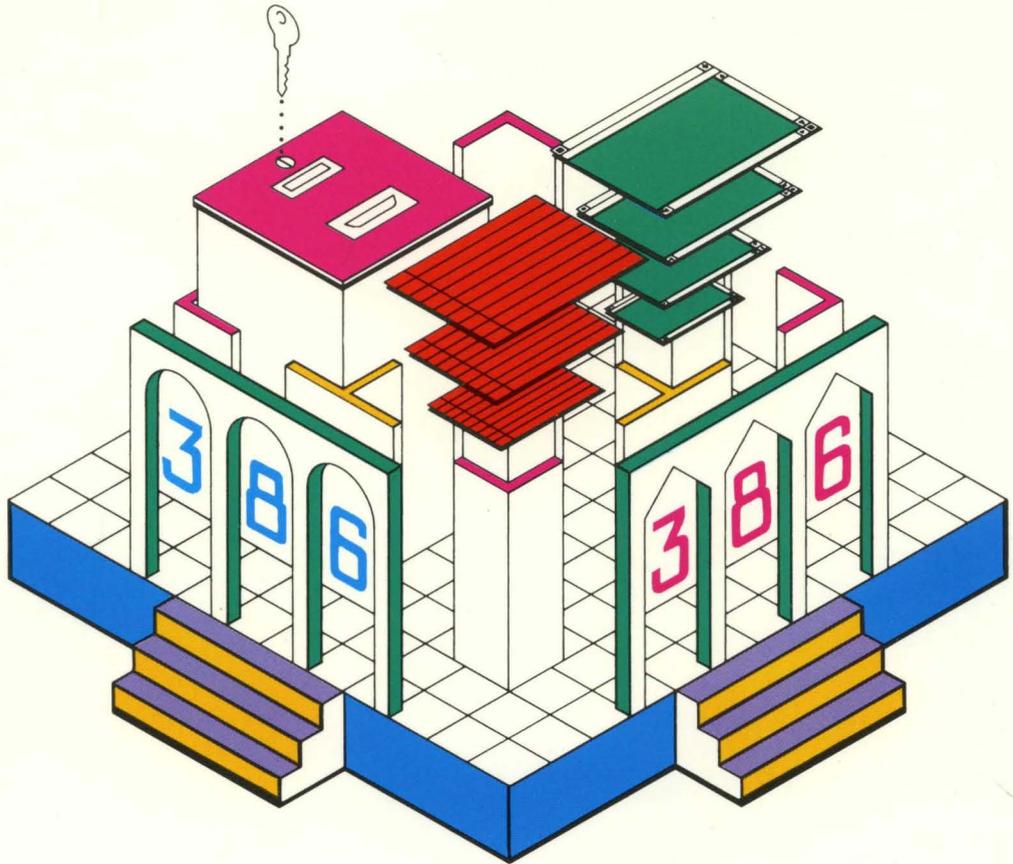


Concurrent™ DOS 386

Multuser/Multitasking Operating System

REFERENCE GUIDE



 DIGITAL RESEARCH®

Concurrent[™] DOS

REFERENCE GUIDE

 DIGITAL RESEARCH®

COPYRIGHT

Copyright © 1987 Digital Research Inc. All rights reserved. No part of this publication may be reproduced, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise without the prior written permission of Digital Research Inc, 60 Garden Court, Box DRI, Monterey, California 93942

DISCLAIMER

DIGITAL RESEARCH MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. Further Digital Research Inc. reserves the right to revise this publication and to make changes from time to time in the content hereof without obligation of Digital Research Inc to notify any person of such revision or changes.

NOTICE TO USER

This manual should not be construed as any representation or warranty with respect to the software named herein. Occasionally changes or variations exist in the software that are not reflected in the manual. Generally, if such changes or variations are known to exist and to affect the product significantly, a release note or READ.ME file accompanies the manual and the distribution disks. In that event, be sure to read the release note or READ.ME file before using the product.

TRADEMARKS

Digital Research and its logo, CP/M, and CP/M-86 are registered trademarks of Digital Research Inc. Cardfile, Concurrent, Concurrent DOS 386, Concurrent DOS XM, DR EDIX, DOS Plus and MP/M-86 are trademarks of Digital Research Inc. "We Make Computers Work" is a service mark of Digital Research Inc.

All rights reserved

The following are registered trademarks of the listed companies.

| Registered Trademark | Company |
|-----------------------------|---------------------------------|
| AST | AST Research Inc. |
| IBM | International Business Machines |
| IBM PC | International Business Machines |
| IBM PC/XT | International Business Machines |
| Intel | Intel Corporation |
| Lotus | Lotus Corporation |
| Microsoft | Microsoft Corporation |
| Quadram | Quadram Corporation |

The following are trademarks of the listed companies.

| Trademark | Company |
|------------------|-----------------------|
| MS-DOS | Microsoft Corporation |
| Quadems+ | Quadram Corporation |
| RAMPage | AST Research Inc. |

First Edition, November 1987

Foreword

This manual, referred to as the Concurrent Reference Guide, is for users of the Concurrent DOS 386 or Concurrent DOS XM operating systems. In this guide, DOS refers to either PC DOS or MS-DOS™. The manual contains reference information about Concurrent DOS commands and should be used with the Concurrent DOS User's Guide, hereafter referred to as User's Guide.

In this Manual

This manual contains detailed descriptions of the Concurrent DOS commands. See the User's Guide for operating system details.

Examples use **boldface type** to represent information that you type into your computer, and ordinary `unit-spaced type` to represent information that your computer displays on the screen.

The figures in this manual might vary from the screen you see on your video display.

Digital Research is interested in your reports on the performance of our software. Please use the Software Performance Reports enclosed in each product package to help us provide you with better software products.

Contents

| | |
|---|----|
| Concurrent Commands | 1 |
| Command-line Symbols | 1 |
| Command Summary | 2 |
| Built-in Commands | 6 |
| Commands that have Changed from Previous Versions | 6 |
| 8087 | 7 |
| APPEND | 8 |
| ATTRIB | 10 |
| AUX | 11 |
| BACKUP | 12 |
| Multiple Floppy Disks | 14 |
| Restoring Files to a Hard Disk | 14 |
| BANK | 15 |
| BATCH | 16 |
| Invoking and Interrupting a BATCH File | 16 |
| Creating a BATCH File | 16 |
| Using Variables in a BATCH File | 17 |
| Environment Variables | 17 |
| CALL Subcommand | 18 |
| ECHO Subcommand | 18 |
| EXIT Subcommand | 18 |
| FOR Subcommand | 19 |
| GOTO Subcommand | 19 |
| IF Subcommand | 20 |
| PAUSE Subcommand | 21 |
| REM Subcommand | 22 |
| SHIFT Subcommand | 22 |
| BREAK | 23 |
| CARDFILE | 24 |
| CHDIR/CD | 25 |
| CHDIR Option | 25 |
| CHDIR Special Characters | 25 |
| Concurrent Directories | 26 |
| Using Floating Drives | 26 |
| Making a Directory the Current Directory | 27 |
| How to Display the Current Path | 27 |
| CHKDSK | 28 |
| CHSET | 29 |
| CLS | 32 |
| COPY | 33 |
| How to Copy Files between Disk Drives and Directories | 34 |
| How to Copy Files to the Console or Printer | 36 |
| How to Create a File with COPY | 36 |
| COPYMENU | 37 |
| CPM | 38 |
| DATE | 40 |
| DELQ | 42 |
| DIR | 43 |
| DISKCOMP | 45 |

Contents

| | |
|--|----|
| DISKCOPY | 46 |
| Drive Types Supported | 46 |
| DISKCOPY Permutations | 47 |
| DREDIX | 48 |
| DSKMAINT | 49 |
| EDITMENU | 50 |
| ERAQ | 51 |
| ERASE | 52 |
| FDISK | 53 |
| Create DOS Partition | 54 |
| Delete DOS Partition | 55 |
| Select Bootable Partition | 56 |
| Select Alternate Fixed Disk | 56 |
| FIND | 57 |
| FM | 58 |
| FORMAT | 59 |
| Drive Types Supported | 59 |
| FORMAT Permutations | 59 |
| FUNCTION | 62 |
| F3 -- Assign Function Keys | 64 |
| F5 -- Save Function Keys in a File | 65 |
| F7 -- Reset Window Switching Keys | 65 |
| Using FUNCTION with an Input File | 66 |
| HELP | 70 |
| Customizing Your HELP Files | 71 |
| LABEL | 73 |
| LIMSIZE | 74 |
| LOADSYS | 75 |
| Installing LOADSYS | 75 |
| MEMSIZE | 76 |
| MKDIR/MD | 77 |
| MORE | 79 |
| PASSWORD | 80 |
| PATH | 82 |
| PIFED | 83 |
| PRINTER | 86 |
| PRINTMGR | 87 |
| START Command | 88 |
| PRINT Command | 88 |
| DELETE Command | 90 |
| HELP Command | 90 |
| STATUS Command | 90 |
| RESET Command | 91 |
| PROMPT | 92 |
| REBOOT | 94 |

| | |
|---------------------------------------|-----|
| RENAME/REN | 95 |
| Using RENAME to move files | 95 |
| REPLACE | 96 |
| RMDIR/RD | 97 |
| RUNMENU | 98 |
| SCEPTER | 99 |
| SCEPTER Options | 99 |
| SET | 101 |
| SETPORT | 103 |
| Menu-driven Mode | 104 |
| Command-line Mode | 105 |
| SETUP | 107 |
| F2 -- Reserve System Space | 109 |
| F3 -- Save System Parameters | 110 |
| F4 -- MDisk | 110 |
| F5 -- Diskette Drive Parameters | 111 |
| F7 -- Select Serial Console | 112 |
| F8 -- MENU TWO | 114 |
| F9 -- Color Monitor Scroll Mode | 115 |
| F10 -- Update and Exit | 116 |
| SHOW | 117 |
| SORT | 118 |
| STOP | 119 |
| How to Terminate a Program | 120 |
| SUBST | 121 |
| SUSPEND | 122 |
| TIME | 123 |
| TOUCH | 125 |
| TREE | 126 |
| TYPE | 127 |
| VER | 128 |
| VERIFY | 129 |
| VOL | 130 |
| WINDOW | 131 |
| VIEW | 132 |
| TOP | 133 |
| FULL | 134 |
| WRITE | 135 |
| CHANGE | 135 |
| WMENU | 136 |
| WMENU Rules of Operation | 137 |
| PLACE | 138 |
| SIZE | 138 |
| SCROLL | 138 |
| TRACKING | 139 |
| COLOR | 139 |
| WRITE | 140 |
| ABORT | 141 |
| XCOPY | 143 |
| XDEL | 145 |
| XDIR | 146 |

Contents

Tables

| | | |
|----|---|-----|
| 1 | Command-line Symbols | 1 |
| 2 | Concurrent Command Summary | 2 |
| 3 | CHKDSK Options | 28 |
| 4 | CHSET Fields and Settings | 30 |
| 5 | COPY Device Names and Descriptions | 33 |
| 6 | COPY Options | 34 |
| 7 | DIR Options | 44 |
| 8 | FORMAT Command Options | 60 |
| 9 | FUNCTION Main Menu Options | 63 |
| 10 | Function Key Identifiers | 67 |
| 11 | Window Switching Number Assignments | 68 |
| 12 | Hexadecimal Values for Selected Keys | 68 |
| 13 | Printers and Number Assignments | 86 |
| 14 | PRINTMGR Commands | 87 |
| 15 | PRINTMGR Options | 89 |
| 16 | SETPORT Options | 104 |
| 17 | SETPORT Parameters | 105 |
| 18 | SETUP Main Menu Options | 107 |
| 19 | SETUP Options | 109 |
| 20 | SHOW Options | 117 |
| 21 | STOP Display Information Descriptions | 120 |
| 22 | WINDOW Command Summary | 131 |
| 23 | WINDOW VIEW Parameter Definitions | 133 |
| 24 | WINDOW CHANGE Parameter Values | 135 |
| 25 | WMENU Command Summary | 137 |
| 26 | WMENU WRITE Command Options | 140 |

Figures

| | | |
|---|--|-----|
| 1 | Sample Hierarchical Subdirectories | 78 |
| 2 | Sample WINDOW VIEW Display | 132 |

Concurrent Commands

This manual describes the Concurrent commands in alphabetical order. Command descriptions assume that you are entering the command at the command line prompt. Each description presents first the command-line format, or syntax, followed by an explanation of when and how to use the command. Example command lines illustrate the form and function of the command. Note that you must press the Enter key after you type a command line.

Command-line Symbols

Table 1 shows the terms, special characters, and symbols used in command-line format.

Table 1. Command-line Symbols

| Component | Description |
|---------------|--|
| filename | Represents the command's filename. Some command filenames can be abbreviated, as explained in the text. You do not need to include the file extension. To run a non built-in command program, you must have a corresponding program file in your current directory or in one of the path directories. |
| filespec | Represents a file specification. The file specification consists of the following elements: [d:] [dirpath] filename [.ext] [;password] "d:" is an optional drive reference. "dirpath" is the location of the file in the subdirectory structure. "filename" is the 1- to 8-character name of the file. "ext" is the optional 1- to 3- character file extension. "password" is the optional 1- to 8-character password. The colon (:), period (.), and semicolon (;) must be used as shown to separate the filespec elements. |
| [] | Items enclosed within square brackets are optional. To include optional items enter only the information inside the square brackets. The brackets are not entered on the command line. |
| | A vertical bar means either/or. Just one of the separated items must be selected. |
| ... | An ellipsis shows that an item may be repeated. |
| , = ? * : \ / | All punctuation (except square brackets) should be entered at the command line as shown. |
| dev: | Represents a peripheral device, such as a printer. |
| n | Indicates that a number should be entered. The command explanation describes the range of valid numbers. |

Table 1. (Cont'd)

| Component | Description |
|-----------|---|
| command | Represents any parameter or option entered on a command line after the command name. |
| fd | Represents a floating drive letter used to access directories. |
| dirname | Represents a directory name. The directory name consists of the following elements: subname.ext where "subname" is the 1- to 8-character name of the directory and "ext" is the optional 1- to 3-character directory extension. |
| dirpath | Represents directory path and is defined as follows: [N] dirname[\ dirname] |

Command Summary

Table 2 lists the commands described in this manual and summarizes their purpose. Built-in commands are indicated. Optional short forms are shown when they exist.

Table 2. Concurrent Command Summary

| Name | Purpose |
|--------|---|
| 8087 | Indicates to Concurrent that a COM or EXE program makes use of an 8087 co-processor. (Built-in; default = OFF) |
| APPEND | Finds files outside the current directory |
| ATTRIB | Displays and modifies a file's attributes. |
| AUX | Selects auxiliary port. The number of auxiliary ports (numbered 0,1,2...) depends on the configuration of your system. The default value is AUX = 0. (Built-in) |
| BACKUP | Copies files from a hard disk drive to floppy disks and restores files from floppy disks to the hard disk. |
| BANK | Controls how a program runs in banked memory. (Built-in; default = ON) |
| BATCH | Executes the commands in a file that have a BAT file extension. |
| BREAK | Allows you to break out of programs running under Concurrent. |

Table 2. (Cont'd)

| Name | Purpose |
|------------------|--|
| CARDFILE | Stores and retrieves names, addresses, phone numbers and comments for persons and businesses you want to remember. |
| CHDIR (CD) | Shows the path to and changes the current directory on DOS media. (Built-in) |
| CHKDSK | Checks the integrity of data on disks, and restores corrupted disks. |
| CHSET | Changes the command header of programs with the CMD file extension. |
| CLS | Clears the display screen. (Built-in) |
| COPY | Copies files between directories and devices. |
| COPYMENU | Copies one or more menus from one file to another. |
| CPM | Provides access to CP/M files |
| DATE | Displays and changes the date. (Built-in) |
| DELO | Erases files but queries you before operation. (Built-in) |
| DIR | Displays the files in a directory. (Built-in) |
| DISKCOPY | Copies entire disks of the same format. |
| DISKCOMP | Compares entire disks of the same format. |
| DREDIX | Text editor used to create and edit text files. |
| DSKMAINT | Menu-driven command for disk formatting. |
| EDITMENU | Creates, modifies, and deletes menus. |
| ERAQ | Erases files but queries you before operation. (Built-in) |
| ERASE (ERA)(DEL) | Erases files. (Built-in) |
| FDISK | Prepare hard disk partitions for Concurrent. |
| FIND | Finds a string of characters within files. |
| FM | The File Manager allows you to select commands from a series of menus. |
| FORMAT | Formats floppy disks. |
| FUNCTION | Assigns the function key commands and window switching keys. |
| HELP | Explains Concurrent commands. |
| LABEL | Assigns a volume label to the disk. |
| LIMSIZE | Limits the amount of EMS memory available to an application. |

Table 2. (Cont'd)

| Name | Purpose |
|--------------|--|
| LOADSYS | Starts Concurrent if DOS has control when you first turn on your computer. |
| MEMSIZE | Limits the amount of conventional memory available to an application. |
| MORE | Causes Concurrent applications' output to the screen to be displayed a screenful at a time. |
| MKDIR (MD) | Creates a subdirectory on DOS media. (Built-in) |
| PASSWORD | Sets password protection to files or paths. |
| PATH | Sets a search path for commands or batch files. (Built-in) |
| PIFED | Associates system parameters with an application program so that Concurrent is automatically configured correctly when the program is run. |
| PRINTER | Changes the current printer number. (Built-in; default = 0) |
| PRINTMGR | Controls the printing of files. |
| PROMPT | Modifies the command prompt. (Built-in) |
| REBOOT | Performs a system reset, is a synonym for Ctrl-Alt-Del. |
| RENAME (REN) | Renames files. (Built-in) |
| REPLACE | Selectively copies files. |
| RMDIR (RD) | Removes subdirectories on DOS media. (Built-in) |
| RUNMENU | Runs a menu. |
| SCEPTER | Initializes the banked memory management routines within Concurrent (Concurrent XM only). |
| SET | Inserts strings into the command processor's environment. (Built-in) |
| SETPORT | Configures the serial ports. |
| SETUP | Modifies the default characteristics. |
| SHOW | Displays current disk status. |
| STOP | Displays memory allocation and terminates programs. (Built-in) |
| SUBST | Allows you to replace a path with a drive. |
| SUSPEND | Suspends EXE and COM programs when their window is switched out. (Built-in; default = ON) |
| TIME | Displays and changes the time of day. |
| TOUCH | Sets a time and date on groups of files. |

Table 2. (Cont'd)

| Name | Purpose |
|-------------|---|
| TREE | Displays the hierarchical path of directories and subdirectories. |
| TYPE | Displays the contents of a text file. (Built-in) |
| VER | Displays the current version number |
| VERIFY | Checks that data has been written correctly to disk |
| VOL | Displays the disk volume label. |
| WINDOW | Shows and modifies window characteristics. |
| WMENU | Allows you to change your window configuration. |
| XCOPY | Selectively copies groups of files. |
| XDEL | Selectively deletes groups of files. |
| XDIR | Extended directory utility. |

Built-in Commands

Several Concurrent commands are built into the operating system. This means that no corresponding program file is needed to run a built-in command. For example, 8087 is a built-in command; therefore, there is no program file named 8087.COM, 8087.EXE, or 8087.CMD. You can run a built-in command even when all the system's memory is in use.

