directory:

MS-DOS uses this shorthand notation to indicate the name of the current directory in all

hierarchical directory listings. MS-DOS automatically creates this entry when a directory is made

The shorthand name of the current directory's parent directory. If you enter:

```
DIR ..
```

then MS-DOS will list the files in the parent directory

of your current directory.

If you enter:

```
DIR ..\..
```

then MS-DOS will list the files in the parent's parent directory.

(Note that the maximum length of a pathname is 63 characters.)

# PATHS AND EXTERNAL COMMANDS

External commands reside on disk as program files. They must be read from the disk before they execute.

When you are working with more than one directory, it is convenient to put all MS-DOS external commands into a separate directory so they do not clutter your other



directories. When you issue an external command to MS-DOS, MS-DOS immediately checks your current directory to find that command. You must tell MS-DOS in which directory these external commands reside. This is done with the PATH command.

For example, if your current directory is `\BIN\PROG`, and all MS-DOS external

commands are in  
\BIN\COMMANDS, you  
must tell MS-DOS to choose  
the \BIN\COMMANDS path  
to find an external command.  
Enter the command:

PATH \BIN\COMMANDS

3-10

MS-DOS USER GUIDE

This tells MS-DOS to search

first your current directory and then the `\BIN\COMMANDS` directory for all commands. You only have to specify this path once to MS-DOS during your working session. The `PATH` command can be in `AUTOEXEC.BAT`. If you want to know what the current path is, enter the `PATH` command without a parameter and the current value of `PATH` will be

displayed.

You may also specify several paths. Refer to the `PATH` command in Chapter 5 for more information.

# **HOW TO DISPLAY YOUR CURRENT**

# DIRECTORY

You can find out the name of the directory you are in by issuing the MS-DOS command CD or CHDIR (Change Directory) with no parameter. For example, if your current directory is \USERS\DENISE, when you enter:

CD

you will see:

**C:\USERS\DENI**

This is the complete path of your current directory and comprises your current drive designation plus the current directory (\USERS\DENISE). If you now want to see what is in the \USERS\DENISE directory, you can issue the MS-DOS command DIR. The

following is an example of the display you might receive for the DIR command for a subdirectory:

Table captionDirectory of  
C:\USERS\DENISE

```
8-
<DIR> 09- 10:09a
82
```

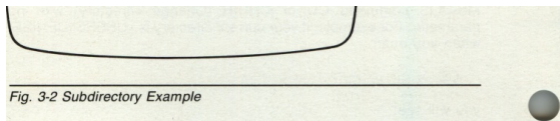
8-  
<DIR> 09- 10:09a  
82

8-  
FORMS <DIR> 09- 10:09a  
82

8-  
TEXT TXT 04- 9:30a  
5243 82



4 File(s) 8376320 bytes free



Note that MS-DOS lists files and directories in this output. As you can see, DENISE has another directory in this tree structure named FORMS. The indicates the current directory `\USERS\DENISE`, and the is the shorthand notation for the parent

directory USERS.

TEXT.TXT is a file in the  
\USERS\DENISE directory.

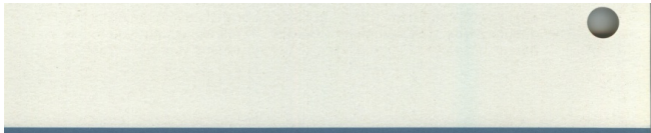
All of these directories and  
files reside on the disk in  
drive C.

Because files and directories  
are listed together (see  
previous display), MS-DOS  
does not allow you to give a  
subdirectory the same name  
as a file in that directory. For  
example, consider again the

path

USERS\DENISE\FORMS.

FORMS is a subdirectory of DENISE, therefore you cannot create a file named FORMS in directory DENISE.



3-12

MS-DOS USER GUIDE

# HOW TO CREATE A DIRECTORY

To create a subdirectory in your current directory, use the MD or MKDIR (Make Directory) command. For example, if you are in directory DENISE and you want to create a new

directory named NEWDIR within your current directory, simply type:

```
MD NEWDIR then press  
ENTER
```

and a new directory will exist in your tree structure under your current directory.

You can also create directories anywhere in the tree structure by specifying

MD and then a path. For example, if your current directory is DENISE and you want to create a directory named SPECS in directory MARY you would type:

```
MD \USERS\MARY\SPECS  
then press ENTER
```

MS-DOS will automatically create the . and .. entries in the new directory.

To create text files in the new directory, use either the Video File Editor (see Chapter 6) or the MS-DOS line editor EDLIN (see Chapter 7).

# **HOW TO CHANGE YOUR CURRENT**

# DIRECTORY

To change your current directory to another directory simply issue the CD or CHDIR (Change Directory) command and supply a path. For example type:

```
CD \USERS then press  
ENTER
```

to change the current



directory to \USERS. You can specify any path after the command to "travel" to different branches and leaves of the directory tree. The command "CHDIR .." will always put you in the parent directory of your current directory (unless you are in root).

## **HOW TO**

# REMOVE A DIRECTORY

To remove a directory from the tree structure, use the MS-DOS RD or RMDIR (Remove Directory) command. For example, to remove the directory NEWDIR from the current directory, type:

RD NEWDIR then press  
ENTER

Note that the directory  
NEWDIR must be empty  
except for the . and .. entries  
before it can be removed;  
this will prevent you from  
accidentally deleting files  
and directories. You can  
remove any directory by  
specifying its path. To  
remove the \USERS\JOE  
directory, make sure that it

has only the . and .. entries,  
then type:

```
RD \USERS\JOE then press  
ENTER
```

If the directory is not empty  
an error is reported. To  
remove all the files in a  
directory (except for the . and  
.. entries), enter DEL and  
then the path of the directory.  
For example, to delete all  
files in the \USERS\DENISE

directory, type:

`DEL \USERS\DENISE` then  
press `ENTER`

and MS-DOS will display:

**Are you sure?**  
**(Y/N)**

Press `Y` and the files will be  
deleted. You cannot delete

the . and .. entries (except by removing the directory).

# HOW TO CHECK THE VALIDITY OF YOUR FILES

The CHKDSK command is used to check your disks for

consistency and errors. The CHKDSK command analyzes the directories and the File Allocation Table on the disk that you specify. It then produces a status report of any inconsistencies, such as files which have a nonzero size in their directory but really have no data in them.

MS-DOS USER GUIDE



# FILES AND DIRECTORIES

To check the disk in drive A type:

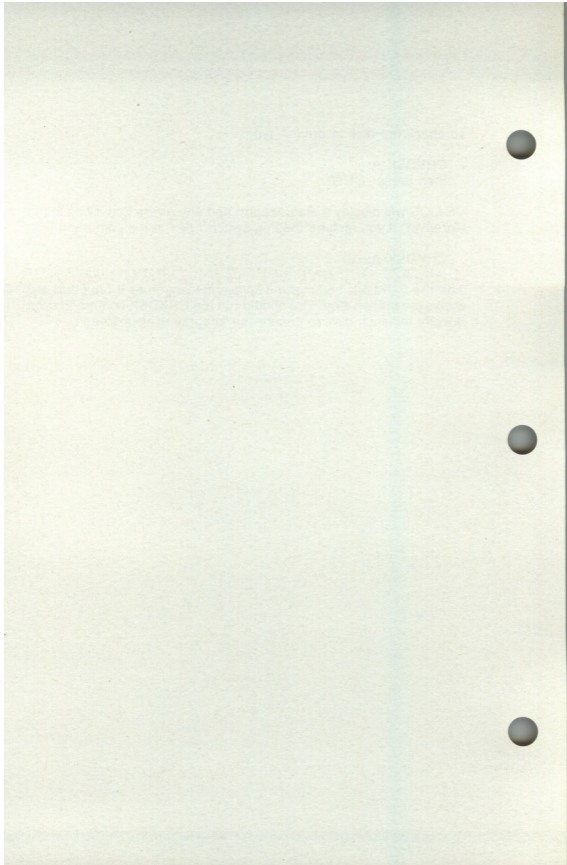
CHKDSK A: then press  
ENTER

MS-DOS will display a status report and any errors that it has found. Moreover, if you

include the flag switch /V in the command line:

```
CHKDSK A: /V
```

then the CHKDSK command displays messages as it runs and also shows the hidden files. You should run the CHKDSK command occasionally for each disk to ensure the integrity of your files.



# ABOUT THIS CHAPTER

This chapter defines the syntax for a command, explains how commands can be grouped into batch files, and how the output from a command can be redirected

to some device other than the VDU. It also describes the concept of "piping", whereby the output from one command becomes the input to another.

For further details of commands mentioned in this chapter refer to Chapter 5.

## CONTENTS

COMMAND SYNTAX 4-  
1

PARAMETERS 4-  
1

INTERNAL AND  
EXTERNAL  
COMMANDS 4-  
6

EXTERNAL 4-

COMMAND SYNTAX 8

INFORMATION  
COMMON TO ALL MS-  
DOS COMMANDS 4-  
8

BATCH PROCESSING 4-  
10

HOW TO CREATE 4-

AND EXECUTE A 10  
BATCH FILE

THE AUTOEXEC.BAT 4-  
FILE 13

BATCH FILES WITH  
REPLACEABLE 4-  
PARAMETERS 14



INPUT AND OUTPUT

4-  
15

REDIRECTING YOUR

OUTPUT

4-15

REDIRECTING YOUR

INPUT

4-16

FILTERS 4-16

COMMAND PIPING 4-17

# COMMAND SYNTAX

The general command format is defined as follows:

KEYWORD [parameter] ...

Table captionWhere

SYNTAX

# ELEMENT MEANING

A one to eight character mnemonic that specifies the **KEYWORD** command to be executed. It must exclude any file name extension.

parameter

A parameter to the command defining the command action. The number of parameters depends on the command executed.

Remarks

If the **KEYWORD** is the name of an executable file, it may not reside on the default drive and directory. In this case the general command format can be extended (see the section on "External Command Syntax").

# **PARAMETERS**

Parameters are user-selected strings of alphabetic

characters and of integers which can be optional. They are recognized by their position in the command line. The types of parameter are described in the following table:

PARAMETER TYPE	MEANING
-------------------	---------

Either a one to

character string  
one to eight characters  
string followed by  
period (.) and  
three character  
extension. It must  
be made up from  
the following characters

filename

A-Z 0-9 \$ & #

or

0/0 ; ( ) - \_

directory

@ ( ] ! a-z



Note: Lower-case letters are converted into upper case

For example:

NEWFILE

NEWFILE.TXT

[drive\]filename

A file specific

filespec

(filespec) can  
name with or w  
drive specifier  
For example:

NEWFILE.TXT

B:NEWFILE.TXT

[drive:][\][[dir  
[\directory]...\  
or

pathname [drive:][\][dire  
[\directory ]...



4-2

## MS-DOS USER GUIDE

PARAMETER	MEANING
TYPE	

a pathname may

- a drive specification specifying the specified drive

C:

- one or more directories separated by backslashes \ directory path of the directory of the example:

MARY\PROG

- a drive specification, followed by a backslash, and a path starting from the root directory of the drive. For example:

C:MARY\PROG

- a backslash, followed by a path starting from the root directory of the drive. For example:

- a drive specification and a backslash directory of the example:

C:\

- a backslash for directory name backslashes specify path starting from the default drive

\USER\MARY

## MEANING

- a drive specification consisting of a drive letter, a backslash and a directory name separated by backslashes, specifying a directory starting from the root directory of the drive. For example;

## PARAMETER TYPE

C;\USER\MAIL

- any of the above

by a file name

- a file name

drive

A single letter drive or hard c

switch

An option whi  
command execu  
form of a sing  
slash. For exam



/P

Provides more  
DOS commands

argument

ON or OFF



4-4

MS-DOS USER GUIDE

# PARAMETER TYPE

## MEANING

volume label

One to eleven character string. Spaces may be included but not tabs. A volume label must be made up from the follow/ing characters:

A-Z 0-9 \$ & #

0/0 ' ( )

@ [ ) ! a-z

Note; Lower case letters are converted to uppercase.

For example:

Hard Disk -> HARD DISK  
SYSTEM DISK

# **Nil Parameters**

Some parameters are optional and take default values if they are not specified in the command line. For example, if you enter:

```
MODE COM1:11
```

then the last three parameters of the MODE COM command, which takes four parameters in all, take default values.

If default parameters are required in the middle of a command line commas must be entered separating omitted parameter. For example, if you enter:

```
MODE COM1:11,,1
```

then the omitted second and third parameters take default values.

# INTERNAL AND EXTERNAL COMMANDS

There are two types of MS-DOS command;

- Internal commands

- External commands

Internal commands are the simplest, most commonly used commands. You cannot see these commands when you do a directory listing on your MS-DOS disk; they are part of the command processor. Therefore they reside in memory whenever MS-DOS is loaded. When you enter these commands, they execute immediately.

This class of command  
comprises:

BREAK

CHDIR (CD)

CLS

COPY

CITY

DATE



DEL (ERASE)

DIR

ECHO

EXIT

FOR

GOTO

IF

MKDIR (MD)

PATH

PAUSE

PROMPT

REM

REN (RENAME) RMDIR  
(RD)

SET

SHIFT

TIME

TYPE

VER

VERIFY

VOL

External commands reside on disk as program files. Any

file name with a file extension of .COM, .EXE, or .BAT is considered to be an external command. They must be read from disk and loaded into memory before they can execute. With most of the external commands, following execution they are removed from memory. However a few of these external commands (GRAFTABL, GRAPHICS, PRINT and SHARE) remain

resident in memory after they have executed.

4-6

## MS-DOS USER GUIDE

When you enter an external command, do not include its file name extension. External commands include:



## ASSIGN

ATTRIB

BACKUP

CHKDSK

FC

FDISK

FIND

FORMAT

GRAFTABL

GRAPHICS

GWBASIC

HEXDUMP

JOIN

LABEL

LINK

MODE

MORE

PRINT

RECOVER

REPLACE

RESTORE

SELECT

SHARE

SORT



SUBST

SYS

TREE

XCOPY

COMMAND

COMP

DEBUG

DISKCOMP

DISKCOPY

EDIT

EDLIN

EXE2BIN

So for example invoking  
FORMAT executes the  
command file  
FORMAT.COM and

invoking ATTRIB executes the executable file ATTRIB.EXE. .EXE files have to be located in memory when they are loaded. Some .EXE files can be converted to .COM files using the MSDOS utility EXE2BIN. .COM files are in memory image format and always load starting at location 100H in a memory segment, therefore .COM format is more compact and loads

faster. Because all external commands reside on disk, you can create commands and add them to the system by writing programs in assembler or high level languages and compiling them. MS Compilers and the assembler MASM produce object code (.OBJ) files. These .OBJ files have to be linked, using the linker LINK (see Chapter 8). The LINK produces .EXE (executable)

files. If the .EXE cannot be converted to .COM files the following error message appears:

# **File cannot be converted**

Refer to chapter 5 for more details on EXE2BIN.

When you specify an external

command simply as **KEYWORD**, MSDOS first looks in the default directory of the default drive. It then searches the paths set in the **PATH** variable of the environment. If the **KEYWORD** command file is not found, it cannot be executed and the following error message appears:



# Bad command or filename

For external commands the general command format can be extended by preceding the **KEYWORD** with the Drive where the command file resides and/or the path leading to its directory.

# EXTERNAL COMMAND SYNTAX

The general format of external commands is therefore defined as follows;

```
[dnVe:][paf/7]KEYWORD[  
parameter ]...
```



# PARAMETER MEANING TYPE

drive:

A one character drive specifier followed by a colon, specifying the drive where the **KEYWORD** is to be found.

[[\]directory[\  
rectory ]... \j

path

If the path  
consists of the  
root directory,  
only one  
backslash  
should be used  
for example:

C:\

# **INFORMATION COMMON TO ALL MS- DOS COMMANDS**

The following information  
applies to all MS-DOS

commands;

- Commands are usually followed by one or more parameters.
- Commands and parameters may be entered in upper case or lower case, or a combination of both. MS-DOS will convert all lower case letters to upper case.

# MS-DOS USER GUIDE

- Commands and parameters must be separated by delimiters. A space is usually used; for example:

```
COPY A:MYFILE  
B:YOURFILE
```

You can also use the comma (,), semicolon (;) or the equal sign (=) as delimiters in MS-DOS commands.

For clarity, this manual will use a space as the delimiter

- When you are instructed to "Press any key", you can press any alphabetic (A-Z) or numeric (1 -9) key.
- You must include the file name extension when referring to a file that already has one.
- You can abort commands

that perform input/output by pressing CTRL C or CTRL BREAK.

- Commands take effect only after you have pressed ENTER.

- Wild cards (global file name characters) and device names (for example, PRN or CON) are not allowed in the names of any commands.

- When commands produce a large amount of output on the screen, the display will automatically scroll to the next screen. You can press CTRL S or CTRL NUMLOCK to suspend the display. Press any key to resume the display on the screen.
- MS-DOS control keys and editing function keys can be used when entering



commands. Refer to Chapter 2 for a description of these keys.

- The default prompt from the command processor is the default drive designation plus a greater-than sign; for example, A>. You can change this prompt using the PROMPT command (see Chapter 5 for details).

- Disk drives will be referred

to as source drive and target drive. A source drive is the drive you will be transferring information from. A target drive is the drive you will be transferring information to.

# BATCH PROCESSING

Often you may find yourself entering the same sequence of commands over and over to perform some common task. With MS-DOS, you can put the command sequence into a special file called a batch file, and execute the

entire sequence simply by entering the name of the batch file. "Batches" of your commands in such files are processed as if they were entered at the keyboard. Each batch file must be named with the .BAT extension, and is executed by entering the file name without its extension.

## **HOW TO**

# CREATE AND EXECUTE A BATCH FILE

You can create a batch file by using the Video File Editor, the Line Editor (EDLIN) or the COPY command.

The MS-DOS command library contains a sub-set of batch processing commands.

Among the more commonly used are REM and PAUSE. REM permits you to include remarks and comments in your batch files without these remarks being executed as commands. PAUSE prompts you with an optional message and permits you to either continue or abort the batch process at a given point.

Batch processing is useful if you want to execute several

MS-DOS commands with one batch command, such as when you format and check a new diskette. For example, a batch file for this purpose might look like this:

```
REM This is a file to check  
new diskettes REM It is  
named NEWDiSK.BAT  
PAUSE Insert new diskette in  
drive B FORMAT B:
```

# CHKDSK B:

Place an unprotected working copy of your system diskette in drive A.

Make sure that you have the prompt A> and proceed as follows to create the example batch file:



# MS-DOS USER GUIDE

STEP IF you enter... THEN

1

COPY CON:

NEWDISK.BAT

the co:

awaits

from t

keybo:

(CON:

”REM

2 REM This is a file to  
file to check new di  
new diskettes is ente  
first li

3 REM It is  
named  
NEWDISK.BAT

”REM  
named  
NEWI  
.BAT”  
entere  
secon

4

PAUSE Insert  
new diskette in  
drive B

”PAU  
new di  
drive l  
entere  
third l

• FORMAT B:

”FOR  
is ente  
fourth

”CHK

6           CHKDSK B:           is ente  
fifth li

7           CTRLZ           type C  
the en  
charac  
entere  
sixth l

press ]  
the fil  
is com

8

ENTER

•

the me

1 File()

appear

screen

The fi

NEWI

is crea

system

li

---

To execute this batch file, simply enter the file name without the extension:


NEWDISK

The result is the same as if each of the lines in the batch file were entered at the terminal as individual commands. That is, the first

three commands are executed successively and the following messages are displayed on the screen:

```
r A> NEWDISK
```

```
A>REM This is a file to  
check new diskettes A>REM  
It is named NEWDISK.BAT  
A> PAUSE Insert new  
diskette in drive B Strike a  
key when ready
```



## Fig. 4-1 Sample Batch File Display

After striking a key, the diskette in drive B is formatted. You will then be asked if you wish to format another diskette. Following a negative reply, the diskette will be checked.



# Remarks

1. Only the file name should be entered to execute the batch file. Do not enter the file name extension.

2. Do not name batch files with internal command names.

3. If you name batch files with external command

names, will be executed in preference to the .BAT file, .EXE or .COM file.

## 4-12 MS-DOS USER GUIDE



4. If you press CTRL C or CTRL BREAK while in batch mode, this prompt appears:

**terminate batch**

# job (Y/N)?

If you press Y, the remainder of the commands in the batch file are ignored and the system prompt appears.

If you press N, only the current command is terminated and batch processing continues with the next command in the file.

5. If you remove the diskette containing a batch file being executed, MS-DOS prompts you to insert it again so the next command can be read.

6. The last command in a batch file may be the name of another batch file. This allows you to call one batch file from another, when the first is finished. However there is no return to the calling batch file.

7. Input and Output can be redirected (the ” < ”, ”) sym

bols. See later in this chapter for more information.

# **THE AUTOEXEC.BA FILE**

An AUTOEXEC.BAT file is

a batch file that allows you to automatically execute programs when you start MS-DOS. Automatic Program Execution is useful when you want to run a specific package (for example, Multiplan) under MS-DOS, and when you want MS-DOS to execute a batch program automatically each time you start the system.

When you start MS-DOS, the

command processor searches the MSDOS system diskette for a file named AUTOEXEC.BAT. The AUTOEXEC.BAT file is a batch file that is automatically executed each time you start the system.

The AUTOEXEC.BAT file is created in exactly the same way as any other batch file. It must, however, reside in the root directory of the MS-

DOS system disk.

# Example

If your AUTOEXEC.BAT file contains the following:

**DATE**

**TIME**



# **GWBASIC**

then on initializing your system the date and time prompts will appear and the system will automatically enter GWBASIC.

**BATCH FILES  
WITH  
REPLACEABLE**

# PARAMETERS

You may want commands within a batch file to have replaceable parameters. For example, if your batch file contains a COPY command, you may wish to supply a different parameter to the COPY command each time you run the batch file. You can do this by specifying dummy parameters to the

commands within the batch file. These parameters, named %0 to %9, can be replaced by values supplied when the batch file executes. For example, you may have created the following batch file named "MYFILE.BAT";

```
COPY 0 / 0 I.MAC %2.MAC  
TYPE %2.TXT TYPE  
%0.BAT
```

To execute this file you must

enter the file name without extension, which is the value for parameter %0, followed by the replacement values for 0 / 0 I and %2. For example, if you type:

MYFILE A:PROG1  
B:PROG2 then press the  
ENTER key

then:

- %0 is replaced by

”MYFILE”

- %1 is replaced by

”A:PROG1”

- %2 is replaced by

”B:PROG2”

4-14

## MS-DOS USER GUIDE

The effect is to execute the following sequence:

COPY A:PROG1.MAC  
B:PROG2.MAC TYPE  
B:PROG2.TXT TYPE  
MYFILE.BAT

# Remarks

1 . Up to 10 dummy parameters (%0-%9) can be specified in this way. Refer to the SHIFT command if you wish to specify more than 10 dummy parameters.

2. If you use the percent sign as part of a file name within a batch file, you must enter it twice. For example, to specify the file ABC%.EXE, you must enter it as ABC%%.EXE in the batch file.

# INPUT AND OUTPUT

MS-DOS normally assumes that input comes from the keyboard and that output goes to the screen. However, the flow of command input and output can be redirected. Input can come from a file rather than the keyboard, and



output can go to a file or to a printer instead of to the screen. In addition, "pipes" can be created that allow output from one command to become the input to another. Redirection and pipes are discussed in the next sections.

# **REDIRECTING YOUR**

# OUTPUT

Most commands produce output that is sent to the screen. You can send this information to a file by using a greater-than sign (>) in your command. For example, the command:

DIR

displays a directory listing of

the current directory on the screen. The same command can send this output to a file named MYFILES instead of the screen by designating the output file In the command line:

```
DIR > MYFILES
```

If the file MYFILES does not already exist, MS-DOS creates it and stores your directory listing in it. If

MYFILES already exists,  
MS-DOS overwrites what is  
in the file with the new data.

Two greater-than signs (> >) can be used to tell MS-DOS to append the output of the command (such as a directory listing) to the end of a specified file. For example, the command:

```
DIR >> MYFILES
```

appends your directory listing to a currently existing file named MYFILES. If MYFILES does not exist, it is created.

# **REDIRECTING YOUR INPUT**

It is often useful to have input for a command come from a file rather than from

the keyboard. This is possible in MS-DOS by using a less than sign (<) in your command. For example, the command:

```
SORT < NAMES >LIST1
```

sorts the file NAMES and sends the sorted output to a file named LIST 1.

# FILTERS

A filter is a command that reads your input, transforms it in some way, and then sends the output, usually, to the screen or to a file. In this way, the data is said to have been "filtered" by the program. Since filters can be put together in many different ways, a few filters

can take the place of a large number of specific commands.

MS-DOS filters include FIND, MORE, and SORT. Their functions are described below:

FIND

MORE

SORT



Searches for a particular string of text in a file.

Takes standard output and displays it, one screen at a time. Sorts text.

Refer to Chapter 5 for details of these commands.

4-16

MS-DOS USER GUIDE

# **ENTERING AND USING MS-DOS COMMANDS**

You can see how these filters are used in the next section.

# COMMAND PIPING

If you want to give more than one command to the system at a time, you can "pipe" commands to MS-DOS. For example, you may occasionally need to have the output of one program sent as the input to another program.

A typical case would be a program that produces output in columns. You might want to have this columnar output sorted.

Piping is done by separating commands with the pipe separator, which is the vertical bar symbol (`|`). For example, the command:

```
DIR | SORT
```

will sort your directory into alphabetical order. The vertical bar causes all output generated on the left side of the bar to be sent to the right side of the bar for processing.

Piping can also be used when you want to send output to a file. If you want your directory sorted and sent to a new file (for example, DIREC.FIL), you could enter:

```
DIR I SORT > DIREC.FIL
```

MS-DOS will create a file named DIREC.FIL on your default drive. DIREC.FIL contains a sorted listing of the directory on the default drive, since no other drive was specified in the command. To specify a drive other than the default drive, enter:

```
DIR I SORT >B:DIREC.FIL
```

This sends the sorted data to a file named DIREC.FIL on drive B. A pipeline may consist of more than two commands. For example:  
DIR | SORT | MORE

will sort your directory, show it to you one screen at a time, and put -MORE- at the bottom of your screen when there is more output to be seen.

# Warning

If you use diskette.

'command piping' do not  
write protect your default  
drive

4-18

MS-DOS USER GUIDE

5. COMMANDS



# ABOUT THIS CHAPTER

This chapter gives the syntax and use of all the MS-DOS commands. The commands are presented in alphabetical order.

# CONTENTS

INTRODUCTION	5- 1
MS-DOS 3.20 AND NETWORKING	5- 5
COMMANDS	5- 8

ASSIGN

5-  
8

ATTRIB

5-  
10

BACKUP

5-  
12

BREAK

5-  
16

CHDIR 5-  
17

CHKDSK 5-  
19

CLS 5-  
23

COMMAND

5-  
23

COMP

5-  
27

COPY

5-  
30

CTTY

5-  
36

DATE

5-  
38

DEL

5-  
41

DIR 5-43

DISKCOMP 5-45

DISKCOPY 5-48

ECHO 5-51

ERASE 5-52

EXE2BIN 5-52

EXIT 5-55

FC 5-56

FDISK 5-66

FIND 5-66

FOR 5-70

FORMAT 5-72

GOTO 5-76

GRAFTABL 5-78

GRAPHICS 5-79

5-82

GWBASIC



HEXDUMP 5-83

IF 5-85

JOIN 5-87

LABEL 5-92

MKDIR 5-94

MODE 5-95

MODE COM: 5-96

MODE n 5-98

MODE LPT 5-100

MORE 5-103

PATH 5-104

PAUSE 5-105

PRINT 5-107

PROMPT 5-111

RECOVER 5-114

REM 5-115

REN 5-116

REPLACE 5-118

RESTORE 5-122

RMDIR 5-126

SELECT 5-127

SET 5-129

SHARE 5-131

SHIFT 5-133

SORT 5-135

SUBST 5-139

SYS 5-143

TIME 5-144

TREE 5-146

TYPE 5-147

VER 5-149

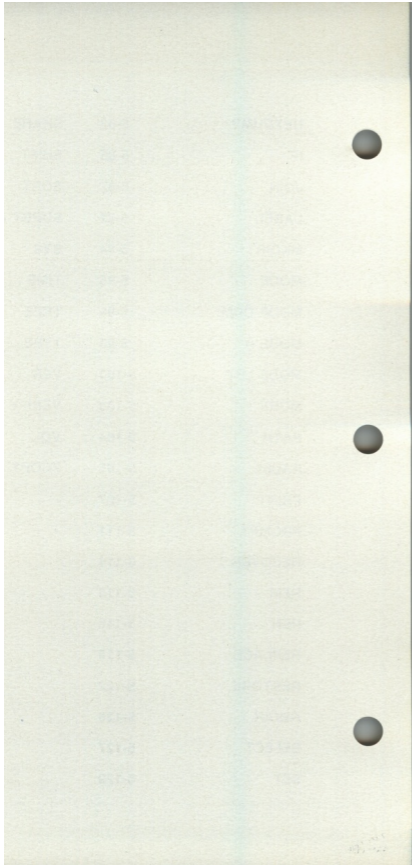
VERIFY 5-149

VOL 5-150

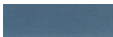
XCOPY 5-151







- 
- 



# INTRODUCT

The table below lists the commands included in this chapter and gives a brief description of each. It also tells you which commands are internal (I), which are external (E), and which it is not possible to use over a network (U).

# COMMAND CLASS FUNCTION

ASSIGN

E

Instructs  
MS-DOS  
to route all  
requests to  
one drive  
instead of  
another  
drive.

Sets or  
the read

ATTRIBS

E

attribute  
file. Set  
resets th  
archive  
attribute  
file.

BACKUP

E

Creates  
back up  
one or n  
disk file  
a series  
disks.

		Turns of and on t abort fe provided
BREAK	1	
		CTRL C CTRL BREAK
		Changes current
CHDIR	1	

		director,
		Analyze
		contents
CHKDSK	E U	the disk
		the spec
		or defau
		drive.
CLS	1	Clears th
		screen.

COMMAND E

Starts a  
commar  
process

COMP E

Compar  
the cont  
of a file  
group of  
files wit  
contents  
another  
or group



files.

Copies of  
or more  
to another  
file or to  
device.

Alternat  
several  
can be  
concaten  
and copi  
to a

COPY

1

destinat  
file.

CITY

1

Changes  
input/ou  
console  
which y  
issue  
commar

COMMAND CLASS FUNCT

DATE	1	Displays and sets date kno to the system.
------	---	---

DEL	1	Deletes specifie file(s).
-----	---	---------------------------------

Lists the  
requeste

DIR 1 directory entries.

DISKCOMP E U Compare the content of two diskettes of the same type.

Copies t

DISKCOPY E U

contents  
one disk  
onto  
another  
diskette

Turns th  
batch fil  
echo  
feature c  
and on.  
Text giv  
as a

ECHO

1

parameter  
will be  
output to  
the stand  
output  
device.

ERASE

1

Is the same  
as DEL.

Convert  
executal

EXE2BIN	E	files to binary format.
		Exits from a second command process and returns to a parent program.
EXIT	1	and returns to a parent program.
		command process.

FC	E	Compar the cont of two f
FDISK	E	Sets up MS-DO partltlor for the h disk.



FIND

E

Searches for a specific string of text in file(s).

FOR

1

Allows iterative execution of MS-DOS commands.

FORMAT	E U	Formats disk to receive MS-DOS files.
--------	-----	---------------------------------------

GOTO	1	Jumps to specified position a batch :
------	---	---------------------------------------

# MS-DOS USER GUIDE

## COMMAND CLASS FUNCTION

GRAFTABL E

Loads the non-BIC ASCII character graphics modes.

GRAPHICS E

Enables  
graphics  
currently  
displayed  
the screen  
be printed  
a compatible  
printer,  
with any  
when the  
SHIFT F  
SCR key

pressed.

GWBASIC E

Enters the  
GW-BASIC  
interpreter.

HEXDUMP E

Displays  
contents  
file, byte  
byte, in  
hexadecimal.

IF

1

Causes  
conditio  
executic  
commar  
batch fil

JOIN

E U

Joins a c  
drive to  
empty  
director  
another  
to produ

		single directory structure
LABEL	E U	Creates, changes deletes a volume
MKDIR	1	Creates directory

MODE

E

Sets the monitor mode, so transmits and prints in environment

MORE

E

A filter sends output to the terminal screen and



time.

PATH

1

Sets a  
command  
search p

PAUSE

1

Pauses u  
key is p  
In a batc  
file.

Queues

PRINT	E	files for backgro printing
-------	---	----------------------------------

PROMPT	1	Sets the DOS commar prompt.
--------	---	--------------------------------------



# COMMAND CLASS FUNCTION

RECOVER E U

Recover  
file or a  
entire di  
containi  
faulty bl

		A null command which can be used for putting remarks into a batch file
REM	1	
REN[AME]	1	Rename

REPLACE E

Replace previous versions files.

RESTORE E

Restores number files from back up. The back disks must have been created.

the BAC  
commar

RMDIR

1

Remove  
empty s  
director,

Copies y  
MS-DO  
diskette  
create a

SELECT E working  
for your  
selected  
country  
keyboard

Assigns  
string va  
the  
environ  
to anoth  
string; f  
in progr

SET 1

SHARE

E

batch fil

Installs  
network  
and recc  
locking.  
installs  
diskette  
change  
checking

Allows :



SHIFT

1

to more  
the defa  
number  
replaced  
paramet  
batch  
processi

SORT

E

A filter,  
sorts dat  
alphabet  
in forwa  
reverse

SUBST	E	Substitu dummy specific pathnam
-------	---	--

SYS	E U	Updates specific with the hidden s files, wh come fro
-----	-----	--

5-4

## MS-DOS USER GUIDE

## COMMAND CLASS FUNCT

TIME	1	Displays sets the system t
------	---	----------------------------------

TREE

E

Displays  
the direc  
and path  
the spec  
drive. It  
has an o  
to list th  
in each  
director

Displays

TYPE	1	contents specifie on the v screen.
		This commar displays your scr
VER	1	the vers: number MS-DO system y

are using,

VERIFY

1

Verifies  
writes to

VOL

1

Displays  
volume  
of the di  
the spec  
or defau  
drive.

XCOPY

E

Copies 1  
and  
subdirec

# MS-DOS 3.20 AND NETWORKING

MS-DOS 3.20 supports networking using the MS-Network extension software.

The file/record locking mechanism installed when



using the following command only works when networking is active.

## COMMAND DESCRIPTION

SHARE	This program loads, then terminates, but stays resident in the Random Access
-------	--

Memory. It installs the file/record locking mechanism.

The following command(s) are useful for networking;

**COMMAND DESCRIPTION**

## ATTRIB

This command sets or resets the read-only attribute of a file or displays the attributes of that file. If any application opens a file with read/write permission, ATTRIB can set the file to read-only,

allowing  
certain  
application  
programs to be  
run and shared  
over the  
network in  
compatibility  
mode.

This command  
can be used to  
copy files from

COPY

a network disk to your own or to another network disk.

DIR

This command can be used to display information about files on network disks.

## PRINT

For a printer server, use PRINT with the network printer as a logical device.

This command can be used to update or to add files to network drives.

## REPLACE

Use this command with the /A switch to restore deleted files, which have been previously backed-up with the XCOPY command.

## XCOPY

This command can be used to recursively copy directory structures and the files contained within those directories, from or to network drives. Use this command with the /M switch



to carry out incremental backups of network drives; as this switch turns off the archive bit.

5-6

MS-DOS USER GUIDE



Most MS-DOS commands  
can be used over the network.  
But do not use:

COMMAND RESULT...

the error  
message:

CHKDSK Cannot  
CHKDSK a  
CHKDSK Network drive.

If you suspect  
a problem  
contact the  
Network  
Manager.

the error  
message:

Cannot  
DISKCOMP to  
or from a  
Network drive.

DISKCOMP Use COMP \*.\*  
for each  
relevant  
directory  
instead.

the error  
message:

Cannot  
DISKCOPY to

DISKCOPY or from a  
Network drive.

Use COPY \*.\*  
for each  
directory  
instead.

the error  
message:

FDISK  
Cannot FDISK  
a Network  
Drive.

the error  
message:

Cannot  
FORMAT a  
Network Drive.

FORMAT

The Network  
Manager can  
stop the server,  
do FORMAT

then restart the  
server.

the error  
message:

JOIN

Cannot JOIN a  
Network Drive.

the error  
message:

LABEL

Cannot LABEL

a Network  
Drive.

COMMAND RESULT...

the error  
message;

Cannot  
RECOVER to a  
network drive.



RECOVER      The Network  
Manager can  
stop the server,  
do

RECOVER,  
then restart the  
server.

the error  
message:

SUBST              Cannot SUBST

to a network  
drive.

the error  
message:

Cannot SYS to  
a network  
drive.

SYS

The Network  
Manager can  
stop the server,

do SYS then  
restart the  
server.

This command  
will not work  
for files copied  
over a network.

VERIFY

VERIFY ON  
will only cause  
verification of  
writes to local  
files.



# ASSIGN

Instructs MS-DOS to route all requests for one drive to another drive.

Classification

External

```
[d: ][path] ASSIGN [drive1 =  
drive2]
```

5-8

## MS-DOS USER GUIDE

Table caption Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the  
drive where  
ASSIGN is to be

found.

path

Specifies the directory where ASSIGN Is to be found.

drive1

The letter of the drive whose requests are to be rerouted.

drive2

The letter of the drive to which requests for drive1 are to be rerouted.



## Characteristics

Following execution of the ASSIGN command MS-DOS converts all references for drive1 to drive2.



Note that only the drive letter should be entered in the command line. Do not enter the colon.

If you enter the ASSIGN command without parameters all current assignments will be reset.



5-9

Table captionExamples

IF you enter... THEN...

ASSIGN all requests to  
A = C B drives A or B will  
= C be rerouted to drive  
C.

ASSIGN all assignments are  
reset.

## Remarks

Never ASSIGN drives and then use the following commands or unpredictable results will occur and/or error messages will be displayed.

BACKUP DISKCOMP  
DISKCOPY FORMAT JOIN  
LABEL PRINT RESTORE  
SUBST

# ATTRIB

Sets or resets the read-only attribute and/or archive bit attribute of a file.

Classification

External

```
[d:][path] ATTRIB [-hR|-R] [+A|-A] [drive:][file-path]filename
```

^ u ^ ^

# MS-DOS USER GUIDE

5-10



Table captionWhere

SYNTAX	MEANING
ELEMENT	

d

Specifies the drive where ATTRIB is to be found.

path

Specifies the directory where ATTRIB is to be found.

Specifies the drive where

drive

filename is to be found

file-path

Specifies the directory where filename is to be found.

The filename of the files you want to

filename reference.  
Wildcard  
characters (\* and  
?) can be used in  
the filename.



## Characteristics

+ R sets the read-only  
attribute of a file. - R  
disables read-only mode.

-I- A sets the archive



attribute of a file. -A clears the archive attribute of a file.

To display the attribute of files enter:

ATTRIB filename.

Remarks

If an application opens a file with read and write permission, ATTRIB forces read-only mode to allow file

sharing over a network.

5-11

The BACKUP, RESTORE, and XCOPY commands use the archive attribute to control a selective Backup/Restore/Xcopy on files that have been modified. You can use the +A and -A options to select files that you want to back up with the BACKUP /M switch or copy

with the XCOPY /M switch.

## Example

The following example makes the file named MYFILE.TXT read-only:  
ATTRIB-i-RMYFILE.TXT

# BACKUP

Creates a backup of one or more disk files on a series of disks. The source is usually a hard disk, but can be a floppy disk. The target is usually a floppy disk, but can be a hard disk. However the source and the target must be different drives.

## Classification

External

[d:]\path] BACKUP source-  
drive:\pathname] target-  
drive: [/S] [/M] [/A] [/D:  
date]

5-12

MS-DOS USER GUIDE

Where

# SYNTAX

## ELEMENT MEANING

d

Specifies the drive where BACKUP is to be found.

path

Specifies the directory where BACKUP is to be found.

source-  
drive

The disk drive to  
be backed up.

The files you  
wish to back up.  
If you enter only  
the source-drive  
specifier then  
only those files  
In the current  
directory are

pathname

backed up. If you specify a path terminating In a directory name then all files in that directory will be backed up. If the path terminates in a file name (or a group of file names specified using wild card



characters) only the specified file(s) will be backed up.

target-  
drive ■

The disk drive in which the backup copy is to be made.

The files contained in the

/S

subdirectories are to be backed up as well as those In the specified directory (or current directory if no directory is specified). This includes files at all directory levels below the specified directory.

/M

Only those files  
In the specified  
directory that  
have been  
modified or  
created since the  
last backup will  
be backed up.

SYNTAX

ELEMENT

MEANING

/A

The specified files will be added to the disk already inserted in the diskette drive. If /A is not

specified  
you will be  
prompted  
to insert a  
diskette  
once the  
BACKUP  
program is  
in  
memory.

Only those  
files in the

**/D:date**

specified  
(or  
current)  
directory  
that have  
been  
created or  
modified  
since the  
specified  
date are to  
be backed  
up. Refer  
to the

DATE  
command  
for valid  
date  
formats.

## Characteristics

Once you have  
entered the  
BACKUP  
command a

prompt will be issued asking you to insert a target diskette (unless you included /A in the command line). You must use MS-DOS formatted diskettes. Any files that already existed on the target diskette



will be deleted  
unless you

used the /A option. Once the target diskette is full you will be prompted to insert another target diskette. Be sure to label each diskette as the order will be important when you restore your backup to hard disk.

As each file is backed up its name is displayed on the screen.

The exit code is set by the BACKUP command as follows:

- 0 Normal completion.
- 1 No files found.
- 2 Some files not backed up due to file sharing conflicts.

- 3 Command execution terminated by the user.
- 4 Command execution terminated due to an error.

The error level exit code can be used by the batch processing `IF ERRORLEVEL` command.

MS-DOS USER GUIDE



## Note

The files on the backup diskettes cannot be used except for restoring using the RESTORE command.

## Warning

You should not use the BACKUP command if the

drive you are backing up has been ASSIGNed, JOINed, or SUBSTITuted. If you do, you may not be able to restore the files with the RESTORE command.

## Examples

IF you  
enter...

THEN...

BACKUP  
C:\*.COM  
A:

each file with the .COM extension in the current directory of the hard disk drive C is backed up onto a series of diskettes in drive A.

all files on the hard disk drive C will be backed

BACKUP up onto a series  
C:\*.\* A:/S of diskettes in  
drive A.

BACKUP the file named  
C:MYDIR\ MYFILE in the  
MYFILE directory  
A: /A MYDIR is added  
to the backup  
diskette in drive  
A.

BACKUP  
\*.\* A; /M

all files in the current directory on the hard disk that have been created or modified since the last backup was made are backed up onto a series of diskettes in drive A.



BACKUP  
\*. \* A:  
/D:01-01-  
84

all files In the  
current directory  
on drive C that  
have been  
created or  
modified since 1  
January 1984 are  
backed up onto a  
series of  
diskettes in drive  
A.

# BREAK

Turns off and on the abort feature provided by CTRL C and CTRL BREAK.

Classification

Internal

BREAK [ ON | OFF ]

Characteristics

The default setting of BREAK is OFF. With this setting MS-DOS normally checks CTRL BREAK only for input/output operations involving screen, keyboard and printer. CTRL BREAK will not normally abort other functions, such as the assemble or compile operations. Specifying BREAK ON makes CTRL BREAK effective for other functions.

If you specify BREAK without a parameter the current BREAK setting is displayed.

5-16

MS-DOS USER GUIDE

# CHDIR

Changes the current directory; displays the name of the current directory.

Classification

Internal Syntax 1

CHDIR [drive:][path]

Syntax 2

CD [drive:][path]

Table caption Where

SYNTAX ELEMENT	MEANING
-------------------	---------

drive
-------

The letter of the drive where the new directory is to be found.
---

path

A path that terminates with the name of the directory you wish to enter.

## Characteristics

Use CHDIR (or CD) with a path to change to any valid directory.

To change to the parent directory of your current directory enter: CHDiR ..

Used without a parameter CHDIR displays the full path and name of your current directory. This feature is useful if you have forgotten the name of the directory in which you are working.

To change to the root directory enter:



CHDIR \

Table caption Examples



IF you  
enter...

THEN...

CHDIR	MS-DOS puts y
\BIN\USER\	directory
JOE\FORMS	BIN\USER\JOE

CHDIR ..

MS-DOS puts y  
parent of the cu  
working directo  
the above exam  
\BIN\USER\JOI



5-18

MS-DOS USER GUIDE

# CHKDSK

Analyzes the contents of the disk in the specified or default drive.

Classification

External, Non-network

[d:][paff7] CHKDSK [drive:]  
[dir-path][filename] [/F] [/V]

Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the drive where CHKDSK is to be found.

Specifies the

path                    directory where  
CHKDSK is to  
be found.

drive                    The drive  
containing the  
disk to be  
checked.

dir-path                The path to the  
directory to be  
checked.

filename

A file or group of files specified using wild card characters. A status report for the individual files will be displayed, if they are non-contiguous.

# SYNTAX MEANING ELEMENT

/F CHKDSK tries to correct any errors it finds.

/V CHKDSK displays status messages for each directory, subdirectory and

each file  
specified.

## Characteristics

CHKDSK produces a status report on the File Access Tables, Directories, Files, and bad sectors of the disk. Run CHKDSK with the /F switch to attempt to correct these error then the status report informs you of faults



with files and/or directories, the cause may be bad sectors on the disk. If so the first time you run CHKDSK with the /F switch the problems with the files and directories should be fixed, this may release some space on the disk. Run CHKDSK with the /F switch a second time to mark faulty clusters in this freed space. A cluster on the disk consists of one or more sectors, depending on the

cluster and sector size.

If you specify the /V switch, CHKDSK displays messages while it is running and will also list the hidden files.

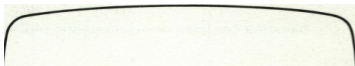
If no parameter is specified then the disk in the default drive is checked.

5-20

MS-DOS USER GUIDE

# Example

The following is an example of a CHKDSK status report:



Volume SYS (3 20) created  
Dec 6,1986 9:20a

362496 bytes total disk space  
36864 bytes in 3 hidden files  
289792 bytes in 34 user files

35840 bytes available on disk

262144 bytes total memory

224320 bytes free



Fig. 5-1 CHKDSK Example

## Remarks

To redirect CHKDSK output,  
use the > redirection symbol

followed by the name of a file. Error messages will be sent to the file specified. Do not use the IF switch if you redirect CHKDSK output.

If you use the IF switch CHKDSK tries to fix any errors found in the directory, file or file allocation table (FAT). When errors are found, due to lost clusters, CHKDSK asks you "Convert lost chains to files (Y/N)?".

If you reply Y then press ENTER, CHKDSK recovers each cluster in the disconnected chain to a file called FILEnnnn.CHK, in the root directory of the specified drive (where nnnn starts at 0000 and increases by 1 for each lost chain). See Appendix D "ERROR MESSAGES" for a list of all the messages CHKDSK issues.

Space on diskettes is allocated in clusters.

Diskettes that have had a lot of file creation and deletion activity become fragmented, because clusters are not allocated sequentially. The first free cluster found is the next cluster allocated regardless of its location on the diskette.

A fragmented diskette can cause poor performance due

to delays involved in reading or writing a file.

CHKDSK will display one of the following messages:

**filename  
contains non-  
contiguous  
blocks**



or

**All specified  
file(s) are  
contiguous**

If the first message appears and you are experiencing poor disk performance, use the COPY command to copy all the files in the directory

to a newly formatted diskette. Then use the copy rather than the original.

Use the command CHKDSK \*.\* to find out whether the files in the current directory are contiguous or not.

5-22

MS-DOS USER GUIDE



# CLS

Clears the screen.

# Classification

Internal

# CLS

# Characteristics

All data on the display screen is cleared. The cursor is moved to the upper left hand corner (the home position). This command has no effect on memory or files.

# COMMAND

Starts a new command processor

## Classification

External

[d\]\path] COMMAND

[drive:shellpath]\cttydev\

[/E: nnnnn] [/P] [/C  
command-string]

# Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the drive  
where COMMAND  
is to be found.

path

Specifies the directory where **COMMAND** is to be found.

drive

A single letter drive-name of the drive containing **COMMAND.COM**

shellpath

The pathname of the directory containing COMMAND.COM

cttydev

An alternative device for standard input and output (see the CTTY command for more details).



`/E'.nnnnn`

This switch specifies the environment size, where nnnnn is the size in bytes. The size may range between 160 and 32768 bytes. The default value is 1000 bytes.

This switch make

/P

this copy of  
COMMAND.COM  
permanent. It is n  
possible to exit to  
the primary  
command  
processor without  
re-booting the  
system. If the  
AUTOEXEC.BAT  
file is present in  
the root directory  
of the current  
drive, it is

executed;  
otherwise you are  
prompted for the  
current date and  
time.

5-24

MS-DOS USER GUIDE

- ' - r

# SYNTAX ELEMENT

# MEANING

This switch causes the command-string to be passed to the new command processor for execution

/C command-  
string

and then  
exit to the  
original  
command  
processor.  
This must  
be the last  
switch if  
used;  
everything  
following is  
part of the  
command-  
string.

## Characteristics:

COMMAND.COM is loaded into memory in two parts: the transient part and the resident part. Some application programs write over the transient part of the command processor when they run. When this happens

the resident part looks into the COMSPEC variable in the environment, to find the command processor file.

COMSPEC = drive.shellpath  
typically

COMSPEC =  
A:\COMMAND.COM

You can use SET without any parameters to examine the environment. (See SHELL in

Appendix C for details on loading the top level command processor). One application of this is to install a copy of COMMAND.COM on Virtual Disk using (VDISK); calling it D:, for example. Then to issue the following call to invoke the second command processor.

```
D:\COMMAND D:\ IP
```



Another application is to call `COMMAND` without any parameters or only with `drive:shellpath`. This invokes a secondary command processor (a child), which inherits the parent command processor's environment and prompt. If this environment and/or prompt is modified, these changes are only known to the child processor and its applications. Using `EXIT` (see `EXIT` command in this

chapter), reinstates the parents environment. For example at the A> prompt enter:

COMMAND



The computer will respond:

Microsoft(R) MS-DOS(R)  
Version 3.20 (C)Copyright  
Microsoft Corp 1981-1986

A>

enter:

PROMPT time =  $T + G$

the computer will respond  
with the prompt  
incorporating the time known  
to the computer:

**time =**

# 17:10:11.80>

If you now enter:

EXIT

the computer exits to the parent processor which has the prompt.

**A>**

In batch files calling

```
COMMAND /C  
batch_filename  
[parameter....]
```

enables the original parent batch file to call a child batch file and command processor as a subroutine. Upon completion of the child batch file, there is an automatic exit to the parent batchfile and command processor.

Normally MS-DOS allocates 160 bytes (10 main memory paragraphs) for the environment table. This may not be enough if you want to set numerous environment variables using the SET or PATH command. A subdirectory can have a pathname of up to 63 characters, each character uses one byte. So with environment variables set to point to long pathnames, you

could easily require more than 160 bytes. With MS-DOS Ver. 3.20 it is now possible to increase the environment size of the command processor.

If you specify the environment size to be less than 160 bytes, the new command processor will have an environment of 160 bytes and the following error message is issued:

## MS-DOS USER GUIDE

# **Invalid environment size specified**

If you specify the environment size to be greater than 32768 bytes, the



new command processor will have an environment of 32768 bytes and the above error message is issued.

Between these two limits environment space is allocated in paragraphs, each paragraph consists of 16 bytes. When you specify an environment size, which is not a multiple of 16 the actual environment size is rounded up to be a multiple of 16, that is to the next

paragraph boundary.

# COMP



Compares the contents of a file or group of files with the contents of another file or group of files. This is useful for checking the results of a COPY operation.

## **Classification**

External

[d:]^ath] COMP [pathname1  
[pathname2]]

Table captionWhere

SYNTAX ELEMENT	MEANING
-------------------	---------

Specifies the  
drive where

d COMP is to be found.

path Specifies the directory where COMP is to be found.

The file or group of files (specified using wild card

characters) that the file or files specified by pathname2 are to be compared with. If the path terminates in a directory name, all files in the specified directory are compared. If the path terminates in a drive

pathname1

specifier the files In the current directory of that drive are compared.

The file or group of files (specified using wild card characters) to be compared with

pathname2

the file or group of files specified by pathname1. If the path terminates in a directory name or drive specifier, only the files with the same file name as those specified by pathname1 will be compared.



# Characteristics

The files you wish to compare may be on the same or different drives, or in the same or different directories.

If you enter the command without parameters, or if you omit the second parameter,

you will be prompted for the missing parameters.

As the COMP command proceeds it displays the files and paths of the files currently being compared.

An error message is issued if a specified directory path is invalid, or the two files to be compared are different in size, or a file specified by pathname2 cannot be found.

## MS-DOS USER GUIDE

If the same location in two files contains information which does not match, a message is issued indicating the offset (in bytes) within the files and the contents of the bytes for each file. If ten such mismatches are found the comparison terminates and issues an appropriate

message.

If, at the end of a comparison, the end-of-file marker cannot be found, the following message will be displayed:

**EOF mark not found**

This is necessary as some

applications create files that are always recorded in multiples of 128 bytes, although the data actually occupies a few bytes less than stated in the directory. The COMP command may therefore find compare errors in the last 128 bytes beyond the last actual data byte. The above message therefore indicates that the actual data in the two files matches.

Following a successful comparison COMP displays a message indicating that the files match and continues comparing the next pair of files. After all the specified files have been compared the following prompt appears:

**Compare more files (Y/N)?**

Press N to terminate the command, or Y if you wish to compare more files. If you press Y you will be prompted for the files you wish to compare.

## Examples

IF you	THEN...
enter...	

COMP  
A:\*.LST  
B:\*.CPY

all the files on drive A with the extension LST are compared with the files of the same name but with extension CPY on drive B

COMP

all the files on drive A with the extension LST are compared with



A:\*.LSI      the files of the  
C:              same name on the  
                 current directory  
                 of drive C

# **COPY**

Copies one or more files.  
Alternatively several files  
can be concatenated and  
copied to a target file.

## **Classification**

Internal

COPY [/A/B] pathname 1  
[/A/B] [ + pathname2  
[/A/B]...] [pathname[/A\B]]  
[/V]

# Where

SYNTAX ELEMENT	MEANING
-------------------	---------

The path of the

path name  
1

file to be copied (excluding the drive only if the file is on the default drive, excluding the directory path only if the file is in the current directory).

The path of any file to be

concatenated  
with the file in  
pathname 1  
(excluding the  
drive specifier  
pathname2 only if the file is  
in the default  
drive, excluding  
the directory  
path only if the  
file is in the  
current  
directory).

pathname

The path of the target directory or file (excluding the drive to place the file in the default drive, or the file name to retain that given in pathnamel).

Verify the target

*/V*

file by a read after a write and COPY then compares this data. An error message is output if this comparison fails

5-30

MS-DOS USER GUIDE

# Characteristics

COPY either makes a copy of a file or concatenates two or more files. The resulting target file may have the same name as the first source file (so long as it is in another directory), or a different name.

To copy a file and retain its name, use the following syntax:



**COPY pathname1 [pathname]**

where **pathname** gives the drive and/or directory in which the file is to be placed.

Omit the **pathname** parameter to place the copy in the current directory of the default drive. Specify simply a drive to place the copy in the current directory of that drive. If you specify a directory path, make sure

that you end the path with the name of an existing directory.

Two files cannot have the same name in the same directory: if you try to copy a file onto itself you will get an error message.

To copy a file and give the copy a different name, use the following syntax:

COPY pathname1 pathname

Where pathname gives the name of the file in which the copy is to be placed.

Omit drive and directory names from pathname to place the file in the current directory of the default drive.

Omit just directory names from pathname to place the file in the current directory of the specified drive.

Because you are renaming the file, the source and target directories may be the same.

If you specify in `pathname` a file that does not exist, MS-DOS creates it for you. If the file already exists, its previous contents are destroyed in the copy operation (warning: before copying be sure that you do not want these contents).

To concatenate two or more files, use the following syntax:

```
COPY pathname1 +  
pathname2[ + pathnames]...  
[pathname]
```

Combine the names of each file to be concatenated with the plus sign ( + ). COPY appends each file in turn to the previous one.

The result of the concatenation is a single file. This file is given the drive, directory, name and extension specified in pathname-, if this is absent, the resulting file replaces the file specified in pathnamet (that is, the first file to be concatenated). If only the drive is supplied, a new file with the name of the file specified in pathnamet is copied into the drive's

current directory. If drive and directory names only are supplied, a new file with the name given in pathnamet is copied into the specified subdirectory. If only the name and extension are supplied, the file is placed in the current directory of the default drive.

Table captionExamples

IF you enter...

THEN...

COPY  
BiSECRETS

the file  
SECRETS  
is copied  
from drive  
B to the  
default  
drive.

all files in  
the current



COPY \*.\* B:

directory of  
the default  
drive are  
copied to  
drive B.


the file  
SECRETS  
is copied to  
the file  
INFO in the  
current

COPY

`\DOCS\SECRETS` directory of  
`B:INFO` drive B (or  
the  
subdirectory  
`INFO` In the  
current  
directory if  
it exists).

5-32

MS-DOS USER GUIDE



IF you enter...

THEN...

COPY  
\DOCS\SECRETS  
\DOCS\NEWS

the file  
SECRETS  
copied to the  
file NEWS  
the  
subdirectory  
NEWS if it  
exists).

COPY INFO +  
NEWS + VIEWS  
ALL.LST

the files NEWS  
and VIEWS  
appended to  
the file INFO  
and the  
resulting  
concatenated  
file  
is copied to  
file ALL.LST

the file NEWS

COPY ALL.LST  
+ NEWS

is appended  
to the file  
ALL.LST,  
resulting in  
an enlarged  
ALL.LST.

all files with  
the extension  
.LST are  
concatenated  
and the res

COPY MST  
COMBIN.PRN

- 

Is placed in  
COMBIN.]  
The source  
target are  
operated on  
ASCII files  
default.

each file w  
the extensi  
.REF is  
appended t

COPY

\*.LST+\*.REF

\*.PRN

the file with  
the same name  
but the  
extension  
.LST, and create  
resulting files  
given the  
extension .

COPY

\*.LST+\*.REF

all files with  
the extension  
.LST then create  
files with the

COMBIN.PRN

extension .  
are placed  
COMBIN.]

COPY

ALL.LST+\*.LST

all files wi  
the extensi  
.LST, with  
exception o  
ALL.LST,  
appended t  
ALL.LST.



COPY  
PROG.COM/B +  
ERRS.TXT/A

the text file  
ERRS.TXT  
appended to  
the binary  
PROG.COM  
leaving the  
result in the  
binary file  
PROG.COM

Table caption5-33

# Remarks

As the examples show, you can use the wild cards \* and ?, to do both simple copy operations (where no target file is specified) and file concatenation.

When wild cards are present in two or more source parameters combined with the concatenation symbol (-

^), the result is a single target file. However if the target parameter itself contains a wild card, a series of concatenated files are produced.

Note that the penultimate example shows the correct way of concatenating files where one of the source files is also the target file. Had the command `COPY *.LST ALL.LST` been entered, the

previous contents of  
ALL.LST would have been  
destroyed and the following  
message would have  
appeared:

# **Content of destination iost before copy**

You may use the following

reserved device names in  
place of standard file name  
parameters (the colons are  
optional):

AUX: LPT2:

COM1: LPT3:

COM2: NUL:

CON: PRN:

LPT1:

For example, to copy text that you are about to enter from the keyboard into a file, use the following format:

COPY CON: filename

You can then enter text

directly into the file you have named, terminating your input by pressing CTRL Z followed by ENTER.

The parameters /A and IB shown in the syntax of the COPY command apply when you wish to regulate the amount of data to be copied. The following table shows the effect of /A and IB on files to which they are attached and on all remaining

files in the command until another parameter is found.

5-34

## MS-DOS USER GUIDE

IF you enter... WITH... THEN

the fi  
regar



/A

a source  
file

text (file, a  
conten  
copie  
but ex  
the fi  
of-fil  
chara  
(CTR

This i  
defau  
conca

/A

a target  
file

the fi  
regar  
text (  
file, a  
end-c  
chara  
(CTR  
addec  
last c  
This i  
defau  
conca

the fi  
regar  
binar  
and tl  
file in

a source any n  
file end-c  
chara  
copie  
is the  
for si  
copy.



/B

a target  
file

the fi  
regar  
binar  
end-c  
chara  
(CTR  
addec  
the de  
simpl

The default  
value is /A  
when you are

using COPY  
to concatenate  
files,

/B when you  
are using  
COPY simply  
to copy files.

See the last  
example  
COPY  
PROG.COM/B  
+

ERRS.TXT/A

which shows  
the use of /A  
and IB to

append a file  
of error

messages to a  
program file.

The default  
for

concatenated  
files being /A  
the IB

attached to the

program file is  
obligatory.

The /A must  
then be  
attached to the  
text file in  
order to cancel  
the previous  
IB parameter.

Table caption5-35

# CTTY

Changes the input/output console from which you issue commands.

## Classification

Internal

CTTY device



# Table caption Where

SYNTAX ELEMENT	MEANING
-------------------	---------

device	The reserved name of the device you wish to use.
--------	--

# Characteristics

A suitable terminal must be connected to the device port. Command Input/ Output is passed to the alternate terminal. The CTTY CON command must be entered at the alternate terminal to restore input/output back to the normal console.

# MS-DOS USER GUIDE



## Examples

IF you  
enter...

THEN...

CTTY  
AUX:

command I/O is  
moved to the

device attached to  
the RS-232-C.

CTTY  
CON:            command I/O is  
                 returned to the  
                 console.

**Note**

You must use MODE to initialize the device before use. There are many programs that do not use MS-DOS for input and/or output, but use the BIOS or hardware ports. The CTTY command will have no effect on these programs. CTTY will only affect programs that use MS-DOS.

# DATE

Displays and sets the date known to the system.

Classification Internal  
Syntax 1 USA

DATE [mm-dd-yy]

or

Syntax 2 Europe

DATE [dd-mm-yy]

5-38

MS-DOS USER GUIDE



h::-^ .V.- -.k,- X

Table captionWhere

SYNTAX

# ELEMENT MEANING

mm

The one or two-digit identifier of the month (1-12).

dd

The one or two-digit identifier of the day (1-31).



yy

The two or four-digit identifier of the year (80-99 or 1980-2099).

# Characteristics

The syntax depends on the COUNTRY setting in CONFIG.SYS.

Separate month, day and year entries by either hyphens (-) or slashes (/)•

If you leave out the parameter, DATE prompts you as in the following example:

**Current date is**

**Tue 11-5-85**

**Enter new date:**

Enter the date in the correct format, without entering a value for the day of the week. To accept the current date simply press ENTER.

## **Example**

IF you enter...

THEN...

DATE 1-2-84

2nd January 1984 is established as the current date.

DATE

the DATE program prompts you to enter the date.

**Remarks**

If the values or separators  
you enter are not valid,  
DATE displays the message:

**Invalid date**

**Enter new date:**

DATE then waits for you to  
enter a valid date.



5-40

# MS-DOS USER GUIDE

# DEL

Deletes the specified file(s).

## Classification

Internal Syntax 1

DEL [drive:] pathname

Syntax 2

ERASE [drive:] pathname

# Where

SYNTAX ELEMENT	MEANING
-------------------	---------

drive

Specifies the drive where the file(s) to be deleted are to be found.



pathname      The specification of the file(s) to be deleted, excluding the directory path to delete file(s) In the current directory.

You may use the wild cards \* and ? in the file name and extension.

To delete all the files in a directory enter the wild cards  
Alter

natively, enter a path ending in a directory. In these cases, MS-DOS prompts you to confirm your choice:

**Are you sure  
(Y/N)?**

Press Y to carry out the deletion, or N to return to the MS-DOS prompt.

To delete all files without a file extension, enter

Table captionExample

IF you      THEN...  
enter...

DEL all files in the  
B:MMP default directory on  
drive B: with the  
extension .IMP are  
deleted.

## Remarks

To delete an actual directory (as opposed to all the files that a directory contains) you must use the RMDIR

command, but first you must delete all the files that directory contains.

The command ERASE is synonymous with DEL.

5-42

MS-DOS USER GUIDE



# DIR

Lists details of the files in the current or a specified directory.

## Classification

Internal

# DIR

**[drive:]\pathnam**

**[ /P ] [ /W ]**

**Where**

SYNTAX ELEMENT	MEANING
-------------------	---------

Specifies the

drive

drive where the file and directory names to be listed are to be found.

pathname

The specification of the file and directory names to be listed, excluding the directory path to



list names in the  
current  
directory.

/P

The directory  
display halts as  
soon as the  
screen becomes  
full. Press any  
key to resume  
the listing.

*/W*

File and directory names only are displayed, five to a line across the screen.

If you do not specify a file name and extension, all files in the specified (or current) directory are listed.

You may use the wild cards \* and ? in the file name and extension. If you omit either the name or the period and extension, the wild card \* is assumed in its place.

To list a file that does not have an extension but exclude any that do in the pathname, enter the file name followed by a period (.).

DIR produces a display in

which the size in bytes and date and time of creation or last modification appear alongside the file name(s).

# Remarks

DIR does not display hidden files in a directory.

Table caption Examples

IF you enter... THEN...

DIR B: (or  
DIR B:\*.\*)

all files in the  
current  
directory on  
drive B are  
listed.

DIR .COM

all files In the  
current  
directory on  
the default

(or DIR  
\*.COM)

drive having  
the extension  
COM are  
listed.

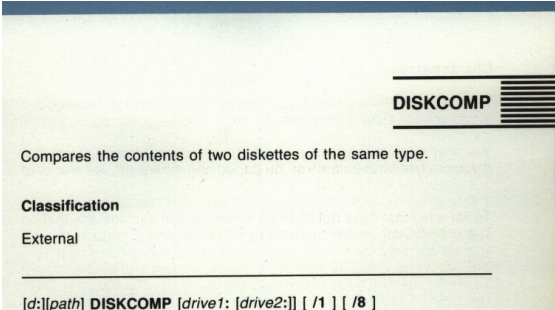
DIR  
AUTHORS  
(or DIR  
AUTHORS.\*)

all files in the  
current  
directory on  
the default  
drive with the  
name  
AUTHORS

and any  
extension are  
listed.

5-44

## MS-DOS USER GUIDE



**DISKCOMP**

Compares the contents of two diskettes of the same type.

**Classification**

External

---

`[d:][path] DISKCOMP [drive1: [drive2:]] [ /1 ] [ /B ]`

# Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the drive where DISKCOMP is to be found



path

Specifies the directory where DISKCOMP is to be found.

drive1

The drive to contain the first of the two diskettes to be compared.

The drive

drive2

containing the  
diskette to be  
compared with  
the diskette in  
drive1.

n

Only the first  
sides of the  
diskettes are  
compared.

IS

Only eight sectors per track are compared, even if the diskette in drive 1 has nine sectors per track.

## Characteristics

The DISKCOMP command can only be used to compare diskettes. If you specify a

hard disk drive an error message will be issued.

If neither drive is specified a single drive comparison is done on the default drive.

If only drive1 is specified then drive2 assumes the default drive.

The DISKCOMP command prompts you to insert the diskettes at the appropriate

time then waits for you to strike any key before continuing.

The DISKCOMP command compares all tracks on the diskettes and indicates mismatched tracks by track and side number.

When using DISKCOMP to compare diskettes in different drives, those drives must be of the same type.

The DISKCOMP command determines the number of sides to be compared and the number of sectors per track from the first diskette. That is:

- If the first diskette is High Density 1.2 MB (80 tracks, 15 sectors and double sided) then the second diskette must also be High Density 1.2MB.
- If the first diskette is

720KB (80 tracks, 9 sectors and double sided) then the second diskette must also be 720KB with the same specification.

- If the first diskette is dual-sided and has nine sectors per track then a nine-sectors-per-track comparison on both sides of the second diskette will be performed (unless /I and/or 18 was specified). If the second diskette is single-

sided or formatted eight sectors per track an error message will be displayed.

- If the first diskette is single-sided then only the first side of the second diskette will be compared regardless as to whether the second diskette is single or double sided.



# MS-DOS USER GUIDE

- If the first diskette is formatted eight sectors per track, then only eight sectors per track of the second diskette will be compared regardless as to whether the second diskette is formatted eight or nine sectors per track.

When all tracks have been compared the following

prompt appears:

**Compare more  
diskettes (Y/N)?**

Press Y to perform another comparison using the same drives, or press N to exit the program.

Table captionExamples

IF you  
enter...

THEN...

DISKCOMP  
A: B:

the diskette in  
drive B is  
compared with  
the diskette in  
drive A.

a single-drive  
comparison is  
done using the

DISKCOMP default drive.

/1 Only the first sides of the diskettes are compared.

an eight-  
sectors-per-  
track  
comparison is  
done between  
the diskettes in

DISKCOMP drive B and the  
B: /8 default drive. If  
drive B is the  
default drive  
then a single-  
drive  
comparison is  
done.

## Remarks

The DISKCOMP command compares entire diskettes. If you wish to compare only files you must use the COMP command.

The DISKCOMP command cannot be used to compare a diskette created using the COPY command with the original because the COPY command copies on a file-by-file (not track-by-track) basis.

On single-drive systems all prompts are for drive A, regardless of the drive letters you entered in the command line.

# DISKCOPY

Copies the contents of a diskette in one drive onto another diskette.

## Classification

External, Non-network

[d:][path] DISKCOPY



[sourcedrive:] [targetdrive:]  
[/I]

# Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the  
drive where  
DISKCOPY is

to be found

path

Specifies the directory where DISKCOPY is to be found.

sourcedrive

The letter of the drive that contains the diskette to be copied.

targetdrive

The letter of the drive that contains the diskette to receive the copy.

n

This switch specifies that only the first side of the

diskette Is to be  
copied.

5-48

MS-DOS USER GUIDE

# Characteristics

The DISKCOPY command  
copies entire diskettes. Use

COPY to copy files, or to copy to a different disk type than the source. For DISKCOPY, the diskettes must be of the same type, i.e. single, double, or quadruple density or high capacity. Also the source drive and the target drive must be able to read and write diskettes of the same type (see Chapter 1 for a table of Diskette and Drive compatibility). DISKCOPY automatically

determines the number of sides to copy, based on the source drive and diskette.

The target diskette is formatted or reformatted if necessary, during the copying. You can use the CHKDSK command to determine the capacity and DISKCOMP A: A: (comparing a disk with itself) to determine the format of the source diskette.

If you have 3 1/2 inch disk drives as drive A: and drive B: the operating system needs configuring to handle this media. Place the following declarations in your CONFIG.SYS:

```
DRIVPARM=/D:0 /F:2
```

```
DRIVPARM=/D;1 /F:2
```

(/D:0 indicates drive A:, /D:1 indicates drive B:, /F:2

indicates 720KB capacity.)

After changing the CONFIG.SYS file your computer must be re-bootstrapped. See the MS-DOS Software Installation Guide for more details.

With DISKCOPY you can specify the same drives or you may specify different drives. If the drives designated are the same, a single-drive copy operation



is performed. You are prompted to insert the disks at the appropriate times.

DISKCOPY waits for you to press any key before continuing. If you omit both parameters, a single-drive copy operation will be performed on the default drive. If you omit the second parameter the default drive will be used as the target drive.

After copying, DISKCOPY prompts:

# Copy complete

Copy another disk (Y/N)?

If you press Y, the next copy is performed on the same drives that you originally specified. You are prompted to insert the proper diskettes.

To end the copy, press N.

# Error Codes

The following error codes are returned by DISKCOPY, these can be

tested by IF ERRORLEVEL in a batch file.

0 Copied Successfully. The last diskcopy was completed

with no errors.

1 Non-fatal read/write error.  
An un-recoverable but non-fatal read or write error occurred.

2 CTRL C error. The user entered CTRL C to terminate DISKCOPY.

3 Fatal hard error.  
DISKCOPY was unable to read the source disk or

format the target disk.

4 Initialization error. There is not enough memory or the DISKCOPY command line syntax is incorrect or an invalid drive was specified.

## Remarks

After an apparently successful DISKCOPY, you can carry out a DISKCOMP

to compare the source and target diskettes.

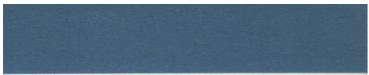
If diskette errors are encountered during a DISKCOPY, you can run CHKDSK with the /F switch to try to correct errors on the source diskette. Use XCOPY or COPY \*.\* for each directory, instead of DISKCOPY to copy the suspect diskette.

Do not use DISKCOPY when a directory on the source disk is JOINED to another drive. DISKCOPY does not acknowledge an ASSIGNED drive: the DISKCOPY parameters refer to physical drives.

MS-DOS USER GUIDE

5-50

# COMMANDS





# 9 ECHO

Turns the batch file echo feature off or on or outputs a message to the standard output device.

## **Classification**

Internal

ECHO I ON I OFF

[messafife]

## Characteristics

Normally, commands in a batch file are displayed (“echoed”) on the screen when they are interpreted by the command processor.

ECHO OFF turns off this feature. ECHO ON turns the echo back on.

If ON or OFF is not specified, then the current setting is displayed.

ECHO message outputs the message. Note that this message can be redirected using (> or >>). However when appending to a file, problems can arise if the original file is delimited by an End-Of-File (CTRL Z).

# Remarks

ECHO message can be used as a command outside of batch files.



5-51

# ERASE

Deletes the specified file(s)

See the "DEL" command.

# **EXE2BIN**

Converts files from .EXE format to binary format.

## **Classification**

External

[d:][path] EXE2BiN

pathname1 [pathname2]

5-52

MS-DOS USER GUIDE

COMMANDS

Where

SYNTAX

ELEMENT

MEANING

d

Specifies the drive where EXE2BIN is to be found.

path

Specifies the directory where EXE2BIN is to be found.

path name 1

The file specification of the file to be converted (excluding the drive if it is in the default drive, excluding a directory path if it is in the current



directory,  
excluding the  
extension to  
use the  
default  
extension of  
.EXE).

The file  
specification  
of the output  
file  
(excluding

pathname2

the drive  
and/or file  
name to  
accept the  
drive and/or  
file name in  
pathname1  
excluding a  
directory path  
to accept the  
current  
directory,  
excluding the  
extension to

use the  
default  
extension of  
.BIN).

## Table caption Characteristics

The input file must be in valid .EXE format produced by the linker. The resident, or actual code and data part of the file must be less than 64K. There must be no

# STACK segment.

Two kinds of conversions are possible, depending on whether the initial CS:IP (Code Segment: Instruction Pointer) is specified in the .EXE file:

1. If CS:IP is not specified in the .EXE file, a pure binary conversion is assumed. If segment fixups are necessary (that is, the program contains

instructions requiring segment relocation), you will be prompted for the fixup value. This value is the absolute segment at which the program is to be loaded. The resultant program will be usable only when loaded at the absolute memory address specified by a user application. The command processor will not be able to load the program.

2 If CS:IP is 0000:100H, it is assumed that the file will run as a .COM file with the location pointer set at 100H by the assembler statement ORG; the first 100H bytes of the file are deleted. No segment fixups are allowed, as .COM files must be segment relocatable; that is, they must assume the entry conditions explained in the MS-MACRO ASSEMBLER User Guide. Once the

conversion is complete, you may rename the output file with a .COM extension. Then the command processor will be able to load and execute the program in the same way as the .COM programs supplied on your MS-DOS disk.

## **Remarks**

If the input file does not meet

one of the two requirements given above, the following message appears:

## **File cannot be converted**

Note that to create a standard .COM file using the assembler you must set the location pointer at 100H using the ORG statement and use the END statement to set



the first location as the start address. For example:

```
ORG 100H START:
```

```
END START
```

Do not have a .EXE file and a .COM file of the same name in the same directory, when you execute the file.

# MS-DOS USER GUIDE

## **EXIT**

Exits from a secondary command processor and returns to a parent program or command processor.

## **Classification**

Internal

# EXIT

## Characteristics

This command can be used when you are running a program and have started a secondary MS-DOS command processor, then want to return to your program. For example, to look at a directory on drive B: while running GW-

BASIC, you must start the command processor by entering SHELL. The system prompt will appear.

**A>**

You can now enter the DIR command and MS-DOS will display the directory. When you enter EXIT, you return to the parent, GW-BASIC.

# Note

The error level exit code if set by a program, is returned to the calling program. For example, the following batch script calls a secondary command processor:

```
A>Type COM.BAT  
COMMAND
```

```
IF ERRORLEVEL 1 ECHO
```

# Error

When you invoke this batch file by entering COM, the following message will appear:

**Microsoft(R) MS-  
DOS(R) Version 3.20  
(C)Copyright  
Microsoft Corp  
1981-1986**

**A>**

enter for example:

**FORMAT B:**

When formatting is complete  
enter: **EXIT**

If format returns an  
errorcode of 1 or greater,  
then the original batch file  
will echo:

- 
- Error**

- 
- FC**

Compares the contents of two files.

**Classification**



# External

[d:][path] FC [/a] [/b] [/c]  
[/I] [/lb length] [/n] [/t] [/w]  
[#] filename1 filename2



5-56

MS-DOS USER GUIDE

Where

# SYNTAX MEANING ELEMENT

d Specifies the drive where FC is to be found.

path Specifies the directory where FC is to be found.

filename 1

The name of the first file to be compared.

filename2

The name of the second file to be compared.

Table caption Characteristics

The File Comparison utility,

FC, compares the contents of two files. The differences between the two files can be output to the screen or to a third file. The files being compared may be either source files (files containing source statements of a programming language) or binary files (output from the assembler, the MS-LINK Linker utility, or a highlevel language compiler).

The comparisons are either on a line-by-line or a byte-by-byte basis. The line-by-line comparison isolates blocks of lines that are different between the two files and prints those blocks of lines. The byte-by-byte comparison displays the bytes that are different between the two files.

The following table describes the switches available with

the File Comparison utility. It is important to enter the switches in lower case: upper case switches are not recognized.

## SWITCH MEANING

/a	Abbreviates the output to ASCII comparison. displaying all the lines that are different, only the lines that are different, only the lines that are different.
----	--

begin each set of differences are displayed. The individual lines are represented by  
(...)

A binary comparison of files is performed. The files are compared 1 byte, with no attempt to synchronize after a mismatch. The mismatches are as follows:

-ADDRS~~F1--F2 :  
yy zz

/b

(where xxxxxxxx is  
address of the pair c  
from the beginning  
Addresses start at 0  
yy and zz are the m  
bytes from file1 and  
respectively. If one  
contains less data th  
other, then a messag  
out. For example, if  
ends before file F2,



utility displays: fc:  
than F1 . This option  
default when you co  
with extensions of '  
''.COM'', ''.SYS'',  
''.LIB'' or ''.BIN''.

The utility Ignores 1  
letters. All letters in  
are considered upper  
letters. For example  
an underscore repre

white space).

/c

M uch \_ M O R E \_ da  
S \_ N O T \_ F O U N D

will match

much \_ more \_ data \_ is

This switch is used  
source comparisons



## MS-DOS USER GUIDE

## SWITCH MEANING

The utility compare files in ASCII mode. Is the default when compare files that do not have extensions ".EXE", ".COM",

/I

”.SYS”, ”.OBJ”,  
”.LIB” or ”.BIN”.

/lb  
length

Sets the Internal Line Buffer to length lines. The default length of the internal buffer is 100 lines. Files that have more than length consecutive differing lines will abort the comparisons.

In

The line numbers are displayed on ASCII comparisons.

It

Tabs are not expanded to spaces. The default is to treat tabs as spaces 8 column positions.

The utility compares “whites” (tabs and

spaces) during the comparison. Thus, multiple contiguous whites in any line will be considered as a single white space. That although FC compresses whites, does not ignore them. The two exceptions beginning and ending whites in a line, which are ignored. For example (note that :

/w

underscore representer  
white space):

\_\_More  
\_data\_to\_be\_found\_

will match with

More\_data\_to\_be\_f

and with

More data to be fou

but will not match v

\_More data to be found

This switch is used  
in source comparison

/#

Replace # with the number of  
lines required to match for  
the lines within the files to be  
considered as matching



again, after a difference has been found. # can be any number from 1 to 9. If this switch is not specified, the number defaults to 2. This switch is used only In ASCII comparisons.

Note: The default setting for tabs is to convert them into spaces to 8-column positions.

The File Comparison utility reports differences between

the two files you specify by displaying the first file name, then the matching line before the differences, followed by the lines that differ between the files, followed by the first line to match in both files.

FC then displays the name of the second file followed by the matching line before the lines that are different, followed by the first line that matches. The default for the number of lines that must

match before FC recognizes a match is 2. (If you want to change this default, specify this number with the /# switch.) For example:

```
*****filename1
```

matching line before  
differences difference

1st line to match file2 in file1  
after difference

```
*****filename2
```

matching line before  
differences difference

1st line to match file1 in file2  
after differences

If there are too many  
differences (involving too  
many lines), the program  
simply reports that the files  
are different and stops.

**MS-DOS USER GUIDE**

5-60

If no matches are found after the first difference is found, FC displays:

**resynch failed. Files  
are too different**

and returns to the MS-DOS default drive prompt.

The comparison report is sent

to the screen unless you specify output redirection to a file.

FC uses a large amount of memory as buffer (storage) space to hold the source files. If source files are larger than available memory, FC compares only what can be loaded into the buffer space. If no lines match in those portions of the files that have been loaded into the buffer

space, FC simply displays the message:

**resynch failed. Files are too different**

For binary files larger than available memory, FC compares the files piece meal, overlaying the portions in memory with the next portions from disk. All differences are output in the

same manner as those files that fit completely in memory.

## **Examples**

Assume these two ASCII files are on disk:

ALPHA.DOC BETA.DOC



A

A

B

B

c

C

D

G

E

H

F 1

G J

H 1

1 2

M P

N

Q

0

R

P

S

Q

T

R

U

S

V

T

4

U

5

V

W

w

X

X

Y

Y

Z

Z

The following examples show three possible ways of using FC to compare the contents of these two files.

5-62

## MS-DOS USER GUIDE

IF you enter...

THEN...

FC ALPHA.DOC BETA.DOC

FC compares ALPHA.DOC with BETA.DOC and displays the differences on the screen. All the defaults

remain intact. The output appears on the screen as follows (the Notes do not appear):

```
***** ALPHA.DOC
```

C

D

E

F

G

\*\*\*\*\* BETA.DOC

C

G

NOTE: ALPHA file contains  
CDEFG, BETA contains CG

\*\*\*\*\* ALPHA.DOC I

M



N

O

P

\*\*\*\*\* BETA.DOC

1

J

1

2 P

NOTE: ALPHA file contains  
IMNOP, BETA contains  
IJ12P

\*\*\*\*\* ALPHA.DOC

V

W

NOTE: ALPHA file contains  
VW BETA contains V45W



THEN...

\*\*\*\*\*

BETA.DOC

V

IF you  
enter...

4

5

W

\*\*\*\*\*

FC compares  
ALPHA.DOC  
with  
BETA.DOC. 4  
lines have to  
be the same in  
the  
comparison,  
for FC to

regard the lines within the file as matching again. The output is redirected to the line printer (PRN).

\*\*\*\*\*

ALPHA.DOC

C

D

E

FC M

ALPHA.DOC F

BETA.DOC G

>PRN

H

I

M

N

0

P

\*\*\*\*\*

BETA.DOC

NOTE: P is

the

C 1st of a

string

G of 4

matches.

H

1

1

J

1





# 5- MS-DOS USER GUIDE

64 ■:

IF you enter.

THEN.

V

W \* \* 1

V

4

5 W

ALPHA.DOC

BETA.DOC

NOTE: W is the 1st of a  
string of 4 matches

FC 1b ALPHA.DOC

# BETA.DOC

the following binary comparison report appears:

NOTE: The first field is the relative address bytes from the beginning of the file. The second field Is the mismatching byte from ALPHA.DOC. The third field is the mismatching byte from BETA.DOC.

00000009: 44 47

0000000c: 45 48

0000000f: 46 49

00000012 47 4a

00000015 48 31

00000018	49	32
----------	----	----

0000001b	4d	50
----------	----	----

0000001e	4e	51
----------	----	----

00000021	4f	52
----------	----	----

00000024	50	53
----------	----	----

00000027	51	54
----------	----	----

0000002a	52	55
----------	----	----

0000002d	53	56
----------	----	----

00000030	54	34
----------	----	----

00000033	55	35
----------	----	----

00000036                    56    57

00000039                    57    58

0000003c                    58    59

0000003f:                    59    5a

fc: alpha.doc longer than

# **FDISK**

Sets up the MS-DOS partition for the hard (fixed) disk.

## **Classification**

External, Non-network



# **FDISK**

## **Characteristics**

The FDISK command allows you to set up the MS-DOS partition(s) on the fixed disk. Refer to the “MS-DOS Software Installation Guide” for operational details.

# **FIND**

Searches for a specific string of text in a file or files.

## Classification

External

```
[d:][paf/7] FIND [/V] [/C]
[/N] "string" [pathname]
```

Where

# SYNTAX MEANING ELEMENT

d

Specifies the drive where FIND is to be found.

path

Specifies the directory where FIND is to be found.

string            A string of valid characters contained in quotes (”).

pathname        The path of a file to be searched.

Table caption Characteristics

**FIND** displays all lines that contain the specified string from the file or files listed in the command line.

You cannot use wild cards in your file specifications.

If no files are specified, **FIND** takes the standard input and displays all lines that contain the specified string.

Put in two sets of quotes  
where the string itself  
contains quotes.

That is:

FIND "this is a quote (""")"  
finds the string this is a quote  
(")

Switches for FIND are;

SWITCH MEANING

/V

Causes FIND to display all lines not containing the specified string.

1C

Causes FIND to print only the count of lines in each file that contain a match.

/N

Causes each line to be preceded by its relative line number in the file.

## Example

IF you  
enter...

THEN...



FIND displays  
each line in  
"COLOR" BOOK1 and  
BOOK2 (in that  
order) that  
contains the  
string COLOR.

FIND displays  
each file name  
on the disk in  
drive B that does  
not contain the

string DAT.

DIR

B:|FIND/V Note: when using  
”DAT” piping in this  
manner your  
default drive  
system diskette  
must not be  
write-protected.

5-68

MS-DOS USER GUIDE

# Remarks

If you use the FIND command on BASIC text files, the text must have been saved in ASCII format.

If you use more than one switch with the FIND command, you will get the results summarized in the following table:

# COMBINATION RESULT

ail three  
switches or /V  
and /C

The count  
of lines  
that do not  
contain the  
string is  
displayed.

The count  
of lines

/C and /N

that  
contain the  
string is  
displayed.

/V and /N

The lines  
not  
containing  
the string  
are  
displayed,  
together  
with their

line  
numbers.

# **FOR**

Allows iterative execution of  
MS-DOS commands.

## **Classification**

Internal

Interactive entry:

```
FOR %char IN (item ...) DO  
[command] %char
```

Batch entry:

```
FOR %%char IN (item ...)  
DO [command] %o/ochar
```

**Where**

# SYNTAX MEANING ELEMENT

char	Any single character other than the digits 0-9.
------	---

item	A parameter valid for the command required
------	--



(separated from another such item by a space).

The command you wish to invoke. External commands may optionally be preceded by the drive and/or the path to the directory where

command

the command is  
to be found.

5-70

# MS-DOS USER GUIDE

COMMANDS

# Characteristic

Use the `%c/7a/` variable in a command line you enter interactively. Use the `%char` variable in a command line within a batch file. The `%char` or `%char` variable is assigned the value of each item listed in turn. The command specified is activated for each resulting

parameter.

You may include the wild cards \* and ? in an item.

Remember to separate each item with a space, and to surround the complete item list by parentheses.

Table captionExamples

IF you enter in THEN...

a batch file...

```
FOR %%f IN  
(* .ASM)
```

```
DO MASM  
%%f
```

All .ASM files are submitted to the assembler.

```
FOR %%f IN  
(report memo  
address) DO  
DEL %%f
```

The files report, memo and address are deleted.



# Remarks

A FOR command cannot call another FOR command directly. However a FOR command can call a secondary command processor, which in turn processes another FOR command (see details of COMMAND for a full explanation).

For example:

```
FOR %x IN (1 2) DO  
COMMAND 1C FOR %y IN  
(o/ox) DO REM %y
```

Table caption produces the  
output:

```
A> COMMAND o/oy IN REM  
/C FOR (1) %y  
DO
```



A> REM 1

A>

A> COMMAND %y IN  
/C FOR (2) REM  
DO %y

A> REM 2

# FORMAT

Formats a disk to receive MS-DOS files.

Classification

External, Non-network

[d:]lpath] FORMAT drive:  
[/1] [IS] [/O] [/V] [/8] [/4]

Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the drive where **FORMAT** is to be found.

Specifies the

path                      directory where  
FORMAT is to  
be found.

drive                      The name of the  
drive that  
contains the  
disk.

*5-72*

MS-DOS USER GUIDE

# COMMANDS

## Characteristics

You must run `FORMAT` on any new diskette you wish to use with `MSDOS`. Any information already on the diskette is destroyed.

`FORMAT` places a bootstrap loader directory and file

allocation table at the beginning of the diskette. It also checks for any faulty sectors on the diskette.

If you use the `FORMAT` command on a hard disk it will format the logical drive specified. If you are formatting an existing formatted logical drive on a hard disk, `FORMAT` prompts you with the following message:

# **Enter current Volume Label for drive (x:)**

This is a security feature, to prevent you accidentally formatting an existing formatted logical drive. However if that logical drive has no Volume Label, press ENTER in response to the

message. If the volume label that you enter does not match the label on the hard disk, `FORMAT` displays the message:

**Invalid Volume  
ID Format  
failure**

Otherwise `Format` continues:



**WARNING,  
ALL DATA ON  
NON-  
REMOVABLE  
DISK DRIVE x:  
WILL BE  
LOST!**

Proceed with Format (Y/N)?

If you want to format your hard disk, type Y and press the ENTER key. If you do not want to format your hard disk, type N and press the ENTER key.

Switch options available with the FORMAT command have the following effect:

**SWITCH MEANING**

/1

Formats a diskette single-sided. You would use this option for preparing 180 Kbyte diskettes on a double sided drive (or 160 Kbyte diskette if the 16 option is also specified). This option is not valid on a 1.2 MB, 3 1/2 inch, or hard

disk.

/S

Copies the hidden system files and COMMAND.COM to the disk being formatted.

Can only be used in conjunction with the IS option to leave a place in

the directory for the operating system of MS-DOS version 1.1. But the operating system is not placed on the disk.

Note: This option causes the FORMAT program to take significantly longer.

N

Allows you to enter a volume label. The **FORMAT** command issues a prompt that enables you to enter a unique volume label of up to 11 characters. This label will

appear in  
subsequent  
directory listings.

Formats diskettes  
8 sectors per track  
instead of the  
default.

IS 9 sectors per track.  
Diskettes  
formatted in this  
manner are  
compatible with

# MS-DOS Ver. 1.XX.

Formats 48 tpi  
diskettes in High  
Capacity drives.

Note: Diskettes  
formatted with this  
switch cannot be  
reliably used In  
Normal Capacity  
drives.



## Remarks

Refer to the section on "Disks", in Chapter 1 for charts of "Diskette Capacities" and "Diskette Type Compatibility in Different Capacity Drives".

5-74

MS-DOS USER GUIDE

Unless you use a switch to specify otherwise, the default format depends on the diskette type and the drive capacity.

Table caption Example

IF you  
enter...

THEN...

the diskette in

drive B is  
FORMAT formatted and  
B: /S operating system  
files are copied  
onto it.

## Remarks

For diskette drives,  
FORMAT prompts you with  
a message such as:

**Insert new  
diskette for  
drive B: and  
strike ENTER  
when ready**

When you have struck  
ENTER to continue, MS-  
DOS formats the disk  
cylinder by cylinder and

displays the following information:

**Head: x**

**Cylinder: y**

Where the head-value can be 0 or 1, and the cylinder-value increases from 0 to the number of cylinders formatted. When format has finished you will receive a

message such as:

**Format complete**

**362496 bytes**

**total disk space**

**362496 bytes**

**available on disk**

# Format another (Y/N)?

Press Y to format another; N to return to MS-DOS.

If you have 3 1/2 inch disk drives the operating system may need

configuring to handle this media. If both the "A;" and

B; are 3 1/2

inch drives, place the following declarations in your CONFIG.SYS;

```
DRIVPARM = /D:0/F:2
```

```
DRIVPARM=/D:1 /F:2
```

(/D:0 indicates drive A:, /D:1 indicates drive B:, /F:2 indicates 720KB capacity.)

After changing the CONFIG.SYS file your computer must be re-



bootstrapped. See the "MS-DOS Software Installation Guide" for more details.

## **Error Codes**

The following error codes are returned by FORMAT, these can be tested by IF ERRORLEVEL in a batch file:

0 Normal completion

3 Terminated by user ( CTRL BREAK or CTRL C)

4 Fatal Error

5 N response to hard disk prompt. Proceed with Format (Y/N)?



# GOTO

Jumps to a specified position in a batch file.

## Classification

Internal



GOTO label

5-76

# MS-DOS USER GUIDE

# COMMANDS

## Where

SYNTAX ELEMENT	MEANING
-------------------	---------

A string of  
characters, the  
first eight of

which are significant (there is no need for quotes around the string). In a batch file, when the GOTO command is executed, the next command executed is on the line following the

label

label. Any line in a batch file can start with a :label. The contents of this line are not displayed by the MS-DOS batch processor. So preceding batch lines by colon (: ) is useful for placing comments in a

batch file. If the first eight characters of two labels are identical, GOTO that label will cause a jump to the first of the two labels. If no label is found the batch file terminates, with the message:



Label not found.

# Characteristics

To define a label in a batch file, precede a sequence of characters by a colon (:).

Batch processing then ignores the line until it encounters the GOTO command with the label as parameter. It then jumps to

the line below the one that contains the label.

Do not enter the colon when using the label as a parameter of GOTO.

## Example

IF you execute  
a .BAT file      THEN...  
containing...

:foo

an Infinite  
number of  
”looping...”

REM  
looping...

messages  
are  
produced.

GOTO foo

# GRAFTABL

Loads the non-BIOS ASCII characters for graphics modes.

## Classification

External

[d:][path] GRAFTABL

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the drive where GRAFTABL is to be found.

path

Specifies the directory where GRAFTABL is

to be found.

# Characteristics

After the character table is loaded, the following message is displayed:

**GRAPHICS**

**CHARACTERS**

# LOADED

The routine is resident and occupies Random Access Memory space. If you try to load GRAFTABL again the following message is displayed.