The Concurrent built-in commands are as follows:

| | | |
|----------|------------|---------|
| 8087 | ERAO | SHOW |
| APPEND | ERASE/ERA | STOP |
| AUX | LIMSIZE | SUBST |
| BANK | MEMSIZE | SUSPEND |
| BREAK | MKDIR/MD | TIME |
| CHDIR/CD | MORE | TYPE |
| CLS | PATH | VER |
| DATE | PRINTER | VERIFY |
| DELQ | PROMPT | VOL |
| DATE | RENAME/REN | |
| DELQ | RMDIR/RD | |
| DIR | SET | |

All the BATCH subcommands are also built in:

| | | |
|------|-------|-------|
| CALL | GOTO | SHIFT |
| ECHO | IF | |
| EXIT | PAUSE | |
| FOR | REM | |

Commands that have Changed from Previous Versions

If you have used previous versions of Concurrent you will notice that some commands are no longer present. These have been replaced by other commands as follows:

ADDMEM and COMSIZE are replaced by MEMSIZE

FSET is replaced by ATTRIB and PASSWORD

PIP is (largely) replaced by XCOPY.

SDIR is replaced by XDIR.

The rest of this manual gives detailed descriptions of all the Concurrent commands.

8087

Format: 8087=ON
8087=OFF

Explanation: The 8087 command allows EXE and COM programs to use the 8087, 80287, or 80387 co-processor. 8087=ON signals Concurrent that a program requiring the co-processor is going to be started. 8087=OFF signals Concurrent that no co-processor is available. The default setting is 8087=OFF.

The 8087 command is necessary only for DOS 80x87-dependent programs. See the CHSET command description for more information on running 80x87-dependent CMD programs.

To allow a program to use the co-processor, type the following:

```
A>8087=ON
```

Concurrent checks to see if the co-processor is present in your computer. If it is, the command prompt is returned to the screen. If it is not, the message

```
8087 not installed
```

is displayed.

The 8087=ON command allows co-processor use for the current window only. If you run an 80x87-dependent program from two windows, you must enter 8087=ON from each.

APPEND

Format: APPEND [/X]
APPEND d:path [;[d:]path]
APPEND [;]

Explanation: APPEND enables Concurrent to look in other specified directories for files that it can't find in your current directory. It is similar to the PATH command: PATH is used when you want Concurrent to search for .COM, .EXE, .CMD and .BAT files. APPEND is used when you want Concurrent to search for any other types of files.

The two main forms of the command are:

- * APPEND followed by the option /X
- * APPEND followed by pathnames

APPEND with /X

Before you enter an APPEND command to specify a search path, you can issue APPEND with the option /X. This causes APPEND to work with Concurrent utilities that use the Search First, Find First, and Exec functions. Use this option with care, as it can lead to problems when used with certain applications programs.

Note also that before you use BACKUP you should cancel the APPEND search string by entering APPEND ;

APPEND with Pathnames

APPEND followed by pathnames is used to specify a search path. Once typed in, APPEND specifies where Concurrent looks for files if it can't find them in the current directory. You can specify several search paths provided each is separated by semicolon (;), and provided the total length of the APPEND command is no more than 128 characters.

To return to the default value of no search path, type

A>APPEND ;

Warning:

Some applications cannot use the APPEND command; you will find that they read files successfully from directories other than the current directory, but they can only create files in the current directory. This results in two copies of files: an edited version on the current directory and the unedited version left unchanged on the original directory.

Examples: The following command sets up a search path such that if Concurrent can't find the file you want in your current directory, it looks in the directory c:\spreads\cashflow.

```
A>APPEND c:\spreads\cashflow
```

The following example extends the search path so that Concurrent also looks in the directory \oldones on the A: drive

```
A>APPEND c:\spreads\cashflow;a:\oldones
```

ATTRIB

Format: ATTRIB [/H] [+|-A] [+|-R] [+|-S] filespec [/P] [/S]

Explanation: ATTRIB displays a file's attributes, and can be used to switch the attributes on or off. The attributes included are the archive, read-only and system attributes.

The archive attribute shows that a file has been modified. When the computer is started, all the files have archive attribute switched off. Whenever a file is written to and then closed, Concurrent sets (switches on) the archive attribute automatically. The ATTRIB command enables you manually to switch the archive bit on or off for any given file.

The system attribute, when switched on, prevents DIR from displaying the file and COPY from copying it. It's used for the Concurrent system files that you don't need to see in everyday use of the computer. (Note: XCOPY has an option that allows you to copy system files.)

A file that is set to be read-only can't be erased, renamed, copied or altered in any of the normal ways. Setting a file with the read-only attribute should protect it from being inadvertently altered. However, the REPLACE, TOUCH and XCOPY commands all have [/R] options that allow you to alter read-only files.

The ATTRIB options are as follows:

| | |
|----|---|
| /H | Displays help information. |
| +A | Sets the archive attribute (-A switches it off) |
| +R | Sets the read-only attribute (-R switches it off) |
| +S | Sets the system attribute (-S switches it off) |
| /P | Pauses after each screenful in a long display |
| /S | Sets the attributes in the subdirectories associated from the chosen directory. |

Examples

A>ATTRIB +A *.NEW

The above example switches on the archive attribute for all the files with the extension .NEW in the current directory. You could then use a command like XCOPY /M to copy all the files marked for archiving.

A>ATTRIB -R *.BAS /S

This example switches off the read-only attribute from all the files with the BAS extension in the current directory and any associated subdirectories.

AUX

Format: AUX
AUX=n

Explanation: The AUX command selects one of the auxiliary ports. The number of auxiliary ports in your system depends on the system configuration. Each window can have a different AUX assignment, as follows:

AUX 0 = Serial Com1
AUX 1 = Serial Com2
etc.

Examples: When you enter AUX with no parameters, Concurrent displays your window's current AUX number. For example:

```
A>AUX  
AUX number = 0
```

Enter AUX with a parameter and Concurrent changes the status of your current window and displays the new AUX assignment. If you enter:

```
A>AUX=1
```

Concurrent displays:

```
AUX number = 1
```

BACKUP

Explanation: The Concurrent BACKUP utility copies files from a hard disk drive onto floppy disks, and it can restore files from BACKUP floppy disks to that hard disk drive. If the amount of space on a floppy disk is insufficient to hold all the files to be copied from a hard disk drive, BACKUP can copy files onto a series of disks. Each floppy disk is numbered serially starting from 1. When files are restored to a hard disk drive, BACKUP checks that the floppy disks are inserted in the correct order.

Floppy disks to be used by BACKUP must be formatted as DOS media. If disks previously used for BACKUP are recycled for use by BACKUP again, they must be reformatted.

To backup files from a hard disk drive C of a system onto floppy disks in drive A, proceed as follows. This procedure assumes that drive C is your default drive.

1. Enter

BACKUP

at the prompt and press the Enter key. The following message appears on the screen:

```

BACKUP Backup/Restore   dd/mm/yy   Version n.n
Serial No. XXX-999-999999   All rights reserved
Copyright (c) 1985         Digital Research Inc.
    
```

Do you want to backup or restore files (enter B or R)?>

2. Press "B". The following message appears:

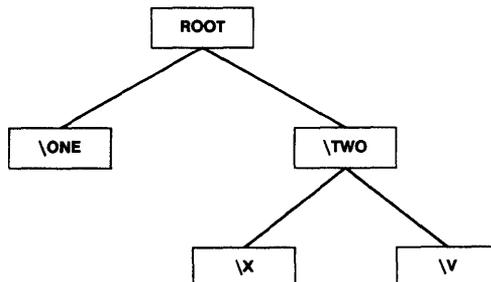
Which drive do you want to backup (A-P)?

Enter "C" (or the letter that corresponds to your hard disk drive).
The next prompt is

Which directories do you want to backup:

- A - All directories on the specified drives
- S - Current directory and its subdirectories only
- C - Current directory only

Please select (A/S/C):



3. Suppose that the above diagram represents the file directory structure for drive C and the current directory level is \TWO. Option A copies all the files for all directory levels on drive A. Option S will copy files only from subdirectories \TWO, \TWO\X, and \TWO\Y. Option C only copies files from \TWO.

Note: Before selecting one of these three options, be clear on the effect your choice might have on other users' files during a restore operation. If you choose A or S, you might include a subdirectory of files belonging to another user. When you restore these subdirectories, that user's files could be overwritten with obsolete copies.

4. After selecting your option, this prompt appears on your screen:

Which files do you want to back up?

A - All files in specified directories

N - New and modified files in specified directories

D - Files created or modified after a specified date

Please select (A/N/D)

Option A copies all files for the directory levels selected in 2. above. Option N selects only those files that have been created or modified since the last time BACKUP was run. Option D selects only those files that have been created or modified since a particular date. If you select option D, the following additional prompt appears:

Start date for back-up (format DDMMYY e.g. 311285)

Specify six digits or hit RETURN only for no date.

Enter a valid date in DDMMYY form.

5. The next message asks

Do you want to see the file names (Y/N)

If you press "Y", BACKUP displays a list of all the files that are copied to floppy disks. The display of filenames takes the form:

___Files found___file(s)

A:\directory_1\directory_2\...\filename.ext

When copying is complete, the program ends with the message:

---- Backup/Restore completed ----

Multiple Floppy Disks

If the file to be copied cannot fit on one floppy disk, BACKUP displays the following message when a disk is full:

```
Destination disk full
Insert formatted blank disk and press RETURN when ready
```

When you have followed these instructions, the following message appears to confirm that the backup is continuing on the disk just inserted:

```
Backing up on disk number n, created [date] [time]
```

Restoring Files to a Hard Disk

To restore files,

1. Select option R at step 1 above. The following message appears:

```
Restore to which drive (A-P)
```

2. Enter the drive designation of your hard disk drive (C, in this example). You can restore your files to a different drive than the one from which they were copied.

The next prompt asks you to name the floppy disk drive you will use for the backup disks:

```
Restore from which drive (A-P)
```

3. Enter the name of the floppy disk drive you are going to use. BACKUP then displays the message:

```
Insert backup disk number 1 into drive A:
and press any key when ready:
```

4. After you have inserted the disk and pressed a key, BACKUP restores files to their original subdirectory levels with their filenames unchanged. If the file already exists on the hard disk drive, it is overwritten. As each file is restored, the following message appears:

```
---nnnnn -dd-mm-yy hh:mm X:\level1\level2...\filename.ext
```

where X: is the hard disk drive designation, "\level1\level2" is the full directory path, and "filename.ext" is the filename of the file being restored. When the message is complete, the program ends with the message:

```
----Backup/Restore Complete----
```

BANK

Format: BANK=ON
BANK=OFF

Explanation: The BANK command controls how a program runs in banked memory. The default setting is BANK=ON. With BANK=OFF, Concurrent cannot page the program in and out of conventional memory. This means that the memory used by the program is removed from the total memory available for paging. Using BANK=OFF can severely limit system performance.

The Bank command is overwritten by PIFED information (see the PIFED command).

BATCH programming subcommands

Explanation: BATCH files are files containing a list of Concurrent instructions that Concurrent executes automatically once you run the BATCH file. The principles and special commands used in creating BATCH files are described below.

Invoking and Interrupting a BATCH File

Invoke a BATCH file by entering the filename with or without the file extension, which is always BAT. Variables are optional and are explained below in "Using Variables in a BATCH File."

You can interrupt a BATCH file by entering a Ctrl-C, causing the following prompt to appear:

```
Terminate batch job (Y/N)?
```

If you press Y, batch processing ends and the command line prompt reappears. If you press N, the current command ends, but batch processing continues with the next command in the file.

Creating a BATCH File

Use COPY CON: or an editor such as DR EDIX(tm) to create a BATCH file. COPY CON: copies information typed at the console to the file specified. The following example uses COPY CON: to create the BATCH file ENDWORK.BAT:

```
A>COPY CON: ENDWORK.BAT
ERASE B:*.BAK
DIR B:
SHOW B:
COPY B:*.* A:
DIR A:
SHOW A:
```

Press Ctrl-Z, then press the Enter key to end the file ENDWORK.BAT.

Once ENDWORK.BAT is created, type ENDWORK to execute the six commands in the BATCH file in the order listed.

To invoke a BATCH file from within another BATCH file, include a command line with BATCH followed by the name of a BATCH file as one of the commands to be executed. To chain BATCH files, use the name of the next BATCH file to be executed as the last command in the BATCH file.

Using Variables in a BATCH File

When creating a BATCH file, you can use variables as a part of the commands to be executed. Then the command invoking the BATCH file can contain the actual values for the variables. Thus, you can create general purpose BATCH files.

Variables take the form %n, where n is a number from zero to nine. For example, the variables %0, %1, and %2 are used in the following BATCH file, SAVE.BAT:

```
A>COPY CON: SAVE.BAT
COPY %1.BAS %2.BAS
TYPE %0.BAT
```

Press Ctrl-Z, then Enter, to complete the BATCH file.

The command to invoke SAVE.BAT might look as follows:

```
A>SAVE PROGRAM1 B:PROGRAM1
```

SAVE replaces %0, PROGRAM1 replaces %1, and B:PROGRAM1 replaces %2. The command copies PROGRAM1.BAS from the current drive to drive B, then shows the contents of the SAVE.BAT file on the screen.

Remember that you can specify up to 10 variables (%0 through %9). %0 is always the name of the BATCH file and its drive specification, if included. Separate variable values by a space. Make sure that values are entered in the proper sequence. If you want to use % as a filename within a batch file, you must include % twice. For example, if the filename is test%.dat, the batch file must specify test%%.dat.

Environment Variables

Batch files can also retrieve and use environment variables such as "os" (defining the operating system); "ver" (the current version number) and "path" (the current path). See also the SET command.

The environment variable names should be surrounded by % signs and quotation marks when used in batch files.

In the following example, the batch file identifies the environment as either Concurrent DOS or MS-DOS, and then moves to either label "cdos" or label "msdos", elsewhere in the batch file.

```
If "%os%" == "cdos" goto cdos
If "%os%" == "" goto msdos
```

CALL Subcommand

CALL allows a batch file to load and run a second batch file. The subcommand format is:

```
CALL [d:][path]filename
```

CALL subcommands can be placed at any line in the batch file. You can even use CALL to make a batch file call itself – though you need to be careful to avoid creating an infinite loop.

The following example calls the batch programs second.bat and third.bat and processes them in turn:

```
CALL second  
CALL third
```

ECHO Subcommand

ECHO allows or inhibits the screen display of DOS commands executed from a batch file. It does not affect the messages produced during command execution. The ECHO subcommand format is:

```
ECHO [ON|OFF|message]
```

ECHO is ON by default, and shows all commands as they run. ECHO OFF stops all commands from appearing on the screen.

The ECHO message option displays messages on the screen whether or not ECHO is ON or OFF.

The current ECHO condition of ON or OFF is displayed if ECHO is issued with no parameters.

Note also that preceding a batch subcommand with the @ character prevents the command from being displayed. For example,

```
@ ECHO OFF
```

turns ECHO OFF but the words "echo off" are not displayed on the screen when the batch file is run.

EXIT Subcommand

Exit allows you to terminate the batch program before you reach the end of the file. The command format is simply

```
EXIT
```

FOR Subcommand

FOR allows iterative execution of a command. Each filename in the set is sequentially substituted in the command following DO. Wildcard * or ? characters are valid for members of the set and all matching filenames on the disk are substituted in the command. Only one FOR command can be used on a command line. Path names are not accepted with filenames in the set. The format is:

```
FOR %%variable IN (set) DO command
```

If you enter this command:

```
FOR %%f IN (myprog.asm acct.bas acct2.bas)DO type %%f ]
```

The three specified files are each typed to the screen in turn.

GOTO Subcommand

GOTO transfers control to the line following a label. The GOTO label causes the commands on the line after label to process. If label is not named, the current batch file ends with this message:

```
Label not found
```

To set up a label in a batch file, use a colon (:) followed by a character string. The first eight characters of the string are considered the significant characters, and they must be different from those of all other labels. The format for GOTO is:

```
GOTO label
```

The following batch file sets up an infinite loop of messages on the screen. Your screen displays "rem this is a loop ..." and "GOTO MYLABEL".

```
:MYLABEL  
rem this is a loop..  
GOTO MYLABEL
```

Labels within a batch file are not shown while the file is executing. Unreferenced labels may be used to place comments in your batch file.

IF Subcommand

IF allows conditional processing of DOS commands. The DOS command executes when the IF condition is true. When the condition is false, the DOS command is ignored and the next command executes. The IF format is:

```
IF [NOT] condition command
```

The condition is one of the following:

- * ERRORLEVEL number
- * string1==string2
- * EXIST filespec

ERRORLEVEL number is true if the previous program had an exit code (specified as a binary value) of number or higher. Your programs can set an error code that can then be checked by the IF ERRORLEVEL command.

When string1 and string2 are identical, this condition is true.

If filespec is on the indicated drive, EXIST filespec is true. Path names are allowed with the filespec.

NOT condition is true if the condition is false.

Examples:

IF EXIST filespec command:

```
if exist myfile goto xyz
dir b:
.
.
:xyz
command
```

This condition is true if executed with myfile on the default drive. The command following the label xyz is executed. If myfile is not found, the goto xyz is not executed and processing continues with the dir b: command in the batch file.

IF "string1"=="string2" command:

```
if "%1"=="John" echo What are you doing here?
```

The batch file with this command, assuming John is named as the %1 parameter, performs the ECHO batch command. "What are you doing here?" is displayed. If Joan is given as the %1 parameter, the condition is false, and the ECHO batch command is not executed.

Note that it's advisable to put quote marks around the text strings.

IF ERRORLEVEL number command:

```
tstprog
if errorlevel 1 echo tstprog failure.
dir b:
```

TSTPROG is a program that sets the errorlevel when it ends. Assume TSTPROG sets the errorlevel to 0 if it finishes successfully and sets the errorlevel to 1 if it fails. If TSTPROG fails, the condition is true and the ECHO batch command is processed, displaying the message following the echo command. If TSTPROG is successful, the condition is false and the ECHO batch command does not execute. The next command in the batch file, dir b:, is processed.