# GRAPHICS CHARACTERS

# **ALREADY LOADED**

## **Remarks**

The Random Access Memory space occupied may only be reclaimed by rebooting the system.

**MS-DOS USER GUIDE**



5-78

# GRAPHICS

Enables graphics currently displayed on the screen, along with any text, to be printed, on a compatible printer, when the SHIFT PRT SC keys are pressed.

Classification

External

[d:]lpath] GRAPHICS  
[printer-type] [/B][/R]  
[/D[/U|/H)]

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the  
drive where  
GFtAPHICS is  
to be found

path

Specifies the directory where GRAPHICS is to be found.

Table caption The printer-type parameter can be:

printer-  
type

DESCRIPTION

Olivetti PR-15B or  
PR-17B or Industry  
graphics Standard Graphics  
Printer or Epson  
MX/FX 80

Industry Standard  
col or 1 Color Printer with  
black ribbon

Industry Standard  
Color Printer with

color4 RGB (Red, Green,  
Blue and Black)  
ribbon 4 colors.

colors Industry Standard  
Color Printer with  
CMY (Cyan,  
Magenta, Yellow  
and black) ribbon 8  
colors.

jx80

Epson JX-80 (color printer) using the Epson character set.

dm

Olivetti PR-12B (DM285) or PR-14B (DM295) Color Printer. The character set must be IBM International 2.

# Remarks

If no printer-type parameter is given then graphics is assumed.

The GRAPHICS command can only be used with printers that have graphics capabilities.

# Switches



Switch options have the following effects:

5-80

MS-DOS USER GUIDE

SWITCHES MEANING

Causes the  
background  
color to be

/B

printed,  
otherwise the  
background is  
suppressed.

Causes black on  
the screen to be  
printed black  
and white on  
the screen to be  
printed white.

/R

Without the  
switch the

default is to  
print black as  
white and white  
as black.

/D

For a machine  
with an  
Enhanced  
Graphics Color  
Board (EGC)  
but no DEB INT  
10 filter

installed.

400 scanlines  
(Olivetti High  
Resolution).

/U

This switch  
cannot be used  
unless preceded  
by ID.

200 scanlines  
(Industry

/H

Standard  
Compatible).  
This switch  
cannot be used  
unless preceded  
by /D.

# Characteristics

This command must be entered to install graphics support necessary to print the

screen in graphics modes. SHIFT PRTSC then invokes the printing function. Re-enter the GRAPHICS command with new parameters to reset the existing parameters; graphics support is not reinstalled, only the parameters are changed.

Text modes are printed in the upright position. Graphics Modes are rotated counter-

clockwise 90 degrees on the printout page, so the Visual Display Unit's upper right corner appears on the paper's upper left corner.

The GRAPHICS command can only be used with printers that have graphics capabilities.

## **Remarks**

Use INT 5 to print the screen from a program.

Use the GRAPHICS command before entering GW-BASIC if you want to print graphics and text with the GW-BASIC LCOPY command.

For details of printing with an Enhanced Graphics Color Board, see the EGC Board User Guide.



# Warning

Do not turn the printer off while printing as this may cause unpredictable effects, and force you to reboot the operating system.

# GW BASIC

Enters the MS GW-BASIC interpreter.

## Classification

External

[d:][path] GWBASIC

# SYNTAX MEANING ELEMENT

d

Specifies the drive where GWBASIC is to be found.

path

Specifies the directory where GWBASIC is to be found.

5-82

# MS-DOS USER GUIDE

Table caption5-82

# COMMANDS

## Remarks

For more information on how to initialize GW-BASIC see the "MS GWBASIC Interpreter under MS-DOS User Guide".

# HEXDUMP

Displays the contents of a file, byte by byte, in hexadecimal.

## Classification

External

[d:][path] HEXDUMP

[drive:]filename

Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d	Specifies the drive where HEXDUMP is to be found.
---	---

path

Specifies the directory where HEXDUMP is to be found.

drive

A drive letter, specifying the current directory of the specified drive.



filename      The file whose  
                 contents are to  
                 be displayed.

Each line of the display shows 16 bytes of information. At the left-hand end of the line the hexadecimal address of the first byte in the line is given. The hexadecimal value of each of the next 16 bytes then follows -two hex digits per

byte. The right-hand columns show the ASCII equivalents (if any) of the bytes displayed in that line.

## Example

IF you enter... THEN...

The contents  
of the file

HEXDUMP      named  
B;ALPHABET    ’’ALPHABET  
on the diskette  
in drive B are  
displayed in  
hexadecimal.

HEXDUMP produces a  
display similar to the  
following:

Dumping File:  
BiALPHABET

0000: 20 20 61  $\begin{matrix} 62 \\ 63 \end{matrix}$  64 65 66 67

0010: 6f 70 71  $\begin{matrix} 72 \\ 73 \end{matrix}$  74 75 76 77

0020: 34 35 36  $\begin{matrix} 37 \\ 38 \end{matrix}$  39 2d 5e 40

0030: 2e 7c 41  $42$  44 45 46 47

0040: 4f 50 51  $\frac{52}{53}$  54 55 56 57

0050: 24 25 26  $\frac{27}{28}$  29 3d 7e 60

0060: 20 20 20  $\frac{20}{20}$  20 20 20 20

0070: 20

0080: Oa 20 20 20 20 20 20 20  
20

Table caption"

abcdefghijklmn

"opqrstuvwxyzOI23

'•456789-'\*@[

Table

captionlABCDEFGHIJKLMN

"OPQRSTUVWXYZ !”# "\$%

4 '() = '■ {+\*} <>?

Table captionHEXDUMP  
Complete

Fig. 5-2 HEXDUMP Display

5-84

MS-DOS USER GUIDE

Causes conditional execution  
of a command in a batch file.

IF

# Classification

Internal

IF [NOT] condition  
command

## Where

SYNTAX MEANING  
ELEMENT



condition      One of the three  
valid conditions  
listed below.

The command  
you wish to  
conditionally  
execute.

If the command  
is external, it

command may optionally be preceded by the drive where it is to be found and/or the path leading to the directory where it is to be found.

## **Characteristics**

The specified command is

only executed if the condition is true. If it is false the command is ignored.

Valid conditions are as follows:

CONDITION	MEANING
-----------	---------

	The command is executed only if the
--	-------------------------------------

EXIST [d:]  
[paff)l  
filename

specified  
file exists,  
on drive d:,  
and in the  
directory to  
which the  
path leads.  
The default  
drive is  
searched if  
d: is not  
specified.  
The current  
directory is

searched if  
path is not  
specified.

The  
command is  
executed  
only if the  
two strings  
are identical  
after  
parameter  
substitution.

String 1 =  
=string2

The case of the characters in string1 and string2 is significant.

The command is executed only if the previous

ERRORLEVEL program  
number executed  
had an exit  
code of the  
specified  
number, or  
higher.

The  
command is  
executed  
only if the  
previous

NOT program  
ERRORLEVEL executed  
number had an exit  
code of less  
than the  
specified  
number.

5-86

MS-DOS USER GUIDE



# COMMANDS i V r

## Examples

IF you enter...

THE

If the  
MAF

IF NOT EXIST

does

\SPECIAL\MARKER

the u

C:\BIN\CREATE

progr

C:\SPECIAL\MARKER CRE

run to  
it.

```
IF %1 == OLIVETTI  
ECHO
```

```
PARAMETER 1 IS  
OLIVETTI
```

The c  
displ  
mess  
if par  
1 afte  
subst  
is the  
"OL

IF NOT ERRORLEVEL  
3 LINK


If the  
level  
three  
linke

# JOIN

Joins a disk drive to an empty directory on another drive to produce a single directory structure.

Classification External, Non-network  
Syntax 1 - To join

[d:]\path] JOIN connected-drive splice-drive:\splice-directory



Syntax 2 - To deassign a join

JOIN connected driveID

or

Syntax 3 - To find out which drives are joined

JOIN

Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the drive where JOIN is to be found.

Specifies the

path

directory where  
JOIN is to be  
found

connected-  
drive

The drive which  
is to be  
connected to  
another drive.

splice-  
drive

The drive to  
which reference

is to be made.

splice-  
directory

The connected drive's directory structure is spliced to this directory.

/D

This switch indicates that the connected drive is to be unspliced.



5-88

MS-DOS USER GUIDE

# Characteristics

The JOIN command removes the distinction that physical drives are separately addressable by drive letter. You can refer to all the

directories on the joined drives as a single tree structure on one logical drive. After the JOIN command is executed the connected-drive becomes invalid.

## Remarks

You can only join a connected-drive to a splice-directory which is a sub-

directory of the root of the splice-drive. If the splicedirectory does not exist JOIN will create it. If the splicedirectory does exist it must be empty of files and sub-directories. Do not JOIN a drive, if the drive being JOINed is part of a substitution (SUBST) or assigned (ASSIGN).

Example; To join drive A; (the connected drive) to drive

C: (the splice-drive). Before the join the "C: Drive Directory Structure" is as shown in the following figure.



(ROOT)

JOE DENISE JANET

Fig. 5-3 C: Drive Directory

# Structure

Before the join the "A: Drive Directory Structure" is shown in the following figure.

(ROOT)

SYSTEM

Fig. 5-4 A: Drive Directory Structure

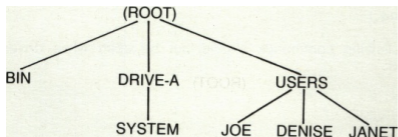
after the command

JOIN A; C:\DRIVE-A

The "Spliced C: Drive Directory Structure" appears to be:

5-90

MS-DOS USER GUIDE



## Fig. 5-5 Spliced C: Drive Directory Structure

Notice that in the "Spliced C: Drive Directory structure" that the subdirectory DRIVE-A has been created; the subdirectory SYSTEM of A: has been spliced into C: and the root directory of drive A: has been replaced by the sub-directory DRIVE-A. The whole directory structure of the connected drive is always

joined. Drive A: is inaccessible while JOINed. If you try to refer to it the error message is output;

## **Invalid drive specification**

To find out which drives are joined enter without parameters: JOIN



in the above example the following message is output:

```
A: => C:\DRIVE-A
```

To deassign the join in the example enter:

```
JOIN A: ID
```

Note: The current directory of the splice-drive should always be \(\root) when JOIN commands are executed.

When joining drives, the connected-drive should not be the default drive.

# Warning

The following commands should JOINed:

- BACKUP
- DISKCOMP

- DISKCOPY

- FORMAT

- RESTORE

not be used while drives are

# **LABEL**

Creates, changes or deletes a disk volume label.

## **Classification**

External, Non-network

[d:]\path LABEL [drive:]  
[volume-label]

5-92

# MS-DOS USER GUIDE

# COMMANDS

Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d	Specifies the drive where LABEL is to be
---	--

found.

j

path

Specifies the directory where LABEL is to be found.

drive

The drive containing the disk you wish to

# LABEL.

volume-  
label

The name you wish to give the disk. Refer to the section

”Parameters” in Chapter 4 for the syntax details of volume-label.

From 1 to 11 characters including spaces



are allowed.

# Characteristics

If you do not specify a volume label, LABEL prompts

**Volume in drive X is**

**XXXXXXXXXXXX**

Volume label (11 characters,  
ENTER for none)?

Type the volume label that  
you want and press the  
ENTER key. See the section  
"Parameters" in Chapter 4  
for the syntax details of  
volume-label. If you want to  
delete the volume label, just  
press the ENTER key.

LABEL prompts with the message:

**Delete current  
volume label  
(Y/N)?**

If you press Y, Label deletes the volume label on the disk, otherwise the volume label remains unchanged.

# MKDIR

Creates a new directory.

## Classification

Internal Syntax 1

MKDiR [drive:] path Syntax  
2

MD [drive:] path

Where

SYNTAX ELEMENT	MEANING
-------------------	---------

drive

Specifies the drive where the directory is to be

created.

path

The path of the directory you want to create.

# Characteristics

Use MKDIR to create or add to a hierarchical directory structure on the disk in the

default or specified drive.

5-94

## MS-DOS USER GUIDE

You may enter either MKDIR or MD to invoke this command.

# **Example**

IF you THEN...  
enter...

MKDIR  
\USER

The subdirectory  
USER is created  
beneath the root  
directory in the  
default drive.



# MODE

This command enables you to:

- Set the protocol for an RS-232-C port.
- Set the monitor mode.
- Set the printing characteristics or redirect printer output to the

communications port.

# Classification

External

# MODE

## COM:

Sets the protocol for an RS-232-C port.

[d:] \path] MODE

COMn:t>>aud[ , parity[ ,  
databits[ , stopbits[ , P]]]]

## Where

# SYNTAX MEANING

## ELEMENT

d

Specifies the drive where MODE is to be found.

path

Specifies the directory where MODE is to be

found.

One of the following values:

n

1 the built-in RS-232-C port

2 the (optional) second RS-232-C port.

baud

The baud rate.  
This must be one  
of the following  
110, 150, 300,  
600, 1200, 2400,  
4800 or 9600.

Only the first  
two digits need  
be specified.

# MS-DOS USER GUIDE

SYNTAX ELEMENT	MEANING
-------------------	---------

One of:

- |        |            |
|--------|------------|
| parity | - E (even) |
|        | - O (odd)  |
|        | - N (none) |

The default is E (even).

databits

The number of data bits. This must be either 7 or 8. The default is 7.

Either 1 or 2. If the baud rate is specified as 110,



stopbits

then the default is 2, otherwise the default is 1.

P

Continuous  
retry on time-  
out errors.

# Characteristics

The baud rate must be

specified. All other parameters are optional and will take default values if omitted.

Parameters must be separated by commas.

Use the P switch with a serial interface printer. Retry loops resulting from a repeated time-out condition can be broken by pressing CTRL BREAK .

# Table caption Examples

IF you enter... THEN...

MODE	the baud rate is set to 110, odd parity is specified, and the data
COM1:11,0,8,1	bits and stop bits are specified as

8 and 1,  
respectively.

MODE  
COM1:96

the baud rate  
is set to  
9600. All  
other  
parameters  
take default  
values.

# **MODE n**

**Sets the monitor mode. Syntax 1**

**[d:][path] MODE n**

**Syntax 2**

**[d:] [path] MODE**

**[n],m[, T]**

5-98

# **MS-DOS USER GUIDE**

# COMMANDS

## Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the  
drive where  
MODE is to be

found.

path

Specifies the directory where MODE is to be found.

An argument that may take one of the following



values:

40 The width of the display is set to 40

characters per line (color monitor only).

80 The width of the display is set to 80

characters per  
line (color  
monitor only).

BW40 Switches  
the active  
display  
controller to that  
of the color  
display, sets the  
display mode to  
black and white  
and the display  
width to

n

40 characters per line.

•

BW80 Switches the active display controller to that . of the color display, sets the display mode to black and white and the display width to

80 characters per line.

CO40 Switches the active display controller to that of the color display, sets the display mode to color and the display width to 40 characters per line.

CO80 Switches the active display controller to that of the color display, sets the display mode to color and the display width to 80 characters per line.

Is R for right

shift and L for  
left shift.

T

Displays a test  
pattern at the top  
of the screen.

5-99 ■

# Characteristics

Use the m parameter to shift the display one character (with 40 columns) or two characters (with 80 columns) to the left or right. If you specify the T switch, MODE displays a series of numbers at the top of the screen, and prompts you:

**Do you see the  
leftmost 0 (Y/N)**

?

if you have entered R , or

**Do you see the  
rightmost 9  
(Y/N) ?**

if you have entered L .



In response to either prompt  
press N to shift the display  
and redisplay the prompt,  
press Y to return to MS-DOS.

Remember, if you omit the n  
parameter, to precede the m  
parameter by a comma.

# MODE LPT

Sets the mode of operation for a compatible printer, or redirects the output for any printer to a communications port.

Syntax 1


```
[d:][path] MODE LPT#:  
[c/7ars][,spac/77fif] [,P]
```

# Syntax 2

[d:][path] MODE LPT#: =  
COMn

5-100

## MS-DOS USER GUIDE

A screenshot of a document page. The top portion is a dark blue header with the word 'COMMANDS' in white, bold, sans-serif capital letters. Below the header is a light cream-colored area. In the lower-left corner of this area, there is a small dark grey circle followed by the word 'Where' in a dark, sans-serif font.

COMMANDS

Where

# SYNTAX

# ELEMENT MEANING

d

Specifies the drive where MODE is to be found.

path

Specifies the directory where MODE is to be found.

#

The printer number (1, 2 or 3).

chars

The number of characters per line.

spacing

Vertical spacing in lines per inch. Its value must be

either 6 or 8.

n

The number of the communications port to which printer output is to be redirected.

P

Continuous retry on time-out errors.



5-101



# Example

IF you  
enter...

THEN...

MODE

LPT1:132,8

The mode of operation of printer 1 is set to 132 characters per line and 8 lines per inch.

MODE

LPT1: =

Output that would normally be sent to the printer is redirected to the



COM1: first RS-232-C  
port.

## Remarks

**The printer must be connected before using this command.**

# MS-DOS USER GUIDE

# COMMANDS

# MORE

Sends output to the terminal one screen at a time.

## Classification

External

[d:][path] MORE

# Characteristics

Either redirect input through MORE or use MORE as a filter added at the end of a command line. Screen output is displayed one screen at a time, instead of scrolling through its entire contents. At the bottom of each screenful, the prompt — MORE — is displayed. Press any key to display the next

screenful.

Table caption Example

IF you enter...

THEN...

The file  
MYFILE  
on the  
default  
drive is  
displayed

TYPE one screen  
MYFILE|MORE at a time.

or

Note: The  
MORE<MYFILE diskette  
on the  
default  
drive  
cannot be  
write-  
protected.

# PATH

Sets a command search path in the environment.

## Classification

Internal

```
PATH[ = ] [: I  
[pathname1,pathname]...]]
```



# Table captionWhere

SYNTAX ELEMENT	MEANING
-------------------	---------

pathname	The path of a directory you wish MS-DOS to search, including optionally a drive letter. Do not include a
----------	--

filename.



# Characteristics

PATH tells MS-DOS which directories, and in what order, to look for external commands after it has searched your current directory.

You can specify a single directory path or a number of paths each separated by a semicolon (;), (there must be no spaces in this string). The default is no search path; in this case MS-DOS only searches your current directory.

If you enter PATH with no parameter, MS-DOS displays the current search path. If you enter PATH; any

previously established path is cancelled and only your current directory is searched.

5-104

## MS-DOS USER GUIDE

You only need to set the search path once in any terminal session.

Path only finds executable files: e.g. .COM, .EXE, .BAT

files. Path ignores files with any other extension. The paths are searched in the order specified, so place the most frequently accessed directories first.

## **Example**

IF you enter...

T

PATH

C:\BIN\USER;C:\BIN\DEV

M  
s  
f  
c  
d  
t  
\  
a  
\  
o  
d

# Remarks

Non-existent directories specified in the PATH variable in the environment are ignored.

# PAUSE

Suspends execution of the batch file in which it is contained.

## Classification

Internal

PAUSE [comment]



# SYNTAX MEANING ELEMENT

comment      A string of up to  
121 characters.

## **Characteristics**

When PAUSE is encountered during the execution of a

batch file, any comment you have entered shows on the monitor followed by this prompt:

**Strike a key  
when ready...**

At this point the batch file is suspended, allowing you to change disks or perform any other necessary action.

To resume batch execution  
press any key with the  
exception of CTRL C or  
CTRL BREAK .

Press CTRL C to cancel  
processing of the batch file.  
The following prompt  
appears:

**Abort batch job  
(Y/N)?**

Press Y to cancel the batch operation and return to the MS-DOS prompt. Press N to return to the previous prompt.

Table caption Example

IF you enter...	When the batch file runs...
--------------------	--------------------------------

PAUSE	the batch job is
-------	------------------

insert	suspended and
target	”insert target disk
disk in	in drive B:”
drive B:	displayed.

Table caption5-106

MS-DOS USER GUIDE

# PRINT

Queues text files for background printing, while other MS-DOS commands are obeyed.

## Classification

## External Syntax 1

The first time PRINT is called

[d:][path] PRINT [/Didevice]  
[/Bibuffsize] [/liibusyticks]  
[/Mimaxticks] [/S'.timeslice]  
[/Qiqueuesize]  
[\pathname]...

## Syntax 2

Subsequent calls to PRINT

[d:][path] PRINT [[ /C//P]  
[[pathname]...] [/C//P]...]

## Syntax 3

Subsequent call to terminate  
PRINT

[d:][path] PRINT [/T]

## Where

SYNTAX ELEMENT	MEANING
-------------------	---------

Specifies the



d drive where  
PRINT is to be  
found.

path Specifies the  
directory  
where PRINT  
is to be found.

The file  
specification  
of a file to be

pathname

printed,  
optionally  
preceded by  
the drive and  
the path to the  
directory  
where the file  
is to be found.

Use to specify  
the print  
device. If not

ID'.device

used PRINT  
will ask for a  
print device.

/B:buffsize

Use to set the  
internal print  
buffer size in  
bytes. The  
normal size is  
512 bytes.  
Increasing the  
size may  
increase

performance.

/U.busyticks

Specifies the number of MS-DOS clock ticks that PRINT will wait if the printer is busy. Otherwise PRINT gives up its

timeslice. The default is 1 tick.

`M'.maxticks` Specifies how many MS-DOS clock ticks print can have to print a file, maxticks can be from 1 to 255 clock ticks (the default is

2).

/S.timeslice

Specifies the time slice value, timeslice can be from 1 to 255 (the default is 8). The lower the value the higher the priority of the

print queue.

Specifies the number of files allowed in the print queue, queuesize can be from 1 to 32 (the default is 10).

/Q:queuesize

O

# MS-DOS USER GUIDE



# COMMANDS

## Characteristics

You may use global and wildcard characters.

When you run PRINT for the first time in a terminal session, you are prompted as follows:

# **Name of list device [PRN:]**

Type the name of a valid line printer device driver, or simply press ENTER to accept the default line printer device PRN:.

The following switches are possible with this command:

# SWITCH      MEANING

/T	<p>TERMINATE: this switch cancels all files in the print queue (those waiting to be printed). A message to this effect will be printed.</p>
----	---

/C

CANCEL: This switch turns on cancel mode.

The preceding filespec and all following filespecs will be suspended in the print queue until /P switch is encountered on the command line.

IP

PRINT: This switch turns on print mode. The preceding filespec and all following filespecs will be added to the print queue until a /C switch is encountered on the command

line.

PRINT

with no

parameters

dispiays

the

contents of

the print

queue

on your screen without

affecting the queue.

# Examples

IF you enter... THEN...

PRINT U the print queue is emptied.

PRINT/T  
\*.ASM

the print queue is emptied then all the \*.ASM files are queued to the printer.

PRINT  
TEMP1/C  
TEMP2  
TEMP3

the three files indicated are removed from the print queue.

TEMPI is



PRINT removed from the  
TEMP1/C queue, whereas  
TEMP2/P TEMP2 is added.

## Warning

When PRINT is active do not  
print screen (SHIFT PRT  
SCR) or turn the printer echo  
on (CTRL PRT SCR).



5-110

# MS-DOS USER GUIDE



# PROMPT

Sets the MS-DOS command prompt.

## Classification

Internal

PROMPT [ {meta-character\character}...]

Where

SYNTAX MEANING  
ELEMENT

meta-  
character

A special character you wish to use to create the MS-DOS prompt, preceded by a \$

sign.

character

A character you wish to appear in your prompt, but this cannot be a \$ or any of the characters described in the following table.



# Characteristics

If no argument is entered, the prompt will be set to the default prompt, which is the default drive designation plus the > symbol. You can set the prompt to something different such as the current time, by using the meta-characters indicated below.

The following

meta-characters can be used  
in the prompt command



5-111

.

to specify special prompts.

They must all be preceded by a dollar sign (\$) in the prompt command:

SPECIAL

MEANING

CHARACTER

\$

The '\$'  
character.



t The time.

d The date.

P The default  
drive and the  
path to the  
current  
directory.

The version

V

number.

n

The default  
drive.

g

The ' > ' character.

I

The ' < ' character.

b

The ' I ' character.

—

A carriage return-linefeed sequence.

s

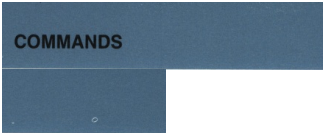
A space (leading only).

h

A backspace.

5-112

# MS-DOS USER GUIDE



COMMANDS

SPECIAL

MEANING

# CHARACTER

e ASCII escape  
(Hexadecimal  
1B).

q The ' = ' character.

## Example

IF you enter... THEN...

PROMPT  
HELLO

the prompt  
becomes  
HELLO

PROMPT  
\$p\$g

the default  
drive and the  
current  
directory is  
established as

a prompt, E.g.  
C:BIN>

PROMPT  
HELLO \$g

the prompt  
becomes  
HELLO >

the current  
time is shown  
in reverse  
video. The  
device

PROMPT      ANSI.SYS has  
\$e[7m\$t\$e[0m to be in  
CONFIG.SYS  
see Appendix  
B and  
Appendix C  
for more  
details.





# RECOVER

Recovers a file or an entire disk containing faulty sectors.

## Classification

External, Non-network

[d:]\path] RECOVER

[drive:]pathname]

Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d	Specifies the drive where RECOVER is to be found.
---	---

path

Specifies the directory where RECOVER is to be found.

drive

A drive containing a disk with faulty sector(s).

A file containing faulty sector(s).

pathname    Optionally preceded by a drive and directory path.

# Characteristics

Specify a file name to recover a particular file. RECOVER reads the

## MS-DOS USER GUIDE

file sector by sector and marks any faulty sectors it finds. MS-DOS will no longer allocate data to such sectors.

Specify a drive name to recover a complete disk. RECOVER reads the contents of the disk sector by sector

and marks any faulty sectors  
It finds.

When you run RECOVER on  
a disk any directory tree that  
may be present is destroyed.  
Files are placed in the root  
directory and renamed  
FILE0001 .REC,  
FILE0002.REC and so on. If  
there is not enough space in  
your root directory for  
information on all the files  
on the disk, the following

message is displayed:

Warning - directory fuui

If this happens copy these recovered files to another disk, delete the FILE\*.REC files and run RECOVER again with the same parameters as before. You can neither run RECOVER on a directory nor use a list of files or wild cards.

# REM

Displays a remark during the execution of a batch file.

## Classification

Internal

REM [remark]

Table captionWhere



# SYNTAX ELEMENT MEANING

remark	A string of up to 123 characters.
--------	-----------------------------------

## **Characteristics**

A remark inserted in a batch file shows on the screen as

soon as it is encountered  
during batch execution.  
ECHO OFF prevents display  
of remark.

# REN

Renames files.

## Classification

Internal

REN[AME] pathname  
filename

Table captionWhere

# SYNTAX MEANING

## ELEMENT

pathname      The path of the file to be renamed (excluding the drive only if the file is on the default drive).

filename

The new name including any extension you wish to give the file.



j II jifum' VI V

5-116

MS-DOS USER GUIDE

# COMMANDS



## Characteristics

REN changes the name and extension of the file specified in the first parameter to those given in the second parameter. REN cannot be

used to move a file from one drive or directory to another; therefore the second parameter must only be a filename.

The wildcard filenames, using \* and ?, may be used in either parameter. If wild cards appear in the second parameter, the corresponding characters in the first parameter remain unchanged.

# Table caption Examples

IF you  
enter...

THEN...

REN

BrPRESENT

PAST

the file  
PRESENT in  
the current  
directory in  
drive B is  
renamed  
PAST.



RENAME

\*JON \*.\*A?

any file in the  
current  
directory in the  
default drive  
with the  
extension JON  
has its  
extension  
changed to  
JAN.

# Remarks

An attempt to give a file a name already in the file directory results in the following message:

**Duplicate file  
name or file not  
found**

# REPLACE

Updates or adds files according to the criteria set by the option switches.

## Classification

External

[d:]\path] REPLACE [source-

drive:][source-path]source-  
file [target-drive:][target-  
path] [/A][/D][/P][/R][/S]  
[/W]

# Where

SYNTAX ELEMENT	MEANING
-------------------	---------

Specifies the

d drive where  
REPLACE is to  
be found.

path Specifies the  
directory where  
REPLACE is to  
be found.

source- Specifies the  
drive containing  
drive the source files.

source-  
path

Specifies the directory containing the source files.

source-file

Specifies the source file(s) that are to be added or replaced in the target directory.

The files can be specified by wild cards.

5-118

## MS-DOS USER GUIDE

COMMANDS

SYNTAX ELEMENT	MEANING
-------------------	---------

target-drive Specifies the target drive.

target-path Specifies the target directory.

Switches for REPLACE are:

SWITCH MEANING



/A

Causes REPLACE to add new files to the target directory instead of replacing existing ones. You may not use this switch with either the ID or IS switches.

Causes REPLACE to replace files in the target directory

ID

only if the source files are newer than the corresponding target files. This switch is incompatible with the /A switch.

Causes REPLACE to prompt you before replacing a

IP target file or  
adding a source  
file:

Replace  
(filename)? (Y/N)

This switch causes  
REPLACE to  
replace read-only  
files as well as  
unprotected files.  
If you do not  
specify this

/R

switch, any  
attempt to replace  
a readonly file  
causes an error and  
stops the replace  
pro

cess.

**SWITCH MEANING**

/S

This switch causes REPLACE to search all subdirectories of the target directory while it replaces matching files.

This switch is incompatible with the /A switch.

REPLACE never searches subdirectories in the source path.

/w

This switch causes REPLACE to wait before replacing or adding files. This is useful for changing diskettes.

## **Characteristics**

By default, it replaces files in

the target directory with files in the source directory that have the same name.

However when you specify the /A switch, REPLACE adds files that exist in the source directory (but NOT in the target directory) to the target directory.

As files are replaced or added. Replace displays the filenames on the screen; then at the conclusion of the

replace operation, it displays  
a summary line:

**NNN file(s)**

**added/replaced**

**or No files**

**added/replaced**

**Note**



You cannot use the REPLACE command to update hidden files or system files.

## Examples

Replacing Files:

Suppose your hard disk, drive "C:", contains several files of client

5-120

## MS-DOS USER GUIDE



names and phone numbers. To replace these files with the latest version of this file that exists on the disk in drive A, you would enter:

```
REPLACE A:\PHONES.CLI  
C:\ IS
```

This command replaces every file on drive "C:" that is named PHONES.CLI with the file PHONES.CLI from the root directory on drive "A".

## Adding Files:

Suppose you want to add some new printer device drivers to a directory called C:\MSTOOLS, which already contains several printer

driver files for a word processor. To do this, you would enter:

```
REPLACE A:*.PRD  
C:\MSTOOLS /A
```

This command adds any files from the default directory of drive "A:" with an extension of PRD (that do not currently exist in the \MSTOOLS directory on drive "C:" to C:\MSTOOLS.

# Error Codes

If Replace encounters an error, it returns one of the following error level codes:

1 Command line error

2 File Not Found


3 Path Not found

5 Access Denied

# 8 Insufficient Memory 1 5 Invalid Drive

Other MS-DOS internal  
extended error number.

You can test for these codes  
by using the IF  
ERRORLEVEL command  
within a batch file.



# RESTORE

Restores a number of files from back-up disks. The backup disks must have been created using the BACKUP command.

## Classification

External

[d:][path] RESTORE source-  
drive: [target-drive:]  
[pathname] [/S] [/P]

# Where

SYNTAX ELEMENT	MEANING
-------------------	---------

Specifies the  
drive where



d RESTORE is to  
be found.

path Specifies the  
directory where  
RESTORE is to  
be found.

source- The drive  
containing the  
disk with the  
backup

drive

information to be restored. Typically a floppy disk drive.

target-  
drive

The drive containing the disk to which the backup information is to be restored. Typically a hard

disk drive.



5-122

## MS-DOS USER GUIDE

SYNTAX	MEANING
ELEMENT	

The file(s) you

wish to restore.  
If you enter only  
the hard disk  
specifier, then  
all files backed  
up from the  
current directory  
are restored. If  
you specify a  
path terminating  
in a directory  
name then all  
files backed-up  
from that

pathname

directory will be restored to it. If the path terminates In a file name (or a group of file names specified using wild card characters) then only the specified file(s) will be restored.

IS

Files in all subdirectories, as well as those In the specified (or current) directory will be restored. This Includes all levels of subdirectory below the specified (or current) directory.

/p

You will be prompted before restoring files that have been modified since the backup was made, or those files that are read only. This switch is recommended when restoring

files backed up  
from MS-DOS  
Ver. 2.11 or Ver.  
3.10 disks.

# Characteristics

Once you have entered the command line you are prompted to insert the backup diskette. It is up to



you to ensure that you insert the diskette(s) containing the file(s) you wish to restore in the correct order. If you are unsure as to which diskette(s) contain the files you require, start with the first backup diskette then insert each backup diskette in turn in the order in which they were made. The RESTORE command will prompt you to insert the next diskette.

# Note

If you have backed up the root directory of a disk running under MSDOS Ver. 2.11 or Ver. 3.10, before you restore the root carry out the following procedures at the MS-DOS prompt:

1. C:

2. CD \

3. COPY CON 10.SYS

4. 10.SYS

5. Press F6

6. Press ENTER

7. COPY 10.SYS  
MSDOS.SYS

You have now created two dummy files called 10.SYS and MSDOS.SYS. These

have the same name as the hidden system files used by MS-DOS Ver. 2.11 and Ver. 3.10. When you restore the root directory of the C: drive, use the /P switch. The use of this switch is shown in the following example;

5-124

MS-DOS USER GUIDE

# Examples

IF you  
enter...

THEN...

All files on a series of backup diskettes in drive "A:" are restored to the hard disk drive "C:" in the

same directory structure. Before restoring a file which was changed after it was backed up, RESTORE asks you the following question: filename

Warning! The file above was  
RESTORE changed after it was backed up.

A: C:\  
/S/P

Replace the file  
(Y/N)?

Answer N, if  
filename is  
10.SYS,  
MSDOS.SYS,  
COMMAND.COM  
or any of the other  
files supplied on  
your MS-DOS  
Ver. 3.20 System  
or Supplementary  
Diskettes. When

you have finished restoring files to the hard disk, the dummy IO.SYS and MSDOS.SYS can be deleted.

all files on the backup diskette(s) having the file name extension .COM that were  
RESTORE backed up from



A: the current  
C:\* .COM directory are  
restored into the  
current directory  
on the hard disk  
drive "C:".

The RESTORE command  
sets the exit code as follows:  
0 Normal completion 1 The  
specified file(s) was not  
found

3 Command execution terminated by the user

4 Command execution terminated due to error

The exit codes can be used by the batch processing IF ERRORLEVEL command.

# RMDIR

Removes an empty sub-directory

Classification Internal

Syntax 1

RMDIR [drive:]path

Syntax 2

RD [drive:]path

Where

SYNTAX ELEMENT	MEANING
-------------------	---------

drive

The letter of the  
drive containing  
the directory  
you

wish to remove.

path

The path of the directory you wish to remove.



5-126

MS-DOS USER GUIDE

# Characteristics

RMDIR removes a directory that is empty apart from the . and .. special files. If the directory contains subdirectories or files, these must first be removed by means of the RMDIR or DEL command respectively.

You may either enter RMDIR or RD to invoke this command.

Table captionExample

IF you enter...

THEN...

RMDIR

C:\BIN\USER\JOE is

the  
specified  
empty  
directory

removed,  
on the C:  
drive.

# SELECT

Formats the target disk and installs MS-DOS, configured for your selected country and keyboard.

## Classification

External

```
[d:][path] SELECT [[s-drive]  
t-drive:[t-path]W country-
```



code keyboard-code

# Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the drive where SELECT is to be found.

path

Specifies the directory where SELECT is to be found.

s-drive

Specifies the source drive, which can only be A: or B:.. If this parameter is not specified the

default source  
drive is A:.

t-drive

Specifies the  
target drive. If  
this parameter is  
not specified the  
default target  
drive is B:. The  
source drive and  
the target drive  
must be  
different.

t-path

Specifies the target directory. If this is not specified, the root directory is the default.

country-

A three digit number which is the telephonic

code

international  
country code.

keyboard-  
code

A two character  
alpha code  
indicating your  
national  
keyboard.



**Note**

See the "MS-DOS Software Installation Guide" for more details.

5-128

MS-DOS USER GUIDE

# **WARNING**

This command **FORMATS** the target drive. Only use it for installing MS-DOS for

the first time onto unused  
disks.

# SET

Assigns one string value in the environment to another key string; for use in programs or batch files.

## **Classification**

Internal



SET [key = value]

Where

SYNTAX  
ELEMENT MEANING

key

The key string  
you wish-to  
assign a value to.

value

The string you wish to assign to the key string.

Note; leading spaces are significant.

# Characteristics

Use SET to assign a value to a standard parameter

included in an application program. The value remains operative during a working session until another SET command is issued.

When the SET key command is executed, with a key assignment, it inserts the entire string into a part of memory reserved for "environment" strings. If the name already exists in the environment, it is replaced

by the new string. If you enter the SET command with only the first string, the associated string name is removed from the environment. If you enter SET with no options, MS-DOS displays the current environment settings.

A program can get a listing of all the environment values that have been set by examining its environment.

(A pointer to the environment is passed in the Program Segment Prefix). Refer to the "MS-DOS System Programmer Guide" for more information.

You can also use the SET command with batch files. Instead of passing string values to a batch file by means of replaceable parameters (see Chapter 4) in the command line, you can

use SET to assign string values to string keys. Within the batch file the form of the key must be as follows:

`%key%`

That is, the string must be preceded and followed by a percent sign (whereas replaceable parameters are only preceded by a percent sign).

Use of the SET command with no parameters causes all the current SET assignments to be displayed on the screen.

Table captionExample

IF you enter... THEN...

SET the TTY value is set to VT52, in all batch

TTY = files up on execution  
VT52 %TTY% is replaced  
by VT52.

## MS-DOS USER GUIDE

5-130

# Remarks

The case of the key is converted to uppercase. The



case of the value is left in the case input. Be careful, because some programs require values in uppercase, these values must be input in uppercase..

For example:

IF you THEN...  
enter...

```
SET      in the environment tty
tty =    -> TTY
vt52
```

```
SET      PATH =
```

```
COMSPEC =
A:\COMMAND.COM
```

```
TTY = VT52
```

If the program expects VT52 in uppercase vt52 in lowercase will not be recognized.

# SHARE

Installs network file and record locking. It also installs a resident facility which checks for diskette removal during reading and writing to diskettes.

## Classification

External

[d:][path] SHARE [/F:  
memory-space] [/L: locks]

# Where

SYNTAX ELEMENT	MEANING
-------------------	---------

Specifies the

d drive where  
SHARE is to be  
found.

path Specifies the  
directory where  
SHARE is to be  
found.

When this  
switch is used,  
the memory-

space parameter indicates how much memory space is reserved for recording file sharing information. Each open file should be allocated space for its pathname plus eleven bytes; so the

/F:memory-space

space allocated should be between 32 bytes and 74 bytes per file. 74 bytes will allow for a full-length pathname of 63 characters. The default memory-space for filesharing information is



2048 bytes.

/L:locks

When this switch is used, memory space is allocated for the maximum number of locks it is possible to apply to a file. The default value for the

number of locks  
it is possible to  
apply is 20.

# Characteristics

If used it should be included  
in the system disk's  
AUTOEXEC.BAT file. Once  
the command has been called  
the support utility becomes  
resident. It takes about 5K

bytes of memory (with the default switch settings).

After SHARE is resident, all read and write requests are checked for lock violations. See the "MS-DOS System Programmer Guide" for details on file locking and unlocking.

5-132

MS-DOS USER GUIDE

After SHARE is resident, removal of a diskette during a read or write, causes a check of the replaced diskette's volume label. If the volume label has changed, MS-DOS issues one of the following error messages:

**Invalid disk  
change**

**reading/writing**

**drive x:**

# SHIFT

Allows access to more than 9 replaceable parameters in batch processing.

## Classification

Internal

# SHIFT

# Characteristics

Usually, batch files are limited to handling 10 parameters, %0 through %9. Initially %0 contains the name of the batch file. Initially %1 to %9 contain the values of the first nine replaceable parameters. To allow access to more than

nine replaceable parameters,  
use SHIFT to alter the  
numbering of your command  
line parameters. For  
example:

calling a batch file  
SUPER.BAT with the 12  
replaceable parameters:

```
SUPER pi  
,p2,p3,p4,p5,p6,p7,p8,p9,p10,]  
,p12
```



initially;

`%0 = SUPER o/o1 = p1`

`%9 = p9`

to access the other parameters SHIFT is used within the batch file one SHIFT will result in:

`%0 = p1`

`o/o9 = p10`

Successive SHIFT calls will result in:

$$\%9 = p11$$

and another SHIFT call will result in:

$$\%9 = p12$$

so finally:

$$\%0 = p3 \quad \%1 = p4 \quad \%2 = p5$$

$$\%3 = p6 \quad \%4 = p7 \quad \%5 = p8$$

$\%6 = p9 \quad \%7 = p10$

$\%8 = p11 \quad \%9 = p12$

So if you have entered more than nine replaceable parameters on the command line, those that appear after the ninth ( $\%9$ ) will be shifted one at a time into  $\%9$  by successive shifts. You can then refer to these parameters in your batch file.

5-134

# MS-DOS USER GUIDE

... !

# **SORT**

A filter which sorts data alpha-numerically in forward or reverse order.

## **Classification**

External

```
[d:][path] SORT [pathname]
```

[/R] [/ +number]

# Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the drive where SORT is to be found.

path

Specifies the directory where SORT is to be found.

pathname

The path of the file to be sorted.

This switch indicates a

/R

reverse ASCII  
sort.

/ + number

The column  
number from  
which to begin  
the sort.

# Characteristics

Sort is a filter which works



on text lines, the case of the characters is ignored.

Sort takes place in ASCII order unless you specify the /R switch, in which case a sort in reverse ASCII sequence is done. It starts with the first column of input unless you specify otherwise using the /+number switch.

If you do not specify a file, SORT takes the standard

input and outputs to the screen, unless you specify otherwise with the redirection symbols ">" and "<" or the pipe symbol "|".



## **Note**

The maximum size of file that can be sorted is 63 KB.

# Examples

IF you enter...

THEN...



```
SORT/R < UNSORT.TXT >  
SORT.TXT
```

the file UNSORT.TXT is  
sorted in reverse order and  
the result placed in the file

`SORT.TXT.`

`DIRSORT /+141 MORE`

the directory listing produced by the `DiR` command is sorted starting with the fourteenth column (the column that contains the file size), and output a screen at a time.





## Collating Sequence

So as to be able to sort  
National Character Sets the  
following character  
mappings are effected:

ORIGINAL CHARACTER	MAPPED CHARACTER
--------------------	------------------

C	C
---	---

0	U
---	---

e	E
---	---

a	A
---	---

a

A

a

A

a

A

c

C

e E

e E

e E

V 1

1 1

T 1



1

A

A

E

ae

1

A

A

E

A

/E

A

ORIGINAL CHARACTER

MAPPED CHARACTER

6

6

6

u

u

y

6

u

(P

£

¥

O

O

O

U

U

Y

O

U

\$

\$

\$

Pt

f

a

6

u

n

N

t

\$

A

I

O

U

N

N

6

r

1

1/2

1/4

»

?

n

n

1/2

1/4

3



S



5-138

# MS-DOS USER GUIDE

# SUBST

Substitute a dummy drive specifier for a pathname.

Classification External, Non-network

Syntax 1

[cl:]\path] SUBST dummy-drive: pathname or

## Syntax 2

[d:][path] SUBST dummy-  
drive: /D

or

## Syntax 3

[d:][path] SUBST

Table captionWhere

# SYNTAX MEANING ELEMENT

d

Specifies the drive where SUBST is to be found.

path

Specifies the directory where SUBST is to be found.

dummy-  
drive

The dummy drive that is to be used to refer to pathname.

pathname

The drive and/or directory path to which the dummy drive refers.

ID

A switch which indicates that the specified dummy drive substitution should be deleted.

## **Characteristics**

Enter SUBST with no parameters to display the current substitution, for

example the resulting output could be:

**M = >**

**C:\USR\MIKE**

The dummy-drive must be within the range of drives recognized by the system. If you use real drive letters for the dummy-drive, you will not be able to use the real

drive. Do not use the default drive as a dummy-drive. It is recommended that you do not use real drives. Increase the availability of drive letters by setting in CONFIG.SYS `LAST DRIVE = dummy-drive` or greater (see Appendix C).

Suppose you have a configuration with a hard disk "C:" and two floppy disk drives ("A:" and "B;"), and



are using a dummy drive =  
M. In CONFIG.SYS set  
LASTDRIVE = M. Call the  
command

5-140

## MS-DOS USER GUIDE



```
SUBST M; C:\USR\MIKE
```

where MIKE is a directory.  
You would now be able to

refer to the directory by using the alias "M:" instead of the path C:\USR\MIKE.

After substitution the command

```
DIR M;
```

could produce the following display.

```
Volume in drive M has no label
Directory of M:\

<DIR>                1-29-86   11:29a
<DIR>                1-29-86   11:29a
REALTEST OBJ         717      8-20-85   4:59a
TESTREAL OBJ         654      8-21-85   10:23a
REALTEST MAP        24242    8-20-85   4:59a
REALTEST PAS         207      8-20-85   4:57a
TESTREAL PAS         167      8-21-85   10:20a

7 File(s)          327680 bytes free
```

## Fig. 5-6 Substituted Directory Display

Substituting is particularly useful for programs that do not recognise paths, or you can use a letter as shorthand for a long path.

To undo a substitution use  
SYNTAX 2.

In the above example:

SUBST M: ID

will undo the substitution.



## Remarks

Never use the SUBST

command and then use the following commands on the dummy drive or unpredictable results and/or error messages will occur.

ASSIGN

BACKUP

DISKCOMP

DISKCOPY

FDISK

FORMAT

JOIN

LABEL

PRINT

RESTORE

Pay attention to the  
substitutions in effect when

using the following  
commands.

CHOIR

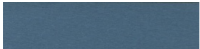
MDIR

MDIR

PATH

5-142

MS-DOS USER GUIDE





# **SYS**

Updates the specified disk with the hidden system files, which come from the default drive.

## **Classification**

External, Non-network

ld:][path] SYS drive:

Table captionWhere

SYNTAX ELEMENT	MEANING
-------------------	---------

d	Specifies the drive where SYS is to be found.
---	---

path Specifies the directory where SYS is to be found.

drive The drive that contains the target disk.

# Characteristics

Bootstrap off the system disk

containing the SYS.COM file. With this disk in the default drive, enter the SYS command.

The target disk must either be formatted, but without files in the root directory, or have been formatted with the IS parameter to contain previous versions of the system files. If this is not the case one of the following

messages appears;

**SYS cannot  
install MS-DOS  
on this disk**

or

**Not enough  
room for MS-**

# DOS on this disk

## Remarks

The file COMMAND.COM is not transferred.

The hidden files will not appear in any directory listing.

# TIME

Displays or set the system time.

## Classification

Internal

Table caption TIME  
[hh[:mm]]

Where

SYNTAX  
ELEMENT

MEANING

hh

Hours (0-  
23).

mm

Minutes (0-  
59).



Table caption5-144

**MS-DOS USER GUIDE**

# COMMANDS

## Characteristics

If you type one or more value(s) and then press ENTER, TIME sets the remaining value(s) to zero.

For example, if you enter "8" the time is set to 8:00:00.00

If you leave out the complete

parameter, TIME prompts you as in the following example:

**Current time is  
16:36:00.00**

**Enter new time:**

You can then enter a new time in the correct format. To accept the current time

simply press ENTER .

Table captionExample

IF you  
enter...

THEN...

TIME  
8:30

half past eight in  
the morning is set  
as the current time.

## Remarks

Note that the format of the time output varies depending on the COUNTRY configuration in CONFIG.SYS (see Appendix C).

# TREE

**Displays all the directories and paths on the specified drive. It also has an option to list the files in each directory.**

# Classification

## External

[d:][path] TREE [drive:] [/F]

## Where

SYNTAX  
ELEMENT MEANING

d

Specifies the drive where TREE is to be found.

path

Specifies the directory where TREE is to be found.

The drive whose



drive

directory

structure is to be  
examined.

IF

The files  
contained in  
each directory  
are also to be  
listed.

**MS-DOS USER**

# GUIDE

5-146

COMMANDS

## Characteristics

For every directory on the specified drive the TREE command will display;

- The directory path starting from the root directory of the drive.
- The subdirectories within the directory.
- The files in the directory (only if /F is specified).

Example

IF you  
enter...

THEN...

TREE  
A: /F

the path to each  
directory on drive  
A is displayed  
along with the  
subdirectories and  
files defined within  
each directory.

# TYPE



Displays the contents of the specified file on the video screen.

## **Classification**

Internal

TYPE pathname

Table caption Where

SYNTAX ELEMENT	MEANING
-------------------	---------

pathname	The path to the file to be displayed, including the filename.
----------	---

Filename can not  
be wildcarded.

# Characteristics

Use this command to examine a file without modifying it. Press CTRL NUMLOCK (or CTRL S ) to suspend output, press any key to recommence typing. Press CTRL BREAK (or CTRL C )

to terminate output. Press CTRL PRTSC (or CTRL P ) to turn the printer on, press CTRL PRTSC again to turn the printer off.

The complete contents of the file, including any non-alphabetic and non-numeric characters, appear on the screen. As such, the file may appear unreadable.

Tab characters are expanded



on the screen to tab stops every eighth column.

# Remarks

It is advised that you can only type (and print) text files.

5-148

MS-DOS USER GUIDE

# VER

Displays the MS-DOS version number.

## Classification

Internal

# VER

# Characteristics

This command displays on your screen the version number of the MSDOS system you are using.

# VERIFY

Sets an internal switch which causes disk writes to be verified.

## Classification

Internal

# VERIFY [ON I

# OFF]

## Characteristics

This command has the same purpose as the N switch in the COPY

command. If you want to verify that all files are written correctly to disk, you can use the VERIFY

command to tell MS-DOS to verify that your files are intact (no bad sectors, for example). MS-DOS will perform a VERIFY each time you write data to a disk. You will receive an error message only if MS-DOS was unable to successfully write your data to disk.

VERIFY ON remains in effect until you change it in a program (by a SET VERIFY

system call), or until you issue a `VERIFY OFF` command to MS-DOS.

If you want to know what the current setting of `VERIFY` is, enter `VERIFY` with no options.



Displays the volume label of the disk in the specified or default drive.

# Classification

Internal

VOL [drive:]

Table captionWhere

SYNTAX  
ELEMENT MEANING



drive

The drive that contains the disk to be examined.

5-150

## MS-DOS USER GUIDE

### Characteristics

If the disk does not have a volume label, VOL displays the following message:

Volume in drive x has no label

# **XCOPY**

Copies files and directories, including lower level directories if they exist.

Classification

External

[d:]lpath] XCOPY [source-  
drive:][source-path]source-  
filename [target-drive:]  
[target-path][target-  
filename] [/A][/D:mm-dd-  
yy]

# mmmi/smm

## Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the  
drive where  
XCOPY is to be

found.

path

Specifies the directory where XCOPY is to be found.

source-  
drive

Specifies the drive containing the source files.

source-  
path

Specifies the directory containing the source files.

source-file

Specifies the source file(s) that are to be copied. The files can be specified by wild cards.

target-  
drive

Specifies the  
target drive.

target-path

Specifies the  
target directory.

target-  
filename

Specifies the  
target file(s).  
The files can be  
specified by wild  
cards.

5-152

## MS-DOS USER GUIDE

Switches for XCOPY are:

### SWITCH MEANING

Causes XCOPY to copy source files that have their archive bit set. It



/A

does not modify the archive bit of the source file.

Refer to the ATTRIB command for information on how to set the archive attribute.

Causes XCOPY to copy source files that have been modified on or

/D

after the date specified by mm-dd-yy. Note that the date format may vary depending on the country code that you are using.

Causes XCOPY to copy empty subdirectories.

**/E** You must use this switch with the **/S** switch.

This switch is similar to the **/A** switch since it copies archived files only; however, it turns off the archive bit in the source file. Refer to the

**/M**

ATTRIB command  
for information on  
how to set the  
archive attribute.

IP

Causes XCOPY to  
prompt you before  
copying each file:  
target-filename  
(Y/N)?

IS

Causes XCOPY to recursively copy lower level subdirectories and their files. Empty subdirectories are not copied unless the IE switch is used with this IS switch. If you omit the IS switch, XCOPY works only within a single directory.

N Causes XCOPY to verify each file as it is written to the target to make sure that the target files are Identical to the source files.

/W This switch causes XCOPY to wait before it starts

copying files. This is useful for changing diskettes.

## **Characteristics**

When this command is used without parameters, it is the equivalent of simple or wildcarded file copy. When the /A or /M switch is used the command is useful for backing up disks, as an

alternative to the BACKUP command. When the IS switch (and optionally the /E switch) is used with the source-directory being root (\), this command can copy whole disks; in contrast to COPY, which can only copy files, one directory at a time. When the IS switch (and optionally the IE switch) is used with the source-directory being a subdirectory, a directory sub-



tree is copied.

## Examples

```
ATTRIB -I- A CABIN  
XCOPY cabin A: /A
```

The above example copies the whole of the BIN directory on the hard disk to a diskette in drive A;; setting the archive bit in each file with ATTRIB prepares for

the use of the /A switch. If there are too many files to fit onto the diskette in A:, use the /M switch instead as in the following example;

```
ATTRIB + A C:\BIN
```

```
XCOPY C:\BIN A: /M
```

When the target disk becomes full, XCOPY finishes and the following message is displayed:

# Disk Full

Put another diskette into the A: drive and repeat:

```
XCOPY C:\BIN A: /M
```

Those files that had been copied onto the original diskette had had their archive bit turned off, so they will not be copied onto the second diskette. If the "Disk Full"

message is displayed again, repeat the copy operation with fresh target diskettes, until no message is received. However, this technique will not cope with files that are too big to fit on the target diskette, in this case use the BACKUP command instead.

5-154

MS-DOS USER GUIDE

The following examples illustrate recursive copying. The source drives for both examples have their directory structure illustrated in the following figure:

Ds<sup>^</sup>.iDdua

(root)

Aij-

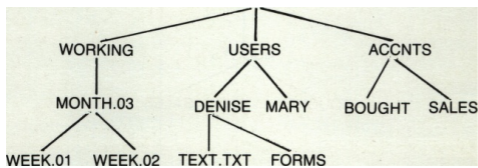


Fig. 5-7 Source Drive Directory Structure

If the target drive directory structure before copying is as illustrated in the following figure:

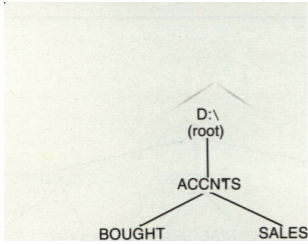


Fig. 5-8 Target Drive Directory Structure Before Copying

Example: The source and the target subdirectory trees have different structures:

```
XCOPY C:\USERS\*.* D: IS
```

The result on the target is to create subdirectories with the same names as those in the source directory tree being copied. All the files that exist in the source subdirectory and its subtree on "C:" will be copied into the directories of the same names on "D:". Note that empty subdirectories will not be created on the target unless the "/E" switch was specified. The resultant



directory structure is illustrated in the following figure:

5-156

## MS-DOS USER GUIDE

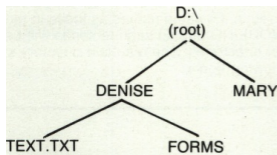


Fig. 5-9 Target Drive Directory Structure After Copying

Example: The source and target drives have the same structure:

```
XCOPY C:\ACCNTS\*.*  
D:\ACCNTS IS
```

The result on the target is to leave the Target Drive Directory Structure the same as before copying. All the files that exist in the source subdirectory and its subtree on C: will be copied into the

directories of the same names on "D;".

Example: Copying From one directory to another:

```
XCOPY C:\WORKING  
D:\ARCHIVE /S/M
```

As the directory "ARCHIVE" does not exist on the target, it will be created. However "XCOPY" does not know whether the

name "ARCHIVE" is intended to be a file or a directory. Therefore you will be asked:

Does ARCHIVE specify a file name or directory name on the target (F = file, 0 = directory) ?

Answer D as "ARCHIVE" is intended to be a directory.

The subdirectory tree of the

directory "WORKING" will be copied and the subdirectories will retain the same names as those in the source directory. All files in WORKING and its subdirectories will be copied to the equivalent target directories. The resultant directory structure is illustrated in the following figure:

D;\

(root)

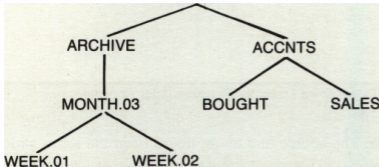


Fig. 5-10 The Target Directory Structure After Copying

Remarks

XCOPY does not provide for copying to or from reserved

device names such as CON;  
or AUX:.

5-158

MS-DOS USER GUIDE

# COMMANDS

XCOPY does not provide for pathnames of longer than 63 characters. For deep trees use SUBST to substitute a logical

drive letter for long pathnames.

XCOPY cannot copy to or from hidden or protected files.

## **Error Codes**

If XCOPY encounters an error, it returns one of the following error level codes;



0 Copy without error

1 No files found to copy

2 CTRL C entered by user to terminate XCOPY

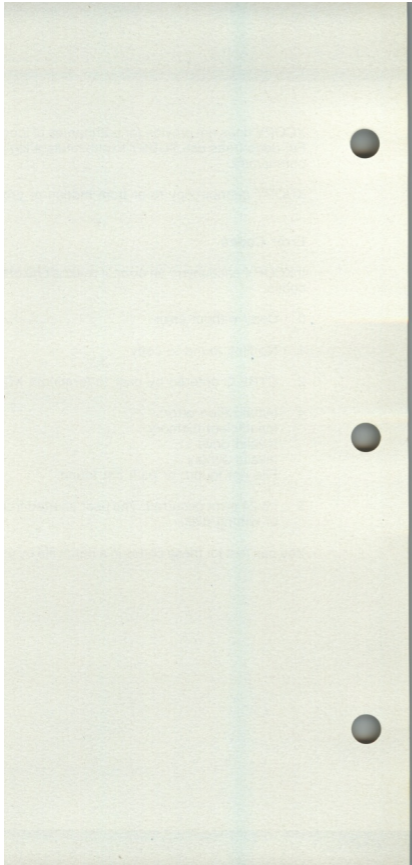
4 Initialization error:

Insufficient memory Invalid  
drive Invalid syntax

File not found, or path not  
found.

5 Int 24 error occurred. The user aborted from INT 24 error reading or writing disk.

You can test for these codes in a batch file by using IF ERRORLEVEL.



# 6. VIDEO FILE EDITOR (EDIT)

# **ABOUT THIS CHAPTER**

This chapter tells you how to use the Video File Editor (EDIT)

## **CONTENTS**

**INTRODUCTION**

**6-1**

HOW TO INVOKE THE VIDEO FILE EDITOR	6-1 iO!
THE DISPLAY	6-3
THE KEYBOARD	6-4
EDIT	6-7

GENERAL EDITING  
FUNCTION KEYS 6-8

WINDOW MOVING  
FUNCTION KEYS 6-  
16

EXITING AND  
SAVING FUNCTION  
KEYS 6-  
18

COMMANDS AND SEARCHING 6-19

STRING SEARCHES 6-19

COMMANDS 6-21

GOTO 6-21



DELETE

6-  
22

FILE

6-  
22

# INTRODUCTION

The Video File Editor

enables you to create and edit files of text. A text file is a file of records containing printable ASCII characters, and each record is separated from the next by a carriage-return/line-feed pair.

The Video File Editor displays a 21-line "window" within which you can perform editing functions via the keyboard. A subset of these functions enables you

to move the window to access any part of the file.

In addition to the functions mentioned above the Video File Editor can also perform an extensive set of line editing and cursor moving functions and can operate in overstrike, insert text or command mode. The latter enables a subset of high level commands.

Each text line in a newly created file can contain up to 80 characters. Existing files created by means other than the Video File Editor can be edited with it, even if its lines extend beyond 80 characters. However, all characters after column 80 are overstruck on column 80.

## **HOW TO**

# **INVOKE THE VIDEO FILE EDITOR EDIT**

The EDIT command is used to enter the Video File Editor.

```
[d:]\path] EDIT [/B][/T][/R]
```

pathname

## Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d	specifies the drive where EDIT is to be found.
---	--

path

specifies the directory where EDIT is to be found.

A backup of the file is to be made when the Video File Editor is entered. This backup is

IS

named  
filename.bak  
where filename  
Is the same as  
that specified In  
the command  
line.

The size of the  
file will be  
minimized by  
automatically  
replacing



n multiple spaces  
with TAB  
characters  
wherever  
possible.

/R The read-only  
option and is  
used when you  
only wish to  
examine the  
contents of the  
file. This

protects the file from accidental damage while examining it.

pathname      The path to the file to be edited.

## **Characteristics**

If the file does not already

exist the prompt "OK to Create?" appears on the screen, to which you must type "Y" to create the file.

The Video File Editor is initially in "overstrike" mode. That is, you can enter text and overwrite whatever is already written on the file. The methods of entry into other modes of operation are described later.

# MS-DOS USER GUIDE

DITOR

## THE DISPLAY

Once the Video File Editor has been invoked the monitor shows a display such as;

file textfile

- ^

Lines Read 10

— + — + — + — + — ^ — +  
-^t + — + — +

-TOP

This file contains text.

BOTTOM



## Fig. 6-1 Video File Editor Screen Layout

Line 1 indicates the file name and the current message.

Line 2 is used for high level commands and search strings and is therefore only used when in command mode.

Refer to the section entitled "Commands and Searching" for details.

Line 3 shows the tab stop settings (4 character positions per tab).

Lines 4 to 24 contain the text window.

Line 25 is not used.

On entering the Video File Editor the beginning and end of the file are marked by two display lines containing the words TOP and BOTTOM,

respectively. The former, known as the TOP bar, always appears immediately before the first line of text in the file. And the BOTTOM bar always appears immediately after the last line of text. They are not actual lines of text and are there merely as markers. The cursor is initially positioned on the TOP bar.

The cursor changes shape



when switching between certain modes of editing. It is represented here as underline.

Note that the screen mode for the Video File Editor is 80x25 lines, even if it is invoked from a terminal set to 40x25 lines.

## **THE KEYBOARD**

The keyboard functions in a different manner once the Video File Editor has been invoked. This provides the means by which the required editing functions are entered. The following tables show for each function key, the function name and the key-stroke combination that executes that function.

Table caption Using the  
Numeric Keypad

KEY- STROKE	FUNCTION KEY NAME
----------------	----------------------

HOME	TOP
------	-----

END	BOTTOM
-----	--------

PGUP	FULL SCREEN UP
------	-------------------

PGDN

FULL SCREEN  
DOWN

CURSOR LEFT

CURSOR RIGHT

t

CURSOR UP

# MS-DOS USER GUIDE

KEY- STROKE	FUNCTION KEY NAME
----------------	----------------------

i	CURSOR DOWN
---	-------------

## **Using the Function Keys**

Note that these function keys are summarized on a

template supplied with your system. Keep this template by your keyboard, for quick reference during working sessions.

KEY- STROKE	FUNCTION KEY NAME
----------------	----------------------

FI	COMMAND MODE
----	-----------------

SHIFT F1 ABORT

F2 RESTORE  
LINES

SHIFT F2 DELETE LINE

F3 JOIN LINES

SHIFT F3 SPLIT LINE

F4                      END OF LINE

SHIFT F4    START OF LINE

KEY-                      FUNCTION KEY  
STROKE                    NAME

F5                      SAVE



SHIFT F5 SAVE AND  
EXIT

F6 NEXT LINE

SHIFT F6 ERASE TO END

F7 GOTO MARK

SHIFT F7 INSERT MARK

F8 SEARCH DOWN

SHIFT F8 SEARCH UP

F9 LINE DOWN

SHIFT F9 LINE UP

F10	HALF SCREEN DOWN
-----	---------------------

SHIFT F10	HALF SCREEN UP
--------------	-------------------

6-6

MS-DOS USER GUIDE

**Using Control Keys**

KEY	KEY FUNCTION
STROKE	NAME

CTRL H	BACKSPACE
--------	-----------

jj

CTRL 1	TAB
--------	-----

CTRL K ERASE TO END  
OF LINE

CTRL L REFRESH

CTRL R RECALL LINE

or BACKSPACE

BS

INS            INSERT MODE

DEL           DELETE CHAR

ESC           ESCAPE

1            TAB

-1

SHIFT REVERSE TAB

INSERT LINE or  
ENTER J EXECUTE  
COMMAND

**GENERAL  
EDITING**

# FUNCTION KEYS

The keys whose functions are described below perform general editing functions such as moving the cursor and inserting and deleting text.



CLASS FUNCTION j|

KEY )

MEANIN

Moves the  
cursor or  
line up the  
screen but  
keeps the  
same  
position  
within the  
line. If the  
cursor w

to f  
move  
the (CURSOR  
cursor UP)

on the  
second li  
of the  
window  
then the  
window i  
moved o  
line up th  
file and t  
cursor  
remains  
the secur  
line.

i

(CURSOR  
DOWN)

Moves the cursor or line down the screen but keeps the same position within the line. If the cursor was on the penultimate line of the window,

stays the  
and the  
window  
moved  
down one  
line.

(CURSOR  
LEFT)

Moves the  
cursor or  
character  
position  
the left

within the  
same line

6-8

# MS-DOS USER GUIDE



CLASS	FUNCTION	MEANING
	KEY	

to  
move  
the  
cursor

(CURSOR  
RIGHT)

Moves the  
cursor or  
character  
position  
the right  
within the  
same line

“◆|3M

or

Moves the  
cursor or  
tab position  
j!(four  
character

CTRL 1 to the rig  
..»/

(TAB)

SHIFT Moves th  
cursor or  
tab posit  
(REVERSE (four  
TAB) character  
to the lef

SHIFT F4 Moves th

(START OF cursor to  
LINE) start of the  
current li

F4

(END OF  
LINE)

Moves the  
cursor to  
character  
position  
immediately  
following  
the last non-  
space  
character



the current  
line.

Is entered  
from  
overstrike  
mode. The  
cursor  
changes  
shape to  
show that  
new mode  
has been

to

INS

entered. .  
character  
which is  
subseque  
entered i  
inserted  
immedia  
before th  
cursor  
position,  
the  
remainde  
the text i  
the line a

insert	(INSERT	the cursor
text	MODE)	are moved
		one
		character
		position
		the right.
		Any
		character
		that was
		the last
		character
		position
		the line is
		discarded

Striking  
INSERT  
MODE k  
second ti  
returns tl  
Video Fi  
Editor to  
overstri  
mode and  
the origi  
cursor is  
restored.

CLASS	FUNCTION	MEANING
	KEY	

		Insert
		blank
		line
		immediately
		after
to	ENTER J	current
insert		and print
text	(INSERT	the contents
	LINE)	the
10 -ijji		beginning
oejyieD ri i .noitieoq		that line

i nmuloo

Subs  
text i  
push  
line o  
the s

If the  
was a  
on th  
botto  
of th  
then

wind  
move  
line o  
the f  
the b  
line i  
inser  
the l  
of th  
wind

Mov  
cursc

to  
delete  
text

^ (or BS) or

CTRL H

(BACKSPACE)

chara  
posit  
the lo  
delet  
chara  
unde  
curs  
Subs  
chara  
the li  
not r  
The c  
chara  
are r



with

This  
func  
usua  
for  
corre  
typir  
error  
enter  
new

DEL

(DELETE  
CHAR)

SHIFT F6

Dele  
chara  
unde  
curs  
shif  
sub  
chara  
In th  
one p  
to th

Dele  
cont

or

CTRL K

(ERASE TO  
END)

the c  
line :  
the c  
curs  
posit  
the e  
the li

6-10

MS-DOS USER GUIDE

# CLASS FUNCTION MEANING

## KEY

to  
delete

SHIFT F2

Deletes  
current line  
and moves

.on-  
»(5(

■■(DELETE  
LINER<sup>^</sup>

subsequ  
text <sup>^</sup> or  
line up  
screen.  
position

text

the cursor

not character  
it remains  
the same

column  
position  
The deletion  
line of text  
is placed  
a holding

area called  
the rest of  
buffer. This  
action  
overwrites  
the previous  
contents of  
the rest of  
buffer  
except  
where  
DELETED  
LINE  
function

.099108 eni  
nwoD

immedi  
follow e  
other, Ir  
which ca  
subsequ  
deleted  
are  
appende  
the buff  
This ena  
you to n  
a block  
text from  
the file

the buffer  
from which  
it can be  
reinserted  
into the  
same or  
another  
using the  
RESTORE  
LINES  
function



to	CTRL R	Restores
restore		contents
		the curro

line to its  
original  
state. The  
contents  
restored  
those that  
existed  
before the

text

(RECALL  
LINE)

cursor w  
moved t  
this line  
Once th  
cursor is  
moved c  
particula  
line the  
contents  
that line  
cannot b  
recalled  
using th  
function

F2

Inserts the  
contents of  
the restore  
buffer

Transfer into  
file starting  
at the third  
line below  
the current  
cursor

(RESTORE  
LINES)

position  
The curs  
Is move  
the start  
the Inse:  
llne(s).  
restore  
buffer it  
is not  
changed  
This  
function  
used in  
conjunc

with the  
DELETE  
LINE  
function  
move ar  
copy blo  
of text.

CLASS FUNCTION MEAN  
KEY

to split

and **SHIFT F3**

join **(SPLIT**

lines of **LINE)**

text

Divide  
current  
into two  
moving  
text un  
and to  
right o  
cursor  
the nex  
The cu  
does no  
move.  
on  
subseq

lines is  
shifted  
line do  
the scr

Combines  
two lines into  
one. The text  
on the  
subsequent  
line is placed  
immediately  
after the last

non-space  
character on  
the current  
line. The  
cursor does  
not move. If  
the current  
line cannot  
accommodate  
the entire text  
of the next  
line then only  
that amount  
which fits is

F3

(JOIN  
LINES)



moved and  
the remaining  
text stays on  
the same line  
but is moved  
to the left  
hand edge of  
the screen.

Causes  
marker  
Inserte

the text  
Immed  
followi  
the cur  
line. T  
marker  
dotted  
contair  
the tex  
'MAR  
the MA  
line wa  
previou  
located

somew  
else in  
text it  
moved  
where  
to the  
positio  
Note th  
this is  
actual  
of text  
will ne  
written  
the file

SHIFT F7

to

insert a (INSERT  
marker MARKER)

placement  
therefore  
only  
significant  
during  
current  
editing  
session  
used in  
conjunction  
with the  
GOTO  
MARK  
function

place n  
(for de  
see the  
section  
entitled  
’’Wind  
Movin  
Functio

Keys’’)  
in  
conjun  
with th  
high-le

comma  
DELETE  
(see the  
section  
entitled  
"Com  
and  
Search

— ^ —  
—  
—————  
^ ————

Table caption6-12

Table captionMS-DOS USER  
GUIDE



CLASS

FUNCTION KEY

# MEANING

to enter

control

characters

ESC

(ESCAPE)

The Video File Editor allows you to enter only the



printable ASCII character set  
(hexadecimal codes 20

to 7E). To force the  
generation of "control"  
codes (hexadecimal 00 to 1F  
and 7F) the ESCAPE  
character must be used.

When you type the ESCAPE  
key a special character (a  
reverse video symbol) is  
placed on the screen. This is  
treated like any any other  
character except that the

following character becomes a control character. This means that only the lower five bits of code are written to the file thereby generating a code in the range 00 to 1F. An exception is the following: to generate a code of 7F you must enter ESC ?; this sets the seventh bit.

To insert the Escape ASCII character (ESC, hexadecimal 1B), type ESC [



# Examples

The following table shows some examples of how text can be modified using the functions discussed above. It assumes a text file called EXAMPLE1 on the B: drive.

Table captionpnoitnniiV “ ’

IF you

The

STEP enter...

screen  
displays...

EDIT B:

EXAMPLE1

The  
purpose  
of this  
text is to  
act as an  
example  
of how to  
use the  
editing  
functions

of the  
Video  
File  
Editor

as an  
example  
of how to  
use the  
editing  
functions  
of the  
Video

CURSOR  
DOWN  
DELETE  
LINE

# File Editor

as an  
example  
of how to  
use the  
editing  
functions  
of the  
Video  
File

CURSOR  
UP  
  
ENTER

Editor

This is \_

as an  
example  
of how to  
use the  
editing  
functions  
of the  
Video  
File

This is  
SPACE

3

## JOIN LINES

Editor

This Is <sup>^</sup>  
an  
example  
of how to  
use the  
editing  
functions  
of the  
Video  
File  
Editor



5  
DELETE  
CHAR  
DELETE  
CHAR  
DELETE  
CHAR

This Is an  
example  
of how to  
use the  
editing  
functions  
of the  
Video  
File  
Editor

6-14



STEP IF you enter... The screen displays.

This is an example of how to use Uie editing

6 NEXT LINE

## DELETE LINE

functions  
of the  
Video Fi  
Editor

^ This is  
an  
example  
of how to  
use the  
Video Fi  
Editor

8

RESTORE  
LINES NEXT  
LINE

This is an example of how to use the Video File Editor to edit the functions of

This is an example

9

T

of how to  
use the  
Video Fi  
Editor

The  
editing  
function  
of

This is a  
example

10

END OF  
LINE

of how to  
use the  
Video Fi  
Editor

The  
editing  
functions  
of\_

This is a  
example  
of how to  
use the

11 BACKSPACE Video Fi  
Editor

BACKSPACE

The  
editing  
functions

This is an  
example  
of how to  
use the

12 RECALL  
LINE

Video Fi

Editor the  
editing  
functions  
of

This is an  
example  
of how to  
use the  
Video Fi  
Editor

13 SPLIT LINE

the



editing  
functions:

of

STEP IF you  
enter...

The screen  
displays...

This is an  
example of  
how to use

CURSOR

14 UP the Video  
File Editor\_

the editing  
functions of

15 INSERT  
LINE This is an  
example of  
how to use  
the Video  
File Editor

the editing  
functions of

Note: To delete a character in the 80th column you should move the cursor to that position in overstrike mode and type SPACE .

# WINDOW MOVING FUNCTION KEYS

The function keys described in the following table enable you to move the window up and down the file.

FUNCTION KEY	MEANING
-----------------	---------

HOME	
------	--

	Moves the window to the top of the text file. The
--	---

(TOP)	
-------	--

	cursor is placed on the top bar of the file.
--	--

END

Moves the window to the end of the file. The cur

(BOTTOM) sor is placed on the last line of text.

6-16 ?

# MS-DOS USER GUIDE

FUNCTION KEY	MEANING
-----------------	---------

PG UP  (FULL	Causes the window to be moved up the file by 20 lines. This allows one line of overlap between the old
--------------------	--

SCREEN  
UP)

and new  
displays. The  
cursor remains  
on the same  
screen line.

PG DN

(FULL  
SCREEN

Causes the  
window to be  
moved 20 lines  
down the file.  
This allows one  
line of overlap  
between the old



DOWN)

and new displays. The cursor remains on the same screen line.

SHIFT F10

(HALF  
SCREEN  
UP)

Causes the window to be moved half a screen (10 lines) up the file. The cursor

remains on the same screen line.

F10

(HALF  
SCREEN

DOWN)

Causes the window to be moved half a screen (10 lines) down the file. The cursor remains on the same screen line.

SHIFT F9  
(LINE UP)

Causes the window to be moved one line up the file. The cursor remains on the same screen line.

Causes the window to be moved one line

F9

(LINE  
DOWN)

down the file.

The cursor  
remains on the  
same screen  
line.

F6

(NEXT  
LINE)

Moves the  
window one  
line down the  
file and places  
the cursor at the  
start of the next

text line.

F7

(GO TO  
MARK)

Moves the window up or down the file such that the cursor lies on the MARK line.

# EXITING AND SAVING FUNCTION KEYS

Table caption The function keys described in the

following table enable you to exit from the Video File Editor and/or save the file you have been working

on.

FUNCTION KEY	MEANING
-----------------	---------

Causes the

SHIFT F5

(EXIT AND  
SAVE)

revised text to be written back to the file and the Video File Editor to be terminated. The screen is erased and control is returned to MS-DOS.

Causes the revised text to



F5

(SAVE  
TEXT)

be written to the file. The Video File Editor does not terminate.

SHIFT F1  
(ABORT)

Causes the Video File Editor to terminate without writing the revised text to the file.

If text has been altered or added since starting the editor you are asked to "Confirm Abort?". To confirm press Y. Any other action causes the Video File Editor to ignore

the ABORT.  
Control Is  
returned to MS-  
DOS.

6-18

MS-DOS USER GUIDE

# COMMANDS AND SEARCHING

The second line of the screen (above the scale line) is called the editor command line and is used for entering high level commands and search strings.

To enter text on the editor command line you must first press the COMMAND MODE function key. This moves the cursor to the second line. You can now enter text there. All line editing operations such as INSERT MODE, BACKSPACE and DELETE CHAR - now apply to the editor command line. The RECALL LINE function when used in command mode

restores the editor command line to its previous contents. The ENTER key performs EXECUTE COMMAND when used in this mode.

Repeating the COMMAND MODE key returns the cursor to the next window without performing any command operation.

## STRING SEARCHES

This feature enables you to search the file for a particular combination or characters. Before searching for a text you must enter command mode by striking the **COMMAND MODE** function key. Then enter the text to be searched for followed by the appropriate function key, as described in the following table:

# FUNCTION MEANING KEY

F8  (SEARCH DOWN)	Searches for the text string starting from the the current cursor position and moving down the file until the first occurrence of the string is
----------------------------	---



encountered. If found, the window and cursor are moved to it.

Searches for the text string starting from the cursor position and moving up the file. If the

SHIFT F8  
(SEARCH

UP) string is found then the window and cursor are moved to it.

## Examples

The following table shows some examples of the use of the searching functions. It assumes a text file called

EXAMPLE2 on the B: drive.

When you enter at the A>

prompt: EDIT B:EXAMPLE2

If you

enter on

the editor

command

line...

Then

strike

function

key...

The screen

dispiays..

This is an

example

F1

how to use  
the search  
function  
keys of the  
Video File  
Editor to  
find a  
particular  
combination  
of  
character

This is an

tunc

SEARCH  
DOWN

example  
how to use  
the search  
function  
keys of the  
Video File  
Editor to  
find a  
particular  
combination  
of  
character

e SEARCH  
SPACE of UP

This is an  
example  
how to use  
the search  
function  
keys of the  
Video File  
Editor to  
find a  
particular  
combination  
of  
character

6-20

## MS-DOS USER GUIDE

### COMMANDS

The Video File Editor commands are a set of special commands that enable you to perform a number of high level functions. Before entering a command you must press the

COMMAND MODE function key (Ft) to move the cursor to the command line. You can then enter the command which is subsequently displayed on the editor command line. To execute the command you must then press the ENTER key. If you decide not to execute the command press F1 again to return the cursor to the edit text.



# GOTO

This command enables you to move the window to a specific line number in the file.

GOTO line

Where

# SYNTAX

## ELEMENT MEANING

line

A decimal integer that is the desired line number in the file. If this number is greater than the number of lines in the file then the window is

moved to the end  
of the file.

## Characteristics

Each line of the text file is automatically numbered.

That is, the first line of the file is line 1, the TOP bar is line 0 and the MARK bar does not count.

# DELETE

This command removes all text between the current line and the MARK line and places the removed text in the restore buffer from where it can be re-inserted at will. If the MARK line does not exist an error message is given.

DELETE



# FILE

The FILE command allows you to suspend processing of the current file and invoke the editor on another file.

When editing of the new file is terminated by a SAVE AND EXIT or ABORT function, the old file is recalled at the point at which it was exited.

FILE pathname

6-22

MS-DOS USER GUIDE

# VIDEO FILE EDITOR

Table caption Where

SYNTAX  
ELEMENT

MEANING

pathname

The path of  
the new file



to be edited.

## Characteristics

The command line option flags (/B, /T or /R) used by the old file remain the same for the new file.

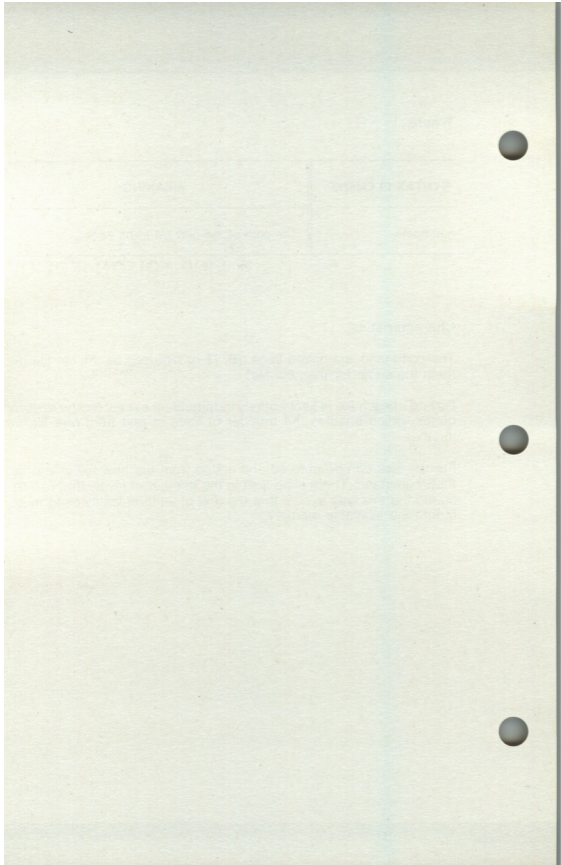
Editing of each file is kept entirely independent except for the restore buffer, which

enables the transfer of lines of text from one file to another.

Further files can be entered and edited from the new file using the FILE command.

There is no limit to the number of levels that can be created in this way except that the text of all the files invoked must fit into the available memory.

---



# ABOUT THIS CHAPTER

This chapter tells you how to use the Line Editor EDLIN.

## CONTENTS

INTRODUCTION 7- W (WR  
1 LINES)

HOW TO  
INVOKE THE 7- INTRA-  
EDLIN 1 COMM  
PROGRAM

COPY1

EDLIN	7- 2	COPYT
-------	---------	-------

INTER-LINE COMMANDS	7- 5	COPYL
------------------------	---------	-------

line (EDIT LINE)	7- 6	SKIP1
------------------	---------	-------

A (APPEND LINES)	7- 8	SKIPTC
---------------------	---------	--------

C (COPY LINES) 7- KILL  
9

D (DELETE  
LINES) 7- INS  
12

E (END  
EDITING) 7- NEWTE  
15

1 (INSERT LINES)	7- 16
---------------------	----------

L (LIST TEXT)	7- 20
---------------	----------

M (MOVE LINES)	7- 23
-------------------	----------

P (PAGE)	7- 25
----------	----------



Q (QUIT EDITING)	7- 26
---------------------	----------

R (REPLACE TEXT)	7- 27
---------------------	----------

S (SEARCH TEXT)	7- 30
--------------------	----------

T (TRANSFER	7-
-------------	----

LINES)

34

# INTRODUCT

The Line Editor (EDLIN) can be used to edit files that contain lines of text, where each line is a maximum of 255 characters, the last of which must be ENTER. The files can contain ASCII text or source programs.

Within EDLIN, two types of commands may be used:

- Commands that enable you to perform editing operations on specified lines, a range of lines or an entire file in order to:

- list, edit, delete and insert lines of text

- search for a specified text string

- search for and replace a

specified text string

- create, edit and save new files

- edit an existing file, save the modified file and keep a back-up of the original file

These commands are termed "inter-line" commands.

- Commands that enable you to perform editing operations

within a line of text. These are termed "intra-line" commands and utilize the source line facility as described in Chapter 2.

The control keys described in Chapter 2 can also be used within EDLIN.

In the disk files, the lines of text are not numbered. But when a file is displayed, lines are numbered dynamically.

When you create or edit a file, line numbers begin at 1 and are incremented by one through to the end of the file. If you insert new lines between existing lines, all line numbers following the inserted text are automatically incremented by the number of lines inserted. When lines are deleted, all line numbers following the deleted text are decremented automatically

by the number of lines deleted. Consequently, lines are always numbered consecutively, starting from 1, through to the last line in the file.



# HOW TO INVOKE THE EDLIN PROGRAM

# EDLIN

The line editor (EDLIN) is invoked as follows:

[d:]\`path`] EDLIN filespec

Where



SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the drive where EDLIN is to be found.

path

Specifies the directory where EDLIN is to be found.

The file specifier

filespec

of the file to be edited. This must include the drive specifier unless you want to default to the disk containing the EDLIN command, In which case you need to remove any writeprotection.



7-2

## MS-DOS USER GUIDE

### Characteristics

When you invoke EDLIN, the back-up file to the file you specify (if it exists) is erased to ensure there is sufficient room on disk for the output file.

If the file specified exists and is smaller than 75% of the available memory, then the entire file is loaded into memory and the following is displayed:

End of input file \*

You can then edit the file.

Note that the EDLIN prompt is an asterisk (\*), and the cursor is an underline (  ).

In the file specified exists and is larger than 75% of the available memory, then only the first part of the file is loaded, until 75% of the

available memory is full. The EDLIN prompt (\*) and cursor ( ) will then

appear but not the "End of input file" message. You can then edit that part of the file loaded into memory. To

access unloaded lines you must use the Write Lines and Append Lines commands described later in this chapter.

If the specified file does not exist on the drive then a new file is created with the specified name. But note that the drive you wish the output file to be written to must be specified when you invoke EDLIN, otherwise the output



file will be written to the default drive. The following message is displayed:

New file

\*

You can then begin to create the file.

There are two edit commands that can be used to terminate the edit session:

- End Edit, which terminates EDLIN, renames the input file filename.BAK and writes the edited file in memory to the output file which is given the same name as the input file. See the "E (END EDIT)" command later in this chapter.

- Quit Edit, which terminates EDLIN without creating a back-up or an output file. The input file remains unchanged.

See the "Q (QUIT EDIT)" command later in this chapter.

Note that a file with the extension .BAK cannot be edited. Any attempt to do so will generate the message:

Cannot edit .BAK file-  
rename file

You must rename the file using the RENAME

command (See Chapter 5), then invoke EDLIN on the renamed file.

If, when attempting to create a new file, the following message appears:

No room in directory for file  
then either:

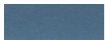
- the file directory is full, or

- you have specified an illegal disk drive or file name.

The latter can be checked by examining the command line. (If the command line is no longer on the screen it can be recalled using the F3 (CQPYLINE) edit key.) To check the former you can run the CHKDSK command on the specified disk drive. See Chapter 5 for details.

7-4

# MS-DQS USER GUIDE



# INTER-LINE COMMANDS

This section describes the EDLIN commands that operate on entire lines of text.

The lines you wish to work on may be specified either by entering a line number as a

parameter to the command, or by entering a period (.). The latter indicates that the "current line" is to be worked on.

The current line is the location of the last change to the file. It is not necessarily the last line displayed. The current line is indicated by an asterisk between the line number and the first text character. For example:



15:\* This is the current line

Each command description summarizes the purpose of the command, defines the command syntax and explains each syntax element. This is followed, for each command, by a detailed account of the command characteristics and some working examples.

Remarks

1. Commands can be entered in either upper or lower case

2. Command keywords and command parameters can be separated from each other by spaces or commas for readability but need not be, except where two line numbers are entered as parameters, in which case they must be separated by a comma or space. For brevity the syntax of this chapter will

always indicate comma where separation is obligatory, but note that a space can alternatively be used

3. Commands only become effective after entering ENTER

4. If you make a syntax error when entering a command the message "Entry Error" will be displayed. You must

re-enter the command using  
the correct syntax

line (EDIT  
LINE)

Enables you  
to edit a  
specified line.

[//ne| . ]

Where

SYNTAX  
ELEMENT

MEANING

line

The number  
of the line to  
be edited.

The current

line is to be edited.

## Characteristics

When you enter a line number followed by ENTER EDLIN displays the line number and the corresponding text, then, on the next screen line, reprints the line number followed by the EDLIN prompt (\*) and

the cursor ( ) . The displayed line serves as the source line and is

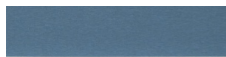
ready to be edited.

If, instead of a number, you enter a period (.), the current line is displayed and ready for editing. If you enter ENTER without a line number or a period, then the line immediately following the current line is displayed,

unless the current line is the last one in the file, in which case the edit prompt (\*) will re-appear.

## MS-DOS USER GUIDE

7-6



To edit the line you can use any of the control or edit (intraline) keys, or re-enter



the entire line by entering text.

If you wish to abort the operation without changing the source line, press CTRLC. Pressing ENTER with the cursor at the start of the line also aborts the operation without changing anything.

To save the edited line and overwrite the original line, type with the cursor at the

end of the edited line. If you type with the cursor in any position other than the start or end of the line, text to the left of the cursor will be written to the file in memory, but all text under and to the right of the cursor will be lost.

## Example

Assuming that the contents of the current edit file are as

follows:

1: This file demonstrates how  
2: the line command can 3:  
be used to edit line 4: four.

IF you  
type...

THEN EDLIN  
displays...

4 ENTER

4:\*four.

4:\*  
\_

INS

4:\*four.

number

SPACE INS

F3

4:\*number  
four.\_

ENTER

\*

# A (APPEND LINES)

Table caption Adds lines from the input file on disk to that part of the file currently

in  
memory.

[n] A

Where

SYNTAX MEANING  
ELEMENT

n

The number of lines to be added to the file in memory from

the input file on disk.

## Characteristics

This command is only useful for files that are too large to fit into the available memory.

When EDLIN is invoked on a file that is too large to fit into memory, it loads as many



lines as possible (as much as will fit into 75% of the available memory). Before using the Append command it is therefore necessary to write some lines of text to the output file on disk by means of the Write Lines command (see later).

If you enter A without a parameter, lines are appended from the disk file until the available memory is

75% full, or until there are no more lines to append.

7-8

## MS-DOS USER GUIDE

Any attempt to read beyond the end of the disk file will yield the following message:

**End of input file**

# Example

IF you  
enter...

THEN...

the next 100 lines  
of the input file on  
disk are read into  
memory and  
appended to that  
part of the file that  
is already there.

100 A



# C (COPY LINES)

Copies a range of lines to a specified line.

[line-a] , [line-b] , line-c[ ,  
count]C

## Where

SYNTAX  
ELEMENT MEANING

line-a                      The first line in  
the range to be  
copied.

SYNTAX  
ELEMENT MEANING

line-b

The last line in the range to be copied.

line-c

The line at which the copied lines are to start.

count

The number of times the range is to be copied.

# Characteristics

If line-a is omitted, then the first line defaults to the current line.

If line-b is omitted, then the last line defaults to the current line.

If both line-a and line-b are omitted, then only the current line is copied.



Following the copy operation, lines that previously followed line-c are moved to follow the copied block.

If line-c is beyond the current end of file then the lines are copied to line numbers contiguous to the end of file.

line-b must be greater than or equal to line-a.

On completion line-c becomes the current line.

# Examples

Assuming the contents of the current edit file are as follows:

```
1:*This is a sample file 2; to  
demonstrate the use 3: of the  
Copy lines command.
```

7-10

# MS-DOS USER GUIDE

— — · ·

IF you THEN the edited  
enter... file becomes...

1: This is a sample  
file

2: to demonstrate  
the use

3: of the Copy lines  
command.

1,3,4C

4:\*This is a sample  
file

5: to demonstrate  
the use

6: of the Copy lines  
command

1: This is a sample  
file

2: to demonstrate  
the use

3: of the Copy lines  
command.

4: This is a sample  
file

„27,3C 5: to demonstrate

the use

6: of the Copy lines  
command.

7:\*This is a sample  
file

8: This is a sample  
file

9: This is a sample  
file

1: This is a sample file

2: to demonstrate the use

3: of the Copy lines command.

4: This is a sample file

5: to demonstrate the use

2,3,8C 6: of the Copy lines  
command.

7: This is a sample  
file

8:\*to demonstrate  
the use

9: of the Copy lines  
command.

10: This is a  
sample file



11: This is a  
sample file



# D (DELETE LINES)

Deletes all  
lines within a  
specified  
range.

[line-a][,line-

b] D

Where

SYNTAX  
ELEMENT

MEANING

line-a

The first line  
in the range  
to be deleted.

line-b

The last line  
in the range  
to be deleted.

## Characteristics

If line-a is omitted, then the first line defaults to the current line.

If line-b is omitted, then the specified line only is deleted.

If D is entered alone, then only the current line is deleted.

After the command has been executed the numbers of the lines following the deleted section are changed to follow the numbers of the lines preceding the deleted section.

7-12

MS-DOS USER GUIDE

The line immediately following the deleted section becomes the current line.

line-b must be greater than (or equal to) line-a .

## **Example**

Assuming the contents of the current edit file are as follows:

1: This is a sample file 2: to  
demonstrate the use 3: of the  
Delete command.

4: "line-a D" deletes just the  
specified line 5: if ",line-b  
D" is specified then 6: all  
lines from the current line 7:  
to line-b, inclusive, are  
deleted.

8: Specifying both line-a and  
9: line-b causes that range  
10: to be deleted.

11: Specifying D alone  
deletes the current line.

12: The line subsequent to  
the 13: deleted line(s)  
becomes the 14: current line.

IF you THEN the edited file  
enter... becomes...

1: This is a sample  
file



2: to demonstrate the use

3: of the Delete command.

4: "line-a D" deletes just the specified line

8,10 D 5: if ",line-b D" is specified then

6: all lines from the current line

7: to line-b,  
inclusive, are  
deleted.

8:\*Specifying D  
alone deletes the  
current line.

9: The line  
subsequent to the

10: deleted line(s)  
becomes the

1 1: current line.

IF you THEN the edited file  
enter... becomes...

1: This is a sample  
file

2: to demonstrate the  
use

3: of the Delete command.

4: \*if ”,line-b D” is specified then

5: all lines from the current line

4 D

6: to line-b, inclusive, are deleted.

7: Specifying D

alone deletes the  
current line.

8: The line  
subsequent to the

9; deleted line(s)  
becomes the

10: current line.

1: This is a sample  
file

2: to demonstrates  
the use

3: of the Delete  
command.

4:\*Specifyng D  
alone deletes the  
current line.

5: The line  
subsequent to the

6: deleted line(s)

,6 D

becomes the

7: current line.

1: This is a sample  
file

2; to demonstrate the  
use

3: of the Delete  
command.

D 4:\*The line  
subsequent to the

5: deleted line(s)  
becomes the

6: current line.