IF NOT EXIST filespec command:

```
if not exist a:%1 copy b:%1 a:
myprog1
```

The batch file is looking for a certain file to be on Drive A. The IF condition is executed before MYPROG1 to ensure that the needed file is on Drive A. If the file is not on Drive A, the condition is true and the copy command is processed, copying the file from Drive B to Drive A. If the file is on Drive A, the copy is not processed and MYPROG1 will run.

PAUSE Subcommand

You can insert a PAUSE subcommand in a BATCH file to pause the processing of the file. The pause allows you to display a message or let an action occur, such as changing disks or aborting the file. The PAUSE subcommand takes the form:

```
PAUSE [remark]
```

After the optional remark or message, PAUSE always inserts the following message:

```
Strike a key when ready ...
```

The optional remark can contain a maximum of 121 characters. The following example shows a PAUSE subcommand in a BATCH file:

```
A>COPY CON: STORAGE.BAT
```

```
COPY A:*. * B:
PAUSE Insert new disk in drive A
COPY A:*. * B:
```

When STORAGE is run, the first disk in drive A is copied. Then the processing pauses and this message appears:

```
A>Pause Insert new disk in drive A
Strike a key when ready ...
```

Two responses are possible. Insert the new disk and strike any key to continue, or press Ctrl-C to terminate the processing.

REM Subcommand

Use REM to display a comment on your screen during BATCH file execution. REM takes the form:

```
REM [remark]
```

The maximum length of a remark is 123 characters. The following is an example of a REM command:

```
REM Archive backup files for security
```

SHIFT Subcommand

SHIFT allows command lines to access more than 10 (%0 through %9) replacement variables. Execute the SHIFT command to get past the tenth parameter on the parameter list. The parameters on the command line are shifted one position to the left; the parameter in %1 position moves to the 0% position. Each shift command moves all the parameters to the left by one more position. For example:

```
%0 = Yellow
%1 = Orange
%2 = Red
%3 = Blue
.
.
%9
```

Execute the SHIFT command and the results are:

```
%0 = Orange
%1 = Red
%2 = Blue
.
.
%9
```

BREAK

Format: BREAK [ON|OFF]

Explanation: BREAK when switched ON allows you to stop a running program by pressing <CTRL><Break>. The program will then stop the next time it tries to call any DOS function.

If BREAK is OFF (the default value), you will only be able to issue a <CTRL><BREAK> when the program you want to stop is performing standard input or output operations such as printing or outputting to the console.

If you type BREAK on its own, you will see the current BREAK value.

BREAK only affects DOS programs.

Example: A>BREAK ON

This switches BREAK ON. BREAK will now remain ON in this window until you issue another BREAK command or restart Concurrent.

CARDFILE

Format: CARDFILE
CARDFILE COLOR

Explanation: Use the CARDFILE command to run the Cardfile™ name and address program. See Section 5 of the User's Guide for detailed instructions on its use.

If you have a hard disk system, and have created a NAME.IDX file, you can start Cardfile from the Startup Menu by pressing F5. You can also run Cardfile from the File Manager by selecting Run a Program and then choosing CARDFILE.COMD. To start Cardfile from the command line prompt, type CARDFILE if you have a monochrome monitor, or CARDFILE COLOR if you have a color monitor.

CHDIR

Format: CHDIR [fd:=][d:][\] [/A]
CHDIR [fd:=][d:]dirpath [/A]

Explanation: The CHDIR (change directory) command provides access to directories. Use CHDIR to change the current directory on a drive, assign floating drives to a directory, and display the current directory path. When no directory or directory path is specified, CHDIR displays the sequence, called the path, of hierarchical directories leading to the current directory. If a path is specified, CHDIR makes the last directory in the path the current directory for the current drive or the drive specified. The short form of the CHDIR command is CD.

CHDIR Option

/A The /A option lists the current directory path for all the current drives.

CHDIR Special Characters

To help you move between directories, CHDIR responds to the following special characters:

* \ (backslash): Enter a backslash alone with CHDIR and you are returned to the current or designated drive's root directory. For example, the command

```
A>CHDIR C:\
```

makes the root directory on drive C the current directory.

* .. (two periods): Enter two periods with CHDIR and you are returned to the next highest (that is, closer to the root) directory. For example, if the directory LEVEL4 in the path LEVEL2\LEVEL3\LEVEL4 is the current directory, the command

```
A>CHDIR ..
```

makes LEVEL3 the current directory.

Concurrent Directories

Concurrent allows you to create named directories with the MKDIR command. Directories appear by name in the next higher directory's DIR file listing and are identified by a preceding backslash (\).

Files in a directory are accessed by preceding the filename with a directory path, or by using CHDIR to change a drive's current directory to the desired path and referencing the files normally (without a path).

Use the second method with programs that do not understand paths. In these cases, floating drives can also be useful (see below).

Using Floating Drives

Concurrent allows you to use up to 25 drives, from A: to Z:, excluding L: (which Concurrent reserves for its own use). All the drives that are not associated with diskette, hard disk, and memory drives are free to be used as "floating drives".

A floating drive temporarily replaces a drive-and-path specification. For example, you could assign drive W: to replace C:\sub1\sub2\sub3. Then, instead of having to type out the path each time, all you need to enter is the floating drive letter. Floating drives remain assigned until you re-assign them or restart the system.

Floating drives have two main uses:

- 1 To save you typing
- 2 To use with programs that don't recognize subdirectories.

Note: Each window keeps a separate list of current directories for each drive.

Examples:

```
A>CHDIR O: =C:SUB1\SUB2\SUB3
A>O:
```

This example assigns the floating drive O: to the subdirectory SUB3 and then makes drive O (actually SUB3) the current directory.

```
C>CHDIR N: =\GENLEDG
```

The above command sequence assigns floating drive N to the subdirectory GENLEDG on drive C. GENLEDG is a subdirectory in the current directory.

```
B>CD O: =C:
```

This command assigns the current path of Drive C to floating drive O:

Making a Directory the Current Directory

There are three ways to make a directory the current directory:

- * Enter CHDIR with the directory's path.
- * Enter CHDIR with a drive specification and a path and then make that the active drive.
- * Use CHDIR to name the directory with a floating drive name and then make that drive the active drive.

If the directory you are currently in is not part of the path to the directory you want, you must specify the complete path from the root, preceding the first directory with a backslash.

Examples: C>CHDIR\GENLEDG\AR

The above example makes AR the current directory. The backslash that precedes GENLEDG indicates that the path to AR begins from the root of drive C, not from the current directory.

C>CHDIR A:\WORDPROC\MEMOS

The preceding command sets drive A's current directory to MEMOS. All references to A: now access this directory.

C>CHDIR\

This command makes the drive C root directory the current directory.

How to Display the Current Path

The path to the current directory is displayed when you enter CHDIR with no parameters. You can display the path to the current directory on another drive by entering the drive's name as the only parameter. Note that a path consisting of just a drive and a backslash (\) indicates that the root directory is the current directory.

CHKDSK

Format: CHKDSK [d:][option]

Explanation: The CHKDSK command checks the integrity of data on disks, and restores corrupted disks. If you omit the drive, CHKDSK uses the default drive.

Table 3 lists the options for CHKDSK.

Table 3. CHKDSK Options

| Option | Description |
|--------|---|
| /B | Mark bad blocks. The /B option searches all the files on a disk to locate any bad blocks. |
| /D | Locate directories. The /D option locates all the directories on a disk so that in the event of a major disk corruption, you can restore individual files and directories using a suitable disk editor. |
| /F | Fix. The /F option allows CHKDSK to write corrections back to the disk. If you do not specify /F, CHKDSK does not modify the disk when errors are found. |
| /L | Rebuild cluster links. The /L option attempts to rebuild a corrupted File Allocation Table (FAT). You must zero any corrupted areas on the disk before using this option. |
| /R | Recover root directory. The /R option recovers lost directories that were contained in the root directory. You must remove any corrupted files in the root directory. |
| /V | Verbose display. The /V option displays additional information while CHKDSK is running and may be useful when trying to locate corrupted data on a disk. |

CHSET**Format:**

```
CHSET [HELP]  
CHSET filespec  
CHSET filespec [field=setting[,field=setting...]]
```

Explanation:

CHSET (Command Header SET) displays or changes the command header of any program file with the .CMD file extension. The command header indicates whether the program requires the math co-processor, whether the program code can be shared, whether the program should be suspended when it is running on a switched-out window, and whether the program uses MP/M-86™ record-locking functions.

The command header also indicates the compatibility attributes. These are defined as attributes F1 to F4 of .CMD files, and they provide a mechanism to modify some of Concurrent's file security rules. This facility is needed because some programs developed under earlier Digital Research operating systems don't run properly under Concurrent. Most of the problems occur because the programs were designed for single-tasking operating systems where file security is not required. For more details see the Programmer's Guide.

File specifications in the CHSET command line require only the filename; the .CMD file extension is assumed. Changes made to the command header are permanent until changed. Table 4 lists the fields and possible settings.

Table 4. CHSET Fields and Settings

| Field | Setting | Description |
|---------|---------|--|
| BANK | ON | The program can be run in banked memory. |
| | OFF | The program cannot be run in banked memory. |
| 8087 | ON | The program requires an 8087 co-processor. |
| | OFF | The program does not require an 8087 co-processor. |
| | OPT | The program determines whether or not an 8087 co-processor is present and adjusts its operation accordingly. |
| SHARED | ON | Concurrent uses the same program code (but not the data) for multiple versions of the program running on different windows. This option must be set by the program's author. |
| | OFF | Concurrent uses separate program code for multiple versions of the program running on different windows. |
| SUSPEND | ON | Concurrent suspends but does not terminate the program's operation when its window is switched out. |
| | OFF | Concurrent does not stop program operation when its window is switched out. |
| MPMLOCK | ON | The program uses the record-locking functions of MP/M-86. |
| | OFF | The program uses the record-locking functions of Concurrent. |
| F1 | ON | If the program opens a file in Locked mode, this will not stop other programs from opening the file at the same time. |
| | OFF | Normal Concurrent file security rules apply. |
| F2 | ON | A program that closes a file and then continues to use it must have this compatibility attribute set. |
| | OFF | Normal Concurrent file security rules apply. |
| F3 | ON | Tells Concurrent to ignore close checksum errors. This attribute is intended for programs that modify an open FCB before closing a file. |
| | OFF | Normal Concurrent file security rules apply. |
| F4 | ON | Tells Concurrent to disable FCB checksum verification for read and write operations. This attribute is intended for programs that modify open FCBs during read and write operations. |
| F4 | OFF | Normal Concurrent file security rules apply. |

To get helpful information about CHSET, type

```
A>CHSET
```

To display a CMD program's current command header settings, type CHSET and a filename. For example,

```
A>CHSET PRINTMGR
```

gives the following display for the PRINTMGR command:

```
CHSET.CMD [8087=OFF, SHARED=OFF, SUSPEND=OFF, MPMLOCK=OFF, BANK=ON]
```

CHSET accepts wildcards when you display file settings but not when you change a field. CHSET requires passwords when you display or change settings for password-protected files.

Examples:

```
A>CHSET MYPROG [BANK=ON]
```

This command tells Concurrent that the program MYPROG can run in banked memory, which means that Concurrent can page the program in and out of conventional memory.

```
A>CHSET EXSORT [8087=ON]
```

This command tells Concurrent that the program EXSORT requires the 8087 co-processor. With this setting, the program does not load unless the 8087 co-processor is available.

```
A>CHSET ANYPROG [SHARED=ON]
```

The next command tells Concurrent that ANYPROG can run on a shared-code basis, which saves memory when multiple copies are run. This is a feature that must be written in by the program's author. Only select the SHARED=ON option if the program's manual states that this option is available.

```
A>CHSET MYPROG [SUSPEND=ON]
```

This command causes MYPROG to suspend when its window is switched out. When the window is switched back in, the program continues.

```
A>CHSET YOURPROG [MPMLOCK=ON]
```

This command tells Concurrent that the program uses the record-locking functions included in MP/M-86.

```
A>CHSET OLDPROG [F1=ON, F3=ON]
```

This command specifies a combination of compatibility attributes set in OLDPROG's command file.

CLS

Format: CLS

Explanation: CLS clears the display screen. If foreground and background colors were selected previously, the colors will be unchanged. Otherwise, the screen is set to white characters on a black background.

COPY

Format: COPY [dirpath\]source[/option][dirpath\]destination[/option]

Explanation: The COPY command copies files between consoles, printers, and drives. The source can be a file in a disk-drive directory or a console keyboard. The destination can be a disk-drive directory, a console screen, or a printer.

A disk-drive directory can be specified by a drive letter followed by a colon, and/or a path specifier. If you omit both, the current drive/directory is assumed.

The printer and console devices are specified by the names shown in Table 5.

The source and destination file specifications are the complete filename and file extension. You can use the wildcard characters (* and ?) to specify a group of files.

COPY can also be used with password-protected files but note:

- * COPY won't prompt you for a password
- * The destination copy of the file won't have any password protection unless you assign a password with the PASSWORD command.

Table 5. COPY Device Names and Descriptions

| Name | Description |
|--|---|
| CON:* | Designates your console screen when it is the destination and your keyboard when it is the source device. |
| LPT1:, LPT2: | Designate your current printers 0, 1 etc. |
| PRN: & LST: | Designate your current printer. |
| NUL: | Designates a dummy output device and is usually used for program testing. |
| COM1:, COM2: | Designate the serial ports 0, 1, etc. |
| AUX: | Designates your current auxillary port. |
| *Colons used with device names are optional. | |

COPY has several options that you can add to a source or destination file specification. Table 6 lists and defines the options.

Table 6. COPY Options

| Option | Description |
|--------|--|
| /A | Causes the file to be treated as an ASCII (text) file. When you specify the /A option with the source file specification, the file contents are copied up to, but not including, the first Ctrl-Z end-of-file character. Remaining file characters are not copied. Using /A with a destination file specification adds a Ctrl-Z as the last character in the file. |
| /B | When applied to the source file specification, causes the entire file to be copied regardless of any Ctrl-Z character found in the file. When you use /B with your destination file specification, no end-of-file character (Ctrl-Z) is added. |
| /V | Causes COPY to compare the destination file with the source file contents to verify the accuracy of the copy. Because of the comparison, it takes longer to complete the copy with the /V option. |
| /S | Includes SYS and hidden files in the copy. Usually, only normal attribute files are copied. |
| /C | Confirms each file before copying it. If combining multiple groups of files, confirms each group by first source name. |
| + | Valid only in the source specification. It is used between file specifications to indicate that the files are to be combined (concatenated). |

The /A and /B options need be entered only when you are combining ASCII and binary files. These options are not necessary to copy data, program, and BATCH files.

How to Copy Files between Disk Drives and Directories

This part explains how to copy files between drives, within a single directory, between subdirectories, between user numbers, how to combine files, and how to change the filename.

Format: COPY filespec[/option] [filespec[/option] filespec[/option]]

Explanation: To copy a file from the current drive to another drive, type COPY followed by the filename, the letter designation of the destination drive, and a colon. The following example copies the file MYFILE from drive A (current drive) to drive B:

A>COPY MYFILE B:

To copy a file to the current drive from another drive, type COPY, then the letter designation of the source drive, a colon, and the filename. The following command line copies MYFILE.TYP on drive B to the current drive A:

```
A>COPY B:MYFILE.TYP
```

To duplicate a file in the same directory, rename the file as you copy it, because you cannot have two files with the same filename in one directory. The following command line duplicates the file MYFILE.TYP in the same directory and names the copy YOURFILE.TYP:

```
A>COPY MYFILE.TYP YOURFILE.TYP
```

Note that you can copy a file from one drive to a different drive by specifying drive designations before the filenames. The following example copies PHONE.LST from drive B to drive D and verifies the destination file against the source file for an exact match:

```
A>COPY B:PHONE.LST D:PHONE.LST/V
```

You can combine files with COPY by putting a plus sign, +, between the source file specifications. The files are concatenated in the order they are listed in the command or, if there is a wildcard character, in the order they are encountered in the directory. The following command line combines MFILE.TYP on drive A with YFILE.TYP on drive B, copies the file to drive C, and names it OFILE.TYP:

```
A>COPY MFILE.TYP+B:YFILE.TYP C:OFILE.TYP
```

When copying a file to a different drive or subdirectory you can rename it on the destination. For example, the following command line copies the file NEWPHONE.LST, from drive A onto drive B, and renames it OLDPHONE.LST:

```
A>COPY NEWPHONE.LST B:OLDPHONE.LST
```

Note that if you enter a destination file specification that matches one already on the destination drive, the copy replaces the existing file. The existing file is erased and then the copy is made.

How to Copy Files to the Console or Printer

You can use COPY to display or print a file.

Format: COPY filespec DEV:

Explanation: COPY can be used instead of TYPE or PRINTMGR to display or print a file. To perform this type of copy, you enter CON: or PRN: as the destination. This command is valid only for text (ASCII) files, not application or command program files.

Examples: A>COPY D:MYFILE CON:

This command reads the file MYFILE from drive D and displays it on the console.

A>COPY *.TXT PRN:

The above command prints all files with the file extension TXT on the current drive.

How to Create a File with COPY

You can create a text file using COPY by specifying the keyboard as the source.

Format: COPY CON: filespec

Explanation: When you specify CON: as the source device, every character you enter is recorded in the destination file you specified except line-editing characters such as Ctrl-H and Ctrl-X. Enter a Ctrl-Z to stop the recording and close the file.

Example: A>COPY CON: MYFILE
This is my file.<Ctrl-Z>

This entry creates the file MYFILE on the current disk and records in it the second line shown. After the Ctrl-Z is entered, the command-line prompt is returned to the screen.

COPYMENU**Format:**

```
COPYMENU  
COPYMENU source-filespec destination-filespec -M  
COPYMENU source-filespec destination-filespec menulist
```

Explanation:

Use COPYMENU to copy one or more specific menus between existing menu files. COPYMENU is one of the three commands--the others are EDITMENU and RUNMENU--provided with Concurrent that allow you to construct and run easy-to-use menus.

See Section 7 in the User's Guide for more information.

CPM

Format: CPM

Explanation: The CPM command is a data transfer utility for CP/M files and media. After you enter, for example,

```
A>CPM
```

you will see a new prompt appear, for example:

```
[CP/M] 0A>
```

This shows that the CPM command is running and that the current drive is A.

When the CPM command is running you can use all the Concurrent commands involved with accessing files such as DIR, ERASE, COPY, and TYPE. Use these just as they are explained in the rest of the manual. When you copy a file from a CP/M disk to a DOS disk (or back again), the CPM command will ensure that the data is in the right format for each disk.