# **E (END EDITING)**

Exits EDLIN and saves the edited file on disk.

E

## **Characteristics**

The edited file is written to the drive selected when EDLIN was invoked, or to the default drive if no drive was specified. The input file is renamed "filename.BAK". If the file was created during the editing session, no back-up file is created.

You must make sure that enough free space is available on disk to take the output file, otherwise only a

portion (at most) will be saved. The remainder will be lost and the message:

# **Disk full-write not completed**

will be displayed, and EDLIN will exit.

Table captionExample

IF you THEN...  
enter...

The current edit file  
is saved on the  
diskette specified  
when EDLIN was  
invoked. The input  
file is renamed  
"filename.BAK",  
and EDLIN is exited.

E

# I (INSERT LINES)

Allows you to insert lines of text before the specified line number.

[line I. I # ] I

Table captionWhere

SYNTAX  
ELEMENT

MEANING

line

The number  
of the line  
before which  
subsequently  
entered text  
is to be  
inserted.

Subsequent

•

text is to be inserted before the current line.

#

Subsequent text is to be appended to the file.

## Characteristics

If a line number is entered, then subsequently entered text is inserted immediately before the specified line. If, instead of a line number you enter a period {.), or you enter I on its own, then subsequent text is inserted before the current line.

Moreover, if you enter # instead of the line number, text will be appended to the file.



## MS-DOS USER GUIDE

EDLIN remains in insert mode until either CTRL Z or CTRL C is entered. While you are in insert mode, successive line numbers appear each time ENTER is pressed.

When you exit insert mode, the line immediately

following the last inserted line becomes the current line. The line numbers of this and all subsequent lines are automatically incremented by the number of lines inserted.

When a file is created, you must enter I before entering text. In this case the line number is 1.

# Example

Assuming that the contents of the current edit file are as follows:

1: This is a sample file 2: to  
demonstrate 3: using the  
Insert Lines 4: command

IF you  
enter..

THEN...

3 I

EDLIN enters  
insert mode and  
displays:

3:\*  
\_

EDLIN displays:

how lines 3:\*how lines of

of text            text

can be  
inserted  
into a file        4:\*can be  
                         inserted into a  
                         file

at a  
specified  
line  
number            5:\*at a specified  
                         line number

6:\*\_

IF you THEN...  
enter...

CTRLZ EDLIN exits insert  
mode.

L The List command  
(see later) is  
invoked and EDLIN

displays;

1: This is a sample  
file

2: to demonstrate

3: how lines of text

4: can be inserted  
into a file

5: at a specified line

number

6:\*using the Insert  
Lines

7: command

1  
or  
EDLIN enters insert  
mode and displays:

.1  
6:\* \_

EDLIN displays:



or  
before  
the  
current  
line

6: or before the  
current line

7:\*  
—

CTRLZ EDLIN exits insert  
mode.

L            The List command  
              is invoked and  
              EDLIN displays:

1: This is a sample  
file

2: to demonstrate

3: how lines of text

4: can be inserted  
into a file

5: at a specified line number

6; or before the current line

7:\*using the Insert Lines

8: command



# MS-DOS USER GUIDE "" ^ ' V

Table caption7-18

IF you enter...	THEN...
-----------------	---------

# 1	EDLIN enters
or	Insert mode and displays:

9

9:\*  
\_

and how  
lines of

EDLIN displays:

9: and how lines  
of

text can  
be

10: text can be  
appended

appended  
to a file.

11: to a file.

12:\*

CTRLZ

EDLIN exits  
insert mode.

The List  
command Is  
invoked and  
EDLIN displays:

1: This is a  
sample file

2: to demonstrate

3: how lines of  
text

4: can be inserted  
into a file

L

5: at a specified  
line number

6: or before the  
current line

7: using the Insert  
Lines

8: command

9: and how lines  
of

10: text can be  
appended

11: to a file.



# L (LIST TEXT)

Table caption Displays a  
specified range of lines.

[line-a][ ,  
line-b] L

Where

# SYNTAX MEANING ELEMENT

line-a                      The first line in  
the range to be  
listed.

line-b                      The last line in  
the range to be  
listed.

## Characteristics

if you specify both line-a and line-b , then the entire range of lines is displayed, unless this is in excess of 23 lines, in which case the display starts from line-a , but this and subsequent lines are scrolled off the top of the screen until line-b appears on the 23rd line.

If line-a is omitted, but line-b is specified, then the display starts 11 lines before the current line and ends at line-b. If this is more than 23 lines, the screen scrolls down the file until line-b appears on the 23rd line.

7-20

MS-DOS USER GUIDE

# LINE EDITOR (EDLIN)

If line-b is omitted, but line-a is specified, then 23 lines are displayed, starting from line-a.

If you enter L on its own,

then the current line appears on the center line of the screen (unless the current line is less than line 12), with the preceding 11 lines displayed before it, and the subsequent 11 lines displayed after it. If the current line is before line 12, EDLIN displays the first 23 lines.

## **Example**

Assuming the contents of the current edit file are as follows:

1: This is a sample file 2: to demonstrate the 3: use of the List 4: command

14: 'This is the current line

23: The List command can be 24: used to examine 25: different parts of the 26: file, up to 23 lines 27: at once.

IF you  
enter...

THEN EDLIN  
displays...

2,4 L

2: to demonstrate  
the

3: use of the List

4: command



IF you  
enter...

THEN EDLIN  
displays...

,24 L

3: use of the List

4: command

14:\*This is the  
current line

23: The List

command can be

24: used to  
examine

24: used to  
examine

24 L 25: different parts  
of the

26: file, up to 23  
lines

27: at once.

L

3: use of the List

4: command

14:\*This is the  
current line

24: used to  
examine

25: different parts  
of the

# MS-DOS USER GUIDE

7-22

f

**LINE  
EDITOR  
(EDLIN)**

# M (MOVE LINES)

Moves a range of lines to a specified line.

[line-a] , [line-b] , [line-c] M

Where

# SYNTAX MEANING ELEMENT

line-a                      The first of the  
range of lines to  
be moved.

line-b                      The last of the  
range of lines to  
be moved.

line-c

The line to  
which the text is  
to be moved.



# Characteristics

If line-a is omitted, then the first line defaults to the current line.

If line-b is omitted, then the



last line defaults to the current line.

line-b must be greater than or equal to line-a.

If line-a is omitted, line-b can be specified as the relative number of lines forward of the current line, by preceding the number with a +.

Following the move, lines are renumbered depending on the direction of the move. For instance, moving lines 10 to 20 to line 100 would effectively delete lines 10 to 20 (thereby causing all subsequent lines to be moved up the file 11 lines), then the moved lines would become lines 79 to 99.

The first of the moved lines becomes the current line.

# Examples

Assuming the contents of the current edit file are as follows:

1: This is a sample file 2: to demonstrate the use 3: of the Move lines command.

4: New first line

100: Next line.

IF you enter...

THEN the edit file becomes...

1,3,100 M

1: New first line

97: This is a sample file 98:  
to demonstrate the use 99: of  
the Move lines command.

100: Next line.

7-24

# MS-DOS USER GUIDE

# P (PAGE)

Lists a specified range of lines.

[line-a][ , line-b] P

Table captionWhere

SYNTAX	MEANING
ELEMENT	

line-a

The first of the range of lines to be displayed.

line-b

The last of the range of lines to be displayed.

# Characteristics

If line-a is omitted, then the

first line defaults to the line following the current line.

If line-b is omitted 23 lines are displayed.

The last line displayed becomes the current line.

The difference between the Page command and the List command is that the Page command changes the current line.



# Q (QUIT EDITING)

Quits the editing session but does not save any changes you have made.

Q

## Characteristics

After entering the Quit command EDLIN replies with the message:

**Abort edit  
(Y/N)?\_**

Pressing N or any key other than Y or CTRL C continues the editing session. Pressing Y terminates the editing session. No BAK file is

created, and any changes made during the editing session are lost. The file on disk remains exactly as it was when EDLIN was invoked. Note also that any previous BAK file is also lost since the current BAK file is always deleted when EDLIN is invoked.

Table captionExample

IF you THEN...  
enter...

the editing session is terminated without saving the changes made during the editing session.

Q

MS-DOS USER GUIDE

7-26

# R (REPLACE TEXT)

Searches a specified range to replace all occurrences of one string with another string.

[line-a][,line-b] [ ? ] R

string-a[ CTRL Z string-b]

Where

SYNTAX  
ELEMENT MEANING

line-a                      The number of  
                                 the first line in  
                                 the range on  
                                 which the  
                                 Replace Text

command is to  
be executed.

line-b

The number of  
the last line in  
the range.

For each  
occurrence of  
the specified  
string the

O.K.?

?

prompt appears,  
enabling you to  
accept or reject  
the replacement.

1

string-a

The string of  
characters that  
are to be  
replaced.



string-b

The string of characters that are to replace string-a.

# Characteristics

For each line in which a replacement occurs the modified line is displayed on

the screen. If you entered the  
? parameter then the prompt:

**O.K.?**

will appear after each  
replacement. You must then  
enter Y or ENTER to confirm  
the replacement, or strike any  
other key to reject it. In  
either case the search will  
recommence for the next  
occurrence of string-a. If ? is

not specified, all occurrences of string-a will be replaced by string-b without confirmation.

If you omit string-b, then all occurrences of string-a are deleted.

If you omit line-a then the search will begin from line 1. If you omit line-b then the search will continue to the end.of the file in memory. If

neither line-a nor line-b is entered, then the entire file in memory will be searched and modified.

Once all replacements have been made, the Replace Text command terminates and the last line in which string-a occurred becomes the current line.

If the replacement string causes a line to expand

beyond the limit of 254 characters then the following message is displayed:

**Line too long**

**Example**

Assuming that the contents of the current edit file are as follows:

1: This is a sample file 2: to demonstrate the 3: use of the Replace Text 4: command.

5: Using this command a 6: specified group of characters 7: can be replaced by 8: another group of characters 9: and can be deleted entirely.

7-28

MS-DOS USER GUIDE

- IF you enter...

THEN...

5,8 R  
group of  
characters  
CTRL Z  
string

all occurrences  
of "group of  
characters" in  
the range line 5  
to line 8 are  
replaced with  
"string".

EDLIN displays:

6: specified  
string

8: another string

and line 8  
becomes the  
current line

each occurrence  
of " and " within



? Rand the entire file  
CTRL Z can optionally  
or be replaced with  
"or". EDLIN  
displays:

4: commor.

O.K.?

• N

5: using this  
commor a

O.K.?

N

9: or can be  
deleted entirely?

O.K.?

\*

Y

and line 9  
becomes the  
current line

L

the List  
command Is  
executed to  
display the file  
and enable you  
to see the  
changes you  
have made.  
EDLIN displays:

1: This is a  
sample file

2: to  
demonstrate the

3: use of the  
Replace Text

4: command.


- 5: Using this  
command a

6: specified  
string

7: can be  
replaced by

8: another string

9: \*or can be  
deleted entirely.



# S (SEARCH TEXT)

Searches a specified range of lines for a specified string.

[line-a][ , line-b] [ ? ] S  
string

Where

# SYNTAX MEANING ELEMENT

line-a

The number of the line from which the search is to start.

line-b

The number of the last possible line to be searched.

On finding a  
matching string  
the;

O.K.?

?

prompt Is to be  
displayed,  
thereby enabling  
you to accept or  
reject the  
particular



occurrence.

string

The string of characters to be searched for.

7-30

MS-DOS USER GUIDE

**Characteristics**

The command searches the range line-a to line-b for the specified string. If line-a is not specified then the search begins from the line after the current line. If line-b is not specified then the last line in the range is the last line of the file in memory.

When a matching string is found, the corresponding line is displayed. Then if the ? parameter is not specified,

the search is terminated and the displayed line becomes the current line. If ? is specified, then the displayed line will be followed by the prompt:

**O.K.?**

To accept the string you must enter either Y or ENTER .

The search will then terminate and the displayed

line becomes the current line. Striking any other key, however, will re-commence the search for the next occurrence of the string.

If no matching string is found, or if ? is specified and all matching strings are rejected, the message:

**Not found**

is displayed.

# Example

Assuming that the contents of the current edit file are as follows:

1: This is a sample file to demonstrate  
2: The use of the Search Text command.

3: The search can either

display the 4: first occurrence of a specified string 5: and terminate, or, if specified 6: to do so, it will enable you to 7: interactively examine each 8: occurrence of a string allowing 9: you to confirm or reject the string.

10: Once a string is accepted the 11:\*search terminates.

IF you THEN...  
enter...

the Search Text  
command will search  
lines 2 to 5 of the  
file, Inclusive, for  
the first occurrence  
of the string  
"string", then

2,5 display the  
Sstring following:

4: first occurrence of  
a specified string

and terminate. Line 4  
becomes the current  
line.

the search will be  
made on lines 5 (one  
after the current line)  
to 8, inclusive, for  
the first occurrence  
of "string". EDLIN



,8 will display:

Sstring

8: occurrence of a  
string allowing

and the search will  
terminate with line 8  
as the current line.

the Search Text  
command searches  
the file for the first

1?

Sstring

occurrence of  
''string''. The search  
starts from line 1.  
The result is:

4: first occurrence of  
a specified string

O.K.?

The string is rejected  
and the search  
continues for the

N next occurrence. The result is:

8: occurrences of a string allowing

O.K.?

Y The search is terminated and line 8 becomes the current line. ^

## MS-DOS USER GUIDE

IF you            THEN...  
enter...

the Search Text  
command searches  
for the string  
"sample file"  
starting from line 9

Sample  
file

(one after the current line), up to the end of the file.

The string is not found, hence the message:

Not found

Is displayed. The search terminates and line 8 remains the current line.

# T (TRANSFER LINES)

Inserts an entire file before a specified line of the current edit file.

[line] T [filespec]

# Table caption Where

SYNTAX  
ELEMENT

MEANING

line

The number  
of the line  
before which  
the file is to  
be inserted.

filespec

The file to be inserted.

## Characteristics

If the line parameter is omitted then the current line is assumed. The specified file must be in the same directory as the edit file.



# MS-DOS USER GUIDE

# W (WRITE LINES)

Writes a specified number of lines from the file being edited in memory to the output file on disk.

[n] W

Where

SYNTAX MEANING  
ELEMENT

The number of lines to be written to diskette, starting from line 1. If this parameter is omitted, then

n

lines of text are written to the output file until the available memory is 25% full.

## Characteristics

The Write Lines command is used in conjunction with the Append Lines command when editing files that are

too large to fit into the available memory. Lines written to the output file are deleted from memory, and the remaining lines renumbered, starting from line 1. This leaves space available at the end of the file in memory for additional lines to be read from the input file on disk using the Append Lines command.

# INTRA-LINE COMMANDS

The intra-line commands are executed using the special editing keys that can be used to perform edits within the current line taking advantage of the source line facility.

They enable you to:

- copy one character from the source line to the current line (COPY1)

- copy a specified portion of the source line to the current line (COPYTO)

- copy all remaining characters in the source line to the current line (COPYLINE)

- delete a specified character

in the source line (SKIP1)

- delete a specified portion of the source line (SKIPTO)

- kill the current input and delete the source line (KILL)

- enter insert mode to insert text into the current line (INS)

- exit insert mode (enter overstrike mode) (INS)



- make the current line the source line (NEWTEMP)

Moreover, you can also use the MS-DOS control keys when in EDLIN.

For details about control keys refer to Chapter 2.

The remainder of this section describes each of the intra-line commands in turn.

Before starting to edit the current line, a copy of the current line exists in the source line. You then begin editing the current line by entering an edit line by entering text and by using the intra-line commands. Not until you complete the edit line by pressing ENTER does the edit line replace the contents of the current line.

You can select a line to work

on using the line command described in the previous section.

7-36

MS-DOS USER GUIDE

# **COPY1**

Copies one character from the source line to the edit line.

## **F1**

## **Characteristics**

Pressing the FI key copies one character from the source line to the edit line. Insert mode, if active, is automatically turned off.

## **Example**

Assuming that the line to be edited is displayed as follows:

**1:\*This is the  
COPY1**

**command 1 :\*\_**

IF you THEN...  
enter...

the first character is  
copied from the

F1

source line into the  
edit line thus:

```
1:*This is the  
COPY1 command
```

```
1:*T_
```

the next character is  
copied from the  
source line into the  
edit line thus:

F1

1:\*Thls is the  
COPY1 command

1:\*Th\_



# COPYTO

Copies up to a given character from the source line to the edit line

F2 character

Table captionWhere

SYNTAX	MEANING
ELEMENT	

character

A character whose first occurrence in the source line will terminate the copy operation. If the character does not appear In the source line nothing will be copied.

# Characteristics

Pressing the F2 key copies all characters up to but not including a given character from the source line to the edit line. The cursor is moved to the position of the given character. The given character is not displayed.

MS-DOS USER GUIDE

# Example

Assuming that the line to be edited is displayed as follows:

```
1:*This is the COPYTO  
command
```

```
1:‘_
```

IF you

enter... THEN...

F2 c the characters 'This  
is the COPYTO' are  
copied from the  
source line to the edit  
line thus:

1:\*This is the  
COPYTO command

1:\*This is the

COPYTO \_

# COPYLINE

Copies the source line to the edit line.

F3

Characteristics

Pressing the F3 key copies all remaining characters from the source line to the edit line regardless of cursor position.

Following the copy, the cursor is positioned after the last character on the line. Insert mode, if active, is automatically turned off.

## Examples

Assuming that the line to be edited is displayed as follows:

```
Table caption1:*This is the  
COPYLINE command
```



1:\*  
\_

IF you THEN...  
enter...

F3 all remaining  
characters are copied  
from the source to  
the edit line thus:

1:\*This is the  
COPYLINE  
command

1:\*This is the  
COPYLINE  
command\_

MS-DOS USER GUIDE

7-40

# SKIP1

Skip one character in the source line.

DEL

Characteristics

Pressing the DEL key skips over one character in the source line without copying it to the edit line. It does not

affect insert mode.

## Example

Assuming that the line to be edited is displayed as follows:

```
Table caption1:*This is the  
SKIP1 command
```

```
1:*
```

IF you  
enter... THEN...

DEL      you skip the letter  
          " T " in the source  
          line.

F3        the remaining  
          characters are copied  
          from the source line  
          to the edit line thus:

1:\*Thls is the SKIP1  
command

1:\*hls is the SKIP1  
command\_

# SKIPTO

Skip to specified character in the source line.

F4 character

Where

SYNTAX	MEANING
ELEMENT	

character

The character in the source line that terminates the string that is to be skipped. ,

## Characteristics

Pressing the F4 key causes characters in the source line to be skipped from the edit character up to but not including the first occurrence



of the given character. If the source line does not contain the given character then no characters are skipped.

Nothing is copied to the edit line by this command. Insert mode remains unaffected.

7-42

MS-DOS USER GUIDE

Examples

Assuming that the line to be edited is displayed as follows:

```
1:*This is the SKIPTO  
command
```

```
1:*  
_
```

IF you THEN...  
enter...

F4 c

all characters in the source line up to the first "c" are skipped over.

F3

the remaining characters in the source line are copied to the edit line thus:

1:\*This is the

SKIPTO command  
1:\*command\_

# KILL

Clear the edit line.

ESC

Characteristics

Pressing the ESC key clears the edit line, but the source line remains unchanged. The KILL command also displays a back-slash (\) and inserts a

carriage return and a line-feed. The cursor is placed immediately under the first character of the terminated line. You can then begin again to edit the line. Insert mode is turned off by this command.

## Examples

Assuming your current and edit lines are displayed as follows:

1:\*This is the KILL  
command 1:\*This Is the\_

7-44

## MS-DOS USER GUIDE



IF you THEN...  
enter...

the edit line is

ESC emptied thus:

1:\*This is the KILL  
command

1:\*This is the\

F3 the source line is  
copied to the edit  
line thus:



1:\*This is the KILL  
command

1:\*This is the\

This is the KILL  
command

# INS

Enters/exits insert mode.

## INS

### Characteristics

Pressing the INS key enters or exits insert mode. On entering insert mode, subsequently entered characters will be inserted

before the character under the cursor when insert mode was entered.

## Examples

Assuming the line to be edited is displayed as follows:

```
1:*This is INS command
```

```
1:*This is_
```

IF you THEN...  
enter...

INS            the characters "the"  
the            are inserted in the  
              edit line thus:

1:\*This is INS  
command

1:\*This is the\_

INS  
F3

insert mode is  
switched off and the  
remainder of the  
source line is copied  
to the edit line thus:

1:\*This is INS  
command

1:\*This is the INS  
command\_

Table caption7-46

Table captionMS-DOS USER  
GUIDE

**LINE  
EDITOR  
(EDLIN)**

# NEWTEMP

Creates a new source line by copying the edit line to the source line.

F5

Characteristics

Pressing the F5 key copies the edit line to the source line. The original contents of



the source line are deleted. An ”@” sign appears at the end of the edit line and a carriage return line-feed is inserted. The edit line is also cleared and and insert mode is exited.

## Example

Assuming the source and edit lines are displayed as follows:

1:\*This is the NEWLINE  
command

and you want to change the  
source line to read "This is  
the NEWTEMP command":

IF you THEN...  
enter...

all characters up to

the first "L" are  
copied from the  
source line to the edit  
line thus:

F2 L

```
1:*This is the  
NEWLINE command
```

```
1:*This is the NEW_
```

the next four  
characters in the  
source line are

replaced in the edit  
line by the characters  
'TEMP' thus:

TEMP

1:\*This is the  
NEWLINE command

1:\*This is the  
NEWTEMP\_

the remaining  
characters in the  
source line are

copied to the edit  
line thus:

F3

1:\*This is the  
NEWLINE command

1:\*This is the  
NEWTEMP  
command\_

the contents of the  
source line are

F5

replaced with those

of the edit line. The display appears thus:

```
1:*This is the  
NEWLINE command
```

```
1:*This is the  
NEWTEMP  
command®
```

```
This is the  
NEWTEMP  
command
```

F3

7-48

# MS-DOS USER GUIDE

# ABOUT THIS CHAPTER

This chapter describes the  
LINK utility.



## CONTENTS



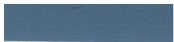
INTRODUCTION	8-1
TEMPORARY FILES	8-1
CHANGING DISKETTES	8-2
SEGMENTS, GROUPS AND CLASSES	8-2

INTERACTIVE ENTRY	8-3
COMMAND LINE ENTRY	8-10
AUTOMATIC RESPONSE FILE ENTRY	8-13



# THE LINKER

.



# INTRODUCT

MS-LINK is an executable program available on your MS-DOS system diskette. It links separate object modules that are the output of the assembler or a compatible compiler, resolving external references by searching multiple library files. Its output is a relocatable run

file, along with a list file that shows external references and error messages.

To run MS-LINK you must provide appropriate object, run, list and library file parameters. In addition, you may enter switches that modify the way in which MS-LINK processes your input. Parameters and switches are fully described in the section 'Interactive

Entry”.

There are three related methods of running MS-LINK. These are described later in this chapter in the following sections:

- Interactive Entry
- Command Line Entry
- Automatic Response File Entry

Interactive entry is the primary method and its section contains all the information common to the three methods.

## TEMPORARY FILES

MS-LINK uses available memory for the link session. If the files to be linked create an output file that exceeds available memory, MS-LINK creates a temporary file on



the default drive and names it VM.TMP. If MS -LINK needs to create VM.TMP, it displays the message:

VM.TMP has been created.

Do not change diskette in drive x:

Once this message is displayed, you must not remove the diskette from the default drive until the link

session ends. If the diskette is removed, the operation of MS-LINK is unpredictable, and MS-LINK might return the error message:

Unexpected end of file on  
VM.TMP

MS-LINK uses VM.TMP as virtual memory. The contents of VM.TMP are subsequently written to the file name following the run file

prompt. VM.TMP is a working file only and is deleted at the end of the linking session.

If the default drive already has a file by the name of VM.TMP, it will be deleted by MS-LINK and a new file will be allocated; the contents of the previous file are destroyed. You should therefore avoid using VM.TMP as one of your own

file names.

## CHANGING DISKETTES

You may wish to change diskettes during the link operation. For example, if MS-LINK cannot find an object file on the specified diskette it prompts you to change diskettes instead of aborting the session. Or if you enter the /PAUSE switch, MS-LINK pauses and

prompts you to change diskettes before creating the run file. You may change diskettes when prompted except in the following cases:

- When the diskette you wish to change has a VM.TMP file created on it (see the previous section)
- When you have requested a list file on the diskette you wish to change

# SEGMENTS, GROUPS AND CLASSES

Some of the terms used in this chapter are explained below to help you understand how MS-LINK works.

Generally, if you are linking object modules compiled from BASIC, Pascal, or any high-level language, you will not need to know these terms. If you are writing and compiling programs in

assembly language, however, you will need to understand MS-LINK and the definitions described below.

In MS-DOS, memory can be divided into segments, classes, and groups. For example:

Group	Segment	Segment
Contents	Names	Class
		Names

Segment  
1            PROG.1   CODE

Segment  
2            PROG.2   CODE

Segment  
12           PROG.3   DATA



# MS-DOS USER GUIDE

## THE LINKER

Note that segments 1, 2, and 12 have different segment names but may or may not have the same segment class name. Segments 1, 2, and 12 form a group with a group address of the lowest address of segment 1 (that is, the

lowest address in memory).

Each segment has a segment name and a class name. MS-LINK loads all segments into memory by class name from the first class encountered to the last. All segments assigned to the same class are loaded into memory contiguously.

During processing, MS-LINK references segments by their

addresses in memory. MS-LINK does this by finding groups of segments.

A group is a collection of segments that fit within a 64K byte area of memory. The segments do not need to be contiguous to form a group. The address of any group is the lowest address of the segments in that group. At link time, MS-LINK analyzes the groups, then

references the segments by the address in memory of that group. A program may consist of one or more groups.

If you are writing in assembly language, you may assign the group and class names in your program. In high-level languages (BASIC, COBOL, FORTRAN, Pascal), the naming is done automatically

by the compiler.

# INTERACTIVE ENTRY

With interactive entry MS-LINK prompts you for each parameter in turn. Type the following:

```
[d:][path] LINK
```

Four prompts appear, one at a time, requesting the

appropriate parameters.

These are summarized in the following table.

PROMPT	RESPONSE
--------	----------

Object	objfile[ +
Modules	objfile]...
[OBJ]:	[switch]...

Run File	[runfile]
----------	-----------

[objfile.EXE]: [switch]...

List File [Iistfile]

[NUL.MAP]: [switch]...

Libraries [libfile] +  
libfile].

[.LiB]:  
..]lswitch]...

Where



# SYNTAX MEANING ELEMENT

d

Specifies the drive where LINK is to be found.

path

Specifies the directory where LINK is to be found.

The file  
specification of  
an object module  
to be linked.

Such  
specifications  
must be  
separated from  
each other with a  
plus sign (- 1 -)  
or a SPACE .

objfile

The default file extension is .OBJ. If any extension is different from .OBJ it must be specified; otherwise it may be omitted. Segments are loaded by class name, from the first class encountered to

the last. The order in which you list object files is therefore significant.

8-4

MS-DOS USER GUIDE

# SYNTAX MEANING ELEMENT

The file  
specification of  
the run  
(executable) file  
that results from  
the link session.  
All run files  
receive the  
extension .EXE.  
Any other

runfile

extension you may enter is ignored. The default run file name is the first object file name entered.

The file specification of the list file that contains an entry for each segment

listfile

in the object files and its offset in the run file. The default list file name is the NUL file. The default extension is .MAP.

The file specification of

libfile

each library. Up to eight libraries may be searched; separate each specification with a plus sign (-i-) or a SPACE . The default is standard library search. The default extension is .LIB. Libraries are searched in



the order they are listed.

The name of any of the switches described in the table below. You may enter any number of switches, at the end of any number of prompt

switch

responses  
(before pressing  
ENTER ).

Switches may be  
abbreviated to  
the / together  
with the first  
letter or any  
sequential  
substring  
starting with the  
first letter.



## Note

That [d:][pat/7] may precede any file name mentioned.

SWITCH

MEANING

/DSALLOCATE

All data de  
to be in DC  
is loaded at  
high end of

group. If the  
switch is not  
MS

LINK loads  
data at the  
end of the g  
At runtime  
data space  
pointer is s  
the lowest  
possible ad  
allowing th

entire storage  
be used. Use  
the  
/DSALLOC  
switch in  
combinatic  
the default  
low (that is  
/HIGH swi  
not used) p  
the user  
application  
allocate  
dynamically

(or /D)

available  
memory be  
the area  
specifically  
allocated w  
DGROUP,  
remain  
addressable  
the same d  
space point  
This dynan  
allocation l  
needed for  
and FORTI

programs. The maximum amount of memory that can be allocated to the application is 64K (or the amount actually available) in the allocated portion of the DGROUP.

/HIGH

(or /H)

MS-LINK :  
the run file  
high as pos  
in memory  
/HIGH is n  
specified th  
file is load

ed as low a  
possible. D  
NOT USE '  
SWITCH V



PASCAL C  
FORTRAN  
PROGRAM

MS-LINK  
includes In  
list file the  
numbers ar  
addresses c  
source stat  
in the in

/LINENUMBERS

(or /L)

put module  
/LINENUM  
is not speci  
line numbe  
addresses a  
included. (C  
compilers  
produce ob  
modules th  
contain line  
number  
information  
these cases  
course MS-

cannot Incl  
line numbe

8-6

## MS-DOS USER GUIDE

SWITCH

ME

MS  
list

/MAP

(glo  
syn  
def

in t  
mo  
/M  
giv  
MS  
list  
(inc  
unc  
glo

(or /M)

The  
are  
alp.  
For  
syn  
LIN  
val  
seg  
off  
in t  
The  
are  
the  
list

**/NODEFAULTLIBRARY**  
**SEARCH**

MS  
doe  
aut  
sea  
def  
to r  
ext  
refe  
For  
linl  
obj

mo  
the

sto] ]  
LIN  
aut  
sea  
file  
PA

(or /N)

MS  
pau

/PAUSE

link  
who

swi  
enc  
No  
MS  
per  
link  
ses  
wit  
sto  
beg



end  
allc  
use  
disl  
bef  
LIN  
the  
file

Wh  
LIN  
enc  
/PA  
swi

(or /P)

disj  
me

Ab  
gen  
file

Cha  
hit

MS  
res  
pro  
wh

pre  
key  
SW  
DIS  
TH  
RE  
LIS  
OR  
US  
TE  
(VT  
FIL

SWITCH

MEANING

/STACK;size  
(or IS)

The size of the stack provided for the load module by the assembler or compiler is overridden. The stack size becomes

that specified  
in the 'size'  
parameter,  
which must  
follow the  
switch name  
and a colon.

If a value  
from 1 to  
511 is  
entered, MS-

LINK uses  
512. At least  
one object  
(input)  
module must  
contain a  
stack  
allocation  
statement. If  
not, MS-  
LINK will  
return a  
**WARNING:  
NO STACK**

SEGMENT  
error  
message.

## Characteristics

After any of these responses, before pressing ENTER , you may enter a comma (,) followed by the answer to what would be the next prompt, without having to

wait for that prompt.

If you conclude any response with a semicolon (;) the remaining responses are all assumed to be the default. Linking begins immediately with no further prompting.

Use the plus sign (+) not only to separate lists of object files and libraries but to extend these lists, where necessary, onto more than



one line. Enter the plus sign followed by ENTER at the end of a physical line. This repeats the object file or library prompt, and enables you to continue the logical line with further file names.

## Example

This sample shows you a typical dialog for an MS-LINK session. In response to the MS-DOS prompt, enter:

# LINK

8-8

## MS-DOS USER GUIDE



IKER

The system displays the following messages, prompts and your responses:



r

Microsoft 8086 Object  
Linker (C) Copyright  
Microsoft Corp.

Object Modules [.OBJ]: 10  
SYSINIT Run File [IO.EXE]:

List File [NUL.MAP]: 10  
/MAP Libraries [.LIB]:



Fig. 8-1 LINK Example

# Notes:

1. By specifying /MAP, you get both an alphabetic listing and a chronological listing of public symbols.

2. By responding PRN to the List File: prompt, you can redirect your output to the printer.

3. By specifying the /LINE

switch, MS-LINK gives you a listing of all line numbers for all modules. (Note that the /LINE switch can generate a large volume of output).

4. By pressing ENTER in response to the Libraries: prompt, an automatic library search is performed.

Once MS-LINK locates all libraries, the linker map

displays a list of

segments in the order of their appearance within the load module. The

list might look like this;

Start Stop Length Name

**00000H**

**009ECH 09EDH**

# CODE

```
009FOH 01166H 0777H  
SYSINITSEG
```

The information in the Start and Stop columns shows the 20-bit hex address of each segment relative to location zero. Location zero is the beginning of the load module.

The addresses displayed are not the absolute addresses where these segments are loaded.

Because the /MAP switch was used, MS-LINK displays the public symbols by name and value. For example:



ADDRESS PUBLICS\_BY\_NAME



009F:0012 BUFFERS

009F:0005 CURRENT\_DOS\_

009F:0011 DEFAULT\_DRIV

009F:0008 DEVICE\_LIST

009F:0013 FILES

009F:0009 FINAL\_DOS\_LO

009F:000F MEMORY\_SIZE

009F:0000 SYSINIT

ADDRESS PUBLICS BY VA

009F:0000 SYSINIT

009F:0005 CURRENT\_DOS\_

009F:0009 FINAL\_DOS\_LO

009F:0008 DEVICE\_LIST

009F:000F MEMORY\_SIZE

009F:0011 DEFAULT\_DRIV

009F:0012 BUFFERS

009F:0013 FILES

Fig. 8-2 /MAP Sample  
Display

# COMMAND LINE ENTRY

With command line entry you enter the MS-LINK command along with its parameters, without waiting to be prompted.

You must separate each complete parameter entry from the next with a comma (,). Apart from this, what you enter is the same as with interactive entry. The command syntax is therefore the following:

8-10

MS-DOS USER GUIDE

■ .. .:■ ■ ■ ' : ■ ■ ■ ■

[d:]\path] LINK objfile[ +  
 objfile]...[switch]...[, [runfile]  
 [switch]...[, [listfile]  
 [switch]...[, [libfile[ +  
 libfile]...][switch]...]]

Table caption Where

SYNTAX ELEMENT	MEANING
-------------------	---------

d:

Specifies the drive where LINK is to be found.

path

Specifies the directory where LINK is to be found.

objfile

An object module to be



linked.

runfile

The run  
(executable) file  
to be created.

listfile

The list file to be  
output.

libfile

A library file to

be searched.

switch

A switch to be applied.



## Note

[d:][path] may precede any filename.

See the section 'Interactive

Entry” for full descriptions of these parameters.

# Characteristics

Use the plus sign (+) as with interactive entry not only to separate lists of object files and library files but to extend those lists, where necessary, onto more than one line.

You may enter switches after any one of the four parameter entries (that is, before any of the commas or the final ENTER ).

To accept the default parameter for a syntax element, enter a second comma with no space between the two commas. To accept a further default enter a third comma and so on. Remember that you must

make an entry for the object file parameter.

If you enter a semicolon (;) at any time, the unspecified parameters all assume default values. Linking begins immediately.

If you enter an incomplete list of parameters and no semicolon is used, MS-LINK prompts you for the next remaining entry. See

”Interactive Entry” for a full list of MS-LINK prompts.

# Examples

IF you enter...



LINK FUN + TEXT + TABLE  
CARE/P/M,,FUNLIST,COBL





LINK FUN,,

Table caption8-12

**MS-DOS USER GUIDE**

# THE LINKER

# **AUTOMATIC RESPONSE FILE ENTRY**

With automatic response file entry you enter the name of a file that already contains the answers to some or all of the

MS-LINK parameter prompts. Precede this file name by the symbol @. The command syntax is therefore the following:

d-.\path] LINK ®[drive:] [f-path] filespec

Table captionWhere

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the drive where LINK is to be found.

path

Specifies the directory where LINK is to be found.

filespec

The file specification of the file that contains responses to the MS-LINK prompts. Use of a file extension is optional; there is no default extension.

Specifies the



drive

drive where  
filespec is to be  
found.

f-path

Specifies the  
directory where  
filespec is to be  
found.

# Characteristics

Automatic response files can contain several lines of text, each corresponding to an MS-LINK prompt. Responses must be in the same order as with interactive input.

Press ENTER to indicate the conclusion of each response and the beginning of the next. Type a plus sign (-H) followed by ENTER to continue a response to the object module or libraries

prompt on a new line. See the section "Interactive Entry" for full details of each MSLINK prompt.

Use switches, commas and colons in an automatic response file just as you would in interactive entry.

You can enter the name of more than one automatic response file on the command line, and combine

response file names with additional parameters. The combined series of resulting parameters must be a valid sequence of MS-LINK prompts.

When the MS-LINK session begins, each prompt is displayed in order with the responses from the response file. If the response file does not contain answers for all the prompts, (in the form of

file names, the semicolon command character or carriage returns), MS-LINK displays the prompt which does not have a response, then waits for you to enter a legal response. When a legal response has been entered, MS-LINK continues the link session.

## **Example**

IF you enter...

THEN...

COPY CON

C:\COB\INC\INC1 FUN +  
TEXT + TABLE + CARE  
/PAUSE/MAP

this response file tells MS-LINK to load the four object modules named FUN, TEXT, TABLE, and CARE. MS

FUNLIST

COBLIB.LIB

F6

LINK pauses before producing a public symbol map to permit you to swap diskettes. When you press any key, the output files are named FUN.EXE and FUNLIST.MAP. MS

C:\COB\LINK@C:\INC\INC1

LINK searches the library  
file

COBLIB.LIB.

8-14

MS-DOS USER GUIDE



# ABOUT THIS CHAPTER

This chapter describes the  
DEBUG utility.

# CONTENTS

INTRODUCTION 9-1

HOW TO INVOKE THE 9-1  
DEBUG PROGRAM

DEBUG 9-1

DEBUGGING COMMANDS  
9-4

COMMAND PARAMETERS

9-4

A (ASSEMBLE) 9-7

C (COMPARE) 9-10

D(DUMP) 9-11

E (ENTER) 9-14

F (FILL) 9-17

G (GO) 9-18

H (HEX) 9-21

I (INPUT) 9-23

L (LOAD) 9-24

M (MOVE) 9-27

N (NAME) 9-29

O (OUTPUT) 9-31

P (PROCEED) 9-32

Q (QUIT)	9-33
R (REGISTER)	9-34
S (SEARCH)	9-37
T (TRACE)	9-39
U (UNASSEMBLE)	9-40

W (WRITE)

9-42

# INTRODUCT

The DEBUG utility is a program that provides a controlled testing environment for binary and executable object files. It eliminates the need to re-assemble a program to see if a problem has been corrected by a minor change.

Moreover, it enables you to

change the contents of a file or CPU register, then to immediately re-execute a program to check the validity of the changes.



# HOW TO INVOLVE THE DEBUG PROGRAM

# DEBUG

The DEBUG program is invoked as follows:

[d:][path] DEBUG [[drive:]  
[file-path] filespec [, arglist]]

Table captionWhere

SYNTAX ELEMENT	MEANING
-------------------	---------

d

Specifies the drive where DEBUG is to be found.

path

Specifies the directory where DEBUG is to be found.

drive

Specifies the drive where filespec is to be found.

file-path

Specifies the directory where filespec is to be found.

The specifier of the program file

filespec to be debugged.



SYNTAX MEANING  
ELEMENT

A list of file  
name parameters  
and switches

arglist

separated by commas. These will be passed to the program specified by the filespec parameter. Thus, when the program is loaded into memory, it is loaded as if it had been invoked with the

command:

filespec arglist

That is, filespec indicates the file to be debugged, and arglist is the rest of the command line that is used when the file is

invoked and  
loaded into

memory.

# Characteristics

On entering the DEBUG  
environment DEBUG  
responds with the



hyphen (-) prompt and underline ( \_ ) cursor. You are then free to enter

any DEBUG command.

If you enter DEBUG without parameters, since no file name has been specified, current memory, disk blocks, or disk files can be worked on using other debugging commands.

If you include the filespec in the command line then the specified file is loaded into memory starting at location 100 (hexadecimal). However, if the file has an EXE extension, then it is relocated to the address specified in the header of the file. Moreover, if the file has the HEX extension, then the file is loaded beginning at the address specified in the HEX file.

## MS-DOS USER GUIDE

Table caption Examples

IF you  
enter...

THEN...

DEBUG

the DEBUG  
environment is  
entered, but

without loading  
a file.

DEBUG  
b:myprog

the DEBUG  
environment is  
entered and the  
file named  
"myprog" is  
loaded into  
memory from  
drive B.

DEBUG  
progs\dev\  
myrtn,  
prog1,prog2

the DEBUG  
environment Is  
entered and the  
file named  
"myrtn" is  
loaded into  
memory from  
the sub-  
directory "dev"  
of directory  
"progs". The  
loaded file also  
takes two file  
name

parameters:  
"prog1" and  
"prog2".

## Remarks

When you  
invoke  
DEBUG, it  
sets up a  
program

header at  
offset 0 in  
the program  
work area.

You can  
overwrite  
this area if  
you enter  
DEBUG

without parameters.

Moreover, if you are  
debugging a file with a COM

or EXE extension you must not tamper with the program header below location 5CH, or DEBUG will terminate.

Do not restart a program after the "Program terminated normally" message is displayed. You must reload the program with the N and L commands for it to run properly.



# DEBUGGING COMMANDS

This section describes the DEBUG commands in alphabetical order for ease of reference. Each such command description summarizes the purpose of the command, defines the command syntax and

explains each syntax element. This is followed, for each command, by a detailed account of the command characteristics and some working examples.

## **Remarks**

1. Commands can be entered in either upper or lower case.
2. Command keywords and

command parameters can be separated from each other by spaces or commas for readability but need not be, except where two hexadecimal numbers are entered as parameters, in which case they must be separated by a comma or space. For brevity, the syntax of this chapter will always indicate comma where separation is obligatory, but note that a space can

alternatively be used.

3. Commands only become effective after pressing ENTER .

4. If you make a syntax error when entering a command the message " Error" will be displayed. You must re-enter the command using the correct syntax.

# COMMAND PARAMETERS

The following DEBUG command parameters require special definition.



9-4

MS-DOS USER GUIDE

# PARAMETER DEFINITION

A  
hexadecimal  
value in one  
of the  
following  
formats:

- A segment register designation and a

hexadecimal  
offset  
separated  
from each  
other by a  
colon. For  
example:

DS:0300

- A  
hexadecimal  
segment and  
offset

address

separated  
from each  
other by a  
colon. For  
example:

9D0:0100

- A  
hexadecimal  
offset value.  
For example:

200



The DEBUG command will append a default segment value from either the DS or CS registers, depending on the command.

byte

A one or two  
character  
hexadecimal  
value.

drive

0, 1 or 2  
depending on  
whether you  
wish to select  
drive A, drive  
B or drive C,  
respectively.



# PARAMETER DEFINITION

A range of  
addresses  
specified as:

EITHER:

address L valu

where address

specifies the start of the range and value specifies the length of the range. For example:

DS:300L30

indicates a range of 48 locations starting at

range

address 300 in  
the sector  
indicated by  
the DS register.

The specified  
range cannot  
be greater than  
1000

(hexadecimal)

To specify this  
value enter  
0000 (or 0) as  
the value

parameter.

OR:

address,address

where the two addresses indicate the limits of the range. Note that space may be used instead of comma.

value

A 1 to 4  
character  
hexadecimal  
value.

9-6

MS-DOS USER GUIDE



# A

# (ASSEMBLE)

Assembles  
assembler  
mnemonics  
directly  
into  
memory.



A [address]

Where

SYNTAX ELEMENT	MEANING
-------------------	---------

The start address  
at which the  
subsequently  
entered line of

address            mnemonics is to be assembled. If this parameter is omitted, location 100 is assumed.

## Characteristics

After you enter the Assemble command, DEBUG displays the specified address followed by the cursor. You

may then enter a line of assembler mnemonics. On terminating the line with ENTER the line will be assembled into memory starting at the specified location. The address of the byte subsequent to the assembled code will be displayed on the next line along with the cursor to enable you enter the next line of code. If, instead of a line of assembler mnemonics,

you simply press ENTER ,  
the Assemble command  
terminates and the DEBUG  
prompt re-appears.

All numeric values are  
hexadecimal and must be  
entered as 1 to 4 characters.  
Prefix mnemonics must be  
specified in front of the  
opcode to which they refer.  
You may also enter them on  
a separate line.

The segment override mnemonics are CS:, DS:, ES; and SS:. The mnemonic for the far return is RETF. String manipulation mnemonics must explicitly state the string size. For example, use MOVSW to move word strings and MOVSB to move byte strings.

The Assemble command will automatically assemble short, near or far jumps and

calls, depending on byte displacement with respect to the destination address.

These may be overridden with the NEAR or FAR prefix. For example:

0100:0500 JMP 502 ;a two-byte short jump

0100:0502 JMP NEAR 505 :a three-byte near jump

0100:505 JMP FAR 50A ;a five-byte far jump

The NEAR prefix may be abbreviated to NE, but the FAR prefix cannot be abbreviated.

DEBUG cannot tell whether some operands refer to a word memory location or to a byte memory location. In this case the data type must be explicitly stated with the prefix "WORD PTR" or "BYTE PTR". Acceptable abbreviations are "WO" and

”BY”. For example:

```
NEG BYTE PTR [128]
```

```
DEC WO [SI]
```

A simple operand is a literal. Whereas operands enclosed within square brackets refer to memory. For example:

```
MOV AX,21 ;Load AX with  
21H
```



MOV AX,[21 ] :Load AX  
with the contents of location  
;21H

Two popular pseudo-  
instructions are available  
with the Assemble command.  
The DB opcode will  
assemble word values  
directly Into memory. For  
example:

DB 1,2,3,4,"THIS IS AN

EXAMPLE”

DB ’THIS IS A QUOTE: ’”

DB ”THIS IS A QUOTE””

DW

1000,2000,3000, ”BACH”

9-8

MS-DOS USER GUIDE

The Assemble command

supports all forms of register indirect commands. For example:

```
ADD BX,34[BP-i-2].[SI-1]
```

```
POP [BP-i-DI]
```

```
PUSH [SI]
```

All opcode synonyms are also supported. For example:

```
LOOPZ 100 LOOPE 100 JA
```

200 JNBE 200

Table captionExample

IF you THEN...  
enter...

A200 DEBUG displays

09AC:0200 \_

the assembler mnemonics are assembled starting at location 200. The byte location subsequent to the assembled code is then displayed thus

MOV  
AX,  
[21]

09AC:0203 \_

the Assemble

command

ENTER terminates and the  
DEBUG prompt re-  
appears.

# C

# (COMPARE)

Compares the contents of two areas of memory.

C range , address

Table captionWhere

## SYNTAX

# ELEMENT MEANING

range

The range of addresses defining the first area to be compared. If no segment is specified then the segment specified in the DS register is assumed.



address

The start of the area to be compared with the area specified by the range parameter.

# Characteristics

The Compare command

compares the area of memory specified by the range parameter with an area of the same size starting at the location specified by the address parameter.

If the contents of the two areas are identical nothing is displayed. If there are differences, then the differences are displayed in the form:

address1 contents1 contents2  
address2

9-10

## MS-DOS USER GUIDE



where; address1 indicates the address in the first area and contents1 its contents: and address2 indicates the corresponding address in the second area and contents2 its

contents.

## Example

IF you  
enter...