Batch files can be run from the CPM prompt, but you can't run applications programs: the CPM command is only designed for data transfer. If you want to process information stored on CP/M media, you need first to copy the data onto a DOS disk and then run the application from the Concurrent prompt in the ordinary way.

The USER subcommand can only be run from the CPM prompt. USER switches the current CP/M user number. User numbers in CP/M range between 0 and 15.

The EXIT subcommand takes you back to the Concurrent prompt.

Examples: In the following example the CPM command is started, the default drive is changed to B and the current user number is changed from 0 to 7.

```
A>CPM
```

```
[CP/M] 0A>
```

```
[CP/M] 0A>B:
```

```
[CP/M] 0B>USER = 7
```

```
[CP/M] 7B>
```

The example below assumes the CPM command is already running, and that a CP/M disk is in drive A: and a Concurrent DOS disk is in drive B: The example shows how to copy all the files from A: to B: The files will be converted to DOS format as they are written to B:

```
[CP/M] OA>COPY *.* B:
```

The next example shows how to return from the CPM prompt to the Concurrent DOS prompt:

```
[CP/M] OA>EXIT
```

```
B>
```

DATE

Format: DATE [mm-dd-yy]
DATE [dd-mm-yy]
DATE [yy-mm-dd]

Explanation: The DATE command displays or sets the month, day, and year.

Concurrent contains its own internal clock, which needs to be set using the DATE and TIME commands each time you start the computer or reset Concurrent. (However, some systems enable Concurrent to read the time directly from a battery-powered clock within the computer.)

Concurrent maintains the internal clock for two reasons:

- * to provide you and your applications continuous access to the date and time
- * to have the time available for file time and date stamping.

(Time and date stamping records the current time and date whenever you create or update a file.)

To be prompted automatically whenever you start or reset your computer, include DATE and TIME commands in your STARTnnn.BAT file.

If you do not enter the date, Concurrent assumes one. The date assumed varies with your Concurrent release. The DATE command always lets you change the current date. When you enter

A>DATE

DATE displays the current date and prompts you as follows:

Current date is day mm/dd/yy
Enter new date:

"day" in the first display line indicates day of the week while mm/dd/yy indicates month, day, and year. You can enter a new date or press the Enter key to keep the current values. To set the date from the command line prompt, enter the following:

A>DATE mm-dd-yy

You can use either a hyphen (-) or a slash (/) to separate your entries. The date values in this form are defined as follows:

- * mm is a one- or two-digit number from 1 to 12 indicating the month.
- * dd is a one- or two-digit number from 1 to 31 indicating the day.
- * yy is a two- or four-digit number indicating the year.

If you enter a date that is incorrect or out of range, DATE returns the following message and prompt:

```
Invalid date  
Enter new date:
```

Enter the date as described above or press the Enter key to keep the previous value.

Note: The way the date is displayed depends on the country that your system is configured for. The US form is mm-dd-yy, the European is dd-mm-yy and the Japanese, yy-mm-dd. You can change your configuration by making a COUNTRY entry in the CCONFIG.SYS file.

DELQ

Format: DELQ [dirpath\] filespec

Explanation: The DELQ and ERAQ commands have the same format and each erases files. Each also queries you to make sure you want to erase each file. See the description of the ERAQ command for more details about the DELQ command.

DIR

Format: DIR [d:][dirpath\][filespec][/W|/L][/D|/S|/A][/P][/V][/R|/C]

Explanation: DIR displays the contents of a disk directory. DIR with no parameters displays all files with the DIR attribute and all subdirectories in the current drive's current directory. Files with the system attribute are not shown. (See the ATTRIB command explanation for a description of attributes.) Subdirectories are separated from each other in the command line by a backslash (\).

DIR shows the contents of the current drive's or the designated drive's current directory unless you specify another drive or path.

You can select a file or class of files for display by entering a specific file specification or a general file specification using the wildcard characters (* and ?). For example, you can enter

```
A>DIR MYFILE.GTM
```

and DIR will search the current directory for the MYFILE.GTM file. If it cannot find it, this message is displayed:

```
File not found
```

The DIR display shows the following information about the current directory:

- * Drive selected.
- * Volume label of current drive.
- * Full path used for file search.
- * List of file names.
- * Number of files found.
- * Amount of storage space left, in bytes.
- * Whether any system files were omitted from the display.

The DIR options can be entered individually or in combination and in any order. Table 7 lists and describes these options.

Table 7. DIR Options

| Option | Description |
|--------|---|
| /D | DIR option. Displays files with the DIR attribute. This is a default option. |
| /S | SYS option. Displays files with the SYS attribute. |
| /A | All option. Displays both the SYS and DIR attribute files. |
| /W | Wide display. Displays only the names of files with Backslashes (\) before directories. A default option. |
| /L | Long display. Includes the size, date, and time. |
| /P | Page option. Pause at end of each full page. The default is no paging |
| /N | No paging. This allows you to return to the default paging option. |
| /R | Remember option. Tells DIR to remember new options you specified as the default option. |
| /C | Change option. Like /R, except the output is not generated. Used to set the default options only. |

Examples: A>DIR C:/S

This command displays all files and subdirectories in drive C's current directory with the system attribute.

The next command displays all files and subdirectories with the CMD file extension in drive A's current directory. It uses the long format, giving information on size and date and time stamping. The display pauses with each full screen. SYS and DIR attribute files are included and the options are remembered for the next DIR command without options. The filespec is not remembered.

A>DIR *.CMD/L/P/A/R

This command displays all files and subdirectories that do not have an extension. If you omit the period, DIR lists all files.

A>DIR *.

The next command sets the default options of DIR (wide, DIR files) without displaying a directory. These are the default options enabled each time you restart Concurrent. This type of command is useful in your Startnnn.bat file. Each window keeps a separate set of default options.

A>DIR/W/D/C

DISKCOMP

Format: DISKCOMP [d: [d:]] [/1] [/8] [/V]

Explanation: DISKCOMP compares two entire diskettes - for example to check that a DISKCOPY operation has worked correctly. If you specify DISKCOMP with no drives, DISKCOMP will assume the current drive. If you don't specify two drives, DISKCOMP will prompt you to swap diskettes as necessary.

Note: if you copy all the files from one diskette to another with the COPY (rather than DISKCOPY command), DISKCOMP will probably return an error, even if the copying is done correctly. This is because COPY may place files on a destination diskette in a different order to the one they had on the source diskette. (The way to check a COPY operation has been performed correctly is to use the COPY /V option.)

The options are:

- /1 This compares only the first side of the diskettes, even if the diskettes are double-sided
- /8 This compares only eight sectors per track, even if the first diskette contains more sectors per track.
- /V This verifies that the whole disk can be read.

Examples: A>DISKCOMP

This compares the first diskette in drive A with a second diskette in drive A, and prompts you to change diskettes where necessary.

A>DISKCOMP a: b:

This compares the diskette in drive A with the diskette in drive B.

DISKCOPY

Format: DISKCOPY [/H] [source drive] [destination drive] [/1]

Explanation: The DISKCOPY command copies entire disks of the same format. If the destination disk has a different format, or is unformatted, DISKCOPY formats it the same as the source disk (where this is possible). If you want to copy a disk to a different format, use COPY.

If you omit the destination drive, DISKCOPY copies to the default drive, not the source drive. If the source and destination drives are identical, DISKCOPY prompts you when you need to swap disks. If you omit both drives, DISKCOPY copies to the default drive.

The /H option displays help information.

The /1 option causes DISKCOPY only to copy side 1 of a diskette. This enables you to copy single-sided diskettes on drives designed for double-sided use.

Example: When you enter the command,

```
A>DISKCOPY A: A:
```

DISKCOPY prompts you to insert the source disk. When the information on the source disk is copied into memory, you are then prompted to insert the destination disk. The information in memory is then copied onto the destination disk.

Drive Types and Diskette Types Supported by DISKCOPY

DISKCOPY only works with physical diskette drives (not hard disk drives or floating drives).

You need to know your drive type and diskette type before you attempt a DISKCOPY. Certain disk types cannot be copied in certain drive types; the DISKCOPY operation will fail. The permutations that do work are explained below:

DISKCOPY supports five different drive types:

- 5¼" single sided single density 40 track 180Kb
- 5¼" double sided single density 40 track 360Kb
- 5¼" double sided double density 80 track 1.2Mb

- 3½" double sided single density 80 track 720Kb
- 3½" double sided double density 80 track 1.44Mb

Between these drive types, diskettes can be formatted seven ways, each of which DISKCOPY supports:

5¼" single sided single density 40 track 160Kb or 180 Kb

5¼" double sided single density 40 track 320Kb or 360Kb

5¼" double sided double density 80 track 1.2Mb

3½" double sided single density 80 track 720Kb

3½" double sided double density 80 track 1.44Mb

DISKCOPY works with the following permutations:

A 180Kb drive can copy 160 and 180Kb diskettes.

A 360Kb drive can copy 160, 180, 320 and 360Kb diskettes.

A 1.2Mb drive can copy 1.2Mb diskettes. It can also copy 160, 180, 320 and 360Kb diskettes, but you may not be able to read these diskettes in a 180 or 360Kb drive.

A 720Kb drive can only copy 720Kb diskettes.

A 1.44Mb drive can copy 720Kb and 1.44Mb diskettes.

DREDIX

Format: DREDIX

Explanation: With DREDIX you can create and edit text files. DREDIX is explained in detail in Section 4 of the User's Guide.

To start DREDIX from the File Manager, select the Run a Program command and then choose file DREDIX.COM. To start DREDIX from the command line prompt, type the following:

```
A>DREDIX
```

Once you start DREDIX, you can press Alt-H for help.

DSKMAINT

Format: DSKMAINT

Explanation: DSKMAINT is a menu-driven utility for formatting floppy disks. It is called by the File Manager user interface for this purpose. If you prefer a menu-driven interface you may use DSKMAINT as an alternative to FORMAT, DISKCOPY and DISKCOMP.

EDITMENU

Format: EDITMENU
EDITMENU filespec

Explanation: EDITMENU, COPYMENU, and RUNMENU construct and run easy-to-use menus for your applications. EDITMENU allows you to create a new menu, or to edit, copy, or delete an existing menu. If you enter EDITMENU without a file specification, the default menu file MENU.DAT is assumed. If you include a file specification, it can be the name of an existing menu file or of a menu file you want to create. You can then create, edit, copy, or delete menus in the menu file.

See Section 7 in the User's Guide for more information.

ERAQ

Format: ERAQ [dirpath\] filespec

Explanation: The ERAQ and DELQ commands have the same form and each erases files. Each queries you before erasing a file. The ERASE command is similar to these commands but in most cases erases files without queries.

ERAQ and DELQ are most useful when used with the wildcard characters * and ? to erase an entire class of files. Instead of erasing every file that matches the file specification, ERAQ and DELQ list the specified files individually, followed by a ? prompt, to which you respond either with a Y for Yes (erase the file) or with an N for No (do not erase the file).

Example: A>ERAQ *.BAK

This example gives you the choice to delete each of the .BAK files in the current directory.

ERASE/ERA

Format: ERASE [dirpath\] filespec
DEL [dirpath\] filespec
ERA [dirpath\] filespec

Explanation: The ERASE, DEL, and ERA commands have the same form and each removes the specified file or files from the directory. The file specification can identify a particular file, or it can indicate a group of files through the wildcard characters * and ?.

After you erase a file, you cannot retrieve it, so you should be very careful when you use wildcards in your file specifications. Generally, ERASE does not query you before erasing files. However, it does query you when you use the asterisk wildcard for both the filename and the extension (*.*). For *.* it displays the following query:

Are you sure (Y/N)?

Type Y to erase all files on the (directory); type N to end the command.

Example: A>ERA *.BAK

This erases all the .BAK files on the current directory, without querying you.

FDISK

Format: FDISK

Explanation: The FDISK command allows you to prepare one or more hard disks for Concurrent DOS and to prepare a hard disk with partitions.

You can pass information between DOS partitions on a hard disk as if between different drives. For example, if your disk is divided into three DOS partitions, you can address these partitions as drives:

C:
D:
E:

When you type the command FDISK, the current partition status of the disk is displayed with a menu of options, as in the following example:

```
Fixed Disk Maintenance Utility
Version 1.00 (c) Digital Research 1986
```

```
Partitions on 1st hard disk (20.3 Mb, 614 tracks):
```

| No | Start | End | MB | Status | Type |
|----|-------|-----|------|--------|---------|
| 1 | 1 | 613 | 20.3 | A | DOS 3.x |

```
Select options:
```

- 1) Create DOS partition
- 2) Delete DOS partition
- 3) Select bootable partition
- 4) Select alternate fixed disk

```
Enter desired option: (ESC = exit)
```

Note: option 4 will only be displayed if your computer has multiple fixed disks

The current partition status of the disk is displayed under the following headings:

| | |
|--------|--|
| No | Each partition in the disk is numbered, in the order in which partitions appear on the disk, beginning at 1. |
| Start | The starting cylinder number for the partition. |
| End | The end cylinder number for the partition. |
| Mb | The size of the partition in megabytes. |
| Status | Either A or N. A indicates that the partition is active. This partition will be searched if Concurrent is booted from the hard disk. N indicates non-active. |
| Type | Indicates the format of the partition. This can be DOS (that is MS-DOS, DOS Plus, Concurrent DOS etc) or non-DOS. |

The options that you can select from the menu are described below.

Create DOS partition

Select this option if you are:

- * Preparing the disk for the first time
- * Adding a new partition to the disk

FDISK will warn you if there is not enough space to create a new partition or if all partitions are already in use.

When you select this option, you will be prompted to:

Enter starting cylinder

Type in the starting cylinder and press Enter. If you want the Concurrent partition to start in the first cylinder of the largest available space in the hard disk, leave this field blank and press Enter.

Note: if FDISK encounters a defective track, it will not create a partition. FDISK prompts you to try again with another starting cylinder number. Similarly, if there is insufficient space at a cylinder FDISK will prompt you to try another cylinder number.

You will then be prompted to

Enter ending cylinder

Type in the ending cylinder number and press Enter. You will need a minimum of 20 cylinders to run Concurrent. If you want the Concurrent partition to end in the last cylinder of the largest available space in the hard disk, leave this field blank and press Enter.

Note: if you plan to add another operating system to the hard disk, check the size requirements for that system before you allocate space to Concurrent.

You will be prompted to :

Enter disk label:

You may give the partition a label of up to eleven characters.

You will then be prompted to return to the options menu. Press Esc. The screen will display details of the new partition you have created.

Note: the partitions you create contain no program or data files. You must load the files you want into the required partition. For example, you may want to load Concurrent into a partition so that you can boot the system from the hard disk. To do this you would probably switch to drive A and copy Concurrent across from a diskette in drive A. Then use the Select bootable partition option (described below) to make the partition active.

Delete DOS partition

Select this option to delete a current partition. When a partition is deleted, all data is removed. You should first backup all files stored in the partition (either on diskettes or in another partition on the hard disk).

When you select this option, you will be prompted to:

Enter number of partition to delete

Type in the number of the partition you want to delete and press Enter.

You will be given a warning message:

Do you really want to erase all data (Y/N)?...

If you are ready to delete the partition, type Y and press Enter.

Select bootable partition

Use this option to select which partition on a hard disk is to be active. The machine will search this partition when Concurrent is booted from the hard disk.

When you select this option, you will be prompted to:

Select partition to make bootable.....

Type in the number of the partition you wish to make active and press Enter. This partition now has an active status (A). The partition that was previously active is now non-active (status N).

The computer will always try to load an operating system from drive A. If there is no diskette in drive A, or if drive A is not engaged, then the computer will try to load an operating system from the active partition on the hard disk.

Select alternate fixed disk

This option will only be displayed on the menu if you have multiple hard disks on your computer. Select this option if you want to switch from working with one disk to the other. The menu screen will display the current partition status of the disk as follows:

```
Fixed Disk Maintenance Utility
Version 1.00 (c) Digital Research 1986

Partitions on 2nd hard disk (20.3 Mb, 614 tracks):
No  Start   End   MB   Status  Type
1   1       304  10.1  A       DOS
2   305     613  10.2  N       Non-DOS
```

The options are exactly as for the first disk.

FIND

Format: FIND [/H] [/C] [/N] [/S] [/U] [/V] "string" filespec

Explanation: FIND searches for a text string in a group of files and displays the lines containing the search string. Multiple files may be specified on the command line. If no files are specified FIND will read from standard input.

The options are:

- /C Show only the total number of lines that contained "string".
- /N Display line numbers
- /S Search files in subdirectories that are derived from the current directory
- /U Case-sensitive search ("A" doesn't match "a" for example)
- /V Display the lines not containing "string".

Examples The following example calculates the number of lines containing the word "sorry" in each of the three files apology1.let, apology2.let and apology3.let.

```
A>find /C "sorry" apology1.let apology2.let apology3.let
```

The following example displays all the subdirectories listed in the current directory. In this example, FIND is acting as a "filter", receiving input from the DIR command and displaying as output only those files that contain the string "<DIR>":

```
A>Dir | find "<DIR>"
```

FM

Format: FM

Explanation: The Concurrent File Manager program allows you to select commands from a series of menus instead of typing them at the command line prompt.

See Section 3 of the User's Guide for a detailed description of File Manager operation.

FORMAT

Format: FORMAT [d:] [options]

Explanation: The FORMAT command formats diskettes in standard formats. (Use FDISK to format hard disks.) The default format for a diskette depends on the drive type: for example, the default format for a 360Kb diskette drive is a 360Kb diskette. This default can be changed by using some of the format options.

Diskette and Drive Types Supported by FORMAT

FORMAT supports five different drive types:

5¼" single sided single density 40 track 180Kb
5¼" double sided single density 40 track 360Kb
5¼" double sided double density 80 track 1.2Mb

3½" double sided single density 80 track 720Kb
3½" double sided double density 80 track 1.44Mb

Between these drive types, diskettes can be formatted seven ways, each of which FORMAT supports:

5¼" single sided single density 40 track 160Kb or 180 Kb
5¼" double sided single density 40 track 320Kb or 360Kb
5¼" double sided double density 80 track 1.2Mb

3½" double sided single density 80 track 720Kb
3½" double sided double density 80 track 1.44Mb

FORMAT works with the following permutations:

A 160/180Kb drive can format 160 and 180Kb diskettes.

A 320/360Kb drive can format 160, 180, 320 and 360Kb diskettes.

A 1.2Mb drive can format 1.2Mb diskettes. It can also format 160, 180, 320 and 360Kb diskettes, but you may not be able to use these diskettes in a 160/180 or 320/360Kb drive.

A 720Kb drive can format copy 720Kb diskettes.

A 1.44Mb drive can format 720Kb and 1.44Mb diskettes.

Table 8 lists the format options:

Table 8. FORMAT Command Options

| Option | Description |
|--------|--|
| /H | Help option |
| /S | Copies Concurrent onto the formatted disk. This option can't be used with 160 and 180Kb disks because of their limited capacity. |
| /V | Allows a volume name to be given to the disk. |
| /1 | Forces single sided formats on a double sided drive. (Only allowed on 5 1/4" drives.) |
| /4 | Forces single density. Use this to format single- density diskettes in a double density "hyper" drive. Note that diskettes formatted this way can't be reliably read by single density drives. |
| /8 | Formats 9 sectors but uses only 8 (only allowed on 5 1/4" drives). The default is to format 9 sectors and use all 9. |
| /T:80 | Sets tracks to 80 - used in conjunction with /N:9 |
| /T:40 | Forces double stepping on 80 track drive. (Only allowed on hyper drives.) |
| /N:9 | Forces 9 sectors per track. (Allowed on 3 1/2" 1.44Mb drives or 5 1/4" hyper drives in conjunction with /T:80 or /T:40. |
| /N:8 | Forces 8 sectors per track (on hyper drives in conjunction with /T:80). |
| /B | Formats 9 sectors but uses only 8. Reserves space for DOS system files. (Only allowed on 5 1/4" drives.) |

Examples: If you enter the command:

```
A>FORMAT
```

the following message appears:

```
Insert disk to FORMAT in DRIVE A
Press <ENTER> when ready (or <ESQ> to quit)
```

If you enter the command:

```
A>FORMAT A:/S
```

FORMAT searches the default drive for CCPM.SYS, and if found, FORMAT displays:

```
READING Concurrent
```

If **FORMAT** cannot find **CCPM.SYS**, it prompts you to insert the Concurrent disk into the default drive and the operation proceeds as before. When finished, **FORMAT** prompts

Format Another? (Y/N)

The following command shows how to format a diskette at 360Kb when the diskette is in a 1.2Mb "hyper" drive

```
A>FORMAT A:/4
```

Note that the 360Kb diskette formatted on a hyper drive cannot be reliably read in a 360Kb drive.

FUNCTION

Format: FUNCTION
FUNCTION [d:] filename.PFK

Explanation: The FUNCTION command allows you to change the commands assigned to your function keys and numeric keypad keys. FUNCTION lets you make the following assignments:

- * Assign a command up to 20 characters long to any of the function keys F1 through F10.
- * Assign a command up to four characters long to any of the key combinations formed by using the function keys in combination with the Ctrl, Alt, and Shift keys.
- * Assign a command up to four characters long to any of the dual-purpose numeric keypad keys. (Only the 5 is not dual purpose.)
- * Change the key combinations used to switch windows.

When you enter FUNCTION without a file specification, Concurrent displays a menu presenting you with options.

When you enter FUNCTION with a file specification, function assignments are made from the file specified. No menu is displayed, and no user interaction is allowed.

Function key assignments generally apply only to command-line entries. Many Concurrent commands and application programs use the function keys for special purposes; they ignore any assignments you make with the FUNCTION command. In some cases, your assignments remain in force even when a command or program is running.

The FUNCTION command allows you to reassign the function keys to perform different commands from within different programs. You can have the function keys perform one set of commands when the command-line prompt is displayed. Then, you can invoke FUNCTION from within a BATCH file that also invokes an application program, and have totally different commands in effect when the program is running. When you exit the application program, the function keys do not always return to their original assignment. You should save your default function key assignments in a file and run FUNCTION with that file after you exit the application program.

Note that to save function key assignments permanently, you should run the SETUP command after running FUNCTION.

If you have several serial terminals in your system, the following rules apply:

Concurrent XM:

- * **FUNCTION** can only be run on the main console, though you can run it on any window in the main console.
- * Programmable function keys can't be used on the serial terminals.

Concurrent 386:

- * **FUNCTION** can only be run on the main console, though you can run it on any window in the main console.
- * A function key assignment that is saved by using **SETUP** will be set for all the function keys on all the terminals when the system is restarted.
- * Serial terminals other than the main console will be able to use, but not change, the function key assignments on their keyboard.

Table 9 describes the options presented when you invoke **FUNCTION** without a file specification.

Table 9. FUNCTION Main Menu Options

| Selection | Description |
|------------------------------------|---|
| F3 -- Assign Function Keys | Select this option to display or change the current function key assignments. |
| F5 -- Save Function Keys in a File | Select this option to save new function key assignments in a file. |
| F7 -- Change Window Switching Keys | Select this option to change the key combination used to switch windows. |

To select an option from the **FUNCTION** Main Menu, press the associated function key. Another screen is then displayed. If you need help, press function key F1. After you have finished with a **FUNCTION** Menu, press Esc to return to the previous menu. Also use Esc to exit the **FUNCTION** Main Menu and return to the command-line prompt.

All assignments you make with FUNCTION take effect as soon as you exit the FUNCTION command. The assignments remain in effect until you turn off or reset your computer, or another program resets them. If you want to save your assignments, select option F5, "Save Function keys in a File" from the Main Menu before you exit FUNCTION. Invoke FUNCTION with this filename when you want to change the default function key assignments to these values.

You can also use SETUP to change the default function key assignments that take effect when you start Concurrent.

F3 -- Assign Function Keys

The Concurrent commands shown for function keys F1 through F10 when you select F3 from the Main Menu are the default commands. To change the command assigned to a particular function key, press that key. FUNCTION then highlights the number of the function key selected and erases the currently assigned command. You can now enter a new command.

To make an entry, type the command exactly as if you were entering it from the command line. Use the backspace key to edit the command as you enter it. When you are satisfied with the command, press Ctrl-@ or enter \00 to conclude the entry. You can then select another function key or exit.

If you want Concurrent to run a command when you press the corresponding function key, press Enter before terminating the command entry. If you want Concurrent to wait for an additional entry, do not press the Enter key at the end of your command. For example, if you want a function key to give you a listing of the files on the current drive, assign the following command to the key:

```
DIR<Enter>
```

If you want Concurrent to wait and let you specify which drive to list, you would make the following assignment:

```
DIR<Space>
```

Afterwards, when you press the associated function key, Concurrent waits for you to press Enter before running the command. This lets you enter a drive specification or other option.

Note: When you press the spacebar during function key assignment, FUNCTION displays \20 instead of a literal space. When you press the Enter key, Concurrent displays \0D. These are the numerical (hexadecimal) representations of the spacebar and Enter keys.

You can program function keys F1 through F10 to invoke the commands you run most often.

When you finish making assignments to the F1 through F10 function keys, press Enter to proceed to the second page of the assign-function-keys display. This screen shows the numeric keypad and Alt-function assignments. Press Enter again to proceed to the third and last page of the display, which shows Shift-function and Ctrl-function assignments.

Make assignments to any of the keys on these two pages in the same way you make assignments to function keys F1 through F10. The only difference is that keys on these pages cannot be assigned commands of more than four keystrokes.

You can scroll through the three pages of the display by pressing Enter as many times as necessary. When you are through making all your assignments, press Esc to return to the FUNCTION Main Menu.

F5 -- Save Function Keys in a File

After you have made your command assignments and returned to the FUNCTION Main Menu, press F5 to save your assignments. FUNCTION displays a new menu that presents you with two options.

One option is to press F5 again. This saves the assignments in a file with the default name KEYS.PFK (highlighted on your screen) in the current user number on the current disk.

Select F3 to change the default filename, KEYS.PFK, to a filename of your choice. The PFK file extension remains on the screen, and there is no cursor. Enter any valid filename, up to a maximum of eight characters. Press Enter to terminate a filename shorter than eight characters. After you complete your entry, FUNCTION updates the screen with the new filename. Select F5 to save your file, or select F3 again to select a different filename.

F7 -- Reset Window Switching Keys

This option lets you define the key combination used to switch windows. The entire range of combinations is shown on your screen when you press F7. The current key combination is indicated by the flashing arrow.

Notice that option F2 allows you to select a window by pressing the numeric keypad key only. All other options require the use of the numeric keypad key in combination with one or more keys.

To select an option, press the associated function key and Esc. This returns you to the FUNCTION Main Menu.

Saving the function-key command assignments in a file also saves your window switching-key selection in a file.

Using FUNCTION with an Input File

When you enter FUNCTION with a file specification, Concurrent reads the file and makes function-key assignments as specified in the file. Note that the specified file must have the extension PFK. The file must also include a line for each function key, whether or not it differs from the default.

The FUNCTION option F5, "Save Function Keys in a File," automatically creates a PFK file for you. You can also create a PFK file with your text editor or word-processing program. It is easier to create the file initially with the F5 option, and then edit it with your word-processing program. This gives you the opportunity to put comments in the PFK file to describe what the command does and why.

When you create your own PFK file, keep the following facts in mind:

- * Each function key is identified in the file by a single specific character. Only one function-key assignment is allowed per line.
- * You can include blank lines between function key assignments.
- * If a command contains fewer than the maximum number of characters allowed, you must terminate it with the characters \00.
- * If a line starts with an asterisk, FUNCTION ignores all characters between that asterisk and the next asterisk it encounters. This allows you to store comments in the file along with the key assignments.

Table 10 lists the identifying character assigned to each function key.

Table 10. Function Key Identifiers

| Function Key | Character | Function Key | Character |
|--------------|-----------|--------------|-----------|
| F1 | ; | Alt-F6 | f |
| F2 | < | Alt-F7 | g |
| F3 | = | Alt-F8 | h |
| F4 | > | Alt-F9 | i |
| F5 | ? | Alt-F10 | j |
| F6 | @ | Shift-F1 | k |
| F7 | A | Shift-F2 | l |
| F8 | B | Shift-F3 | m |
| F9 | C | Shift-F4 | n |
| F10 | D | Shift-F5 | o |
| Home | G | Shift-F6 | p |
| Up Arrow | H | Shift-F7 | q |
| Pg Up | I | Shift-F8 | r |
| Left Arrow | K | Shift-F9 | s |
| Right Arrow | M | Shift-F10 | t |
| End | O | Ctrl-F1 | u |
| Down Arrow | P | Ctrl-F2 | v |
| Pg Dn | Q | Ctrl-F3 | w |
| Ins | R | Ctrl-F4 | x |
| Del | S | Ctrl-F5 | y |
| Alt-F1 | a | Ctrl-F6 | z |
| Alt-F2 | b | Ctrl-F7 | { |
| Alt-F3 | c | Ctrl-F8 | |
| Alt-F4 | d | Ctrl-F9 | } |
| Alt-F5 | e | Ctrl-F10 | ~ |

The window switching keys can also be set in your PFK file by using the special character identifier exclamation point, !. The specific combination of numeric keypad keys and Ctrl, Alt, and Shift keys is set by a number. Table 11 lists the number-to-key-combination assignments. Use this table to construct the desired control string in the PFK file.

Table 11. Window Switching Number Assignments

| Number | Key Combination |
|--------|-----------------------------|
| 0 | Keypad only |
| 1 | Keypad + Shift |
| 3 | Keypad + Ctrl |
| 5 | Keypad + Shift + Ctrl |
| 2 | Keypad + Alt |
| 4 | Keypad + Alt + Shift |
| 7 | Keypad + Alt + Ctrl |
| 6 | Keypad + Alt + Ctrl + Shift |

Use the ASCII hexadecimal values to include a space, Enter character, or other special character. Table 12 lists the hexadecimal values for the keys listed. Be sure to precede each value with a backslash (\).

Table 12. Hexadecimal Values for Selected Keys

| Key Name | Hex Value | Key Name | Hex |
|--------------|-----------|------------|-----|
| Backslash(\) | 5C | End | CF |
| Space | 20 | Down Arrow | D0 |
| Enter | 0D | Pg Dn | D1 |
| Up Arrow | C8 | Ins | D2 |
| Pg Up | C9 | Del | D3 |
| Left Arrow | CB | Home | C7 |
| Right Arrow | CD | | |

Examples: The following shows how Function Key Identifiers are used in a PFK file. Notice the use of comments. The first three commands assign character strings to function keys F1, F2, and F3. The last command changes the window switching key combination.

```
*F1      *;HELP\OD\00
*          This command displays the main
           help screen*

*F2      *<DATE\OD\00
*          This command lets me change
           the date*

*F3      *=SHOW\20SPACE\OD\00
*          This shows the remaining space
           on all logged in drives*

*Window *!2\00
*          This changes the window-switching
           key combination to keypad + Alt.* >
```

Notice the following characteristics of the PFK file entries shown:

- * The keys F1 through F3 are identified for FUNCTION by the characters ;, <, and =. For clarity, F1, F2, and F3 are listed as comments.
- * Comments can extend to more than one line.
- * Regular numbers can be used in the command line. They are distinguished from hexadecimal digits because they do not have the preceding backslash.

HELP

Format: HELP
HELP [topic] [subtopic1 subtopic2 ... subtopic8]
HELP [option]

Explanation: The HELP command displays information about Concurrent commands. You can also create your own help screens. The information is organized in a tree-structured fashion that starts with a topic and proceeds down through as many as eight levels of subtopics. The file HELP.HLP contains all the information displayed by HELP.

Many commands within Concurrent also have their own /H (help) option. To use this option, simply type the command name followed by /H.

Two HELP options allow you to modify the HELP.HLP file. They are EXTRACT and CREATE. See "Customizing Your HELP File" below for a description of how to use them.

When you enter

```
A>HELP
```

the entire list of topics is displayed and the command-line prompt is replaced with the following prompt:

```
HELP (Enter topic. ESC to exit)
```

Select a topic by entering a name in the list. You need not enter the entire topic name, but you must enter enough letters to uniquely identify the topic.

There are two ways to display subtopic information from the HELP prompt. One way is to enter the topic and subtopic in a single command:

COPY EXAMPLES

then press Enter. Another way is to enter the topic alone, press Enter, then type a period (.) and the subtopic name before pressing Enter again. Instead of the period, you can retype the topic name and follow it with the subtopic. Subtopic names can be abbreviated in the same manner as the topic names.

Enter a ? to return to the initial list of topics after you have displayed topic or subtopic information. Alternatively, you can enter another topic and/or subtopic. Press Esc to return to the command-line prompt.

You can also display topical information directly from the Concurrent command-line prompt. For example, to show the information in the EXAMPLE subtopic in COPY, type

```
A>HELP COPY EXAMPLE
```

Customizing Your HELP Files

Concurrent has two related HELP files: HELP.CMD and HELP.HLP. The HELP.CMD file is the program file that manages the display of the information. HELP.HLP is the text file containing the information. While you cannot directly edit HELP.HLP to add new information, you can use the CREATE option to change information in HELP.HLP.

First, make sure that the current directory contains HELP.CMD, HELP.HLP, and a text editor. HELP.HLP need not have the DIR attribute but it must have the Read/Write attribute. (See the FSET command for a description of the DIR and Read/Write attributes.) To modify the HELP.HLP file, perform the following steps:

1. Create the file HELP.DAT by entering:

```
A>HELP [EXTRACT]
```

EXTRACT can be abbreviated to E, but the brackets must be entered.

2. Edit the file HELP.DAT with your editor.
3. Create a new HELP.HLP from the edited HELP.DAT file by entering:

```
A>HELP [CREATE]
```

CREATE can be abbreviated to C, but the brackets must be entered.

New topics and subtopics are added to HELP.DAT in step 2 according to specific rules. The format of a topic heading in HELP.DAT is as follows:

```
///ntopicname
```

The three slashes designate that the following topicname is a topic. The n indicates the topic level. Valid numbers are 1 to 9. Topics with a 1 value appear in the HELP list of topics. The remainder are shown as subtopics in the previous level topic's display.

The topicname is any combination of 12 letters and numbers, including spaces. Lowercase letters are translated to uppercase in the topicname.

Under the topic identifier, enter your information. What you enter is what you see when you select the topic from HELP. Be sure to end each line by pressing the Enter key. If the information is more than one screen long, HELP stops and waits for you to enter any character before displaying the next screen. Lowercase letters are not translated to uppercase in the description.

A subtopic must follow immediately after its topic. The rules for specifying the subtopic name are the same as for a topic. To make your HELP file additions easy to use, place new topics in alphabetical order. HELP displays the topics in the initial list in their order of appearance in the HELP.HLP file.

Example:

```
///1NEWAPP  
///2COMMANDS  
///3EASY CMDS  
///3DIFF CMDS  
///2EXAMPLES
```

This example shows a possible arrangement of topics and subtopics for the program NEWAPP. There are two second-level subtopics and two third-level subtopics. Notice the use of spaces in the third-level subtopics. Also notice the sequence of second- and third-level subtopics. This arrangement indicates that EASY CMDS and DIFF CMDS are subtopics of COMMANDS and not EXAMPLES.

LABEL

Format: LABEL [d:] [Label]

Explanation: The LABEL command allows you to create, modify or delete the label on a DOS disk. If no drive is specified the default drive is used. You will be prompted to type a label if you don't specify one at the command line.

Example: A>LABEL b:maggie's

This assigns the name "MAGGIE'S" to the disk in drive B.

LIMSIZE

Format: LIMSIZE [nnnn]

Explanation: LIMSIZE limits the amount of Expanded Memory available to an application, and can be used with either Concurrent DOS 386 or Concurrent DOS XM. (To limit the size of conventional memory, see the MEMSIZE command.)

Each Concurrent window has a default LIMSIZE value of 1008Kb, which should be suitable for most purposes. You will probably only want to alter LIMSIZE if you want to:

- * Make room on one window for a program with unusually large memory requirements.
- * Decrease the available memory space on other windows so that one window can have a very large memory allocation.

Typing LIMSIZE without an associated number shows the current LIMSIZE on the current window. Note that all LIMSIZE values are rounded up to the nearest 16Kb.

Examples: A>LIMSIZE 1990

This sets the LIMSIZE on the current window to 2000Kb.

LOADSYS

Format: LOADSYS
LOADSYS ASK
LOADSYS INSTALL

Explanation: Use the LOADSYS command to start Concurrent if DOS has control when you first turn on your computer. You invoke LOADSYS under DOS, and when you use LOADSYS, you do not need to change existing DOS system files. If you use the form

LOADSYS ASK

you are asked if you want to load Concurrent. Enter Y to load Concurrent; enter N to retain DOS.

Examples: A>LOADSYS

This loads Concurrent if DOS was running when the command was typed.

The following command line, in a batch file such as AUTOEXEC.BAT, tests for the DOS environment and then asks you if you want to load Concurrent:

```
IF "%os%"= "" LOADSYS ASK
```

Installing LOADSYS

To install LOADSYS you need to type

```
C>LOADSYS INSTALL
```

Note that you must install LOADSYS under Concurrent; once installed, LOADSYS works under DOS.