THEN...

the area of  
C100, 1FF, memory from  
300 or 100 to 1FF is  
compared with

C100L100, the area of  
300 memory from  
300 to 3FF.

# D (DUMP)

Displays an area of memory.

D [range | address]

Table captionWhere

SYNTAX  
ELEMENT

MEANING

range

The range of addresses whose contents are to be displayed. If you enter only an offset, then the segment specified in the DS register is assumed.

address

The address from which the display is to start. The contents of this address and the subsequent 127 locations are displayed.



# Characteristics

If D is specified without parameters then the 128 bytes following the last address to be displayed are displayed. If no location has yet been accessed then the dump will start from location DS:100.

If D and the range parameter

are specified then the contents of that range of addresses are displayed. If this takes more than 24 screen lines the display is scrolled until the contents of the final address in the range are displayed on line 24.

If D and a single address are specified, then the contents of the 128 locations starting from the specified address are displayed.

The dump is displayed in two portions;

- A hexadecimal dump, where each byte is represented by its hexadecimal value.
- An ASCII dump, where the equivalent ASCII character for the byte is displayed. If there is no corresponding printable ASCII character then the dump displays a

period (.).

9-12

## MS-DOS USER GUIDE

Each line of the dump begins with an address which is followed by the hexadecimal contents of the 16 bytes starting from the addressed location. The eighth and ninth bytes are separated by a hyphen (-). The right-hand

columns display the equivalent ASCII values. Each line of the display, except possibly the first, begins on a 16 byte boundary.

## Examples

IF you            THEN...  
enter...

D 100,110 or D100L11 a hexadecimal and ASCII dump of lines 100 to to 110 (hexadecimal), inclusive, are displayed.

D a hexadecimal and ASCII dump of the 128 bytes starting from location 111 (hexadecimal) Is

displayed.

D200

a hexadecimal and  
ASCII dump of  
the 128 bytes  
starting from  
location 200  
(hexadecimal) Is  
displayed.

# E (ENTER)

Replaces the contents of memory locations at the byte addresses specified.

E address[ , bytevalue[ , bytevalue]...]

Where

SYNTAX      MEANING



# MEANING

address

The address of the location whose value is to be replaced; or the address of the first of a succession of locations whose contents are to be replaced. If only an offset is

specified then  
the segment  
indicated by the  
DS register is  
assumed.

The hexadecimal  
byte value that  
is to replace the  
contents of the  
specified  
address. The

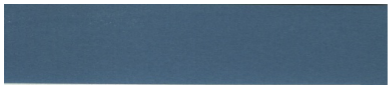
bytevalue

first bytevalue  
parameter will  
replace the  
contents of the  
location  
specified by the  
address  
parameter. A  
second  
bytevalue will  
replace the  
contents of the  
location  
following that

specified by the  
address  
parameter, and  
so on.

9-14

MS-DOS USER GUIDE



# Characteristics

if the command is entered without the bytevalue list, then DEBUG displays the specified address and its contents. The Enter command then waits for you to perform one of the following:

- Replace the displayed byte value by entering another value. You simply enter the new value after the current value. If you enter an illegal

value, or if you type more than two digits then the illegal or extra character is not echoed.

- Advance to the next byte by pressing SPACE . To change the value of this byte simply enter the value as described above. If you advance beyond an eight-byte boundary, DEBUG starts a new display line with the address displayed at the start of the

line. To advance to the next byte without changing the current byte simply press SPACE again.

- Return to the previous byte by entering hyphen (-). On doing so DEBUG starts a new display line with the address of the byte you have returned to and its contents. You can then change the contents of this location as described above. To move back one

byte further without changing this value simply enter hyphen again, and another new display line will be generated.

- Terminate the Enter command by pressing ENTER . This key may be pressed in any byte position.

If you specify bytevalues in the command line then the first of these bytevalues will



replace the contents of the location specified by the address parameter.

Subsequent entries in the list of bytevalues will replace subsequent bytes in memory.

IF you  
enter...

THEN...

E100

DEBUG  
displays

something like:

0580:0100 CD

—

26

the value of  
location 100 is  
changed to 26  
and DEBUG  
displays:

0580:0100

CD.26\_

SPACE

the next byte  
(location 101)  
is displayed:

0580:0100

CD.26 20\_

SPACE

the next byte  
(location 102)

is displayed:

0580:100

CD.26 20. 00\_

the previous  
byte (location  
101) is  
displayed on  
the next line:

0580:0100

CD.26 20. 00.

0580:0101 20. \_

30

the contents of  
location 101 are  
changed to 30  
and the Enter  
command is  
terminated:

0580:0100

CD.26. 20. 00.

0580:0101

20.30

>  
\_

the contents of  
byte locations  
200, 201, 202  
E200,26,0A, and 203 are  
19,23  
changed to 26,  
0A, 19 and 23,

respectively

9-16

MS-DOS USER GUIDE

, ■' ■ ■ ■' ■

# **F (FILL)**

Fills an area of memory with specified byte values.

F range , bytevalue[,  
bytevalue...]

## **Where**



# SYNTAX MEANING ELEMENT

range

The range of addresses whose contents are to be overwritten with the specified byte values. If only the offset is specified then the segment

indicated by the DS register is assumed.

bytevalue

A two digit hexadecimal value that is to overwrite the contents of the specified address.




# Characteristics

If the specified range contains more bytes than the list of bytevalues , then the list of byte values is repeated until the specified range is filled.

If the list of bytevalues is longer than the specified range then the extra bytevalues are ignored.

9-17



Example

IF you enter...

THEN

DEBU

fills

memo

locati

04BA  
to  
04BA  
F04BA:100L100,42,45, FF wi  
the by  
48,37,20  
value  
speci:  
The f  
value  
repea  
until  
256  
locati  
are fi



# G (GO)

Executes the program currently in memory, optionally halting at a specified breakpoint and displaying information about the system/program environment.

G [= address][, address]...

Table captionWhere

# SYNTAX MEANING ELEMENT

The address in memory at which program execution is to start.” = ” must  
= address be entered to distinguish a start address from a



breakpoint  
address.

9-18

## MS-DOS USER GUIDE

SYNTAX  
ELEMENT

MEANING

The

address

breakpoint  
address at  
which  
program  
execution is  
to halt and  
the register  
and flag  
states to be  
displayed  
along with  
the next  
instruction to  
be executed.

## Characteristics

If you enter G without parameters then the program currently in memory is executed starting from the address specified by the CS and IP registers.

If you specify the = address parameter, then the contents

of the CS and IP registers are changed to those specified by the = address parameter, and the program in memory is executed, starting from that point.

If you specify one or more breakpoint addresses then program execution stops at the first such address encountered and displays the contents of the registers, the state of the flags and the next

instruction to be executed (see the Register command for a description of the display)..

You may specify up to ten breakpoint addresses, in any order. If your program has many paths you can use this feature to ensure that your program halts, whichever path it takes.

If you enter more than ten

breakpoints DEBUG will display:

## **BP Error**

Before executing the program the Go command replaces the contents of the breakpoint locations with an interrupt instruction (hexadecimal CC). VWhen program execution halts at such a location DEBUG

restores the original values of all the specified breakpoint locations. However, if the program terminates normally (that is, not at a specified breakpoint), then the breakpoint values are not restored.

Each breakpoint address that you specify must point to the first byte of an 8086 instruction, otherwise unpredictable results will

occur.

The user stack pointer must have six bytes available for this command, otherwise unpredictable results will occur.

## **Example**

IF you  
enter...

THEN...



the program currently in memory is executed starting from location 200. Assuming location 141 is encountered before 1AF, then the program halts

G = 200,1AF,141 at location 141 and the register and flag values are displayed along with the next instruction to be executed. If neither breakpoint location is encountered, then the program

terminates  
normally.

if, in the  
previous  
example, the  
program halted  
at location 141,  
then program  
execution  
continues from  
that address. If  
program

G execution terminated normally In the previous example, then program execution again starts at location 200.

9-20

MS-DOS USER GUIDE

# H (HEX)

Calculates and displays the sum and the difference of two hexadecimal values.

H value-a , value-b

Table captionWhere

SYNTAX  
ELEMENT

MEANING

value-a

The first of  
two  
hexadecimal  
values.

value-b

The  
hexadecimal  
value that Is  
to be added  
to or  
subtracted

from value-a.

## Characteristics

The hexadecimal values may be up to four characters long.

The Hex command displays two four digit values:

- The first is the result of adding value-b to value-a

- The second is the result of subtracting value-b from value-a

## Example

IF you  
enter...

THEN...

H19F,10A

DEBUG  
displays:



02A9 0095

HFFFF,2

DEBUG  
displays:

0001 FFFD

MS-DOS USER GUIDE

9-22



## 9 I (INPUT)

Inputs and displays (in hexadecimal) one byte from the specified port.

I value

Where

SYNTAX  
ELEMENT

MEANING

value

The hex  
address of  
the port from  
which the  
byte is to be

input.



# Characteristics

The port address can be up to 16 bits.

# Example

IF you THEN...  
enter...

I2F8      the byte at the  
            addressed port is  
            input and displayed.



9-23

# L (LOAD)

Loads a file or absolute disk sectors into memory.

L [address[ , drive , sector-a , sectors ]]

## Where

# SYNTAX MEANING ELEMENT

address

The address in memory at which the file or specified sectors, is to be loaded. If only an offset is entered then the segment indicated by the

CS register is assumed. Sectors cannot be loaded across segments.

drive

The drive from which disk sectors are to be loaded. For drive A you must enter 0, for drive B you must enter 1, or for drive C



you must enter  
2.

sector-a

The first of a  
range of sectors  
to be loaded  
from the disk  
specified by the  
drive parameter.

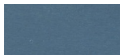
The number of  
sectors to be

sectors

loaded. The maximum number of sectors that can be specified is 80 Hexadecimal.

9-24

MS-DOS USER GUIDE



# Characteristics

If all parameters are specified then DEBUG loads sectors of information from disk into memory. The first such sector to be loaded is that specified by the sector-a parameter. It is loaded at the memory location specified by the address parameter. This sector is the first of a continuous range of sectors

to be loaded, the number of which is specified by the sectors parameter.

If you enter L without parameters, or with just the address parameter, the file whose file control block is correctly formatted at location CS:5C is loaded into memory. The file control block at CS:5C is set either to the filespec specified when the DEBUG command was

invoked, or to the filespec specified by the most recent subsequent Name command.

If L is entered alone, then the file is loaded at location CS:100. If you specify L and the address parameter, the file is loaded at the specified address. In either case DEBUG sets the BX:CX registers to the number of bytes loaded.

- If the file has an .EXE extension, then it is relocated to the load address specified in the loader of the .EXE file. That is, the address parameter to the Load command is ignored. Note that the header itself is stripped off the .EXE file before the file is loaded into memory. Thus the size of the .EXE file on disk will differ from its size in memory.

If the file is a .HEX file, then entering the Load command with no parameters causes the file to be loaded starting at the address specified within the .HEX file. If the address parameter, however, is specified then loading starts at the address which is the sum of the address specified and the address in the .HEX file.

# Examples

The following examples assume the system to be initially in MS-DOS.

IF you enter... THEN...

the debugger  
is entered and  
the



subsequent  
Name  
command sets  
the file  
control block  
at CS:5C to  
debug identify file  
"file.com" on  
Nb'.file.com the diskette  
L inserted in  
drive B. The  
Load  
command  
then loads

this file Into  
memory  
starting at  
CS:100 (the  
default  
address).

file.com is  
loaded into  
memory at  
location CS:  
100 by the  
DEBUG

debug

b:file.com

L300

command. It  
is then  
relocated to  
CS;300 by the  
Load  
command.

109 sectors  
are loaded  
into memory  
from drive B  
starting from

debug

sector OF.

L500,1,OF,6D

They are placed in memory starting at location CS:500.

9-26

MS-DOS USER GUIDE

'■' ^-ic JiiVi ■



# M (MOVE)

Moves the contents of a specified range of memory addresses to the locations starting at a specified address.

M range , address

Where

# SYNTAX ELEMENT

# MEANING

range

The area of memory whose contents are to be moved. If you only entered an offset then the segment indicated in

the DS  
register is  
assumed.

The start of  
the  
destination  
area. If you  
only entered  
an offset  
then the  
segment

address



indicated by  
the DS  
register is  
assumed.

## Characteristics

If the source and destination areas overlap the move is performed without loss of data.

The contents of the source area are not changed by the move, unless the destination area overlaps it.

If you specify an address as the end of the range you must only enter the offset. The segment specified, or defaulted to, in the start address of the range is assumed.

# Example

IF you enter...

THEN.

the 11  
bytes  
starting  
at

location  
CS:100

MCS:100,110,CS:500 through  
to 110

or

MCS:100L11,CS:500

are

copied

to the

11

bytes

starting

at

location:

CS:500

9-28

MS-DOS USER GUIDE



# N (NAME)

Provides file names for the Load and Write commands or file name parameters for the program to be debugged.

Table captionN [drive:][file-path] filespec [arglist]

Where

# SYNTAX

## ELEMENT MEANING

filespec

The file specifier of a file to be loaded into memory, written to diskette, or used as a file name parameter to the file currently in memory.

See Page  
9-1 for the  
definition  
of the  
other  
syntax  
elements.

# **Characteristics**



The name command can be used to provide:

- The name of the disk file to be loaded into memory by a subsequent Load command.
- The name to be assigned to the file currently in memory when the file is subsequently written to disk.
- File name parameters to the file in memory to be

debugged.

The first case enables you to specify the file you wish to debug after entering the DEBUG environment. That is, you can enter DEBUG without specifying parameters, then use the Name command to name the disk file you wish to debug, then load the file into memory using the Load command. This has the same

effect as entering the file name as the first parameter to the DEBUG command. In either case the file control block for the file to be debugged is set up at location CS:5C and the file is loaded.

In the second case the file is already in memory and the Name command sets up the file control block for the specified file name at location CS:5C. When a

Write command is subsequently entered the file in memory is written to disk with the file name whose file control block is set up at location CS:5C.

In the third case the Name command provides file name parameters for the program currently in memory.

Whatever file control block was set at CS:5C is replaced by that of the first such

parameter. If a second file parameter is specified then its file control block is set up at location CS:6C. Only two file control blocks are set up although additional file name parameters may be included if required. All the file names specified are placed in a save area at CS:81, with CS:80 containing a character count. Parameters specified in this way are analogous to file names specified in the

argument list to the DEBUG  
command.

## Examples

IF you  
enter...

THEN...

the system  
enters the  
DEBUG

environment  
and the file  
named  
DEBUG file.COM  
resident on  
Nbifile.com drive B has its  
file control  
L block set up at  
location CS:5C.  
The Load  
command  
subsequently  
loads this file  
into memory.

This sequence  
has the same  
effect as  
entering  
''DEBUG  
b:file.com''.

9-30

MS-DOS USER GUIDE



IF you enter... THEN...

the file control  
block is set up  
location CS;  
for the file  
specifier  
"b:newfile.co

Nb:newfile.com The subsequent

W Write command  
writes the file  
currently in

memory to d  
B and names  
file  
''newfile.com

the DEBUG  
command loc  
the file name  
file1 .com fro  
drive B to be  
debugged. Th  
Name comm

DEBUG b:file1  
.com  
Nfile2.dat,files,  
dat

G

sets up two file  
control blocks  
locations CS:  
and CS:6C for  
the file  
specifiers  
b:file2. dat a:  
b:file3.dat,  
respectively.  
These files then  
become  
parameters to  
file1 .com with  
the subsequent

Go command  
executes  
file1.com.

# O(OUTPUT)

Sends a specified byte to an output port.

O value , byte

---

Table captionWhere

SYNTAX	MEANING
ELEMENT	

value

The address of the output port. It must be specified in hexadecimal and can be up to 16 bits.

byte

A two-digit hexadecimal value to be sent

to the specified  
port.

Table caption Example

IF you enter... THEN...

01E8, 27 the byte value 27  
Hex is output to the  
port IE8.





# P

# (PROCEED)

Proceed past a CALL or INT instruction.

P [= address] [, value]

See Page 9-40 for the definition of the syntax elements.

# Characteristics

This command's key function is to execute, without tracing, all CALL and INT instructions. In addition it displays repeated instructions only once and traces loops (terminated by LOOP XX) only on the first pass.

# MS-DOS USER GUIDE

# Q(QUIT)

Terminates the DEBUG program.

## Characteristics

The Quit command terminates the debugger without saving the file you are working on. Control is

returned to MS-DOS  
command mode.

# Example

IF you THEN...  
enter...

the DEBUG program  
terminates and

Q returns you to MS-

DOS command  
mode.

# R(REGISTER

Displays the hexadecimal contents of the registers and flag settings, or displays the contents of a specified register with the option to change that value, or displays the flag settings with the option of reversing any number of those settings.

R [register-name\ F ]

Where

SYNTAX ELEMENT

MEANING

register-name

Any valid register name whose contents are to be examined and optionally changed. This may be one of:



AX DX SI ES IP

BX SP DI SS PC

CX BP DS CS

F

Note: IP and PC both refer to the Instruction Pointer.

The flag settings are to be displayed and, optionally, changed.

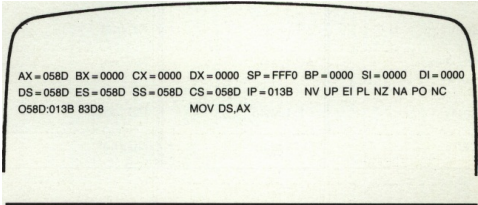
9-34

MS-DOS USER GUIDE

# Characteristics

If you enter R without parameters, then the contents of all registers are displayed

along with the flag settings and the next instruction to be executed. For example:



```
AX=058D BX=0000 CX=0000 DX=0000 SP=FFF0 BP=0000 SI=0000 DI=0000
DS=058D ES=058D SS=058D CS=058D IP=013B NV UP EI PL NZ NA PO NC
058D:013B 83D8          MOV DS,AX
```

*Fig. 9-1 Sample R Display*

If you enter R with a register-name , then DEBUG displays the contents of that register. The command then waits for you to do one of the

following:

- Press ENTER to terminate the Register command without changing the value of the displayed register.
- Change the value of the register by entering the four-digit hexadecimal value then terminate the Register command by pressing ENTER .

The valid flag values are shown in the following table:

^

9-35

FLAG NAME	SET	CLEAR
--------------	-----	-------

Overflow	OV (yes)	NV (no)
----------	----------	---------

	DN	UP
Direction	(decrement)	(increment)

	EI	DI
Interrupt	(enabled)	(disabled)

	NG	PL (plus)
Sign	(negative)	

	ZR (yes)	NZ (no)
Zero		

## Auxiliary

carry	AC (yes)	NA (no)
-------	----------	---------

Parity	PE (even)	PO (odd)
--------	-----------	----------

Carry	CY (yes)	NC (no)
-------	----------	---------

If you enter RF, then the  
current flag settings are

displayed. You can then either:

- Press ENTER to terminate the Register command without changing the flag values, or
- Change the setting of one or more flags by entering the alternate value of the appropriate flags. The new values may be entered in any order, with or without



delimiters.

MS-DOS USER GUIDE

9-36

# Examples

IF you THEN...  
enter...

R

DEBUG displays the contents of all registers, flag settings and the next instruction to be executed.

R IP

DEBUG displays the contents of the Instruction Pointer.  
For example:

IP 0139

013B the contents of the  
Instruction Pointer  
are changed to 013B.

DEBUG displays the  
flag settings. For  
example:

RF

NV UP EI PL NZ NA  
PO NC-  
\_

The Parity flag is set to even (PE), the

PE ZR Zero flag is set (ZR),

DI NG the Interrupt flag is cleared (DI), and the Sign flag is set (NG).

DEBUG displays the new state of the flags:

RF

NV UP D1 NG ZR  
NA PE NC-  
\_

# S (SEARCH)

Searches a specified range for a list of bytes.

S range ,  
list

Where

SYNTAX MEANING  
ELEMENT

range

The range of addresses within which the search is to be made. If you only enter the offset then

the segment  
Indicated by the  
DS register is  
assumed.

list

The list of bytes  
to be searched  
for. Bytes in the  
list must be  
separated by a  
space or a  
comma.



## Characteristics

For each occurrence of the list of bytes within the specified range, DEBUG returns the address of the first byte. If no address is returned, no match was found.

9-38

MS-DOS USER GUIDE

# Example

IF you  
enter...

THEN...

DEBUG  
(displays the  
address of  
every  
occurrence of  
byte value 20  
in the address

range 100 to  
1FF, inclusive.

S100L100,20 For example:

or 058D: 010C

S100,1FF,20 058D: 0110

058D: 0115

058D: 0118

058D: 0120

058D: 0128

058D: 0125

# T (TRACE)

Executes one or more instructions and displays the register contents, flag settings and the next instruction to be executed.

T [= address][ , value]

## Where

SYNTAX  
ELEMENT

MEANING

= address

DEBUG is to  
commence  
execution at  
this address.

The number  
of  
instructions

value

to be  
executed.

## Characteristics

If the = address parameter is not specified then execution begins at CS:IP.

If the value parameter is not specified then only one instruction is executed.

The display generated is of the same format as that of the Register command (without parameters).

# Examples

IF you THEN...  
enter...

five instructions,



starting with the one at location 200 are executed, and the register and flag values following each instruction are displayed along with the next instruction to be executed.

T =  
200,5

the instruction pointed to by CS:IP Is executed and the

T register and flag contents are displayed along with the next instruction to be executed.

9-40

MS-DOS USER GUIDE

# U

## (UNASSEMB)

Disassembles strings of bytes in memory and displays them as assembler-like statements along with their corresponding addresses.

U [range\address]

Table captionWhere

# SYNTAX ELEMENT

# MEANING

range

The range of addresses whose byte values are to be disassembled. If you do not specify the segment then

the segment indicated by the CS register is assumed.

The start of a 32 byte area of memory to be disassembled. If you only enter an

address

offset then  
the segment  
indicated by  
the CS  
register is  
assumed.

## Characteristics



If neither the range nor  
address parameter is

specified, then 32 bytes are disassembled starting at location CS:IP (16 bytes are disassembled in 40 column screen mode).

The number of bytes disassembled may be slightly more than the number you specified. This is because instructions are not always the same length and the final address in a range will not always contain the last byte

of an instruction.

The first address of a range, or the address parameter, must always refer to the first byte of an assembler instruction, otherwise results will be unpredictable.

Table caption Example

IF you enter... THEN...



eight bytes  
starting at  
location  
058D:204 are  
disassembled  
and the result  
displayed:

U058D:204L8 0580:0204  
8D16DFOD  
LEA DX,  
[0DDF]  
05^80:0208  
42 INC OX

0580:0209  
0300 AOO  
OX,AX

0580:0200  
8916E50B  
MOV  
[OBE5],OX

# W (WRITE)

Writes the file being debugged to disk.

W [address[ , drive , sector-a  
, sectors]]

MS-DOS USER GUIDE

9-42

# THE DEBUGGER



SYNTAX MEANING  
ELEMENT

The start address  
of the code in

address

memory that is to be written to disk. If you enter only an offset then the segment indicated in the CS register is assumed.

The drive containing the specified blocks to which code in

drive

memory is to be written. For drive A you must enter 0, for drive B you must enter 1, or for drive C you must enter 2.

The sector number on disk that is the first of a contiguous

sector-a range of sectors to be overwritten with code from memory.

sectors The number of disk sectors blocks to be overwritten with code from memory. The maximum number of

sectors that can be specified is 80 Hexadecimal.



## Characteristics


If you enter the Write command without parameters, then the file is written to disk starting from memory address CS:100. If you specify the address



parameter then the file in memory, starting from the specified address, is written to disk.

In either case, before executing the Write command, BX:CX must be set to the number of bytes to be written. This value was set up correctly when the file was loaded (either by the Load command or the DEBUG command itself).

However, if, since loading the file, you have executed a Go or Trace command, then the value of BX:CX will have been changed. You must be sure this value is set up correctly.



9-43

When the Write command writes a file to disk it obtains

the drive specifier and file name via the file control block set up at CS:5C. If no drive specifier is set up then the default is assumed. This file control block is set up either by the DEBUG command or a subsequent Name command. If it does not indicate the file specifier you require, you must set up this file control block using the Name command.

When the file is written to disk it overwrites the version currently on disk unless the specified file name does not exist, in which case a new file is created.

If all parameters are specified then the code in memory is written to the drive specified by the parameter. The data to be written starts at the memory location specified by the

address parameter, and is written to the blocks on disk specified by the sector-a and sectors parameters. You must therefore be extremely careful to specify the required sectors, since information held there will be destroyed by this operation.

Table captionExamples

IF you enter... THEN...

W

the file in  
memory,  
starting from  
location  
CS:100, is  
written to  
disk with the  
file specifier  
defined by the  
file control  
block set up

at location  
CS:5C.

the file In  
memory,  
starting from  
location  
CS:200, is  
written to  
disk with the  
file specifier  
defined by the

W200

file control  
block set up  
at location  
CS:5C.

20 Hex (32  
decimal)  
sectors on  
drive B  
starting from  
location 1F  
are

W200,1,1F,20 overwritten



with the data  
starting at  
memory  
location  
CS;200.

9-44

MS-DOS USER GUIDE

# ABOUT THIS APPENDIX

This appendix provides a table of ASCII codes and extended keyboard codes.

# CONTENTS

BASIC ASCII CODE A-1

(ISO 7-bit code)

EXTENDED ASCII CODE  
A-5 FOR THE DISPLAY

(ISO 8-bit code)

NATIONAL VARIATIONS  
A-8 IN EXTENDED ASCII

CODE (ISO 8-bit code)

# EXTENDED KEYBOARD

A-14

# **BASIC ASCII CODE (ISO 7-bit code)**

This is a 7 bit code used for Information Interchange. The codes 0 to 31 and code 127 (all decimal numbers) are

Control Codes. These Control Codes are used for Data Transmission, Device Control and Formatting Effects. Note that CTRL shows on the screen as ^ .

DECIMAL	CHARACT
HEXADECIMAL	

0

00



5 05

6 06

7 07

8 08

9 09



10

OA

11

OB

12

OC

13

OD

14

OE

15

OF

16

10

17

11

18

12

19

13

20

14

21

15

22

16

23

17

24

18

25

19

26

1A

27

1B

28

1C

29

ID

30

IE

31

IF

32

20

# DECIMAL HEXADEGMAL CHARACTER CONTROL MEANING

33 21 !

34 22 I >

35 23 #

36 24 \$

37 25 %

38 26 &

39 27 9

40 28 (



41 29 )

42 2A \*

43 2B +

44 2C ,

45 2D

46 2E

47 2F /

48 30 0

49 31 1

50 32 2

51 33 3

52 34 4

53 35 5

54 36 6

55 37 7

56 38 8

57 39 9

58 3A

59 3B I

60 3C <

61 3D =

62 3E >

63 3F ?

64 40 @

65 41 A

66 42 B

67 43 C

68 44 D

69 45 E

70 46 F

71 47 G

A-2

MS-DOS USER GUIDE

DECIMAL HEXADEaMAL  
CHARACTER CONTROL  
MEANING

72 48 H

73 49 I

74 4A J

75 4B K

76 4C L



77 4D M

78 4E N

79 4F 0

80 50 P

81 51 Q

82 52 R

83 53 S

84 54 T

85 55 U

86 56 V

87 57 w

88 58 X

89 59 Y

90 5A z

91 5B [

92 5C \

93 5D ]

94 5E A

95 5F

96 60 '

97 61 a

98 62 b

99 63 c

100 64 d

101 65 e

102 66 f

103 67 g

104 68 h

105 69 i

106 6A j

107 6B k

108 6C l

109 6D m

DECIMAL	CHARACTER
HEXADECIMAL	

111	6F
-----	----

112	70
-----	----

113	71
-----	----

114	72
-----	----



115

73

116

74

117

75

118

76

119

77

120

78

121

79

122

7A

123

7B

124

7C

125

7D

126

7E

127

7F

A-4

# MS-DOS USER GUIDE

**EXTENDED  
ASCII CODE  
FOR THE  
DISPLAY  
(ISO 8-bit  
code)**

This table shows the 256 elements of the extended ASCII character set, together with their decimal and hexadecimal equivalents.

## DEC: HEX CHARACTER DE

000	00	(nul)	016
-----	----	-------	-----

001	01	©(SOH)	017
-----	----	--------	-----

002 02 9 (STX) 018

003 03 V (ETX) 019

004 04 ♦ (EOT) 020

005 05 ♦ (ENQ) 021

006	06	4 (ACK)	022
007	07	• (BEL)	023
008	08	D (BS)	024
009	09	O (HT)	025
010	OA	@ <LF)	026



Oil	OB	CT (VT)	025
012	OC	Of	028
013	OD	/> (CR)	029
014	OE	(SO)	030
015	OF	^ (SI)	031

# Tab. A-1 Extended ASCII Character Set

DEC.	HEX	CHARACTER	DEC.	HEX	CHARACTER	DEC.	HEX	CHARACTER	DEC.	HEX	CHARACTER
000	00	BLANK (NUL)	016	10	▶ (DLE)	032	20	SPACE (SP)	048	30	0
001	01	☺ (SOH)	017	11	◀ (DC1)	033	21	!	049	31	1
002	02	☹ (STX)	018	12	↑ (DC2)	034	22	”	050	32	2
003	03	♥ (ETX)	019	13	!! (DC3)	035	23	#	051	33	3
004	04	♦ (EOT)	020	14	¶ (DC4)	036	24	\$	052	34	4
005	05	♣ (ENQ)	021	15	§ (NAK)	037	25	%	053	35	5
006	06	♠ (ACK)	022	16	▬ (SYN)	038	26	&	054	36	6
007	07	• (BEL)	023	17	↓ (ETB)	039	27	,	055	37	7
008	08	■ (BS)	024	18	↑ (CAN)	040	28	(	056	38	8
009	09	○ (HT)	025	19	↓ (EM)	041	29	)	057	39	9
010	0A	● (LF)	026	1A	→ (SUB)	042	2A	*	058	3A	:
011	0B	♂ (VT)	027	1B	← (ESC)	043	2B	+	059	3B	;
012	0C	♀ (FF)	028	1C	└ (FS)	044	2C	,	060	3C	<
013	0D	♪ (CR)	029	1D	↔ (GS)	045	2D	—	061	3D	=
014	0E	♫ (SO)	030	1E	▲ (RS)	046	2E	.	062	3E	>
015	0F	☼ (SD)	031	1F	▼ (US)	047	2F	/	063	3F	?

DEC: HEX CHARACTER DE

064 40 @ 080

065 41 A 081

066 42 B 082

067 43 C 083

068 44 D 084

069 45 E 085

070 46 F 086

071 47 G 087

072 48 H 088

073	49	I	089
074	4A	J	090
075	4B	K	091
076	4C	L	092

077	4D	M	093
-----	----	---	-----

078	4E	N	094
-----	----	---	-----

079	4F	O	095
-----	----	---	-----

Tab. A-1 Extended ASCII  
Character Set (cent.)

A-6

MS-DOS USER GUIDE

# DEC HEX CHARACTER DEC

128	80	C	144
-----	----	---	-----

129	81	ii	145
-----	----	----	-----

130	82	e	146
-----	----	---	-----

131	83	a	147
-----	----	---	-----

132 84 a 148

133 85 a 149

134 86 a 150

135 87 ? 151



136	88	e	152
137	89	e	153
138	8A	e	154
139	8B	I	155
140	8C		156

141	8D	1	157
-----	----	---	-----

142	8E	A	158
-----	----	---	-----

143	8F	A	159
-----	----	---	-----

Table caption Tab. A-1  
Extended ASCII Character

Set (cent.)

DEC HEX CHARACTER DEC

192 CO L 208

193 Cl -L 209

194 C2 “T 210

195	C3	h	211
196	C4	—	212
197	C5	+	213
198	C6	t=	214
199	C1	Ih	215

200	C8	Ik	216
201	C9	r	217
202	CA		218
203	CB	T	219
204	CC	11=	220

205 CD = 221

206 CE JL  
nr 222

207 CF 223

Table captionTab. A-1

# Extended ASCII Character Set (cont.)

**NATIONAL  
VARIATIONS  
IN  
EXTENDED  
ASCII CODE  
(ISO 8-bit**



# code)

For Denmark, Norway, Greece and Portugal certain characters are displayed differently. These characters and their decimal and hexadecimal codes are shown in the following table.

A-8

MS-DOS USER GUIDE

# DEC HEX CHARACTER DEC

128	80	C	144
-----	----	---	-----

129	81	ii	145
-----	----	----	-----

130	82	e	146
-----	----	---	-----

131	83	a	147
-----	----	---	-----

132	84	a	148
-----	----	---	-----

133	85	a	149
-----	----	---	-----

134	86	k	150
-----	----	---	-----

135	87	?	151
-----	----	---	-----

136	88	e	152
137	89	e	153
138	8A	e	154
139	8B	1	155
140	8C	1	156

141	8D	i	157
-----	----	---	-----

142	8E	A	158
-----	----	---	-----

143	8F	A	159
-----	----	---	-----

Table caption Tab. A-2  
National Characters for  
Denmark and Norway

DEC	HEX	CHARACTER	DEC	HEX	CHARACTER	DEC	HEX	CHARACTER	DEC	HEX	CHARACTER
128	80	Ç	144	90	É	160	A0	á	176	B0	␣
129	81	ü	145	91	æ	161	A1	í	177	B1	␣
130	82	é	146	92	Æ	162	A2	ó	178	B2	␣
131	83	â	147	93	ô	163	A3	ú	179	B3	
132	84	ä	148	94	Ö	164	A4	ñ	180	B4	┌
133	85	à	149	95	ò	165	A5	Ñ	181	B5	┐
134	86	â	150	96	û	166	A6	õ	182	B6	┌┐
135	87	ç	151	97	ù	167	A7	Ô	183	B7	┐┌
136	88	ê	152	98	ÿ	168	A8	ı	184	B8	┐┐
137	89	ë	153	99	Ö	169	A9	ã	185	B9	┐┐┐
138	8A	è	154	9A	Ü	170	AA	Ã	186	BA	┐┐┐┐
139	8B	ï	155	9B	ø	171	AB	ł	187	BB	┐┐┐┐┐
140	8C	î	156	9C	£	172	AC	'n	188	BC	┐┐┐┐┐┐
141	8D	ì	157	9D	Ø	173	AD	ı	189	BD	┐┐┐┐┐┐┐
142	8E	Ä	158	9E	Ł	174	AE	³	190	BE	┐┐┐┐┐┐┐┐
143	8F	Å	159	9F	ı	175	AF	☒	191	BF	┐┐┐┐┐┐┐┐┐

DEC HEX CHARACTER DEC

192 CO L

208

193	C1		209
194	C2	“T	210
195	C3	h	211
196	C4	—	212
197	C5	-1	213

198	C6	1=	214
199	C7	Ih	215
200	C8	L	216
201	C9	r	217
202	CA	JL	218



203	CB	T	219
204	CC	11=	220
205	CD	=	221
206	CE	JL nr	222

Tab. A-2 National Characters  
for Denmark and Norway  
(cont.)

MS-DOS USER GUIDE

A-10

DEC HEX CHARACTER DEC

128	80	A	144
129	81	B	145
130	82	r	146
131	83	A	147
132	84	E	148

133 85 Z 149

134 86 H 150

135 87 e 151

136 88 I 152

137 89 K 153

138 8A A 154

139 8B M 155

140 8C N 156

141 8D 157

142 8E 0 158

Table captionTab. A-3  
National Characters for  
Greece

DEC	HEX	CHARACTER	DEC	HEX	CHARACTER	DEC	HEX	CHARACTER	DEC	HEX	CHARACTER
128	80	A	144	90	P	160	A0	ι	176	B0	␣
129	81	B	145	91	Σ	161	A1	κ	177	B1	␣
130	82	Γ	146	92	T	162	A2	λ	178	B2	␣
131	83	Δ	147	93	Υ	163	A3	μ	179	B3	
132	84	E	148	94	Φ	164	A4	ν	180	B4	┐
133	85	Z	149	95	X	165	A5	ξ	181	B5	≡
134	86	H	150	96	Ψ	166	A6	ο	182	B6	≡
135	87	Θ	151	97	Ω	167	A7	π	183	B7	⌞
136	88	I	152	98	α	168	A8	ρ	184	B8	⌞
137	89	K	153	99	β	169	A9	σ	185	B9	≡
138	8A	Λ	154	9A	γ	170	AA	ς	186	BA	
139	8B	M	155	9B	δ	171	AB	τ	187	BB	⌞
140	8C	N	156	9C	ε	172	AC	υ	188	BC	┘
141	8D	Ξ	157	9D	ζ	173	AD	φ	189	BD	┘
142	8E	O	158	9E	η	174	AE	χ	190	BE	┘
143	8F	Π	159	9F	θ	175	AF	ψ	191	BF	⌞

DEC HEX CHARACTER DEC

192 CO L

208

193	C1	JL	209
194	C2	T	210
195	C3	h	211
196	C4	—	212
197	C5	+	213



198	C6	1=	214
199	C7	II	215
200	C8	IL	216
201	C9	r	217
202	CA	JL	218

203	CB	T	219
204	CC	11=	220
205	CD	=	221
206	CE	JL nr	222

Table caption Tab. A-3  
National Characters for  
Greece (cont.)

A-12

MS-DOS USER GUIDE

DEC HEX CHARACTER DEC

128	80	C	144
129	81	ii	145
130	82	e	146
131	83	a	147
132	84	a	148

133 85 a 149

134 86 a 150

135 87 9 151

136 88 e 152

137 89 e 153

138 8A e 154

139 8B 1 155

140 8C i 156

141 8D 1 157

142 8E A 158

143 8F A

159

## Tab. A-4 National Characters for Portugal

DEC	HEX	CHARACTER	DEC	HEX	CHARACTER	DEC	HEX	CHARACTER	DEC	HEX	CHARACTER
128	80	Ç	144	90	É	160	A0	á	176	B0	☐
129	81	ü	145	91	æ	161	A1	í	177	B1	☐
130	82	é	146	92	Æ	162	A2	ó	178	B2	☐
131	83	â	147	93	ô	163	A3	ú	179	B3	
132	84	ä	148	94	ö	164	A4	ñ	180	B4	└
133	85	à	149	95	ò	165	A5	Ñ	181	B5	═
134	86	â	150	96	û	166	A6	õ	182	B6	┘
135	87	ç	151	97	ù	167	A7	Õ	183	B7	┐
136	88	ê	152	98	ÿ	168	A8	ı	184	B8	┘
137	89	ë	153	99	Ö	169	A9	ã	185	B9	┘
138	8A	è	154	9A	Ü	170	AA	Ã	186	BA	
139	8B	ï	155	9B	Á	171	AB	Ú	187	BB	┐
140	8C	î	156	9C	£	172	AC	Í	188	BC	┘
141	8D	ì	157	9D	À	173	AD	ı	189	BD	┘
142	8E	Ä	158	9E	Ê	174	AE	³	190	BE	┘
143	8F	Å	159	9F	Ô	175	AF	Ó	191	BF	┘

DEC HEX CHARACTER DEC

192 CO L

208



193	C1	J_	209
194	C2	T	210
195	C3	1	211
196	C4	—	212
197	C5	+	213

198	C6	l=	214
199	C7	n	215
200	C8	lk	216
201	C9	r	217
202	CA	JL	218

203	CB	T	219
204	CC	11=	220
205	CD	=	221
206	CE	Hr	222
207	CF		223

# Tab. A-4 National Characters for Portugal (cont.)

**EXTENDED  
KEYBOARD  
CODES  
(FOR USA  
KEYBOARDS**



Certain keys and key combinations do not produce ASCII code (one byte).

Instead they produce two bytes; the first byte is always zero. This zero value indicates an extended keyboard code. The following table shows the value of the second byte, when these key(s) are pressed.



A-14

# MS-DOS USER GUIDE

## DECIMAL HEXADECIMAL

3

03

15

OF

16

10

17

11

18

12



19

13

20

14

21

15

22

16

23

17

24

18

25

19

30

IE

31

IF

32

20

33

21

34

22

35

23

36

24

37

25

38

26

44

2C

45

2D

46

2E

47

2F

48

30

49

31

50

32

59

3B

60

3C

61

3D

62

3E

63

3F

64

40

65

41

66

42



67

43

68

44

71

47

DECIMAL HEXADECIMAL

72

48

73

49

75

4B

77

4D

79

4F

80

50

81

51

82

52

83

53

84

54

85

55

86

56

87

57

88

58

89

59

90

5A

91

5B

92

5C

93

5D

94

5E

95

5F

96

60

97

61

98

62

99

63

100

64

101

65



102

66

103

67

104

68

105

69

106

6A

107

6B

108

6C

109

6D

110

6E

111

6F

112

70

113

71



A-16

# MS-DOS USER GUIDE

**ASCII  
DISPLAY  
AND  
KEYBOARD  
CODE  
TABLES**

# DECIMAL HEXADECIMAL

114

72

115

73

116

74

117

75

118

76



119

77

120

78

121

79

122

7A

123

7B

124

7C

125

7D

126

7E

127

7F

128

80

129

81

130

82

131

83

132

84

## Note

Codes 120 to 131 decimal refer to the top row of keys in the body of the keyboard, not those on the right hand keypad.

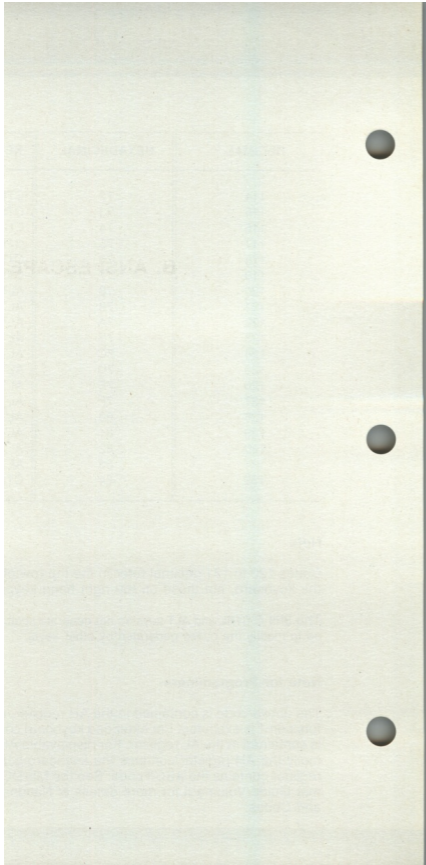
The SHIFT, CTRL and ALT keys do not generate specific codes; they are used to

modify the codes generated by other keys.

## Note for Programmers

The above code is contained in the AH register, when BIOS INT 16H function 1 is executed. For extended keyboard codes 00 Hexadecimal is contained in the AL register. For displayable ASCII characters, normally the AH

register contains the keyboard scan code and the AL register contains the ASCII code. See the MS-DOS System Programmer Guide Volume II for more details of National Keyboard Layouts and Codes.





# ABOUT THIS APPENDIX

This appendix describes the ANSI escape sequences.



## CONTENTS

INTRODUCTION	B-1
CURSOR MOVING FUNCTIONS	B-2
ERASING FUNCTIONS	B-4
GRAPHIC MODE FUNCTIONS	B-5

# KEY REASSIGNMENT

B-  
7



# INTRODUCT

An ANSI escape sequence is a series of characters beginning with the character ESC (decimal 27, hex 1B) that can be generated to define functions to MS-DOS. Specifically you can affect cursor movement, erase specific areas of the screen and set the graphics mode, by

issuing these sequences through standard MS-DOS input/output.

The sequences described in this appendix require the presence of the ANSI console driver. ANSI.SYS is a file included on your MS-DOS system diskette. To install the ANSI console driver the following command must be placed in the CONFIG.SYS file:

DEVICE = [d:]\path]  
ANSI.SYS

Refer to Appendix C for information about the CONFIG.SYS file and the DEVICE command.

## Remarks

The following notes are general to all escape sequences:

1. ESC can be generated in a variety of ways: changing your prompt (PROMPT = \$e); using the Video File Editor (see Chapter 6); or by program.

2. Pn represents a numeric parameter. This is a decimal number.

3. Ps represents a selective parameter. The parameter is still a decimal integer but is

one that must be selected from a list of alternatives.

4. Where no parameter is specified, or where zero is entered, a default value is assumed.

5. No spaces should be typed inside escape sequences; any space shown in the specification is purely for ease of reading.



# CURSOR MOVING FUNCTIONS

The following escape sequences affect the the screen.

position of the cursor on

# MNEMONIC SEQUENCE M

CUP

(Cursor  
Position)

ESC [Pn ■,  
Pn^\

Th  
is  
the  
co  
sp  
the  
se

pa  
res

or

or

Th

va

If

HVP E  
(Horizontal  
and Vertical  
Position)

ESC [Pn -  
.Pni

no

pa

are

the

me

the

po

cuu

(Cursor Up)

ESC [ Pn A

M  
cu  
the  
by  
nu  
ro  
sp  
the  
pa  
If  
pa  
sp  
on  
as

ac  
tal  
cu  
ali  
the  
of  
sci

M  
cu  
the  
by  
nu

CUD

(Cursor  
Down)

ESC [ Pn B

ro  
sp  
the  
pa  
If  
pa  
sp  
on  
as  
ac  
tal  
cu  
alr  
the



B-2

MS-DOS USER GUIDE

MNEMONIC SEQUENCE M

CUF

(Cursor

ESC [ Pn C

M  
cu  
the  
the  
of  
sp  
the  
pa  
If  
pa  
sp  
the



Forward)

co  
as  
No  
tal  
cu  
al  
the  
rig  
co

M  
cu  
by

CUB

(Cursor

Backward)

ESC [ Pn D

nu  
co  
sp  
the  
pa  
If  
pa  
sp  
the  
co  
as  
No  
tal  
cu

DSR  
(Device  
Status  
Report)

ESC [ 6 n

all  
the  
me  
co

Ca  
co  
dri  
pe  
CF  
(C  
Po

Re  
sec

Th  
cu  
po  
rej  
the  
Inj  
de  
fir  
pa  
sp

CPR

(Cursor.

Position  
Report)

ESC [Pn -  
PnR

the  
the  
sp  
the  
Th  
se  
pe  
by  
co  
dr  
re  
DS  
se

# MNEMONIC SEQUENCE M

SCP

(Save Cursor ESC [ s

Th  
cu  
po  
sa  
sa  
ca  
su  
be  
by

Position)

an  
(R  
Cu  
Po  
se

RCP

Th  
cu  
po  
res  
wh

(Restore  
Cursor  
Position)

ESC ( u

at  
tha  
co  
Dr  
rec  
SC  
se



# ERASING FUNCTIONS

The following sequences  
erase specific areas of the  
video display.