MEMSIZE

Format: MEMSIZE [nnnn]

Explanation: MEMSIZE limits the amount of conventional memory available to an application. (To limit the size of expanded (EMS) memory, see the LIMSIZE command.)

Each Concurrent window has a default MEMSIZE set at the maximum possible MEMSIZE, around 512Kb. This provides sufficient room for applications to work and, since each program only takes up the amount of memory it needs, several programs can run concurrently on different windows. However, a few programs are designed to occupy all the memory that they can find. Such a program will occupy all 512Kb of available conventional memory and prevent programs working in other windows. You prevent a program from occupying all available memory by specifying a lower MEMSIZE value for its window.

Typing MEMSIZE without an associated number shows the current MEMSIZE on the current window. The default value can be set from the Maximum Memory per Process option in the SETUP command.

Example: A>MEMSIZE 300

This sets the MEMSIZE on the current window to 300Kb.

MKDIR/MD

Format: MKDIR [d:]dirpath
MD [d:]dirpath

Explanation: Use MKDIR to create subdirectories in the root directory or other subdirectories. You can abbreviate MKDIR as MD in your command line.

If you do not include a backslash (\) before the first subdirectory name in dirpath, MKDIR begins the path to your new directory from the current directory of the current or specified drive. If you do include the backslash, MKDIR begins the path at the root directory of the current or specified drive. For example, if you are in subdirectory LEVEL2 and want to create subdirectory NEWONE from the root directory, enter the following:

```
A>MKDIR\NEWONE
```

If you are in subdirectory LEVEL2 and want to create a subdirectory from LEVEL2, enter the following:

```
A>MKDIR NEWONE
```

The path from the current directory leading to the new subdirectory must be entered and all the subdirectories in the path must exist. If a subdirectory is specified that does not exist, MKDIR displays the following message and terminates:

```
Path Not Found
```

Example: Use subdirectories to organize your data and command files into logical groups. The sequence of MKDIR commands shown below create the hierarchical series of subdirectories shown in Figure 1.

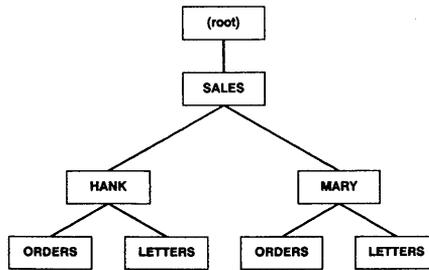


Figure 1. Sample Hierarchical Subdirectories

```
C>MKDIR SALES  
C>MKDIR\SALES\HANK  
C>MKDIR\SALES\MARY  
C>MKDIR\SALES\MARY\ORDERS  
C>MKDIR\SALES\MARY\LETTERS  
C>MKDIR\SALES\HANK\ORDERS  
C>MKDIR\SALES\HANK\LETTERS
```

You can have subdirectories with the same specification, but they must have different paths.

MORE

Format: MORE

Explanation: The MORE command takes input from a standard input device and then display it a screenful at a time. At the bottom of each screen (except the last) is the phrase.

Strike a Key when ready...

You can move to the next screen by pressing any key.

Example: The following example shows MORE used with DIR and illustrates how to display a large directory a screenful at a time. The output from DIR is piped to MORE and then displayed.

```
A>DIR c: | MORE
```

PASSWORD

Format: PASSWORD [/H] filespec [[/R][/W][/D][/P][/G]:password][/N][/S]

Explanation: The PASSWORD command is used to assign passwords to files or paths. Passwords can be up to eight letters long.

The options are:

/R:password The password will be required for reading, copying, writing, deleting or renaming the file.

/W:password You are allowed to read the file without giving a password but a password will be required to copy, delete or rename the file.

/D:password The password will be required only for deleting or renaming the file. You don't need a password to read or modify the file.

/P:password The password operates on paths instead of files. This means that each time you try to use the path you will be prompted for a password. The password levels R, W and D can't be assigned to a path.

/G:password Set global default password. Concurrent remembers this default password and tries it against each password-protected file that you then try to access. This is useful if a group of files have the same password and you want to access the files without typing a password each time. The global default password remains set until it is switched off or until the computer is restarted.

/N: This option removes any password protection. It can also be used in combination with the /P and /G options:

/NP removes password protection from a directory

/NG removes the global default password

/S The password operates on files and subdirectories associated with the current directory.

Example: The following example assigns the password SUZE to the file eyesoff.let, and ensures that the password must be given before the file can even be read.

```
A>PASSWORD eyesoff.let /R:SUZE
```

The example below protects all the files the .BAS extension in the \handsoff directory and in all the associated subdirectories. The level of protection is set at W; the files can be read but not copied, renamed, deleted or altered. Note that you don't need to use the password to access the \handsoff directory or the subdirectories; you only need the password with *.BAS files.

```
A>PASSWORD c:\handsoff\*.BAS /W:SIMON /S
```

The following example assigns the password WINSTON to the path c:\secret\top:

```
A>PASSWORD c:\secret\top /P:WINSTON
```

If you then type, for example:

```
A>CHDIR c:\secret\top;winston
```

you will be able to access the files associated with the path \secret\top without using the password again (so long as the files aren't themselves password-protected).

PATH

Format: PATH [d:]dirpath [;[d:]dirpath....]

Explanation: PATH sets a search path for commands or batch files not found by a search of the current directory. If a command cannot be found in the current directory, Concurrent searches the named directories in the order you entered them, but does not change the current directory.

PATH with no parameters displays the names indicated in the previous PATH command. (The search paths currently defined.) PATH with a semicolon (PATH;) resets the search path to null (no extended search path.) The default search path is the root directory of the drive Concurrent was started from.

In the following example, PROGA.COM is contained in directory PLANDIR on Drive B, and the default drive is Drive A:

```
PATH A:\SUB1;A:\SUB1\SUB2;B:\PLANDIR
```

This command sets the search path so that when you type PROGA at the command line Concurrent will look into the current directory of the drive specified, followed by A:\SUB1, then A:\SUB1\SUB2, then B:\PLANDIR, until it finds the command you entered. If the command entered is not found in any of the directories specified in PATH, the message:

```
Bad command or filename
```

is displayed.

If you enter the command:

```
PROGA
```

Concurrent searches four directories and finds program PROGA in B: PLANDIR.

Incorrect information in the paths will not be found until the specified paths are searched. If a specified path does not exist, Concurrent will ignore that path and go to the next.

PIFED

Format: PIFED [d] [path] filename [.exe|.com] [/T]

Explanation: PIFED allows you to create or modify the Concurrent program information embedded in a DOS .EXE or .COM file. This program information allows Concurrent, for example, to allocate the correct amount of memory for the program, to prevent corruption of interrupt vectors and to stop other processes from accessing devices manipulated directly by a DOS process.

The option /T causes PIFED to work in TTY (teletype) mode, which is useful for running PIFED on very simple serial terminals.

When you run you run PIFED for a particular program, you will see a screen displaying the current Program Information settings. A message at the top of the screen will tell you whether program information has already been installed, or whether Concurrent is using default values. The default values will usually enable your .EXE or .COM file to run successfully, but you can often improve the performance of programs by changing the information settings.

The headings on the screen are described below:

Program name

This is simple a reminder of the name of the program whose program information you are altering.

Memory usage

The memory usage defaults will probably not need to be altered. However, you may find that increasing the 'minimum memory required' enables the program to run more quickly. Similarly, if you know a program always allocates for itself more memory than it needs, you can reduce the 'maximum memory desired' and free more memory for other programs to use.

Screen usage

PIFED allows you to specify one of six different types of screen usage for your program. Refer to your program's documentation to decide which type of screen usage best suits your program. You can toggle through the available options when the cursor is at the 'screen usage' heading by either pressing the spacebar or typing the option number (0 to 5). The six types of screen usage are as follows:

- * Program has no special requirements (24 or 25 lines). Use this for programs that only write simple messages to the screen. You can choose for the program to use 24 lines (option 0) or 25 lines (option 1). Choosing the 24-line option allows the Concurrent status line to be displayed while the program is running.
- * Program uses ANSI escape sequences (24 or 25 lines). All programs that require the ANSI.SYS driver require at least this screen usage type, and may require the more complex usage types described below.
- * Program makes ROS calls (25 lines). Select this option for programs that make calls to the ROM operating system – again, refer to the program's documentation for this information.
- * Program directly writes to screen, makes ROS calls and uses ANSI escape sequences. This option, the default, is used for programs that write directly to screen. The program's documentation should inform you whether or not the program writes directly to screen – most do not.

Options

The other PIFED options are as follows:

- * Maths Co-processor option: This allows you to specify whether the program needs a maths co-processor. Since most programs don't need a maths co-processor, the default is 'off.'
- * 'Program runs only in the foreground' option: This allows you to specify explicitly that a program runs only in the foreground. Note that if you specify 'no' here, your program may still be suspended when it is put in the background because of the option selected at the 'screen usage' heading. (Note: this PIFED option overrides the SUSPEND command.)
- * 'Program may run in banked memory' option: The default here is 'Yes'. If you specify 'No', Concurrent cannot page the program in and out of conventional memory. (Note: this PIFED option overrides the BANK command.)

- * 'Program waits in idle loop' option: Some programs execute loops while they are idle and waiting for user input. This can be wasteful on computer processing time. Concurrent has the ability to look for a program using 'idle loops', and reduce the slice of time that it allocates to that program. However, other programs execute 'idle loops' while they are also doing something else (such as recalculating a spreadsheet), and you do not want Concurrent to look for idle loops in this case. If you are sure the program only uses idle loops when waiting for further user input, you can set this option to 'Yes'. Otherwise leave it at 'No'.
- * 'Program makes direct access to...': Some programs directly access the ports COM1, COM2, LPT1, LPT2 and LPT3. This can cause a problem if another Concurrent window is using the port. Setting the switch to 'Yes' for the specified port will cause Concurrent to check whether or not the port is being used when the program is started up. If the port isn't already being used, Concurrent will reserve the port for the program that directly accesses it. If it is being used, an error message will be issued when the program starts.
- * 'Program uses interrupt vectors...': The default values for interrupt vectors enable most programs to run under Concurrent. Increasing the range of interrupt vectors may enable you to run certain complex programs that write directly to the interrupt vectors. Decreasing the range of interrupt vectors for a program that doesn't write directly to the interrupt vectors may marginally improve the program's performance.

Leaving PIFED

Press the ENTER key to leave PIFED. You can then choose:

- * I to install the changes. These will now be the program defaults until PIFED is used on the program again.
- * Q to leave (quit) PIFED without saving the changes.
- * C to continue using PIFED with the current values.
- * R to continue using PIFED with the last saved values.

PRINTER

Format: PRINTER
PRINTER n

Explanation: The PRINTER command displays or sets the printer assigned to your current window. Each window can have a different printer assignment or all windows can share access to a single printer. When one window is using the printer, no other window can access it. When Concurrent is started, all windows are assigned to printer 0.

The LST: and PRN: device names used by COPY always refer to the current printer.

Table 13 shows the printers and their number assignments for IBM PCs and compatibles

Table 13. Printers and Number Assignments

| Number | Description |
|--------|--|
| 0 | First parallel interface printer and the Concurrent default printer assignment |
| 1 | Second parallel interface printer |
| 2 | Third parallel interface printer |
| 3 | First serial interface printer |
| 4 | Second serial interface printer |

The printer numbers shown are the only valid numbers for the IBM PC computer family (unless you have added further serial ports with an expansion card). Refer to your printer and computer manuals to determine whether your printer is parallel or serial.

Examples: When you enter PRINTER alone, Concurrent displays your window's current printer number.

```
A>PRINTER
Printer Number = 0
```

When you enter PRINTER followed by a number, Concurrent changes your window's current printer and displays the new assignment.

```
A>PRINTER 3
Printer Number = 3
```

PRINTMGR

Format: PRINTMGR command
 PRINTMGR filespec filespec ... [option,option...]

Explanation: PRINTMGR is Concurrent's print spooler. Use PRINTMGR when you want to print two or more files at the same time, when you want to use a window to run another program while you are printing, or when you want to print files from several windows simultaneously.

Note: Concurrent also includes a menu-driven form of PRINTMGR called the Printer Manager. See Section 6 of the User's Guide.

You cannot print more than one file at a time on the same printer. PRINTMGR lets you specify multiple files, so that you do not have to wait for one to finish printing before specifying the next. PRINTMGR numbers each file specification you enter and adds it to a list called the job queue. It then prints the files in the order they are specified. While the files are printing, you can run programs from any window or leave the computer.

Table 14 lists and summarizes the PRINTMGR commands. The PRINT command specifies files to be printed; the other commands control PRINTMGR operation. Some have abbreviated forms.

Table 14. PRINTMGR Commands

| Command | Description |
|-------------|---|
| DEL[ETE] n | Deletes entry n from the job queue. |
| H[ELP] | Displays summary of PRINTMGR commands and options. |
| PRINT | Prints the file or files specified with the options selected. |
| RES[ET] | Resets PRINTMGR system. |
| STAR[T] n | Starts PRINTMGR for use with printer n. |
| STAT[US] | Displays PRINTMGR status. |
| TER[MINATE] | Temporarily stops PRINTMGR. |

START Command

Before you can use PRINTMGR, you must START it. You must START PRINTMGR after any of the following events:

- * Concurrent is started.
- * Concurrent is reset.
- * PRINTMGR is stopped.

If you attempt to use PRINTMGR without first STARTing it, an error message appears. Once you start PRINTMGR, you can use it from all windows.

The START command turns PRINTMGR on and designates which printers are to be used. Printers are designated by the numbers 0, 1, 2, 3, and 4.

You must list all printers you plan to use in a single start command because PRINTMGR can access only those printers that have been specified; for example:

```
A>PRINTMGR START 0 3 4
```

Printer numbers must be separated by a space; do not use a comma.

When you START PRINTMGR, a special job queue file is created to record the contents of the job queue. Every time PRINTMGR is STARTed, it checks this file to see if any jobs remain to be printed. A job is removed from the file only when completed. Thus, there is a permanent record of PRINTMGR's status in case a computer failure or operator error interrupts its operation. When PRINTMGR is STARTed, it begins printing from the beginning of the interrupted job.

Note: When you start PRINTMGR, two programs are invoked: SPL and DSP. If you specify more than one printer, a separate DSP program is invoked for each printer. Do not terminate these programs with the STOP command.

PRINT Command

Files are printed using the PRINTMGR PRINT command. PRINT is assumed if no other command is specified. Thus, the following commands are equivalent:

```
A>PRINTMGR MYFILE  
C>PRINTMGR PRINT MYFILE
```

You can specify more than one file in a single command line. File specifications must be separated by spaces. You can also use the * and ? wildcard characters in your file specifications. For example, the following command prints all files on the current disk with the DAT and TXT file extensions:

```
A>PRINTMGR *.DAT *.TXT
```

All files specified and any matching the wildcard specification are given numbers and added to the job queue.

Several options are available with the PRINT command to control the format of the printout. Enter options in brackets following the file specifications. Table 15 lists and summarizes the options.

Table 15. PRINTMGR Options

| Option | Short | Defaults | Description |
|------------|--------|----------|---|
| COPIES=n | COP=n | 1 | Print n copies of each file in the command line. |
| FORMAT | FOR | OFF | When on, print the file contents with 55 lines per page (page equals 66 lines). |
| FORMSIZE=n | FORM=n | 66 | Set paper length at n lines. |
| MARGIN=n | MAR=n | 8 | Set left margin at n spaces. |
| NOFEED | NOF | OFF | When on, do not send a formfeed to the printer between files. |
| NUMBER | NUM | OFF | When on, print file contents with 55 lines per page and number each page. |
| PAGESIZE=n | PAGE=n | 60 | Print n lines of file per page. |
| PRINTER=n | PRI=n | * | Send output to printer n. |
| TABS=n | TAB=n | 8 | Set a tab at every nth column. |

*Default n=current printer

DELETE Command

The PRINTMGR DELETE (or DEL) command erases a specified job from the job queue. It takes the form:

```
PRINTMGR DEL n
```

Here, n is the job number of the file to be deleted from the queue. Use the PRINTMGR STATUS command to display the job numbers.

DELETE can also be used to terminate a job that is already printing. In this case, the PRINTMGR immediately terminates the job and continues to the next one in the job queue. (Printers that have internal buffers continue to print the terminated job until the buffer is empty. This can give the appearance that PRINTMGR has not terminated the job.)

HELP Command

Use the PRINTMGR HELP (or H) command to display a summary of the PRINTMGR print options and commands. Use the form:

```
PRINTMGR H
```

When you enter this command, a two-screen display appears. The first screen lists the PRINT options, their default values, and describes what effect they have. The display remains on the screen until you press any key. The second screen lists and summarizes the PRINTMGR commands.

STATUS Command

The PRINTMGR STATUS (or STAT) command displays the current contents of the job queue including, for each job, the job number, filename and file extension, destination printer, how many copies remain to be printed, the size of the file, and whether the job is printing or waiting. Use the form:

```
PRINTMGR STAT
```

The PRINTMGR TERMINATE (or TERM) command turns off PRINTMGR. Use the form:

```
PRINTMGR TERM
```

Use TERMINATE to stop the printer or PRINTMGR so you can add another printer. TERMINATE differs from RESET in that it saves the contents of the job queue. All jobs in the queue, including those being printed, are completed when PRINTMGR is STARTed again.

RESET Command

The PRINTMGR RESET (RES) command terminates the current job and erases the job queue file. Use the form:

```
PRINTMGR RES
```

Normally, use TERMINATE or turn off your computer to stop PRINTMGR. RESET is provided for extreme situations in which a computer or PRINTMGR failure prevents PRINTMGR from working properly. You can also use RESET to remove all jobs from the job queue.

If the printer is not working, confirm that the cable between the computer and the printer is in place before using the RESET command. If the printer still does not work, confirm that you have selected a printer that actually exists. If PRINTMGR still does not work, enter the PRINTMGR RESET command and then reissue the START command.

PROMPT

Format: PROMPT prompt string

Explanation: The PROMPT command modifies the command prompt. The prompt string can contain special meta-strings that are in the form \$c. The following are meta-strings that can be included in the prompt string.

Where c is one of the following:

- \$ The "\$" character.
- t The time.
- d The date.
- p The current directory of the default drive.
- v The version number.
- n The default drive letter.
- g The ">" character.
- l The "<" character.
- b The "|" character.
- q The "=" character.
- h A backspace; the previous character is erased.
- e The ESCape character.
- _ The CR LF sequence (go to beginning of new line of the display screen).

Any other value for c is treated as a null character and is ignored by PROMPT.

Type PROMPT with no parameters to reset the prompt to the default DOS prompt. For example, the default prompt (if the user has not enabled a prompt) is \$n\$g which will display the standard DOS prompt, A>. The prompt indicates the default drive letter A, and the character >.

Examples: The following example sets the DOS prompt to "HELLO".

```
A>prompt hello
hello
```

The following example sets the DOS prompt to the default drive and path plus the character >.

```
A>prompt $p$g
A:\wordproc\letters>
```

The following example sets the DOS prompt to display the time and date:

```
A>prompt time = %t%_date = %d
```

The DOS prompt would appear as follows:

TIME = (current time)

DATE = (current date)

REBOOT

Format: REBOOT

Explanation: The REBOOT command is a synonym for <Ctrl><Alt>, and performs a system reset.

Note: If you have a multi-user system you may like to password protect this command (use the PASSWORD command), so that individual users can't reset the whole system without your permission.

RENAME/REN

Format: REN[AME] [dirpath]oldfilespec [dirpath]newfilespec

Explanation: The RENAME command lets you change the name of a file. If the file to be renamed is not on the current drive, you must precede the old file specification with the correct drive/path specification. The following command line renames MYFILE.TYP as FILE1.TXT:

```
A>REN B:MYFILE.TYP FILE1.TXT
```

On password-protected files, you must enter the default password before renaming a file. The same file password is then required to access the new file specification. You can remove the password protection on the file before renaming it. (See the PASSWORD command.)

The wildcard characters * and ? can be used in file specifications as follows:

```
B>REN *.IN *.OUT
```

The above command renames all files with the file extension IN, so that they have the file extension OUT.

REN does not let you rename a file to an existing name or rename a Read/Only file. If you attempt to do either, the message

```
Access Denied
```

is displayed and the command-line prompt is returned. Any files designated to be renamed remain unchanged.

Using RENAME to Move Files

REN can also be used to move files between subdirectories on the same disk. For example the command

```
A>REN MYFILE.LST \wordproc\memos
```

will move MYFILE.LST to the directory A:\wordproc\memos. MYFILE.LST will no longer exist on the original directory.

REPLACE

Format: REPLACE [/H] filespec [filespec] [options]

Explanation: REPLACE copies selected files from the source to the destination. For example, you can select to copy only those files that don't already exist on the destination, or only copy files that have been changed since the last copy.

The first file specification on the command line is the drive, path and name of the source files; the second filespec shows the drive and path of the destination.

The options are:

- /H Displays help information. (Make sure no filespec is entered when you use the help option.)
- /A Only copy files that don't already exist on the destination.
- /H Don't ignore files with hidden or system attributes.
- /M Merge changed files on source with unchanged files on destination.
- /P Prompt before copying each file.
- /R Overwrite read-only files on the destination with fresh copies from the source.
- /S Look also in the subdirectories of the destination for files that match the filespec.
- /W Wait for disks to be changed before starting to look for the source files, and wait also before starting to copy them to the destination.

Example: The REPLACE option is especially useful for, for example, updates of software. Imagine you have a program called WCOUNTER.BAS in various subdirectories on your hard disk. Suppose further you make improvements to the copy of WCOUNTER.BAS in the \wordproc directory. You can copy the improved version of the file to all the other places in the hard disk by using the following command:

```
C>REPLACE c:\wordproc\WCOUNTER.BAS c:\ /s
```

RMDIR/RD

Format: RMDIR [d:]dirpath
RD [d:]dirpath

Explanation: The RMDIR command removes a specified subdirectory. The subdirectory to be removed cannot contain any files or subdirectories, except for the (.) and (..) subdirectories. RMDIR can be abbreviated as RD in your command line.

To remove a subdirectory, the following conditions must be satisfied:

- * The subdirectory cannot contain any files, or additional subdirectories.
- * The subdirectory cannot be a current directory on any drive on any window.
- * The subdirectory cannot be assigned to floating drives on any window.

Examples: A>RMDIR SUB1\SUB2\SUB3

This command erases subdirectory SUB3 on drive A.

A>RMDIR C:GL.DAT\ACCTS.RCV

This command erases subdirectory ACCTS.RCV on drive C.

RUNMENU

Format: RUNMENU
 RUNMENU filespec
 RUNMENU filespec menuname

Explanation: RUNMENU, COPYMENU, and EDITMENU allow you to create and run menus for your applications. See Section 7 of the User's Guide for detailed instructions on the use of these commands.

The RUNMENU command invokes a menu from a menu file. Replace filespec in the command line with the name of the menu file you want to use. Replace menuname with the name of the specific menu you want within that menu file. RUNMENU then runs the specified menu.

If you do not specify a menu file, the file MENU.DAT is assumed. If you do not specify a menuname, the menu MAIN is assumed. The menu file you specify must be in your current directory.

SCEPTER

Format: SCEPTER

Explanation: The SCEPTER command initializes the banked memory management routines within Concurrent. These routines enable Concurrent to support the Enhanced Expanded Memory Specification (EEMS) memory provided by a Quadram^(R) QuadEMS+TM board and AST RAMpageTM board.

SCEPTER is only used with Concurrent DOS XM. It isn't supplied or needed with Concurrent DOS 386.

You must run SCEPTER before using the QuadEMS+ board to supply banked memory. If you use the EEMS board to supply conventional memory only, you do not need to run SCEPTER.

Do not run SCEPTER while any other program is running. SCEPTER does not run under DOS.

See also the the LIMSIZE command for how to limit the memory available to an application.

SCEPTER Options

SCEPTER supports two command line options:

```
SCEPTER /M=hh  
SCEPTER /S=hhhh
```

where h is a hexadecimal digit.

The /M option supports applications written for the Expanded Memory Specification (EMS) also called the LotusTM/IntelTM/MicrosoftTM (LIM) specification. The /M option specifies how many available pages of memory are reported to the application. It is useful for preventing one application from using too much memory, thereby preventing other applications on other windows from having enough memory to run.

If you run SCEPTER without this option, the default is a maximum of 20H pages (512KB). You can specify a 16KB page count as two hexadecimal digits. For example, the command:

```
SCEPTER /M=18
```

specifies 18H pages, or 384KB.

The /S option specifies the address where SCEPTER begins converting "conventional" memory to "banked" memory. The default is 4000H (256KB). You can specify a paragraph address as four hexadecimal digits to start bankable memory windows at a different address.

SET

Format: SET [name=[parameter]]

Explanation: SET inserts strings into the command processor's environment. All the strings in the environment are available to all commands and applications in the current window.

The string (beginning with name) is inserted into a portion of memory reserved for environment strings. Lowercase letters in the name are converted to uppercase when added to the environment. If the name was in the environment, it is replaced by the new parameter.

If you type SET with no options you will see a list of the strings already included in the environment. Among those set automatically by Concurrent when Concurrent is started are:

```
COMSPEC=C:\CDOS.COM
```

and

```
OS=CPCDOS
```

COMSPEC defines the path to the command processor. (The command processor in Concurrent is called CDOS.COM.) COMSPEC is used by programs that on occasions look for the command processor - typically, programs that allow you to access operating system commands without leaving the program. CDOS.COM is shipped with Concurrent DOS; you can, however, insert your own command processor instead by setting a new COMSPEC.

OS describes the operating system type. This environment string is useful for distinguishing between operating systems. For example, if you run two or more different operating systems at different times on one computer you can devise a batch file that looks for the OS environment string, finds which operating system is present and then takes appropriate action.

If you specify SET with a name with no parameter, the current occurrence of name=parameter is removed from the environment.

Use SET only if you want to set values to be used by your programs. For example, SET TOT=VT86 sets your TOT value to VT86 until you change it with another SET command.

Note that the amount of memory set aside for the environment strings is fixed at system start time. You can change the amount available by putting an ENV_SIZE= command in CCONFIG.SYS.

Examples: The following command adds string CAL=abc to the other strings in the environment. The lowercase "cal" is converted to uppercase. With SET, you can enter keywords and parameters not meaningful to Concurrent, but can be found and interpreted by applications that examine the environment.

```
SET ca1=abc
```

The next command adds the string WXYZ=\FILES2 to the environment. An application program can search the environment for the name WXYZ, and use the supplied parameter as the directory name to use for its files.

```
SET WXYZ= FILES2
```

The following command removes WXYZ=FILES2 from the environment:

```
SET WXYZ=
```

SETPORT

Format: SETPORT
 SETPORT parameter . . .

Explanation: Use SETPORT to configure the serial ports COM1 and COM2 without having to restart your system.

Serial ports are used to connect the computer to a serial printer or plotter or to another keyboard and screen, making Concurrent a true multi-user system.

Concurrent provides two commands for configuring serial ports. The SETUP utility is used for initially configuring your system. SETPORT on the other hand is only used for COM1 and COM2 but it allows you to reconfigure these ports without having to restart Concurrent. (And when you do restart Concurrent the values return to those set by the SETUP command.)

Note: Concurrent XM supports up to 5 serial ports (COM1 and four others) and Concurrent 386 supports up to 9 serial ports (COM1 and eight others). You can add the extra serial ports by fitting an expansion card to your computer. COM2 is disabled when you fit an expansion card.

Before you run SETPORT, refer to the manuals provided with your serial devices to find out what configuration values you need. If you have no serial ports, and you invoke SETPORT, the following message is displayed:

```
No serial communications options installed.
```

Note: If you have fitted a multiport expansion card, COM2 is disabled and SETPORT adjusts its menus so that only the options for COM1 are displayed: The same applies if your system only has one serial port.

SETPORT has two modes of operation: menu-driven and command-line. When you enter SETPORT alone, you see the SETPORT Main Menu with the options shown in Table 16. You can also enter SETPORT and specify the parameters you want to change directly from the command line. Parameters not specified remain unchanged. The second form is provided so that you can configure the ports from a BATCH file.

Table 16. SETPORT Options

| Option | Description |
|--------------------------|---|
| F3 -- Port Configuration | <p>Use this option to display and change the port characteristics for both serial ports. The definable characteristics are as follows:</p> <ul style="list-style-type: none"> * baud rate: number of bits of input or output per second * word length: number of bits in each word * parity: the use of odd, even, or no parity * stop bits: number of stop bits added to each word |
| F5 -- Port 0 Protocol | Use this option to set port 0's input and output communication protocol. |
| F7 -- Port 1 Protocol | Use this option to set port 1's input and output communication protocol. |

The changes you enter are implemented immediately after you exit SETPORT. These values remain in effect until you turn off or reset your computer. To record these values permanently as Concurrent's defaults, run the SETUP command afterwards (or instead).

Menu-driven Mode

The first SETPORT screen is the Main Menu and lists the options shown in Table 16. Press F1 if you need help. Press the Esc key to exit SETPORT.

From the SETPORT Main Menu, press F3 to configure either or both serial ports. A new menu is displayed listing the current baud rate, word length, parity, and stop bits for each port as well as the function keys assigned to change those values.

Function keys F3 and F4 set the baud rate for serial ports 0 and 1 respectively. The baud rate is the speed at which the computer sends data to and receives data from external devices. Pressing F3 or F4 repeatedly increases the baud rate incrementally as follows:

110 150 300 600 1200 2400 4800 9600

The baud rate returns to 110 after 9600.

Word length is the number of data bits in each character transmitted. Word length is usually 7 or 8, but you can select as few as 5. Press F5 and F6 to select the word length for COM1 and COM2 respectively.

Parity checking verifies that a character was transmitted correctly. You can select odd, even, or no parity checking by pressing function keys F7 and F8.

One or two stop bits are always added to every byte transmitted. Press F9 or F10 to select 1 or 2 stop bits, respectively.

Press Esc to return to the SETPORT Main Menu. From the Main Menu, press F5 and F7 to define the communication protocol for ports 0 and 1, respectively. The Main Menu is then replaced by a screen listing the different input and output options available. The screen also shows whether the options are ON or OFF. To change a setting, press the corresponding function key.

Command-line Mode

All serial port parameters can be defined from the command line. Table 17 lists the valid command-line values for each parameter. These values must be separated by spaces. They can be entered in any order. If you do not want to change a particular parameter, omit its value from the command line.

Note that P3 designates COM1 and P4 designates COM2.

Table 17. SETPORT Parameters

| Port Number | Baud Rate | Word Length | Parity | Stop Bits | Input Protocol | Output Protocol |
|-------------|-----------|-------------|--------|-----------|----------------|-----------------|
| P3 | 110 | 5 | ODD | 1 | IDRX | ODRX |
| P4 | 150 | 6 | EVEN | 2 | IDRN | ODRN |
| | 300 | 7 | NONE | | IDNX | ODNX |
| | 600 | 9 | | | IDNN | ODNN |
| | 1200 | | | | INRX | ONRX |
| | 2400 | | | | INRN | ONRN |
| | 4800 | | | | INNX | ONNX |
| 9600 | | | INNN | ONNN | | |

The protocol's options for the input and output side of each port must be entered separately. When you set the protocol, use letters in your command-line entry to identify the options. The letters are defined as follows:

- I -- selects the input side of the port
- O -- selects the output side of the port
- D -- turns ON the DSR/DTR option
- R -- turns ON the RTS/CTS option
- X -- turns ON the XON/XOFF option
- N -- used in the position of any other option; turns that option OFF

The letters must be entered in the order shown in Table 17 above. Note that the protocol must begin with either I or O, followed by the options, then the opposite letter, followed by the options. If you want to turn a particular option OFF, enter N instead of the letter. You must make an entry for each input or output option.

If you make an invalid entry in a SETPORT command line, an error information screen appears. Use this information to identify the inappropriate entry and review your options. The command-line prompt is returned at the end of the error message so that you can enter another SETPORT command.

Examples: A>SETPORT P3 9600 7 EVEN 1 IDNN ODNN

This command sets COM1 (printer 3) to 9600 bits per second, with a 7-bit word length, even parity, and 1 stop bit. The protocol selected for the input and output sides of COM1 is DSR/DTR.

A>SETPORT INNX ODNX 110 P4

This command sets COM2 (printer 4) input side for the XON/XOFF protocol, the output side for the DSR/DTR and XON/OFF protocols, and the baud rate at 110.

SETUP

Format: SETUP
SETUP [option]

Explanation: SETUP is a menu-driven program that lets you change certain system characteristics and save your changes as Concurrent's startup values. SETUP modifies the file CCPM.SYS, which contains Concurrent and is read every time you start or reset your computer. To save changes you make with SETUP, CCPM.SYS must be available and set to Read/Write.

To run SETUP from the command line, type SETUP and press Enter. The SETUP Main Menu appears with the options listed in Table 18.

Table 18. SETUP Main Menu Options

| Option | Description |
|-----------------------------------|--|
| F1 -- Help | |
| F2 -- Reserve System Space | Controls the amount of memory in the system space reserved for use by DOS compatible device drivers. |
| F3 -- Save System Parameters | Saves the current window's function key assignments, and/on the current serial port configuration, to use as the new startup values. |
| F4 -- MDisk | Makes an MDISK from a portion of memory. |
| F5 -- Diskette Drive Parameters | Provides for customization options to support non-standard diskette drives. |
| F6 -- Verify After Diskette Write | Enables or disables the Concurrent Disk-Write verification feature. |

Table 18. (Cont'd)

| Option | Description |
|---------------------------------|--|
| F7 -- Setup Serial Console | Enables you to set up each serial port, including ports you have added with an expansion card. |
| F8 -- Menu Two | Provides menu choices for: <ul style="list-style-type: none">* Setting the Maximum Memory per Process (that is, the default MEMSIZE)* Clock Calibration* Selecting a Hard Disk Driver (for certain machines only)* Setting Expanded Memory parameters |
| F9 -- Color Monitor Scroll Mode | Allows you to change the speed and appearance of scrolling on a color monitor. |
| F10 -- Update and Exit | Records your changes in the CCPM.SYS file. |

Once you select an option, the Main Menu is replaced by another menu for the chosen option. Press Esc to return to the Main Menu.

Changes you make using SETUP become system defaults only when you exit the Main Menu by selecting F10, "Update and Exit," and then restart Concurrent. If you want to exit SETUP without saving any changes, exit the Main Menu by pressing Esc.

You can bypass the Main Menu by entering an option as part of the command line. When you do this, you go directly to the menu for that option. Table 19 lists and explains the available options. Note that a slash separates an option from the SETUP command.

Table 19. SETUP Options

| Option | Definition |
|---------------|--|
| /PARAM | Specifies the Save System Parameters Menu |
| /MDISK | Specifies the MDISK Menu |
| /DISK | Specifies the Diskette Options |
| /VERIFY | Specifies the Verify After Diskette Write Menu |
| /SERIAL | Specifies the Select Serial Console Menu |
| /MEMORY | Specifies the Maximum Memory per Process Menu |
| /PFKSAVE | Selects Save System Parameters, saves the current programmable function key assignments, and leaves you at the SETUP Main Menu |
| /SYSDAT | Controls amount of memory in the system space reserved for use by DOS compatible device drivers |
| /CLOCK | Permits calibration of status line clock for machines with non-standard clock hardware |
| /LIM | Allows LIM memory parameters to be set up |

F2 -- Reserve System Space

This option controls the amount of memory in the system space reserved for use by DOS compatible device drivers. Concurrent stops the device driver installation with an error message if the device being driven does not fit into the available space. Many drivers will not require extra space, but if the driver will not load, you can increase this value by 128 bytes and retry. Space reserved here reduces the memory available to the applications programs. The amount of System Space currently reserved is zero bytes.

The F2 option gives you a new menu with these choices:

- * F1 Help
- * F3 Enter a new value
- * Esc Return to Main Menu

F3 -- Save System Parameters

Select F3 (or /PARAM) to save the current window's function key assignments.

Before you select this option, run the FUNCTION command to prepare your function key commands. The parameters in effect when you select this option become your new startup values.

When you select the F3 option, a new menu is displayed and you are presented with the following option:

- * F3. Save Programmable Function Keys: Press F3 to save the current function key assignments. The word "Saved" is displayed when you press F3. Press F3 again to deselect this option.
- * F5. Save Serial Port Configuration: Press F5 to save the serial port values. "Saved" appears and disappears as you continue to press F5.

Press Esc to exit the option and return to the SETUP Main Menu. Only the options saved when you exit are incorporated into Concurrent.

F4 -- MDisk

Select F4 (or /MDISK) to create disk drive M in your system memory.

Disk drive M is a block of memory with all the characteristics of a disk drive, with one important exception: all information on drive M is lost when the computer is turned off. You can COPY files to and from drive M; it can be your system drive; you can run programs stored on it. Drive M has a significant speed advantage over floppy-disk drives. If you want to save information permanently, you must COPY the files to a floppy disk or hard disk partition before you turn off your computer.