MNEMONIC SEQUENCE M

ED

ESC [ 2 J

Is  
an

(Erase  
Display)

cu  
mo  
the  
po

EL

ESC I K

Th  
cu  
lin  
the

(Erase Line)

cursor position  
to the end of  
the line is  
erased,  
including  
the

cu  
po  
its

B-4

# MS-DOS USER GUIDE

# GRAPHIC MODE FUNCTIONS

MNEMONIC

SEQUENCE

MEANING

SGR

## (Set Graphics Rendition)

ESC [ Ps ; ... ;Ps m

The graphic functions specified by the parameters are invoked. The functions set by this sequence remain in effect until another SGR sequence is issued. The parameter values are as follows:

0 - all attributes off (normal

display)

1 - high intensity (bold)

5 - sets blink on

7 - reverse video on

8 - concealed on (makes

display invisible)

30 - black foreground

31 - red foreground

32 - green foreground

33 - yellow foreground

34 - blue foreground on col

or display; underline on  
monochrome display

35 - magenta foreground

36 - cyan foreground



MEANING

MNEMONIC

SEQUENCE

37 -white foreground 40 -  
black background

41 - red background

42 -green background

43 -yellow background

44 -blue background

45 -magenta background

46 -cyan background

47 -white background

SM

(Set Mode)

ESC [ = Ps h

The screen width and type

are set by the parameter specified. The possible parameter values are as follows:

0 - 40 column by 25 line

black and white

1 - 40 column by 25 line

color

2 - 80 column by 25 line

black and white

3 - 80 column by 25 line

color

4 - 320 X 200 color



B-6

MS-DOS USER GUIDE

# ANSI ESCAPE SEQUENCES

---

MNEMONIC SEQUENCE M

an

wh

6 -

20

an

wh

7 -

au

tex

at

RM

ESC [= Ps\

(Reset  
Mode)

en

Re

att

set

the

se

Pa

va

the

as

SM  
sec

Except for

7 -  
wr  
en  
(ex  
ch  
are  
dis



# KEY REASSIGNMENT

An ANSI escape sequence can be used to assign an ASCII code, a string, or a combination of ASCII codes and strings to any key or value or valid key-stroke combination.

Valid control sequences are:

ESC [ Pn ■, Pn p

ESC [ Pn ; "string" p

ESC [ Pn ; "string"] Pn] Pn]  
"string"] Pn p

or any other combination of strings and decimal numbers.

The first parameter defines which ASCII code is to be

mapped, unless it is zero, in which case the first and second parameters comprise the extended keyboard code. Refer to Appendix A for a complete list of ASCII and extended keyboard codes.

Note that the ASCII codes must be entered in decimal.

## Examples

If your program  
issues the  
sequence...

THEN...

ESC[66;82p

pressing  
becomes  
instead of  
B.

a string  
"dir" is  
entered

ESC[0:67:”dir”r13p

followed  
by a  
carriage  
return  
whenever  
you press  
F9. The  
initial 0  
indicates  
that the F  
function  
key is  
represented  
by an

extended  
keyboard  
code 67.

B-8

MS-DOS USER GUIDE

# ABOUT THIS APPENDIX

This appendix describes how you can configure MS-DOS to suit the requirements of your application.

# CONTENTS

INTRODUCTION C-1

CONFIGURING C-1

COMMANDS

BREAK c-1



BUFFERS C-2

COUNTRY C-3

DEVICE C-5

DRIVPARM C-6

FCBS C-8

FILES

C-9

LASTDRIVE

C-10

SHELL

C-11

STACKS

C-12

# INTRODUCT

Whenever MS-DOS is initialized, it searches the root directory for a file named CONFIG.SYS. If this file exists, it is read and the configuring commands within it are executed. If CONFIG.SYS is not found the default setting for each command is used.

You can create or modify the CONFIG.SYS file using the Video File Editor or Line Editor, and include in it any of the commands described in this appendix.

Also if a particular command within CONFIG.SYS is not found its default setting is used.

# CONFIGURATION COMMANDS

# BREAK

Sets or resets the CTRL C break facility.

BREAK = ON | OFF

## Characteristics

The action of the BREAK command is exactly the same as described in Chapter 5. However, it can only occur

once in the CONFIG.SYS file. The state set by including this command in the CONFIG.SYS file can later be reversed by entering a BREAK command with the opposite argument at the keyboard.

The default is OFF.

# BUFFERS

Sets the number of buffers to be allocated.

BUFFERS = number

Table captionWhere

SYNTAX  
ELEMENT

MEANING



number

A decimal number in the range 1 to 99 that defines the number of buffers to be allocated.

The default varies between 2 and 10, depending on

the computer  
and its  
configuration.

## Characteristics

The optimum number of buffers depends on factors such as:

- the kind of disk drives attached to the computer

- whether external devices such as spooling tape are attached
- the type of applications most commonly used
- the amount of main memory
- whether the computer is acting as a network server

A minimum setting of

**BUFFERS = 5** is recommended for all computer configurations with disk drives, unless you have severe main memory constraints.

C-2

## MS-DOS USER GUIDE

If your applications perform a lot of random accesses you will probably benefit by

using a larger number of buffers. However, beyond a certain number it may take MS-DOS longer to scan the buffers than it would to access a value from disk, resulting in a drop in performance. For most data base applications between 10 and 20 buffers is a reasonable number.

Memory size affects the optimum number of buffers

because each additional buffer requires 528 bytes of memory.

The optimum number of buffers can only be determined by trial then timing the result.

# COUNTRY

Sets the country to allow MS-DOS to use the correct national time, date, currency and decimal separators.

COUNTRY = number

Where

SYNTAX ELEMENT

# MEANING

number

A three digit number which is the telephonic international country code:

001 United States of America

002 Canadian-French

031 Netherlands



032 Belgium

033 France

034 Spain

039 Italy

041 Switzerland

044 United Kingdom

045 Denmark

046 Sweden

047 Norway

049 Germany

061 Australia

351 Portugal

358 Finland

Note: If your country is not supported, choose the country which uses your

national conventions.

The default is COUNTRY =  
001

C-4

MS-DOS USER GUIDE

DEVICE =

Installs a device driver.

DEVICE = filespec

Where

SYNTAX ELEMENT	MEANING
-------------------	---------

filespec	The file containing the device driver.
----------	--

Characteristics

If filespec is the file ANSI.SYS, the escape sequences described in Appendix B are supported.

If filespec is DRIVER.SYS, refer to Appendix G for further details.

If filespec is VDISK.SYS, refer to Appendix F for further details.

Alternatively, you may enter

the file name of any device driver written for your system.

## Important

The device driver file must be in the root directory of the drive used for bootstrapping the computer.

Note: DEVICE = filespec can be repeated on several lines of the CONFIG.SYS file with

different device driver  
parameters.

# DRIVPARM

Enables you to override the default settings for predefined block devices.

DRIVPARM

= /D: drive-

no [/C] [/F:

form-factor]

[/H: head ]



[/N] [/S:  
sectors] [/T:  
tracks]

Where

SWITCH      PARAMETER N

S  
t  
d

/D

drive-no

n  
b  
a  
S  
=  
e

I  
c  
s  
r  
S  
,

1C

i  
5  
d

S  
t

/F

form-factor

f  
k  
s

0

K

1

2

3

s

d

4

d

d

5

I

6

I

7

C-6

MS-DOS USER GUIDE

SWITCH PARAMETER MEANING

/H

heads

Spec  
the r  
of he  
the c  
drive  
valu  
rang  
1 to  
The  
Is 2

Spec  
non-

/N

remo  
bloc  
devi  
such  
hard

Spec  
the r  
of se  
per t  
Its v  
can 1

/S

sectors

from  
99.7  
defa  
nine  
secto  
track

Spec  
the r  
of tr  
per s  
Its v



/T	tracks	can 1
		from
		throu
		999.
		defa
		80 tr
		per s

## Note

DRIVPARM can be repeated on several lines of the CONFIG.SYS file with different parameters.

## Example

You might have a computer with an internal tape drive unit on drive "D:" that is configured at boot time to write 20 tracks of 40 sectors per track. If you want to reconfigure this tape drive to write 10 tracks of 99 sectors each, you can put the following line in your CONFIG.SYS file:

DRIVPARM = /D:3 /F:6 /H:1  
/S:99 /T:10

This overrides the default device driver settings, and supports a tape drive as drive "D:" (in this case the logical and physical drive numbers are identical). This tape drive has 1 head, and supports a tape format of

10 tracks and 99 sectors per track. (This assumes that the

device driver for the tape device supports this configuration of tracks and sectors). You might want to use this method to create a tape that you can read on another computer that can only read this alternate format.

See the “MS-DOS Software Installation Guide” for more details.

== FCBS

Table caption Defines the number of FILES opened with File Control Blocks that can

be open at any one time.

FCBS=

maxopen,number

Where

SYNTAX  
ELEMENT

MEANING

This is a  
number from  
1 to 255  
which  
represents

maxopen

the maximum number of files that can be opened with File Control Blocks. The default value is 4.

This is a number from

number

0 to 255 that specifies the files that MS-DOS cannot close automatically if the application tries to open more than maxopen. The first files opened are the protected



files. The default value of number i is 0.

Note: This command is only applicable when you are connected to a network. When not connected to a network, the number of files that can be open with File Control Blocks is 255.

# MS-DOS USER GUIDE

C-8

# FILES

Defines the number of file handles that can be open at any one time.

FILES =  
number

Where

# SYNTAX

## ELEMENT MEANING

number

A decimal number in the range 1 to 255 defining the number of files that can be open concurrently.

The default is 8.

# Characteristics

Each additional file above the default value of 8 requires an extra 48 bytes of memory. These 8 file handles include the 5 predefined file handles for: the console, standard input, standard output, auxiliary output and printer output.

The maximum number of file handles a program can have

open is 20, including the 5 predefined file handles.

When there are foreground and background processes, or in a multi-tasking environment, or on a server; each process can have 20 handles open. In these cases it is useful to set the number of handles larger than 20, the exact setting depends on the number of processes and the requirements of each process.

# LASTDRIVE

Sets the maximum number of drives that you may access

LASTDRIVE = drive-letter

Table caption Where

SYNTAX	MEANING
ELEMENT	

drive-letter Any letter from A through Z, defining the last valid drive that MS-DOS will accept.

Characteristics The default is:

LASTDRIVE = E

In a network environment,



you may use the NET USE command to assign the extra drive letters to a volume on a remote computer.

## Remarks

You cannot set LASTDRIVE less than the number of drives on your computer.

C-10

MS-DOS USER GUIDE

# SHELL

Loads a top-level command processor, which can be COMMAND.COM or an alternative command processor.

SHELL = pathname path [/E:  
nnnnn ] /P

Where

# SYNTAX MEANING ELEMENT

o

pathname      The drive, path  
and file name  
containing the  
toplevel  
command  
processor to be  
loaded.

path

The drive and path of the directory containing the top-level command processor. This sets the environment variable COMSPEC to point to the command processor for

reloading  
purposes.

This switch  
specifies the  
environment size,  
where nnnnn is  
the size in bytes.  
The size may  
range between  
160 and 32768

/E.nnnnn

bytes. The default  
value is

160 bytes.

This switch must  
be specified. This  
is to indicate that  
the command  
processor is to be  
used for system  
startup. If omitted  
AUTOEXEC.BAT

IP

will not be  
executed and an  
EXIT causes a  
system crash.



---

## Example

```
SHELL =  
C:\BIN\COMMAND.COM  
C:\BIN /E:16384 /P
```

This command causes the top level command processor to be loaded from the BIN directory of the C: drive. The second parameter indicates that the transitory part of the command processor is to be reloaded from the BIN directory of the C: drive. The /E switch sets the environment size to be 16384 bytes. The /P switch must be specified to indicate this is the top level command



processor.

## Remarks

If you are writing an alternative command processor, remember to duplicate COMMAND.COM's internal commands, batch processor and program loader.

# STACKS

Starting with MS-DOS Version 3.20 hardware interrupt stacking uses a separate stack, rather than the user stack as in previous versions. This command sets the number and size of stack frames available for hardware interrupt stacking.

`STACKS = frames,size`

C-12

# MS-DOS USER GUIDE

# CONFIGURATI MS-DOS

Where

SYNTAX  
ELEMENT MEANING

frames

Specifies the number of stack frames available.

This number can range from 8 through 64. The default number is 9.

Specifies the size in bytes of each stack frame. This

size                      number can  
range from 32  
through 512. The  
default is 128.

## Characteristics

When a hardware interrupt occurs, MS-DOS allocates a stack frame from the available frames. When the BIOS has processed the interrupt, MS-DOS frees the

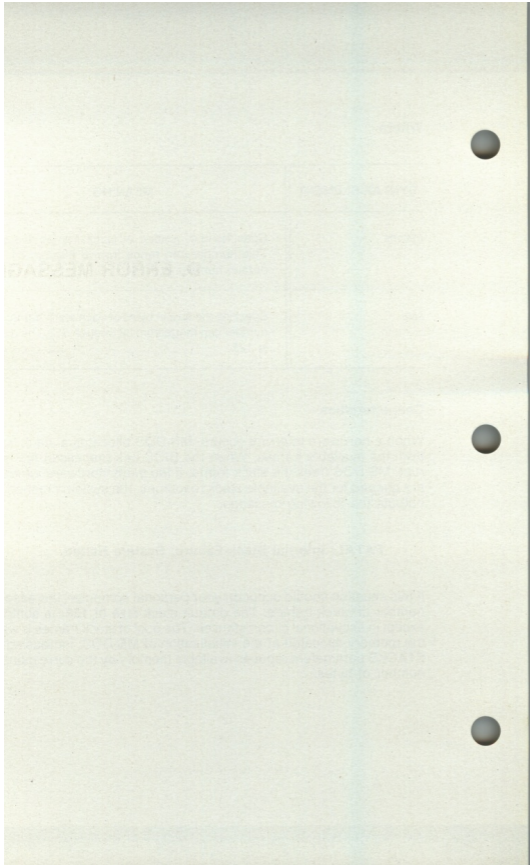
stack frame. If too many hardware interrupts are queued for the available stack resources, the system crashes and outputs the following message:

**FATAL: Internal Stack Failure, System Halted.**

If this situation should occur on your personal computer, increase the number of stack frames. The default stack

size of 128, is sufficient except in exceptional circumstances. The pool of stack frames is within the memory allocated at the initialization of MS-DOS: increasing the STACKS parameters reduces available memory by the corresponding number of bytes.





# ABOUT THIS APPENDIX

This appendix explains the various error messages that can be displayed by MS-DOS and its utilities.

# CONTENTS

DEVICE AND DISK DRIVE  
ERRORS D-1

ERROR MESSAGES IN  
ALPHABETICAL ORDER  
D-3

# **ERROR MESSAGES**

# DEVICE AND DISK DRIVE ERRORS

Errors may occur when reading from or writing to devices and disk drives. These errors cause the

system to stop and output a message of this form:

type error action unit Abort,  
Retry, Ignore?

Where

type error

action

Specifies the possible causes  
of device or disk drive

failure. The Table D-1  
contains the message type.

Can be reading or writing.

unit

Can be either:

or

device device-name drive  
drive-letter

It specifies the device or disk drive in error.

device-name See the section "Reserved Device Names" in Chapter 3 for a list of device-names.

drive-letter A single letter in the range "A" through "Z".

## Response

When you receive one of



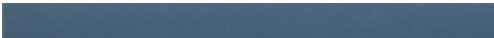
these messages, do one of the following:

Enter A for Abort. The system ends the program that requested the read or write.

Enter R for Retry. The system tries the read or write operation again.

Enter I for Ignore. The system ignores the error and attempts to continue the

program. (This method may cause loss of data).



# Important

For disk drive error messages, do not change disks before responding with A, R or I. The only exception is "Invalid Disk Change".

The following table lists possible causes of device and disk drive errors; these are described along with the rest of the error messages in the section "Error Messages In Alphabetical Order".

Bad call format Bad  
command Bad unit Data

Disk error FCB unavailable  
General Failure Invalid Disk  
Change Lock Violation No

paper Non-DOS disk Not  
ready Read fault Sector not  
found Seek

Sharing buffer exceeded  
Sharing Violation Write fault  
Write protect

Tab. D-1 Possible Causes of  
Device and Disk Drive Errors

# Examples

The following example is a typical error message displayed when the printer connected to the computer is switched off:

**Not ready error  
writing device  
PRN Abort,  
Retry, Ignore?**

## MS-DOS USER GUIDE

### ERROR MESSAGES

The following example is a typical error message displayed when there is no floppy diskette in the disk drive:

**Not ready error**

**reading drive A**  
**Abort, Retry,**  
**ignore?**

# **ERROR MESSAGES IN ALPHABETIC ORDER**

The following list contains a description of the possible



cause and meaning of the message and where possible suggests remedial action.

ERROR MESSAGE MEANING

MS-DOS displays this message when you choose the Q (Quit)

Abort edit  
(Y/N)?

(EDLIN)

command in  
EDLIN. The  
Quit command  
exits the editing  
session without  
saving any  
editing changes.  
Specify Y (for  
"Yes") or N (for  
"No").

MS-DOS  
displays this

## Access denied (MS-DOS)

message when you tried to write to or delete a file marked as read only. If you really want to carry out this action, use the ATTRIB command to give the file a read/ write attribute.

All files  
canceled  
by operator  
(PRINT)

MS-DOS  
displays this  
message when  
you specify the  
/T switch with  
the PRINT  
command.

All  
partitions  
are

Self-

currently explanatory.  
in use  
(FDISK)

All All files are  
specified allocated  
files are contiguously on  
contiguous the disk without  
(CHKDSK) fragmentation.

The size of the

file indicated in the directory was not consistent with the amount of data actually allocated to the file.

Allocation

error size

adjusted

(CHKDSK)

Adjustment

actually takes

place only, if

you specify the

/F switch with

CHKDSK, the file is truncated at the end of the last valid cluster.



D-4

MS-DOS USER GUIDE

# ERROR MESSAGES

ERROR  
MESSAGE

MEANING

Ambiguous  
switch:z

The  
characters in  
z identify



(LINK)

more than  
one linker  
parameter.

Amount read  
less than size  
in

The file is  
smaller than  
its header  
indicates.

header

Recompile  
(or

(EXE2BIN)

reassemble)  
and relink the  
program.

An internal failure has occurred (LINK)

The linker program has failed. Report the conditions of the failure to your Olivetti Dealer.

Attempt to access data

outside of  
segment  
bounds,  
possibly bad  
object module  
(LINK)

An invalid  
object file  
has been  
specified.

Attempt to  
write on  
writeprotected

You cannot  
format a  
write-  
protected  
diskette. Use  
another disk

diskette  
(FORMAT)

or remove the  
write-  
protection  
tag.

Backup file  
sequence  
error  
(RESTORE)

The file being  
restored is  
backed up on  
more than  
one diskette.  
The wrong  
diskette in  
the sequence

has been  
inserted.

Bad call  
format  
(device error)

A request  
header of  
Incorrect  
length was  
passed to a  
device driver.  
Contact your  
Olivetti  
dealer.

# ERROR MESSAGE MEANING

Bad  
command  
(device  
error)

A device driver issued an incorrect command to the device specified in the error message

You typed neither

Bad  
command  
or file  
name (MS-  
DOS)

an Internal  
command nor an  
external command  
(executable  
filename).

You either  
mistyped the  
command or the  
file does not exist  
in the specified  
disk directories.

Bad  
numeric  
parameter  
(LINK)

The value specific  
with the /STACK  
parameter is not a  
valid numeric  
constant.

The MS-DOS disk  
being loaded does  
not contain the file  
COMMAND.COM  
in the root  
directory.



Bad or  
missing

Command  
Interpreter  
(MS-DOS)

This message may also appear if an error occurs during loading of the system disk or if the COMSPEC = parameter does not point to a directory containing COMMAND.COM

One of the

following  
conditions  
occurred during  
startup;

Bad or  
missing  
filename  
(MS-DOS)

The device driver  
named In the  
DEVICE =  
parameter does not  
exist in  
CONFIG.SYS.

A break address  
has been set which

is out of bounds for  
the machine.

An error occurred  
during loading of  
the driver.

D-6

## MS-DOS USER GUIDE

ERROR MESSAGES ■ V : ^

ERROR

MEANING

# MESSAGE

Bad  
Partition  
Table  
(FORMAT)

You have either  
tried to  
FORMAT a  
hard disk  
partition which  
does not exist or  
the partition's  
partition table is  
bad or invalid.  
If there is an  
existing

partition, delete it using FDISK. Create a new partition and try formatting it again.

Bad unit  
error  
(device  
error)

An invalid  
subunit number  
has been sent to  
a device driver.  
Contact your

dealer.

Batch file  
missing  
(MS-DOS)

A file being  
processed in  
batch mode is  
no longer  
present. It may  
have been  
removed or  
erased during  
processing.  
Batch  
processing stops

and control  
returns to MS-  
DOS.

Bad clusters  
have been  
marked in the  
File Access  
Table (FAT). A  
cluster consists  
of one or more  
sectors. This

xxxxxx  
bytes in  
bad sectors  
(CHKDSK,  
FORMAT)

stops the bad sectors on the disk from being used by MS-DOS for files. On a hard disk it is a normal occurrence for there to be a small percentage of bad sectors. If the percentage grows too large



on a hard disk  
call your service  
engineer for  
advice.

bf Error  
(DEBUG)

The specified  
flag code setting  
is Invalid. Re-  
enter the  
Register (RF)  
command with  
the correct code.

ERROR MESSAGE	MEANING
------------------	---------

bp Error (DEBUG)	More than ten breakpoints were specified for the Go (G) command. Reenter the GO command with ten or fewer breakpoints.
---------------------	---

br Error  
(DEBUG)

An invalid register name has been specified. Re-enter the Register (R) command with a valid register name.

Cannot

CHDIR to root Processing cannot continue The disk^ou are checking is faulty. Reboot MS-DOS and try to RECOVER the disk.

(CHKDSK)

Cannot CHKDSK Network drive You cannot check drive's which are redirected over

(CHKDSK) the Network.

Cannot

CHKDSK a

SUBSTed

or

ASSIGNed

drive

You cannot

check drives

which are

SUBSTed or

ASSIGNed.

(CHKDSK)

Cannot

COPY

filename to  
from a

reserved  
device

(XCOPY)

You cannot  
XCOPY files to  
or from a  
reserved device,  
such as CON: or  
PRN:.

Cannot do  
binary

You have tried  
to use the /B  
switch with the

reads from name of a  
a device. Place a  
device /A switch after  
(COPY) the device name  
to copy in ASCII  
mode.

D-8

MS-DOS USER GUIDE

ERROR MEANING

# MESSAGE

You attempted to edit a backup copy created by EDLIN. Either rename the file or copy the .BAK file and give It a different extension.

Cannot edit .BAK file  
rename file (EDLIN)



Cannot  
execute  
(SELECT)

An error  
occured when  
SELECT tried  
to call the  
FORMAT or  
XCOPY  
commands:  
make sure these  
commands are  
present on the  
Source Drive  
and reenter the  
SELECT  
command.

Cannot find  
file object  
file.

Change  
diskette hit

ENTER  
(LINK)

The specified  
object module is  
not present on  
the diskette.

The specified

Cannot find library is not  
library file. present on the  
ENTER current drive.  
new drive Enter the drive  
letter: containing the  
library.  
(LINK)

Cannot  
format an  
ASSIGNED  
or  
You attempted  
to format a  
drive which is  
actually mapped  
to another drive  
by the ASSIGN

SUBSTed or SUBST  
drive command. Run  
(FORMAT) ASSIGN or  
SUBST again  
and clear all  
drive mappings.

Cannot You cannot  
FORMAT a format drives  
Network that are  
drive redirected over  
(FORMAT) the Network.



ERROR  
MESSAGE

MEANING

Cannot load  
COMMAND,  
system

halted

(MS-DOS)

One of the  
following  
conditions  
occurred while  
loading the  
command

processor:

- The available memory map has been destroyed.

- The command processor specified by the COMSPEC

parameter  
does not exist.

- Reboot MS-DOS.

Cannot nest  
response file  
(LINK)

It is not  
possible to use  
an @ filespec  
within an  
automatic  
response file.

Cannot open  
filename  
(PRINT)

Either MS-DOS cannot find the specified file to print or the file does not exist. Check the command for a valid filename.



Cannot open  
filespec  
filespec2  
(FC)

Either FC  
cannot find the  
specified  
file(s) or the  
file(s) do not  
exist. Check  
the command  
for a valid  
filename.

A list file  
cannot be

Cannot open list file (LINK) opened because the disk or directory is full.

Cannot open overlay (LINK) opened because the disk or directory is full.

D-10

# MS-DOS USER GUIDE

# ERROR MESSAGES

ERROR MESSAGE	MEANING
------------------	---------

Cannot open response file (LINK)	The specified response file does not exist.
--	---

Cannot open  
temporary  
file (LINK)

The directory  
or disk is full,  
hence the  
linker cannot  
create the  
VM.TMP file.  
Insert a new  
disk. Do not  
remove the  
disk that will  
receive the  
List.MAP file.

Cannot  
perform a  
cyclic copy  
(XCOPY)

You have used  
the /S switch  
and have  
specified a  
target  
directory,  
which is a  
subdirectory of  
the source.

Cannot  
recover .

The . entry

entry,  
processing  
continued  
(CHKDSK)

(working  
directory) Is  
defective.

Cannot  
recover ..  
entry,  
processing  
continued  
(CHDSK)

The .. entry  
(parent  
directory) is  
defective.

Cannot RECOVER Network drive  
(RECOVER)

You cannot recover files on drives that are redirected over the Network.

CHDIR .. failed, trying alternate

In traveling the tree structure, CHKDSK was not able to return to a parent directory. It



method  
(CHKDSK)

will try to  
return to that  
directory by  
starting over at  
the root and  
traveling down.

COM port  
does not  
exist  
(MODE)

You have  
specified an  
invalid  
"COM" port



## ERROR MESSAGE MEANING

Compare Error(s)  
(DISKCOMP)

Different  
informa  
has been  
on one c  
more dis  
location

Compare error at  
offset

XXXXXXXX

(COMP)

While  
compari  
files, dif  
values w  
found at  
XXXXXX  
(hexade  
The valu  
found ar  
displaye  
hexadec

Answer

Compare more files you wish  
(Y/N)? compare

(COMP) files,  
otherwise  
enter N.

This message  
indicates  
your file  
fragment  
Fragment  
files tak

Contains XXX  
noncontiguous  
blocks  
  
(CHKDSK)

longer to  
COPY b  
fragmen  
files to a  
newly  
formatted  
disk. Us  
new disk  
result in  
reading  
files.

A file to

Content of  
destination lost  
before copy  
  
(MS-DOS)

used as a  
source for  
the

Copy  
command  
been  
overwritten  
prior to  
completing  
the copy  
Example  
COPY F  
F2 F2 de

F2 before  
can be c

If you ha  
your pro  
Include  
paramet  
have spe  
an inval  
drive yo  
first get  
message

Current drive is no longer valid >  
(COMMAND.COM)

”Not re  
error rea  
drive dri  
letter” A  
Retry, Ig

Press 1 i  
response  
the Curr  
drive me  
is displa  
Type in  
drive-let  
change t



current c  
to a vali  
drive.



D-12

MS-DOS USER GUIDE

ERROR  
MESSAGES

ERROR  
MESSAGE

MEANING

Data error  
(device error)

Data could  
not be  
read/written  
correctly  
because of a  
faulty disk.

One of the

DEVICE  
Support Not  
Present  
(DISKCOMP)  
(DISKCOPY)

floppy disk  
drives  
specified in  
the  
DISKCOMP  
command  
does not  
support MS-  
DOS Ver. 3.2  
device  
control.

df Error  
(DEBUG)

Conflicting codes have been specified for a single flag. A flag can be changed only once for each Register (RF) command.

VDISK has adjusted the

Directory  
entries  
adjusted  
(VDISK)

number of  
directory  
entries in the  
parameters of  
DEVICE =  
VDISK.SYS  
in the  
CONFIG.SYS  
command.

Directory  
error in TREE  
(TREE)

Self-  
explanatory.

Directory is joined, tree past this point not processed. (CHKDSK) CHKDSK will not process directories which are joined.

Directory is totally empty, no . or tree The specified directory does not contain references to working and

past this point parent  
not processed. directories.

(CHKDSK)

Delete the  
specified  
directory and  
recreate it.

Directory not  
empty (JOIN)

You can only  
JOIN onto a  
directory  
which is  
empty.



ERROR  
MESSAGE

MEANING

Disk already  
has an MS-  
DOS

partition

(FDISK)

You cannot use  
the "Create  
DOS Partition"  
option on a  
fixed disk  
which already  
has a DOS



partition.

Disk error  
(device  
error)

An error has  
occurred  
reading from  
or writing to a  
disk.

Disk error

An error  
occurred while  
CHKDSK was  
trying to read

reading FAT the file  
copy allocation  
(CHKDSK) table copy has  
the value 1 or  
2.

Disk error  
writing FAT  
copy update the file  
(CHKDSK) allocation  
table copy has

the value 1 or  
2.

Disk full.  
Edits lost  
(EDLIN)

EDLIN was  
not able to  
save your file  
due to lack of  
disk space.

Disk

The diskette  
contains a  
defective track

unsuitable  
for system  
disk  
(FORMAT)

where MS-  
DOS files must  
reside. The  
disk may only  
be used for  
data.

Diskette bad  
or  
incompatible  
(COPY)

You will get  
this message,  
if there is a  
read error from  
your source  
diskette or if  
there is a write

error to your target diskette.

Divide  
overflow  
(MS-DOS)

A divide by zero was attempted, or an internal logic error has occurred. The system continues as if CTRL BREAK

had occurred.



- W/

D-14

MS-DOS USER GUIDE

ERROR

# MESSAGES

ERROR  
MESSAGE

MEANING

Drive types or  
diskette types not  
compatible

(DISKCOPY,  
DISKCOMP)

The source  
and target  
diskettes may  
have the same  
format  
capacity. See  
the respective

command  
specific

(.)(..) Does not  
exist (CHKDSK)

This message  
indicates  
either the .  
.. directory  
entry is  
invalid.

Either an  
attempt has



Duplicate  
filename or file  
not  
found  
(RENAME)

been made  
rename a fi  
with a file  
name that  
already exi  
in the  
directory, c  
the file to b  
renamed  
could not b  
found on th  
specified (c  
default) dri

Dup record too  
complex (LINK)

There is a  
problem with  
an object  
module  
created from  
an assembly  
source  
program. A  
single DUF  
requires  
  
1024 bytes  
before

expansion.  
Debug the  
source  
program th  
return to th  
linker.

The entire :  
was read in  
memory.

End of Input file

If the file is  
read in

(EDLIN)

sections, the  
message  
Indicates the  
last section  
the file is in  
memory.

Entry error  
(EDLIN)

You have  
entered an  
EDLIN in  
command  
correctly. F  
enter the

command.

This message may be preceded by one or two periods indicating which subdirector is invalid. If you have specified the

Entry has a bad link/attribute/size (CHKDSK)

/F switch,  
CHKDSK v  
try to corre  
the error. ^

ERROR  
MESSAGE

MEANING

The end of  
valid data in  
the last block  
of two files

Eof mark not found (COMP) being compared has not been found. Most likely to occur in nontest files.

Error in EXE The file contains erroneous relocation information

file (MS-DOS) created by LINK. The file may have been altered after creation

Error in EXE or HEX file (DEBUG)

The EXE or HEX file contained invalid characters or records.



Error in IOCTL  
call  
(FORMAT)

You are trying  
to format a  
device, that  
does not need  
formatting.

ERROR  
Incorrect DOS  
version

You are not  
boot-strapping  
MS-DOS Ver  
3.20. The  
device  
DRIVER.SYS

(DRIVER.SYS) will only work  
with MS-DOS  
Ver. 3.20.

ERROR No  
Drive Specified  
(DRIVER.SYS)

You did not  
specify a  
drive number  
when you  
declared  
DRIVER.SYS  
In your  
CONFIG.SYS

Error loading  
system from  
fixed  
disk  
(FDISK)

The operating  
system cannot  
be loaded  
from the fixed  
disk. Retry, or  
if that fails,  
boot the  
system from  
diskette and  
put a new  
copy of MS-  
DOS onto the  
fixed disk  
using the SYS

command.

Error reading  
drive x  
(RECOVER)

Self-  
explanatory.



D-16

MS-DOS USER GUIDE

ERROR

MESSAGE

MEANING

Error reading  
file (PRINT)

Self-  
explanatory.

Five  
unsuccessful  
attempts  
have been  
made to  
read the

Error reading  
partition table  
(FDISK)

startup  
record from  
hard disk.

Retry  
FDISK, or if  
that fails,  
try running  
your  
Customer  
Test as  
Instructed  
by your  
Installation  
and

# Operations Guide.

The device  
issued an  
Error writing to I/O error  
device and your  
(COMMANDS) data was not  
written.  
Retry.

Error writing  
partition table  
(FDISK)

Five  
unsuccessful  
attempts,  
have been  
made to  
write the  
startup  
record on  
the fixed  
disk. Retry  
FDISK, or if  
that fails,  
try running  
your



Customer  
test as  
Instructed  
by your  
Installation  
and  
Operations  
Guide.

As the /F  
parameter  
was not  
used, an  
Errors found, F

parameter not specified.  
Corrections will not be written to disk (CHKDSK)

analysis of the disk will be made and the results displayed, but no corrections will be written to the disk.

Errors on list device indicate

that it may be offline. Please check it.  
(PRINT)      Your printer is off-line.

ERROR MESSAGE      MEANING

One of the following conditions occurred while

EXEC  
failure  
(MS-DOS)

reading a file  
from disk: Read  
error occurred.

The FILES =  
parameter in the  
configuration  
file is not large  
enough.

Increase the  
value and restart  
MS-DOS.

EXE and  
HEX files  
cannot be  
written  
(DEBUG)

The data would  
require a  
backward  
conversion that  
DEBUG does  
not support.

File  
allocation  
table bad  
drive x  
Abort,

See "Device  
and Disk Drive  
Errors" at the  
beginning of  
this appendix. If

Retry,  
Ignore?  
  
(MS-DOS,  
CHKDSK)

the error  
persists, the disk  
should be  
reformatted.

Files are  
different  
sizes  
(COMP)

The specified  
files are not of  
the same length  
and cannot be  
compared.

File XXX canceled by operator (PRINT) When the operator cancels printing, this message appears on the screen.

File cannot be converted (EXE2BIN) The input file you have specified does not have the correct format for conversion.

File cannot  
be copied  
into itself  
(COPY)

A request was  
made to COPY a  
file and place  
the copy (with  
the same name)  
in the same  
directory as the  
source file.

Either change  
the name given  
to the copy or  
put it on another



diskette or  
directory.

D-18

MS-DOS USER GUIDE

ERROR  
MESSAGE

MEANING

You have a

File Error  
(SELECT)

problem  
with your  
source or  
target  
diskette.  
Run  
CHKDSK  
on them to  
determine  
the cause of  
the error.

An

File creation  
error (MS-DOS  
and commands)

unsuccessful  
attempt was  
made to add  
a new file to  
the  
directory.

Run  
CHKDSK to  
determine  
the cause of  
the error.

You may

File Is READ-  
ONLY  
(EDLIN)

not change  
this file  
because the  
file is  
designated  
read-only. If  
you really  
want to  
write to this  
file, use the  
ATTRIB  
command to  
give the file  
a read/write

attribute.

File name must  
be specified  
(EDLIN)

You did not  
specify a  
file name  
when you  
started  
EDLIN.

File not found

The file you  
named in  
the Transfer

(COMMANDS) command  
does not  
exist.

File not found  
(MS-DOS and  
commands)

A file  
named in a  
command  
does not  
exist on the  
disk In the  
specified (or  
default)  
drive.

File not In  
PRINT queue  
(PRINT)

The file you  
want to  
remove  
from the  
print queue  
is not in the  
queue.

A non-  
existent file  
name was

FIND: File not specified  
found (FIND) when  
issuing a  
FIND  
command.

ERROR  
MESSAGE      MEANING

FIND:  
Invalid      A string was not  
number of      specified when



parameters issuing a FIND  
command.

(FIND)

FIND:

Invalid

parameter

option-  
name

(FIND)

You specified  
an invalid  
parameter to the  
FIND command.

**FIND:**  
Read error  
in filename  
(FIND)

An error  
occurred when  
FIND tried to  
read the file  
specified in the  
command.

**FIND:**  
Syntax  
error  
(FIND)

You entered an  
Illegal string  
when issuing the  
FIND command.

First  
cluster  
number is  
invalid, file  
truncated  
(CHKDSK)

An invalid  
pointer to the  
data area has  
been found in  
the file whose  
name precedes  
this message. If  
IF was  
specified, the  
file is truncated  
to zero length.

An assembly

Fixup  
offset  
exceeds  
field width  
(LINK)

language  
instruction  
refers to an  
address with a  
short instruction  
instead of a long  
instruction. Edit  
the assembler  
source program  
and process it  
again.

The source

Fixups  
needed-  
base  
segment  
(hex):

(.EXE) file  
contained  
Information  
indicating that a  
load segment is  
required for the  
file. Specify the  
absolute

(EXE2BIN) segment address  
at which the  
finished module  
is to be located.

# MS-DOS USER GUIDE

ERROR  
MESSAGE

MEANING

FOR cannot  
be nested  
(Batch)

It is not  
possible to  
have more  
than one FOR  
subcommand

on one  
command  
line.

Format  
broken.

(FORMAT)

Self-  
explanatory.

A disk error  
occurred on  
the diskette

Format  
failure  
(FORMAT)

being  
formatted.  
The diskette  
cannot be  
formatted.

If you have  
tried to format  
a virtual disk,  
you will have  
received this  
error message.



Format not supported on drive drive:  
(FORMAT)

You do not need to format a virtual disk. If you are using a block device declaration of DEVICE or DRIVPARM in your CONFIG.SYS, try changing the parameters or removing

the  
declaration. If  
you have no  
device  
declaration for  
that drive in  
your  
CONFIG.SYS  
and have  
received this  
error message,  
try  
reinstalling  
MS-DOS.

Formatting  
target while  
copying  
(DISKCOPY)

Unformatted  
tracks have  
been found on  
the target  
diskette.

These tracks  
will be  
formatted as  
copying  
proceeds.

ERROR  
MESSAGE

MEANING

General  
Failure (device  
error)

This message  
displayed when  
no other device  
error message  
suitable. Check  
the device or  
disk drive  
referred to in  
message. If you  
have found an

corrected the cause of the error press R : Retry, otherwise press A for abort.

Has invalid cluster, file truncated

The specified file contains an invalid data area. If the /F parameter was used, the file is

(CHKDSK)

truncated at the  
last valid data  
block.

Permitted device  
names are:

Printers LPT1  
LPT2, LPT3.

Illegal device

The  
Asynchronous  
Communications

name (MODE) Adapter must  
either COM1  
COM2.

Only one spac  
is allowed  
between MOI  
and Its  
parameter(s).

Illegal  
parameter

Self-  
explanatory.

(MODE)

Many DOS utilities will not run on older versions of MS-DOS. For example the utilities CHKDSK, PRINT, and S will only run under the exact version of

Incorrect DOS version

(Various EXTERNAL

MS-DOS COMMANDS)



MSDOS with  
which they were  
distributed.

D-22

## MS-DOS USER GUIDE

ERROR  
MESSAGE

MEANING

Incorrect  
number of  
parameters  
(JOIN)  
(SUBST)

You  
specified too  
many or too  
few options  
in the  
command  
line.

Incorrect  
parameter  
(ASSIGN)  
(SHARE)

One of the  
options you  
specified is  
wrong.

Incorrect  
Version of  
MODE  
(MODE)

Use the  
version from  
the MS-DOS  
System disk  
you used to  
install your  
operating  
system.

Input file read

An invalid  
object file  
has been

error (LINK)

entered in  
the command  
line.

Insufficient  
disk space  
(MS-DOS and  
commands)

The disk  
does not  
contain  
enough free  
space to  
contain the  
new file.

Insufficient  
memory  
(COMMANDS)

There Is not  
enough  
memory to  
perform the  
specified  
operation.

Insufficient  
memory for  
system

Your  
memory  
configuration  
Is  
insufficient  
to transfer

transfer  
(FORMAT)

the MS-DOS  
hidden  
system files,  
when you  
specified the  
/S switch.

ERROR  
MESSAGE

MEANING

CHKDSK  
cannot create

	an entry in the root directory for saving lost chains as files (see message
Insufficient room in root directory.	”X lost clusters found in Y chains Convert lost chains to files (Y/N?)”
Erase files in root and repeat	because the
CHKDSK	root directory
(CHKDSK)	is full. You should copy

some files  
from the root  
directory to  
another disk,  
then reexecute  
CHKDSK.

Insufficient  
space on  
disk  
(DEBUG)

A write  
command was  
issued to a disk  
that does not  
have enough  
free space to



hold the data  
being written.

One of the  
following  
errors has  
occurred:

- One or both  
intermediate  
MS-DOS files  
cannot be  
created because

the root  
directory is  
full.

Intermediate  
file error  
during pipe  
(MS-DOS)

- The piping  
files do not  
exist.

- There is  
insufficient  
space on the  
disk for the  
data being  
piped.

Remove some files from the root directory and retry. If this fails, a piping file has been erased. Correct the program and retry.

INTERNAL ERROR in MODE application

If this error occurs, report the circumstances to your dealer.

(MODE)

Invalid baud rate specified

(MODE)

Self explanatory

# MS-DOS USER GUIDE

ERROR  
MESSAGE

MEANING

Invalid characters  
in volume label

Volume label  
may contain  
11

(FORMAT)

printable  
characters  
a period (.  
[

The progr  
have just 1  
up almost  
memory. ]  
must now  
the transie  
of  
COMMA  
file from c

Invalid

COMMAND.COM

In drive

X

(MS-DOS)

However,

DOS cannot

COMMAND

on the disk

copy found

invalid. It

disk into t

drive whic

contains t

version of

COMMAND

that you s

with.

Press any  
commenc  
reloading.

You have  
specified a  
number in  
SELECT  
command  
CONFIG.S  
which is n  
configured

Invalid country



code (MS-DOS)  
(SELECT)

Implement  
MS-DOS.  
codes must  
the range  
are the same  
International  
dialing code  
the selected  
country.

Invalid current

Your disk  
Replace the

directory (MS-DOS)

or make a copy from backup sy disk.

Invalid current directory.  
Processing cannot continue  
(CHKDSK)

CHKDSK  
found an e  
the disk's  
directory.  
the system  
rerun CHI

ERROR

# MESSAGE MEANING

Invalid  
date (MS-  
DOS,  
DATE)

You specified an invalid date in response to the date prompt when starting MS-DOS or when using the DATE command.

Invalid  
device  
(CITY)

The specified  
device name is  
invalid in MS-  
DOS.

FORMAT  
displays this  
message when  
the number of  
hidden sectors is  
not evenly  
divisible by the  
number of  
sectors per track

Invalid

device parameters from device driver (FORMAT)

(i.e., the partition does not start on a track boundary). This might happen if you tried to format a hard disk that previously had been formatted with MS-DOS Version 2.11 without first running FDISK.

Invalid  
directory  
(MS-DOS)

The directory  
you specified  
either does not  
exist or is  
Invalid. Check  
to see that you  
entered the  
directory name  
correctly.

Invalid  
disk change

You changed the  
disk In a drive  
when it was not  
allowed. Put the  
disk back in the  
drive and press  
R for Retry.

Invalid  
drive in

One of the paths  
specified In the  
PATH command  
contains an  
invalid drive

search path name. This error (MS-DOS) occurs during execution, not during the PATH command.

D-26

## MS-DOS USER GUIDE

ERROR MESSAGE	MEANING
------------------	---------



Invalid drive  
or filename  
(EDLIN.FC)

You did not  
specify a valid  
drive or file  
name when  
invoking  
EDLIN or FC.

Invalid drive  
specification  
(MS-DOS  
and

You have tried  
to enter an  
invalid drive  
specifier in a

commands) command line.

Invalid drive

was

specified

(MS-DOS

and

commands)

Self-

explanatory.

Invalid

The

environment

contains

environment invalid  
(MS-DOS) characters.

Invalid  
environment  
size  
specified  
(MS-DOS)

Either you  
specified an  
invalid  
parameter to  
the  
environment  
switch  
"/E:s/ze" or  
you specified

size to be a  
number less  
than 160 or  
greater than  
32768

Invalid  
format  
(LINK)

An error has  
been found in a  
library.

Invalid

A program has  
attempted to

handle (MS-DOS) access a file using an an invalid file handle.

Invalid media or Track 0 bad-disk unusable (FORMAT) You cannot use the disk you are trying to format because track 0 is damaged or the disk is of the wrong type.

Invalid  
memory  
block  
address  
(MS-DOS)

A program has  
specified an  
invalid  
memory block  
address.

I

ERROR  
MESSAGE

MEANING

Invalid number  
of parameters  
(COMMANDS)

You have  
specified  
the wrong  
number of  
parameters  
on the  
command  
line. Check  
the syntax  
of the  
command  
you are  
using.

Invalid numeric  
parameters  
(LINK)

A character  
other than a  
digit has  
been  
included in  
a numeric  
parameter.

Invalid numeric  
switch

You  
specified a  
character  
for a switch



specification  
switch error:  
"s:XXX"

(LINK)

requiring a  
numeric  
parameter.  
LINK will  
abort.

Invalid object  
module (LINK)

Object  
module(s)  
are  
incorrectly  
formed or  
incomplete

Invalid  
parameter  
(Commands)

The  
parameter  
entered for  
a command  
was  
incorrect.  
Check the  
syntax of  
the  
command  
you are  
using.

This  
message  
Indicates  
one of:

- No  
parameters  
entered.
- First letter  
not L or C.
- First

Invalid  
parameters  
(MODE)

parameter  
not one of:  
40, 80,  
BW40,  
BW80,  
CO40,  
CO80,  
MONO,  
  
L, R.

- The  
referenced  
display

adapter is  
not present.



D-28

## MS-DOS USER GUIDE

ERROR MESSAGE	MEANING
------------------	---------

Invalid  
partition  
table on  
fixed  
disk  
(FDISK)

Invalid  
partition  
information  
has been  
detected during  
startup from  
fixed disk.  
Restart DOS  
using diskette  
and correct the  
fixed disk  
partition  
information

using the  
FDISK  
command.

A directory  
whose name is  
in another  
directory  
cannot be  
accessed by  
TREE. Use  
CHKDSK to  
determine the

Invalid path  
(TREE)

error In the  
directory  
structure.

Invalid path,  
not  
directory, or  
directory not  
empty  
(RMDIR)

The directory  
cannot be  
removed as the  
path contains  
an invalid  
name, or the  
directory is not  
empty. The  
current



directory  
cannot be  
removed.

Invalid path,  
or filename  
(COPY)

A directory or  
filename that  
does not exist  
has been  
specified.

The

Invalid  
subdirectory  
entry  
(CHKDSK)

subdirectory  
whose name  
precedes this  
message  
contains  
invalid  
information.  
For more  
detailed  
information  
rerun  
CHKDSK with  
the A/ switch.

Invalid  
switch: z  
(LINK)

The characters  
in z do not  
identify a valid  
linker  
parameter.

Invalid  
switch  
specification  
(FORMAT)

Self-  
explanatory.

# ERROR MESSAGE MEANING

Invalid  
time (MS-  
DOS)  
(TIME)

You specified  
an Invalid time  
in response to  
the time prompt  
when starting  
MS-DOS or  
using the TIME  
command.

When you tried to format an existing hard disk partition and were prompted for the existing volume label, you entered an incorrect

Invalid  
Volume ID  
(FORMAT)

volume label. If you really intend to format that hard disk

partition, use the VOL command to find out the correct volume label, and re-enter the FORMAT or SELECT command.

filename is You have two files cross

cross  
linked on  
cluster

(CHKDSK)

linked. Make a  
copy of the file  
you want to  
keep, and then  
delete both files  
that are cross  
linked.

Keyboard  
routine not

You Indicated  
that the  
keyboard  
routine  
"KEYBxx" was

found  
(SELECT)

to be found on  
the source  
diskette. It was  
not found.

Label not  
found  
(Batch)

A non-existent  
label has been  
specified In a  
GOTO  
command  
within the batch  
file.



Last  
backup  
diskette not  
inserted  
(BACKUP,  
RESTORE)

The diskette in  
the drive is not  
the last one in a  
series created by  
BACKUP.

**D-30**

**MS-DOS USER**

# GUIDE

ERROR  
MESSAGE

MEANING

During a  
Replace (R)  
command, the  
string given as  
the replacement  
caused the line

Line too long  
(EDLIN)

to expand  
beyond the 253  
character limit.  
The Replace  
command is  
ended  
abnormally.  
Split the long  
line into shorter  
lines; then  
reissue the  
Replace  
command.

Lock  
Violation  
(device  
error)

A program tried  
to access part of  
a file that is  
being used by  
another  
program. Press  
A for Abort or  
wait a while and  
press R for  
Retry.

If you respond

XXX lost  
clusters  
found In  
YYY chains  
Convert lost  
chains to  
files (Y/N)?

Y to this  
prompt and had  
specified the /F  
switch,  
CHKDSK will  
recover the lost  
blocks it found  
when checking  
the disk.  
CHKDSK will  
create a  
directory entry  
and a file for  
you with the

(CHKDSK) filename  
FILEnnnn.CHK  
If you respond  
N and had  
specified /F  
switch,  
CHKDSK frees  
the lost blocks  
so they can be  
reallocated.

Memory  
allocation      The available

error. Cannot memory map  
load has been  
COMMAND, destroyed. You  
system must restart  
halted (MS- MS-DOS.  
DOS)

Memory  
control  
blocks  
destroyed  
(MS-DOS)

Self-  
explanatory.

Missing  
country code  
(SELECT)

When you  
entered the  
SELECT  
command, you  
forgot the  
country code  
parameter.



ERROR  
MESSAGE

MEANING



Missing  
keyboard  
code  
(SELECT)

When you  
entered the  
SELECT  
command, you  
forgot the  
keyboard code  
parameter.

MS-DOS  
incorrect on  
default

Self-

drive

explanatory.

(Startup)

MS-DOS  
not found  
on default  
drive

Self-  
explanatory.

(Startup)

Must  
specify  
destination  
line  
number  
(EDLIN)

A destination  
line number  
was not  
specified for a  
Copy or Move  
command.  
Reenter the  
command  
correctly.

This message Is  
printed if  
EDLIN does

New file  
(EDLIN)

not find a file with the name you specified if you are creating a new file, ignore this message. If you do not intend to create a new file, check to see that you correctly typed the filename of the file you

wish to edit.

No COM:  
ports

You have no  
"COM" ports  
attached to your  
computer

No files  
added  
(REPLACE)

You specified  
the /A switch,  
however all the  
files on the  
source exist on

the target.

# MS-DOS USER GUIDE

D-32

ERROR MESSAGE	MEANING
------------------	---------

No files replaced
----------------------

No files on the target were
--------------------------------

(REPLACE) replaced.

You cannot run the FDISK program due to one of the following conditions:

No fixed disks present

- No fixed disk attached.
- Fixed disk is

(FDISK)

present in the expansion unit but the unit is not switched on.

- Fixed disk Is incorrectly Installed.

Reload MS-DOS. If this message



No free file handles  
Cannot start COMMAND FILES =  
exiting (MS-DOS)  
persists, increase the size of the parameter in the CONFIG.SYS file, and reload MS-DOS.

You have tried to load a second copy of

No free file handles  
(MS-DOS)

the command processor, but too many files are currently open. Increase the size of the FILES = parameter in the configuration file and reload MSDOS.

No MS-DOS  
partition to  
delete  
(FDISK)

You have used  
the Delete DOS  
Partition  
option when no  
such partition  
exists on the  
current fixed  
disk or you  
have tried to  
delete a non-  
existent  
partition.





# ERROR MESSAGES

ERROR MESSAGE	MEANING
------------------	---------

No MS- DOS partition.
-----------------------------

An attempt has been made to format a hard
---

Use FDISK disk that does not have an MSDOS partition.  
to correct (FORMAT)

Non-DOS disk (device error)  
The file allocation table contains invalid information.  
The disk must be reformatted.

No object  
modules  
specified  
(LINK)

You have not  
specified any  
object modules  
for the linker

No  
operating  
system on  
fixed  
disk

Self-  
explanatory.

(FDISK)

No paper  
(device  
error)

The printer is  
either not  
switched on or  
is out of paper.

No path  
(MS-DOS)

You typed  
PATH to display  
your search  
path. There is no  
current  
command



search path.

No room in directory for file (EDLIN)	The directory of the specified diskette Is already full, or the specified disk drive or file name Is illegal.
--	---

No space  
for a XXX  
cylinder  
partition at  
cylinder  
YYYY  
(FDISK)

There is not  
enough space on  
disk to  
accommodate a  
partition with  
the specified  
number of  
cylinders at the  
specified  
position.



# MS-DOS USER GUIDE

eSiSi^ . . . .

ERROR MESSAGE	MEANING
------------------	---------

Non-system disk or disk error. Replace	Restart the system with another
---	---------------------------------------

disk and  
strike any  
key.

(Startup)

MSDOS disk in  
drive A.

Not a block  
device  
(FORMAT)

You are trying  
to format a  
serial device.  
You cannot  
format serial  
devices.

Not enough  
memory  
(JOIN,  
SHARE)

(Other  
External  
Commands)

There is not  
enough memory  
for MSDOS to  
perform the  
command.

Not enough  
room for  
MS-DOS

There Is not  
enough room on  
the target disk

on  
this disk

(SYS)

Not enough  
room to  
merge the  
entire file

(EDLIN)

for SYS to  
transfer the  
system files.

There was not  
enough room In  
memory to  
enable a  
Transfer  
command to  
merge the entire  
contents of a

file.

Not found  
(EDLIN)

The search string was not found in the specified line range, or no further occurrences were found after resuming.

Not ready  
(PRINT)

If this message occurs when PRINT attempts a disk access, PRINT will keep trying until the drive is ready. Any other error causes the current file to be cancelled. An error message would



be output on  
your printer in  
such a case.



ERROR  
MESSAGE

MEANING

Not ready

The named  
device is not  
available for

(device  
error)

the read/ write  
operation  
required.

Out of  
environment  
space (MS-  
DOS

There is not  
enough room in  
the program  
environment to  
accept more  
data.

Out of

memory  
(MS-DOS  
and  
commands)

Self-  
explanatory.

Out of space  
on list file  
(LINK)

There is not  
enough space  
on disk to hold  
the list file.

Out of space  
on run file

There is not  
enough space

(LINK) on disk to hold the run file.

Out of space on VM.TMP (LINK) There is not enough space on disk to allow the VM.TMP file to be expanded.

Parameters not compatible You have specified switches which

(FORMAT, REPLACE) cannot be used together.

Parameters not compatible with fixed disk (FORMAT) The FORMAT command was called with parameters which are only applicable to floppy diskettes.

Parameters not supported (FORMAT)	You have specified parameters that FORMAT does not support.
--	---

D-36

MS-DOS USER GUIDE

ERROR MESSAGE	MEANING
------------------	---------

Parameters not supported by Drive (FORMAT)

Format displays this message when the device driver for this drive does not support Generic IOCTL function requests.

Path not found

You specified an invalid

(PRINT) pathname.

(MS-DOS)

Pathname The maximum  
too long pathname  
(PRINT) permitted is 63  
(TREE) characters.

PRINT There are no  
queue is files waiting to  
empty be printed.

(PRINT)



PRINT  
queue is  
full  
(PRINT)

You have tried  
to place more  
than ten files In  
the print queue.

The printer  
mode has not  
been set because  
of one of the  
following  
conditions:

Printer

error  
(MODE)

- I/O error
- Out of paper
- Power off
- Time out

ERROR  
MESSAGE      MEANING

The disk you are

Probable

using is not recognized by this version of MS-DOS. The disk either was created by another system with a format that is not supported on this version of MS-DOS or is not an MSDOS disk. Do not

non-DOS  
disk.  
Continue  
(Y/N)?  
(CHKDSK)

continue  
processing if  
CHKDSK  
returned this  
message for a  
removable disk,  
if this message  
is returned for a  
hard disk, the  
information  
describing the  
characteristics  
of the disk to  
MS-DOS has

been destroyed.

In this case, you may continue CHKDSK processing.

Processing in your machine cannot continue (CHKDSK) There is not enough memory to process CHKDSK for this disk. You

(FDISK) must obtain more memory to run CHKDSK.

You must acquire more memory to run your application. It is possible that some programs you have run are still using some

Program  
too big to  
fit in  
memory  
(MS-DOS)  
(LINK)

memory. Try to  
restart MS-DOS,  
and retype the  
command. If  
you still receive  
the error  
message and  
have used the  
BUFFERS =  
parameter  
directive in the  
systems  
CONFIG.SYS  
file reduce the

number of buffers, re-boot the system and retry the command.

However, If you still receive this message, you must acquire more memory.





# MS-DOS USER GUIDE

ERROR MESSAGE	MEANING
------------------	---------

Read fault (device error)	MS-DOS cannot read the requested data from the named device.
---------------------------------	--

Read error      The command  
in: filename    could not read  
(Commands)    the entire file.

Requested  
drive is not    Self-  
available       explanatory.  
(FORMAT)

Requested  
stack size      You have tried  
exceeds        to specify a

64K

(LINK)

stack size

greater than 64  
Kbytes.

If this message  
occured when  
you tried to  
boot from hard  
disk, the most  
likely cause is  
that there is no  
active partition

on the hard  
disk. Bootstrap  
from floppy  
disk, run  
Rom BASIC FDISK to  
not available make the  
Press bootable  
partition  
reset to re- active.  
boot

(BIOS)  
If this message  
occured when  
you are  
running a

program, that  
program may  
have been  
trying to  
access Rom  
BASIC or a  
spurious soft  
interrupt 18  
hexadecimal  
occured.

Sector not

The sector  
containing the

found  
(device  
error)

data cannot be  
found, usually  
due to a  
defective area  
on the disk.

ERROR  
MESSAGE

MEANING

Sector size  
too large in  
file

The device  
sector size  
defined in the

filename device driver  
filename  
(Startup) exceeds the  
system limit.

Seek The disk drive  
(device cannot find the  
error) proper track on  
the disk.

You have tried  
to combine

Segment  
size exceeds  
64K (LINK)

identically  
named  
segments  
resulting In a  
segment  
requirement of  
more than the  
addressing  
limit of 64  
Kbytes.

SHARE

Share can only



already  
Installed  
(SHARE)

be installed  
once.

The memory  
space allocated  
by the  
command  
SHARE is  
insufficient.

Sharing

Rebootstrap  
your computer

buffer

and call

exceeded

(device

error)

SHARE with a  
larger memory  
space

parameter. If  
the message re-  
occurs,  
continue to  
increase the  
size of the  
memory space  
parameter.

Sharing  
Violation  
(device  
error)

A program tried  
to access a file,  
but another  
program is  
using that file.  
Press

A to Abort, or  
wait a while  
and press R to  
Retry.

Specify a

Source and target drives are the same  
(BACKUP)  
(RESTORE)

different source and target drive in your BACKUP or RESTORE command.



D-40

MS-DOS USER GUIDE

**ERROR  
MESSAGE**

**MEANING**

Source does  
not contain  
backup files  
  
(RESTORE)

The diskette in  
drive was not  
created by  
BACKUP. The  
file is not called  
BACKUP.®

Source path  
required

You did not specify  
a source path

(REPLACE) parameter.

Specified  
COMMAND  
search  
directory bad  
[access  
denied] (MS-

The SHELL  
directive in the  
CONFIG.SYS  
Is incorrect. Th  
place that you  
told MS-DOS to  
find  
COMMAND.C  
does not exist,  
COMMAND.C  
Is not in that p

DOS)

or access has been denied to the directory containing COMMAND.COM

Specified drive does not exist, or is nonremovable

If you referred your hard disk drive as a parameter of the commands, you will get this error.

(DISKCOMP) message.

(DISKCOPY) Otherwise you referring to a nonexistent drive.

Stack size exceeds 65535 bytes  
(LINK)

The specified stack size exceeds the system limit.

Symbol defined more

Two or more modules have



than once  
(LINK)

defined the same  
symbol name.

Symbol table  
capacity  
exceeded  
(LINK)

Very many, and  
very long names  
were entered  
exceeding the limit  
of approximately  
25 Kbytes.



ERROR  
MESSAGE

MEANING

Syntax error  
(MS-DOS)

You have entered a command in the incorrect format.

System  
Failure  
(MODE)

Self explanatory.

SYS cannot  
install MS-  
DOS on  
  
this disk  
  
(SYS)

You cannot use  
SYS as the  
target disk  
either is not  
empty or was  
not formatted  
with the /S  
option.

Target  
cannot be

BACKUP was  
called with /A

used for  
backup  
(BACKUP)

parameter, but  
the target has  
not already  
been used as a  
backup disk.

Target is  
full  
(RESTORE)

RESTORE  
cannot continue  
for this  
reason.

If you press  
CTRL BREAK  
or CTRL C  
while

Terminate  
batch job  
(Y/N)?  
(MS-DOS)

In batch mode,  
MS-DOS asks  
you whether or  
not you wish to  
end batch  
processing.  
Press Y to end  
processing.  
Press N to

continue the batch job.

If there is not enough room on the target disk to receive a particular backed up file, RESTORE will delete the partially backed up file. Delete

unwanted files  
The last file on the target  
was not disk, then enter  
restored RESTORE  
(RESTORE) filename, where  
filename is the  
name of the file  
previously  
partially  
restored.  
Interruption to  
RESTORE can  
be continued in  
the same

fashion.



Table captionD-42

Table captionMS-DOS USER  
GUIDE

■ iJ ■ - ■ ■ . ' ■ ■ ■ ■ ■ ■ ■ i '  
cj-'-.iJ

ERROR MESSAGE MEANI



Too many external  
symbols in  
one module  
(LINK)

There are  
many ex  
symbols  
one mod  
The limit  
256 exte  
symbols  
module.

Too many  
groups a

Too many groups  
(LINK)

defined.  
limit

Is 10.

Too many libraries  
specified (LINK)

Too many  
libraries  
been nar  
The limit  
eight lib

Try to se

Too many files open  
(MS-DOS)(EDLIN)

this problem  
by Increasing  
the FILE  
directive  
CONFIG

Too many overlays  
(LINK)

The system  
limit of  
overlays  
been exceeded

Too many public symbols (LINK)

There are many public symbols in one module. The limit is 1024 public symbols.

Too many segments

You have many segment classes in your source files. Th

or classes (LINK)

limit is 2  
(segment  
classes t  
together

An  
unrecov  
Abortion  
occured,  
you cho  
ABORT  
response  
disk erro

Top level process

aborted, cannot  
continue

(COMMAND.COM)  
(Commands)

The operating  
system has  
crashed.  
COMMAND.COM must  
bootstrap  
DOS from  
new system  
disk.



ERROR

MEANING

# MESSAGE

Tree past  
this point  
not

processed

(CHKDSK)

Track 0 of the  
disk being  
checked is  
damaged.

CHKDSK

cannot

continue.

SELECT has a  
problem

p  
Unable to  
copy  
keyboard  
routine  
(SELECT)

opening or  
reading the  
KEYBXX.COM  
file on your  
source disk.  
Run CHKDSK  
on the file to  
determine the  
cause of the  
problem

The specified



Unable to  
create  
directory  
(MKDIR)

directory  
cannot be  
created due to  
one of the  
following  
conditions:

- Directory  
already exists.
- One of the  
directory path-  
names could  
not be found.

- The root directory is full.

Unable to shift Screen (MODE) Mode is unable to shift the test pattern on the screen any farther.

The first track

Unable to  
write BOOT  
(FORMAT)

of the diskette,  
or of the hard  
disk partition,  
specified is  
bad. If it is a  
diskette  
problem, use  
another  
diskette. If it is  
hard disk  
problem, rerun  
FDISK  
specifying  
another starting

track for the  
disk partition.

An internal  
MS-DOS error  
has occurred.

The number  
nnn is the  
internal

Unexpected  
DOS Error  
nnn  
(REPLACE)

MSDOS error  
number. See the  
MS-DOS  
System

Programmer  
Guide for a  
complete list of  
MS-DOS error  
numbers.

D-44

MS-DOS USER  
GUIDE

Table captionD-44

# ERROR MESSAGES

ERROR  
MESSAGE

MEANING

Unexpected  
end-of-file on  
library

There Is  
probably an  
error in the

(LINK)

library file.

Unexpected  
end-of-file on  
VM.TMP

The diskette  
containing  
VM.TMP is  
not present in  
the drive.

(LINK)

Unrecognized

You have  
entered an  
Invalid

command In command in  
CONFIG.SYS the  
configuration  
file.

If you press  
Y in response  
to this  
prompt,  
CHKDSK  
will convert  
Unrecoverable the bad  
error in directory into



directory.  
Convert  
directory to  
file (Y/N)?  
(CHKDSK)

a file. You  
can then fix  
the directory  
yourself or  
delete it. If  
you press N,  
you may not  
be able to  
write to or  
read from the  
bad directory.

Unrecoverable

file sharing  
error  
(SHARE)

Caused by a  
file-sharing  
conflict.

Unrecoverable  
read error on  
drive d:  
(DISKCOMP)  
(DISKCOPY)

Self-  
explanatory.  
The diskette  
has probably  
been  
damaged.

Unresolved  
externals: list  
(LINK)

The external symbols listed were not defined in the modules or library files. Do not attempt to run the file created by the linker.

VM.TMP is

an illegal  
filename and  
has been  
ignored  
(LINK)

You cannot  
use VM.TMP  
as an object  
module.



ERROR  
MESSAGE

MEANING

Warning:  
directory full

No more files  
can be

(RECOVER) recovered in  
the directory.

Warning-  
Diskette may  
be  
unusable  
(DISKCOPY) After an  
unrecoverable  
read, write or  
verify error,  
the copy may  
be corrupted.

Warning:

diskette is  
out of  
sequence

Replace  
diskette or  
continue.

Strike any  
key when  
ready.

(RESTORE)

Self-  
explanatory.

When

Warning:  
The file  
above is  
marked read-  
only. Replace  
the file  
(Y/N)?

(RESTORE)

RESTORE /P  
is specified  
and the file  
encountered is  
read-only;  
answer Y if  
you want to  
replace the  
file or N if  
not. When you  
type ENTER,  
RESTORE  
will continue.

Warning:  
The file  
above was  
changed after  
it was backed  
up. Replace  
the file  
(Y/N)?

When  
RESTORE IP  
is specified  
and the file on  
the target disk  
has a later  
time and date  
than the same  
named file on  
the source  
disk; answer  
Y if you want  
to replace the  
file on the



(RESTORE) target disk or  
N if not.  
When you  
type ENTER,  
RESTORE  
will continue.

Warning: no  
files were  
found to  
backup

Check your  
backup file  
specifications  
for incorrect  
input.

(BACKUP)

D-46

MS-DOS USER GUIDE

ERROR  
MESSAGES

W

ERROR  
MESSAGE

MEANING

Warning: No  
files were  
found to  
restore

Self-  
explanatory.

(RESTORE)

None of the

Warning: no  
stack  
segment  
(LINK)

object  
modules  
specified  
contain a  
statement  
allocating  
stack space,  
but you  
entered the  
/STACK  
switch.

Warning:

read error on  
EXE file  
(EXE2BIN)

The input file  
has not been  
read correctly.

Warning:  
Segment of  
absolute or  
unknown  
type

There is a bad  
object module  
or an attempt  
has been made  
to link  
modules that  
the linker  
cannot handle;

(LINK)

for example,  
an absolute  
object  
module.

Write error  
in TMP file  
(LINK)

No more disk  
space remains  
to expand the  
VM.TMP file.

Write error

There is not  
enough disk

on RUN file  
(LINK)

space for the  
Run file.

Write fault  
(device error)

MS-DOS  
cannot  
successfully  
write data  
from/to the  
named device.

You have tried

Write protect to write data  
(device error) to a disk that  
is write-  
protected.

Write  
protected  
(DISKCOPY) diskette for a  
copy.

You cannot  
use a write  
protected  
diskette for a  
copy.



E. GLOSSARY OF TERMS

# ABOUT THIS APPENDIX

This appendix describes some technical terms, whose understanding is necessary for the user of MS-DOS.



**CONTENTS**

**GLOSSARY OF**

**TERMS E-1**



# GLOSSARY OF TERMS

The following table defines the terminology in this manual.

TERM	MEANING
------	---------

The partition on

active  
partition

hard disk which contains the operating system files enabling the bootstrapping of the computer. This happens on system reset or when the computer is turned on.

American Standard  
Code for

ASCII

Information Interchange. A 7 BIT code, which has been extended to an 8 BIT code (a BYTE) to represent graphic characters and international characters.

Part of the operating system

basic  
input  
output  
system  
(BIOS)

which provides an interface with the machine hardware. Most of the BIOS Is In Read Only Memory (ROM), the rest is loaded from the system disk.

In a binary numbering system, only two marks are



binary  
digit  
(BIT)

used 0 and 1. Each  
of these marks is  
called a binary  
digit.

bootable  
file

A file of a specific  
format that the  
bootstrap loader  
can load into  
memory to  
initialize the  
system.

byte

Eight bits, which is normally a code for an ASCII character.

current  
directory

The directory in which you are working.



TERM

MEANING

cylinder

Hard disks usually consist of a number of platters. A cylinder refers to the same track on each surface of the platters which form a notional cylinder.

disk

A diskette or hard disk.

diskette

A single or double-sided 5 1/4 in. floppy disk.

A letter referring to the diskette drive or hard disk

drive In question.  
For example it  
may be:

drive  
specifier

A - first diskette  
drive.

B - second  
diskette drive.

C - hard disk  
drive.

editing  
function  
keys

The keys that  
invoke the intra-  
line commands.

external  
command

A command that  
is not loaded into  
memory at  
initialization.

Such commands  
reside on disk  
from where they  
are loaded,  
executed and

then purged from memory.

Disks must be formatted before they can be used with MS-DOS.

Formatting places tracks, which are split into sectors, onto the surface(s) of formatting a disk. The

sectors are all the same length, typically 512 bytes. Also formatting places a boot record and an empty directory on the disk.

E-2

MS-DOS USER GUIDE



TERM

MEANING

hard disk

A sealed storage unit with non-removable surfaces. A hard disk can store much more information than a floppy disk, and the computer can retrieve information from

it faster.

hardware  
reset

A system  
reinitialization  
caused by pressing  
the physical reset  
button. The  
subsequent  
initialization  
includes  
diagnostic tests  
and a reset of all

system

parameters. Any  
AUTOEXEC.BAT  
file or  
CONFIG.SYS file  
Is executed.

The EDLIN

inter-line commands that  
operate on entire  
lines of text.

intra-line  
commands

The commands invoked by the special editing function keys that perform editing operations within a single line of text.

internal

A command that I embedded in the COMMAND.COM file and resides In

command memory whenever  
MS-DOS is  
booted.

Kilobyte 2 to the power 10  
KB = 1024 Bytes

Mega-byte 2 to the power 20  
MB = 1 048 576 Bytes

nil

parameter

A parameter to a command where the parameter In question is not specified In the command line. The parameter therefore assumes a default value.

TERM

MEANING

partition

A certain number of cylinders of a hard disk, which have been set aside for the use of a particular operating system. That operating system treats the partition like a complete, but smaller, hard disk. The maximum partition size

allowed for MS-DOS is 32 MB. The number of cylinders this corresponds to, depends on how many bytes there are per cylinder.

A sequence of one or more directory names separated by backslashes,



optionally  
beginning with a  
drive specifier and  
pathname optionally  
terminating in a  
file name. It  
specifies a path  
through a directory  
structure to access  
a file or directory.

The track on a disk

sectors

is divided into sectors. MS-DOS disks are soft sectored. The number of sectors per track is typically 8, 9 or 15.

A line of text containing either the last command line entered or the current line in a

source  
line

file being edited. It can be retrieved in whole or part or modified using the special editing function keys.

system

An MS-DOS file that is present on the MS-DOS system diskette that contains

file system software.  
There are three  
such files: Two  
hidden files and  
COMMAND.COM

## MS-DOS USER GUIDE

E-4

# GLOSSARY OF TERMS

TERM	MEANING
------	---------

	A system reinitialization caused by pressing the CTRL, ALT
--	---

system and DEL keys  
reset simultaneously.

Any  
AUTOEXEC.BAT  
file or  
CONFIG.SYS file  
Is executed.

text file An ASCII file  
whose records are  
separated by  
CR/LF.

tracks  
per inch  
(t.p.i.)

A disk track is the circular locus of the head as the disk rotates. The head can be moved to the other tracks; they are concentric circles. A double density diskette has 48 t.p.i. A quad density disk has 96 t.p.i.

virtual  
disk

An emulation of backing store in Random Access Memory (RAM). It is faster than disk backing store, but the information on virtual disk is lost when the computer is turned off.

A name that can be



volume  
label

assigned to a disk  
by the FORMAT  
command. It will  
subsequently be  
displayed In a  
directory listing,  
or by the VOL  
command.

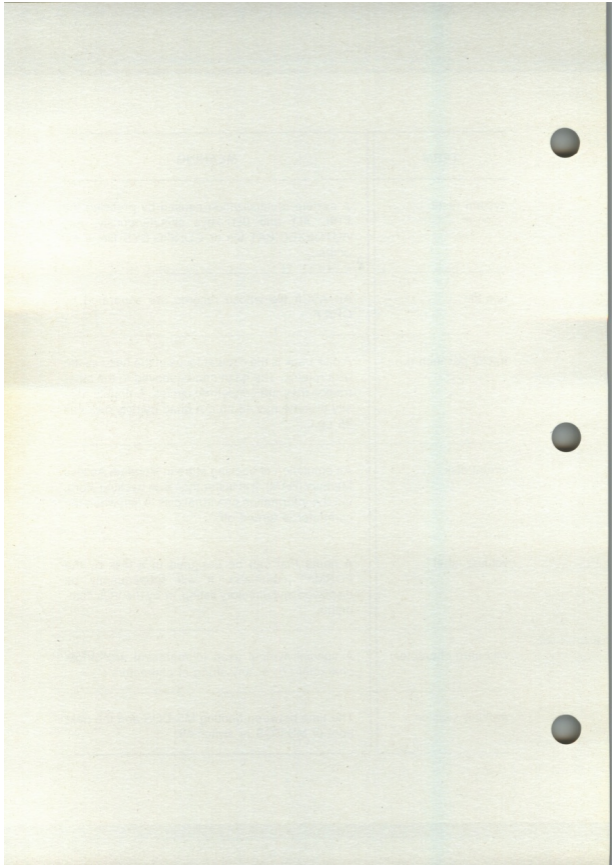
wild  
card

A special symbol  
used to represent  
any single  
character (?), or

character any string of  
characters (\*).

working  
session

The time between  
booting MS-DOS  
and the next boot  
of MS-DOS or  
switch-off.



# ABOUT THIS APPENDIX

This appendix describes how to install VDISK.SYS.



## CONTENTS

# INTRODUCTION F-1

## INSTALLING

VDISK.SYS

F-1



# INTRODUCT

A virtual disk is part of Random Access Memory which emulates a backing store Disk. The VDISK.SYS is a device driver, which when installed, enables a virtual disk drive with the next available drive letter. For example if you have a two physical drive machine,

with two physical drives “A:” and “B;”, when VDISK.SYS is installed, you will have an extra drive C:. The only difference between virtual disk and real disk is that when you turn your machine off, the information on virtual disk will be lost. So remember to COPY all files you want to keep, from virtual disk to a real disk, before you turn your machine off.





# INSTALLING VDISK.SYS

VDISK.SYS is a file included on your MS-DOS Diskettes. To install the VDISK console drive the following command must be placed in the CONFIG.SYS file.

```
DEVICE = VDISK.SYS
```

ldisk-size][sector-size]  
[entries] [/E[: max]]

Table captionWhere

SYNTAX	MEANING
ELEMENT	

A decimal value  
declaring the  
virtual disk size  
in Kilo Bytes.

disk-size

The value be from 1 through the maximum free Random Access Memory on your computer. The default is 64 Kilo bytes.

A decimal value declaring the sector size in

sector-size bytes. The value may be 128, 256 or 512. The default is

128 bytes.

SYNTAX	MEANING
ELEMENT	

entries

A decimal number declaring the maximum number of directory entries required for files. One entry is used for a volume label. Three entries are used for each sub-directory. The value may

be from 2 through 512. The default is 64 entries.

This switch specifies that the VDISK driver uses "extended memory". MS-DOS can only directly address 640 KB. Random

IE

Access Memory installed over 640 KB is "extended memory". This option is only available on Intel 80286 based Personal Computers, do not use this switch for other Personal Computers.

:max

This is a parameter of the /E switch, specifying the maximum number of sectors (of sector-size) to be transferred to/from extended memory. The possible values



are a decimal number In the range 1 through 8. The default value Is 8.

For example place the following assignment in CONFIG.SYS.

```
DEVICE = VDISK.SYS 128  
512 32
```

Reboot your computer and a message similar to the following will be displayed:

VDISK Version 3.20 virtual disk D:

Directory entries adjusted  
Buffer size: 128 KB Sector size: 512 Directory entries: 32

Refer to Appendix C for more details of

CONFIG.SYS.

MS-DOS USER GUIDE

F-2

# THE VIRTUAL DISK SYSTEM

Remarks

The following situations  
prevent VDISK from being

installed.

- less than 64 KB free memory.
- using the /E switch with insufficient extended memory.

In these cases the following message is output on the screen.

**VDISK not installed**

insufficient memory

VDISK.SYS might adjust the parameters you specified in the following ways:

## PARAMETER ADJUSTMEN

Rounded up: 3  
bytes (per file  
entry)  
multiplied by

entries  
(rounded up)  
equals a  
multiple of  
sectorsize. This  
is so as not to  
waste space  
available for  
directory  
entries.

Rounded down  
one sector at a  
time times

entries

sectorsize  
divided by 32  
bytes, until  
there are  
sufficient  
sectors to hold  
the File  
Allocation  
Table, the  
directory  
entries and at  
least two  
sectors for  
files. When th



number of sectors for directory file entries has been rounded down to one, the above error message is issued. In this case redeclare the VDISK parameters In CONFIG.SYS and reboot the

system.

Rounded down  
so as to leave  
64 KB Random  
Access

Memory free.

If this is not  
possible the  
above error

message is  
issued. In this  
case you need

disk-size

to Install more  
Random  
Access  
Memory in  
order to use  
VDISK.

## Remarks

You can install more than  
one virtual disk by placing  
several DEVICE =

VDISK.SYS commands in your CONFIG.SYS. Each virtual disk takes the next available drive letter. If the next available drive letter for the virtual disk is “F;”, place the command:

LASTDRIVE = F

before your DEVICE command in CONFIG.SYS. Refer to Appendix C for more details.

# Warning

Using VDISK in extended memory with networking installed, may lead to problems of lost interrupts and corruption of the virtual disk. If you have such problems, try setting the max parameter of the IE switch to 1. If this does not cure the problems, then you cannot place the declaration `”DEVICE = VDISK.SYS IE”`

in the CONFIG.SYS, when you are going to load networking.

F-4

MS-DOS USER GUIDE

# ABOUT THIS APPENDIX

This appendix describes how to install a device driver using DRIVER.SYS.



# CONTENTS

INTRODUCTION G-1

INSTALLING

DRIVER.SYS G-1





# INTRODUCT

The DRIVER.SYS is a device driver, which when installed either: enables a new block device to be installed; or enables existing block devices to be referred to as a new logical block device with different characteristics.

If you have installed an external floppy disk drive,

you can create a block device driver for this drive using DRIVER.SYS; this will be referred to by the next available drive letter. For example on a twin floppy disk drive machine, when DRIVER.SYS is installed, the extra external drive will appear as drive C:.

The following is an example of the use of DRIVER.SYS for referring to an existing

floppy disk drive by another drive letter. If you have a computer with a 1.2MB floppy disk drive and a hard disk. Before installing DRIVER.SYS for the 1.2MB drive, the drive letters A: and B: refer to this drive. After installing DRIVER.SYS for this drive, the drive letter "D:" can refer to the floppy disk drive specifically for handling 360KB diskettes. So to copy files from a 360KB

diskette to a 1.2MB diskette,  
you could enter the  
command:

```
COPY D:*.* A:
```

MS-DOS will prompt you to  
enter the respective diskettes.

# INSTALLING DRIVER.SYS

DRIVER.SYS is a file included on your MS-DOS Diskettes. To install this driver the following command must be placed in the CONFIG.SYS file.

```
DEVICE = DRIVER.SYS /D
```

drive-no [/C] [/F: form-factor  
]

[/H: heads ]I/N] [/S: sectors ]  
[/T: tracks ]

Where

SWITCH PARAMETER MEANING

Spec  
the

/D

drive-no

phys  
drive  
num  
betw  
and  
Flop  
disk  
start  
hard  
start  
128.

Disk



1C

chan  
supp  
requ

1F

form-factor

Spec  
the r  
form  
kind  
supp

0 32)

KB

1 1.2

2 720  
(default)

3 8 i  
singl  
dens

4 8 i  
dout  
dens

5 Ha  
Disk

6 Ta  
Driv

7 Ot

Spec  
the r  
of he  
the c  
drive

/H

heads

valu

rang

1 to

The

is 2 1

/N

Spec

non-

remo

blocc

devi

such

hard

IS

sectors

Spec  
the r  
of se  
per t  
Its v  
can 1  
from  
99.7  
defa  
nine  
secto  
track

IT

tracks

Spec  
the r  
of tr  
per s  
Its v  
can 1  
from  
throu  
999.  
defa  
80 tr  
per s

G-2

# MS-DOS USER GUIDE

# INSTALLING NEW BLOCK DEVICES

Note

DEVICE = DRIVER.SYS can  
be repeated on several lines



of the CONFIG.SYS file with different parameters.

## Examples

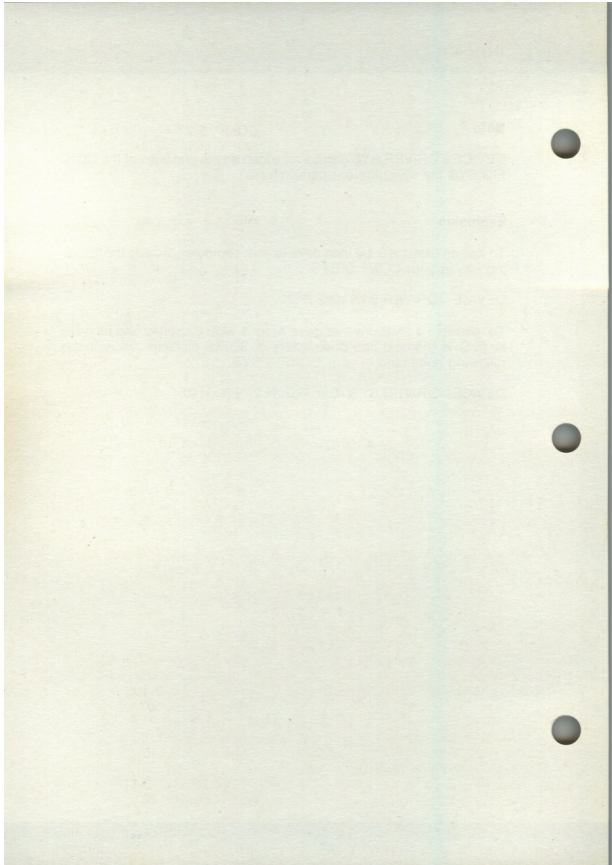
To add an external 3 1/2 inch drive to your computer, include the following line in your CONFIG.SYS: \_

```
DEVICE = DRIVER.SYS  
/D:2 /F:2
```

To refer to a 1.2MB drive as

drive A: for 1.2MB diskettes  
and as drive D: (if D: is the  
next free drive letter) for  
360KB diskettes, include the  
following command in your  
CONFIG.SYS.

```
DEVICE = DRIVER.SYS  
/D:0 /F:0 /H:2 /S:9 /T:40
```



# INDEX

# A

ANSI escape sequences, B-1  
ANSI.SYS device  
(CONFIG.SYS), B-1, C-5  
archive bit, 5-10  
ASCII  
tables, A-1  
ASSIGN  
command, 5-8  
ATTRIB  
command, 5-6, 5-10  
AUTOEXEC.BAT, 1-2, 4-13

# B

BACKUP command, 5-12

batch processing, 1-2, 4-10

batch processing commands,  
4-10

BREAK command, 5-16

BREAK command  
(CONFIG.SYS), C-1

BUFFERS command  
(CONFIG.SYS), C-2

## c

capacities diskettes, 1-3  
changing directory, 3-13  
CHDIR (CD) command, 3-11, 3-13, 5-17

checking disks and files, 3-14  
CHKDSK command, 3-14, 5-19  
CLS command, 5-23  
COMMAND command, 5-23  
command formats, 4-1  
commands ASSIGN, 5-8  
ATTRIB, 5-6, 5-10

BACKUP, 5-12 BREAK, 5-16

CHDIR (CD), 3-11, 3-13, 5-17 CHKDSK, 3-14, 5-19

CLS, 5-23 COMMAND, 5-23

COMP, 5-27 COPY, 3-4, 5-6, 5-30 CTTY, 5-36 DATE, 5-38 DEBUG, 9-1 DEL, 5-41

DIR, 3-8, 5-6, 5-43

DISKCOMP, 5-45

DISKCOPY, 5-48 ECHO, 5-51



EDIT (video file editor), 6-1  
EDLIN (line editor), 7-2  
ERASE, 5-52 EXE2BIN, 4-7,  
5-52 EXIT, 5-55 external, 4-  
6, 4-8 FC, 5-56 FDISK, 5-66  
FIND, 4-16, 5-66 FOR, 5-70  
FORMAT, 5-72 GOTO, 5-76  
GRAFTABL, 5-78  
GRAPHICS, 5-79  
GWBASIC, 5-82  
HEXDUMP, 3-6, 5-83 IF, 5-  
85 internal, 4-6 JOIN, 5-87  
LABEL, 5-92

LINK (the linker), 4-7, 8-3,

3\_“j “j 3-“13

MKDIR (MD), 3-13, 5-94

MODE, 5-95

MORE, 4-16, 4-17, 5-103

PATH, 3-10, 5-104 PAUSE,

4-10, 5-105 PRINT, 5-6, 5-

107 PROMPT, 5-111

RECOVER, 5-114

REM, 4-10, 5-115

REN, 5-116 REPLACE, 5-6,  
5-118 RESTORE, 5-122

## Index-i

RMDIR (RD), 3-14, 5-126  
SELECT, 5-127 SET, 5-129  
SHARE, 5-5, 5-131 SHIFT,  
5-133 SORT, 4-16, 5-135  
SUBST, 5-139 SYS, 5-143  
TIME, 5-144 TREE, 5-146  
TYPE, 3-6, 5-147 VER, 5-  
149 VERIFY, 5-149 VOL, 5-  
150 XCOPY, 5-6, 5-151

COMP command, 5-27  
compatibility diskette, 1-4  
drive, 1-5 COMSPEC, 5-25  
CONFIG.SYS definition, C-1  
CONFIG.SYS commands  
BREAK, C-1 BUFFERS, C-2  
COUNTRY, C-3 DEVICE, C-  
5 DRIVPARM, C-6 FCBS, C-  
8 FILES, C-9 LASTDRIVE,  
C-10 SHELL, C-11 STACKS,  
C-12 CONFIG.SYS devices  
ANSI.SYS, B-1, C-5  
DRIVER.SYS, G-1, C-5  
VDISK.SYS, F-1, C-5 control

keys, 2-1 Control-break, 4-9  
Control-C, 4-9 Control-  
Number Lock, 4-9 Control-S,  
4-9

COPY command, 3-4, 5-6, 5-  
30

COUNTRY command  
(CONFIG.SYS), C-3 creating  
a new directory, 3-13 CTTY  
command, 5-36 cursor  
movement (ANSY.SYS), B-1

# D

DATE command, 5-38

DEBUG command, 9-1

debugging commands, 9-4

definition of terms, E-1 DEL

command, 5-41 DEVICE

command (CONFIG.SYS),

C-5 device driver definition,

G-1 device errors, D-1 device

names, 3-4 DIR command, 3-

8, 5-6, 5-43 directories

changing, 3-13 creating, 3-13

definition, 3-6 displaying, 3-

11 hierarchical, 3-6

removing, 3-14 root, 3-7

sub-directories, 3-6 disk

fixed (hard), 1-3 hard (fixed),

1-3 virtual, 1-9, F-1 disk

operating system, 1-1

DISKCOMP command, 5-45

DISKCOMP command, 5-48

diskette capacities, 1-3

compatibility, 1-4 diskettes,

1-3 handling, 1-5 labeling, 1-

6

## Index-ii

purchasing, 1-6 write-  
protection, 1-6 disks, 1-3

displaying the current  
directory, 3-11 drive

compatibility, 1-5

identification, 1-3

DRIVER.SYS device

(CONFIG.SYS), C-5, G-1

diAU installation, G-1 drives



High-Capacity, 1-4 Normal-Capacity, 1-4 DRIVPARM command (CONFIG.SYS), C-6

E

ECHO command, 5-51 EDIT command (video file editor), 6-1

editing function keys, 2-4

EDLIN (line editor)

command, 7-2 ENTER key,

2-1, 4-9 Environment, 5-26  
size, 5-26

ERASE command, 5-52 error  
level, 5-14, 5-55, 5-86 5-121

error messages (MS-DOS),  
D-1 examining files, 3-6  
EXE2BIN command, 4-7, 5-  
52 exit code, 5-14, 5-55, 5-86  
5-121

EXIT command, 5-55  
external command syntax, 4-

8 external commands, 1-2, 3-10, 4-6

## F

FC command, 5-56

FOBS command

(CONFIG.SYS), C-8 FDISK

command, 5-66 filenames

definition, 3-1 files

examining, 3-6 naming, 3-1

protecting, 3-5 validity, 3-14

- o FILES command (CONFIG.SYS), C-9
- o FIND command, 4-16, 5-66
- o fixed (hard) disk, 1-3
- o font tables, A-1
- o FOR command, 5-70
- o FORMAT command, 5-72

## G

- o GOTO command, 5-76
- o GRAFTABL command, 5-78
- o GRAPHICS command, 5-79
- o GWBASIC command, 5-82

# H

hard (fixed) disk, 1-3

HEXDUMP command, 3-6,  
5-83 hierarchical directories,  
3-6

# I

IF command, 5-85 input, 4-  
15

input redirection, 4-16

internal commands, 1-2, 4-6

## J

JOIN command, 5-87

Index-iii

Key reassignment  
(ANSI.SYS), B-1 keyboard  
code tables, A-1

## L

LABEL command, 5-92

LASTDRIVE command

(CONFIG.SYS), C-10 line editor, 7-1 commands, 7-5 keyboard functions, 7-36 LINK (the linker) command, 4-7, 8-3, 8-11, 8-13 linker

classes, 8-2 groups, 8-2 segments, 8-2 VM. IMP, 8-1

M

MKDIR (MD) command, 3-13, 5-94

MODE command, 5-95

MORE command, 4-16, 4-17  
5-103

MS-DOS commands, 4-8

N

national characters, A-1

networking, 5-5 notation  
conventions, 1-9

O



output, 4-15

output redirection, 4-15

parameters definition, 4-1  
nil, 4-5

PATH command, 3-10, 5-104

pathnames definition, 3-8

maximum length, 3-10

syntax, 3-9

PAUSE command, 4-10, 5-105  
piping, 4-17

PRINT command, 5-6, 5-107  
processor child, 5-26, 5-55  
parent, 5-26, 5-55 secondary,  
5-26, 5-55 programming  
FORTRAN, 1-1 GW-BASIC,  
1-1 Macro Assembler, 1-1  
PASCAL, 1-1

PROMPT command, 5-111  
protecting files, 3-5

R

read-only attribute, 5-10 real-

time clock, 1-2 RECOVER  
command, 5-114 redirecting  
input, 4-16 redirecting  
output, 4-15 REM command,  
4-10, 5-115 removing a  
directory, 3-14 REN  
command, 5-116 REPLACE  
command, 5-6, 5-118

replaceable parameters, 4-14  
5-133

reserved device names, 3-4  
RESTORE command, 5-122

RMDIR (RD) command, 3-14 5-126

root directory, 3-7

Index-iv

S

screen handling (ANSI.SYS), B-1

SELECT command, 5-127

SET command, 5-129

SHARE command, 5-5, 5-131 SHELL command (CONFIG.SYS), C-11 SHIFT command, 5-133 SORT command, 4-16, 5-135 source drive, 4-9 STACKS command (CONFIG.SYS), C-12 sub-directories, 3-6 SUBST command, 5-139 SYS command, 5-143

## T

target drive, 4-9 TIME

command, 5-144 TREE  
command, 5-146 TYPE  
command, 3-6, 5-147

## V

VDISK.SYS device  
(CONFIG.SYS), F-1, C-5  
VER command, 5-149  
VERIFY command, 5-149  
video file editor, 6-1 virtual  
disk, 1-9 definition, F-1  
installation, F-1 visual editor  
commands, 6-21 keyboard

functions, 6-4 searching, 6-19  
VOL command, 5-150

## W

Wild cards, 3-2

the \* wild card, 3-3 the ?  
wild card, 3-2

## X

XCOPY command, 5-6, 5-151

# Index-v



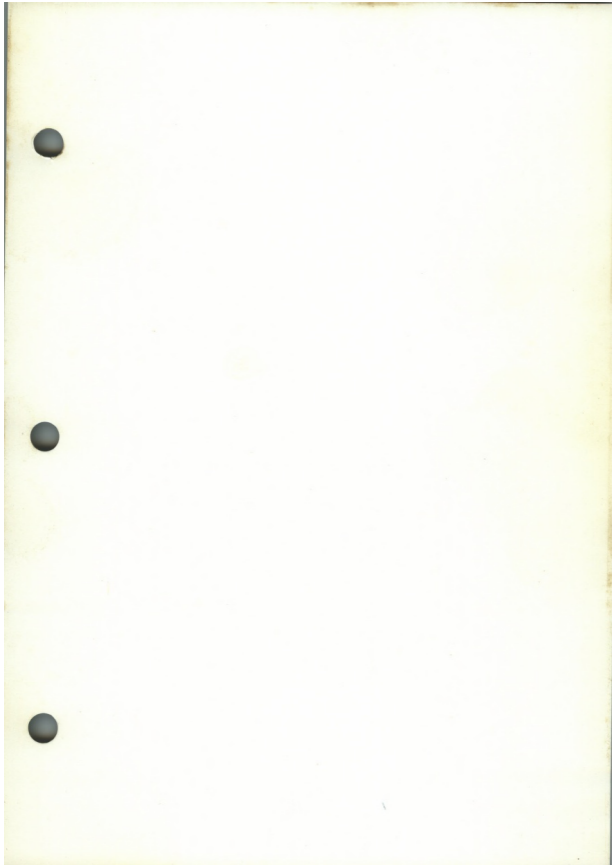


# NOTICE

Ing. C. Olivetti & C., S.p.A. reserves the right to make any changes in the product described in this manual at any time and without notice.

This manual is licensed to the Customer under the conditions contained in the User License enclosed with the Program to which the manual refers.





i.'i.'v 1!'^j . ^

Code 4034460 Z (0) Printed  
in Italy

**oliuelli**

PERSONAL

COMPUTER



# oliuelli