The MDISK menu displays the current MDISK size. The initial value for this parameter indicates how much memory is already reserved for drive M.

Unless an MDISK already exists in your system, the initial value for the MDISK size is zero kilobytes.

The MDISK Menu provides the following options:

- * F3. Increase MDISK size: When you press F3, the size increases in increments of 16K up to a maximum of 512K.
- * F4. Decrease MDISK size. When you press F4, the size decreases and the starting address increases in increments of 16K.

To exit the MDISK option, press the Esc key.

F5 -- Diskette Drive Parameters

This option provides customization to support non-standard diskette drives. Type Esc if your system consists of standard drives. Caution: A customized version of Concurrent may not run correctly on other machines. Be sure that you have a backup copy of CCPM.SYS. The F5 option gives you these choices:

- * F1. Help
- * F2. Motor Startup Time. This parameter specifies how long Concurrent waits before attempting a diskette Read or Write after turning on the drive motor. If this interval is too small, Read and Write errors will result. If the startup time is too high, diskette accesses take longer than required. The current diskette motor startup time is 1000 msec.
- * F3. Head Step Time. This option sets a head step time of 2 to 32 msec for the diskette drive hardware. The smaller the time, the faster the drive head moves from track to track. The default time is set to 8 msec, but times of 6 msec or 4 msec work with some diskette drives. The current diskette head step time is 6 msec.
- * F4. Head Settling Time. Use this parameter to specify how long Concurrent waits before attempting a diskette Read or Write after moving the drive Read/Write head from one track to another. If the head settling time is too low, Read or Write errors will occur. If it is too high, disk accesses will take longer than necessary. The current diskette head settling time is 15 msec.
- * F5. Head Load Time. This parameter specifies how long the floppy disk controller waits before attempting a Read or Write operation after Concurrent gives a command to the controller. Most 5 1/4 inch drives use the minimum setting because the heads are always in a loaded state. Drives with head-load solenoids require a higher setting. The current head load time is 4 msec.

- * F6. Head Unload Time. This parameter specifies how long the floppy disk controller assumes the Read/Write head remains loaded after a Read/Write operation has finished. Most 5 1/4 inch drives use the maximum value of 480 ms because the heads are always in a loaded state. Drives with head-load solenoids require a lower setting. Current head unload time is 480 msec.

F7 -- Select Serial Console

Select F7 (or /SERIAL) to designate your serial ports. You can designate them as either aux/printer ports or as terminal ports. A terminal port is a serial port with a computer terminal (console and keyboard) connected. In other words, this feature allows several people to use the computer at the same time. (Concurrent DOS XM supports up to five serial terminals, each with one virtual console. Programs can be run on the serial terminals so long as they don't directly write to the screen. Concurrent DOS 386 supports up to nine serial terminals. If these terminals are PC emulation terminals, each may have up to two virtual consoles, and programs that write directly to the screen can be run on each terminal.)

The Select Serial Console menu itself leads to other menus, as described below.

F3 from the Select Serial Console menu introduces a new menu that allows you to setup a multiport expansion card (by pressing F3), or to reconfigure Concurrent after removing a multiport card (F5). Page through the list of cards by pressing F3 repeatedly. Then press F7 to select a menu that enables you to setup the characteristics of each serial port available on the chosen expansion card:

F2 selects the individual ports on the expansion card (numbered from 'Serial port 1' upwards).

F3 designates what the chosen serial port will be used for. It toggles between "Multi-user terminal" and "Printer/Aux port".

F5 allows you to select the parity checking convention for the chosen port - odd, even or none.

F6 selects the number of stop bits.

F7 selects the number of data bits.

F8 defines the preferred communications protocol.

F4 from the Select Serial Console menu allows you to switch on or off the "soft-boot" option, <CTRL><ALT>. Normally, pressing <CTRL><ALT> resets the computer. However, if your computer is used by several people at once, you may want to disable this option. If you do disable this option, note that users will still be able to reset the system with the REBOOT command.

F5 from the Select Serial Console menu allows you to reassign the use of COM1 and COM2 and to change their configuration. (Note that COM2 is disabled if a multiport card is fitted). If you choose to change the configuration you will see another menu with the following options:

F3 - Port Configuration. Use this option to display and change the port characteristics. The definable characteristics are as follows:

- * baud rate: number of bits of input or output per second
- * word length: number of bits in each word
- * parity: the use of odd, even, or no parity
- * stop bits: number of stop bits added to each word

F5 - Port Protocol: Use this option to set the chosen port's input and output communication protocol.

The values set here for COM1 and COM2 will become current once Concurrent is restarted. If you want to change COM1 or COM2's configuration without restarting Concurrent, use the SETPORT command.

The following option only applies to Concurrent DOS 386:

If you have Concurrent DOS 386 with multiuser terminals you will see a further option on the Select Serial Consoles menu: F6 - Setup Console Features. This leads to a further menu:

F2 selects the multiuser terminal ports.

F3 toggles between a non-PC emulation terminal or a PC emulation terminal for the chosen serial port. See your terminal's manual if you don't know which option to choose.

F4 selects, for a chosen PC emulation terminal, either the US keyboard or the national keyboard currently setup on the main console.

F5 toggles between one virtual console or two virtual consoles for a chosen PC emulation terminal. Concurrent 386 allows each serial terminal to run programs in two separate windows simultaneously. You switch between the two windows by pressing <CTRL>-1 or <CTRL>-2 in the usual way.

F6 selects either the 84-key keyboard or the enhanced keyboard for the chosen PC emulation terminal.

F8 -- MENU TWO

When you select F8 from the Main Menu, "menu two" is displayed with the following options:

The F3 option sets a limit on the amount of conventional memory available to each program (this is the default MEMSIZE - see the MEMSIZE command). Some programs claim all available memory for their use. This is not a problem when you are running only one program. It is a problem when you change windows and try to run another program. You might get the following message:

Not enough memory

The option menu displays the current limit. Press function F3 to enter a new value. Enter a three-digit decimal value with the number of kilobytes.

The F5 option permits the calibration of Concurrent's status line clock for machines with non-standard clock hardware. If you use this option, be sure and save a copy of CCPM.SYS as the altered version won't now work on standard clock hardware.

The F4, F6, F8, and F10 keys are used differently for 386 and XM versions of Concurrent DOS.

For Concurrent DOS 386:

F4 switches LIM emulation on or off.

F6 defines the LIM page frame address.

F8 defines the maximum available LIMSIZE.

F10 allows you to allocate high memory for Concurrent to use for buffer space allocation; this leaves more room in normal memory for applications to run. When you press F10 you will see a screen that allows you to enable 16Kb segments of high memory. On this screen,

F3 allows you to highlight different specified 16Kb segments of high memory

F4 toggles between 'enable' 'disable' and 'autoscan'. 'Enable' allocates the highlighted 16Kb segment of high memory for buffer space allocation. 'Disable' prevents it being used for buffer space allocation. 'Autoscan' means that Concurrent examines the highlighted segment at the next restart. If the memory segment is free, Concurrent switches it to 'enabled'; if not, Concurrent switches it to 'disabled'.

For Concurrent DOS XM:

F4 switches on or off Concurrent's built-in EMS device driver. Concurrent allows you to choose either this driver or the EEMS driver supplied with your expanded memory card. F6 allows you to allocate high memory for Concurrent to use for buffer space allocation; see the description for the F10 key in Concurrent DOS 386 above. You can only use this facility in Concurrent DOS XM if you run Concurrent's built-in EMS driver.

F9 -- Color Monitor Scroll Mode

Select F9 to change the speed and appearance of scrolling on a color monitor. Your selection determines the appearance and speed of your color monitor. Try each mode and choose the one that best suits your system and your personal tastes.

F10 -- Update and Exit

Select F10 to make your changes Concurrent's default values. The Update and Exit Menu appears. It defines the "Current Update Disk" and gives you two options: F3 to change the current update disk and F5 to update the current disk and exit SETUP.

Select option F3 to change the disk you want updated. SETUP looks to the current user number or directory of the disk you specify to find the CCPM.SYS file. Enter just the letter designation. The colon and Enter keys need not be pressed. You must specify a drive name that exists. After you have specified the drive name, you are returned to the Update and Exit Menu.

Select F5 to update your CCPM.SYS file and automatically exit SETUP to the command-line prompt. If you update the CCPM.SYS file from which you load Concurrent, your changes are implemented as soon as you start or reset Concurrent.

SHOW

Format: SHOW [d:]
SHOW [d:]option [,option...]

Explanation: Use SHOW to display disk-drive characteristics and status. The drive requested can be a floppy disk or a hard disk partition. Table 20 lists the SHOW options.

Table 20. SHOW Options

| Option | Description |
|--------|--|
| SPACE | Displays the remaining space available for file storage on a drive or drives. SHOW SPACE also indicates whether the disk is Read/Write or Read/Only. |
| DRIVE | Displays drive-recording characteristics. The values shown in the following example for a floppy-disk drive represent the total available and not the amount in use. A: Drive Characteristics 2,528: 128 Byte Record Capacity 316: Kilobyte Drive Capacity 64: 32 Byte Directory Entries 64: Checked Directory Entries 256: Records / Directory Entry 16: Records / Block 8: Sectors / Track 1: Reserved Tracks |
| LABEL | Displays a drive's label. |
| HELP | Displays a summary of the SHOW command syntax |

Note: Unless you specify a drive (d:), SHOW displays the option information for all drives accessed since the last computer reset.

SORT

Format: SORT [/R] [/+n]

Explanation: SORT reads data from the standard input device, sorts the data, and then writes it to the standard output device. Uppercase and lowercase letters are not significant.

The options are:

/R Sort in reverse order

/+n Start sorting with column n

Examples: The following example reads and sorts the file INDEX.DIS and writes the output to the printer.

A>SORT <index.dis >PRN:

You can also use SORT to “filter” information from another Concurrent command and present it to you in alphabetical, date or size order. The following example uses SORT to present a directory listing in reverse alphabetical order and then writes it to the file REVERSE.ORD.

A>DIR | sort /r >reverse.ord

STOP

Format: STOP program name Window-#
STOP

Explanation: The STOP command is provided in two forms. The first version is a built-in program and can be run from any console. It is used to stop any program that is running on another window on that same console. Notice that this form requires the entry of the command name and the window number in which it is running. For example, the next command will terminate program MYPROG running in Window 3.

```
A>STOP MYPROG 3
```

The second version of STOP is an external command which is run when you type STOP without any parameters. This second version will:

- * Display all the windows and programs currently being used on all the terminals.
- * Prompt you to terminate any of the programs listed.

This means you can use this second version of STOP to terminate any program on any window throughout the system. To terminate a program, enter the number assigned to it and press Enter.

To exit STOP without terminating a program, press Enter.

(You may like to password-protect the file STOP.CMD so that unauthorized users can't stop other people's programs.)

When you issue the STOP command without any parameters, you see the information listed in Table 21.

Table 21. STOP Display Information Descriptions

| Information | Description |
|--|--|
| Entry Number | Numbers given by the STOP command to all programs running. The numbers are sequential: the highest number indicates the first program invoked and the lowest indicates the most recent program invoked. STOP always appears first on the list with the designation **. |
| Program Name | The name of the program running. |
| Memory Used | The amount of memory used by the program. |
| Window Number | The number of the window in which the program is running. |
| Type of Free Memory | A list of contiguous blocks of memory. The memory can be of two types: conventional memory or banked windows. A banked window can be occupied or empty (specified by "Empty"). Assuming enough physical memory exists (on an expanded memory board), the largest bankable program that can be loaded is equal in size to the largest entry in this list. The largest non-bankable program that can be loaded is equal in size to the amount of conventional memory, or an empty banked window in this list, whichever is larger. |
| Enter the Number to delete or RETURN to exit | This prompt allows you to terminate a program or return to the Concurrent command line prompt. |

How to Terminate a Program

Normally, you should use the termination command procedure described in your program's manual because it saves the information you have entered.

Use the STOP command to terminate a program only if the normal termination command cannot be performed. Change windows if necessary and enter the STOP command. Find the program's number in the display, type that number, and press the Enter key.

STOP terminates the program specified. When you return to the window in which the program was running, the command line prompt is displayed and you can enter another command.

SUBST (Substitute)

Format: SUBST [d: d:path]
SUBST d: /D

Explanation: SUBST allows you to replace a drive-and-path specification with a single drive name. So, for example, you can replace a path like `\wordproc\simon\letters` with the drive S:. This can save typing and it can also be used for programs that don't recognise subdirectories.

Drives assigned by the SUBST command are called "floating drives". Any drive can be a floating drive except:

- * Drives that Concurrent already recognises as physical drives (typically A:, B:, C: ...)
- * Drive L:, which Concurrent reserves for its own use.

Entering SUBST without any parameters displays the current assignments. The /D option removes the assignment for the drive you specify.

Note: You are recommended to avoid using the Concurrent commands that involve whole disks (BACKUP, DISKCOPY, FORMAT, LABEL etc) when you have assigned floating drives.

Examples: The following example assigns drive S: to the path `\wordproc\simon\letters` on drive C:

```
A>SUBST S: C:\wordproc\simon\letters
```

The example below removes the assignment from drive H:

```
A>SUBST H: /D
```

SUSPEND

Format: SUSPEND=ON
SUSPEND=OFF

Explanation: Use the SUSPEND command to temporarily stop a program with the EXE or COM file extension while it is in a background (switched-out) window. (Use CHSET to suspend the operation of programs with the CMD filetype.) Program operation resumes when the program returns to the foreground window.

The most common reason to use SUSPEND is to stop applications that send program output directly to the screen. These applications can overwrite the contents of your foreground window when they are running in the background.

If one of your applications writes directly to the screen, enter the following command before you run the program:

A>SUSPEND=ON

SUSPEND works on a single window basis. Therefore, you must run SUSPEND on each window when you use a program that writes directly to the screen. The SUSPEND=ON state remains in effect on each window until the computer is turned off, reset, or you enter SUSPEND=OFF.

Unless you intend to run another application outputting directly to the console, enter the following command after you exit the program.

A>SUSPEND=OFF

Note: The PIFED command can be used to set .EXE or .COM files to suspend automatically in background.

TIME

Format: TIME
TIME hh:mm:ss

Explanation: Use the TIME command to display or set the time of day. Concurrent maintains an internal clock and displays the time on the status line. Concurrent also uses the time to record the time a file is created or updated when you implement time and date stamping. (See the ATTRIB command explanation for a description of time and date stamping.)

You must set the time whenever you start or reset Concurrent, unless you have a real-time clock with battery backup. (When you do not enter a time, Concurrent assumes the time 00:00:00.00.) Put the TIME command in your STARTnnn.BAT file to prompt you when you start Concurrent. The Concurrent clock works on a 24-hour basis. This means that 1 PM is 13:00 and midnight is 00:00.

A>TIME

TIME displays the time and prompts you as follows.

```
Current time is hh:mm:ss.xx  
Enter new time:
```

You can press the Enter key to keep the current values. To enter a new time, enter the hours, minutes, and seconds and press the Enter key. Note that the last value shown here, xx, indicates hundredths of a second.

The time values are defined as follows:

- * hh is a one- or two-digit number from 0 to 23 indicating the hour.
- * mm is a one- or two-digit number from 0 to 59 indicating the minute.
- * ss is a one- or two-digit number from 0 to 59 indicating the second.

TIME assumes that the first entry specifies the hour. The minute and second values are optional and if not entered are set to zero. You must use a colon to separate the entries.

If you enter a number that is out of range, TIME returns the message and prompt

```
Invalid time  
Enter new time:
```

Examples: A>TIME 22:30:59

This command sets the time for 30 minutes and 59 seconds after 10 PM.

A>TIME 00:12

This command sets the time at 12 minutes after midnight.

TOUCH

Format: TOUCH [/H] filespec [/T:hh:mm:ss] [/D:mm-dd-yy] [/P] [/R] [/S]

Explanation: TOUCH resets the time and date stamps for a single file or a group of files.

The options are:

/T:hh:mm:ss Time to be set. The time is set in 24 hour clock with midnight being 00:00. The default is the current time.

/D:date Date to be set. The date format depends on the current country code, set up in the CCONFIG.SYS file. The default is the current date.

/H Display help information.

/P Prompt before touching each file.

/R Include read-only files in the touch operation.

/S Touch files in subdirectories associated with the current directory.

Example: A>TOUCH *.LET /D:02-02-87 /P

This example resets the date stamps for all the .LET files. The new date is 2 February 1987. Touch will prompt you before changing each file.

TREE

Format: TREE [/H] [d:][filespec] [/F] [/P]

Explanation: The TREE command displays the hierarchical path of directories and subdirectories. When used with a filespec, TREE will find all the files matching the filespec on the disk.

Valid options are:

/H Displays the help information.
 /F Displays all the files in the subdirectories, as well as the subdirectories themselves
 /P Pauses after each screenful in a long TREE display.

Examples: The command:

A>TREE C:

might generate the following listing showing that drive C has four directories, DOS21, Con-Dos, Fred and John, some of which have subdirectories, and listing the space used and number of files belonging to each.

| bytes | files | path |
|-----------------|-----------------------|---------------|
| 273,713 | 23 | c:\ |
| 370,648 | 45 | c:\dos21 |
| 434,598 | 29 | c:\con-dos |
| 1,411,414 | 43 | c:\fred |
| 536,563 | 26 | c:\fred\crg |
| 0 | 0 | c:\john |
| 421,760 | 14 | c:\john\cug |
| 4,352 | 3 | c:\john\memos |
| total files 183 | total bytes 3,453,048 | |

The command:

A>TREE *.GTM

generates, for example, the listing showing two files named *.GTM on drive A. One is in the root directory, and one is in the subdirectory \DR\MEMOS.

| | | | | |
|---------------|-------------------|---------|------|-----------------------|
| --a--- | 1,920 | 1-01-80 | 0:40 | c:\memo1.gtm |
| --a--- | 512 | 1-01-80 | 0:30 | c:\dr\memos\memo2.gtm |
| total files 2 | total bytes 2,432 | | | |

TYPE

Format: TYPE [dirpath\]filespec [/P]

Explanation: The TYPE command displays the contents of a text file on your screen. The file specification is the filename and file extension. If the file is password-protected for READ mode, you must set the default password. The wildcard characters (* and ?) may be used to type multiple files.

The /P option causes the text to be displayed a screenful at a time; move to the next screen by pressing any key.

Press Ctrl-S to stop the display from scrolling. To restart scrolling, press Ctrl-Q. Press Ctrl-C to abort the display.

Press Ctrl-P to have the file contents sent to the printer as well as the screen. Press Ctrl-P again to stop output to the printer. Note that screen output continues to be sent to the printer after the TYPE command has completed until you press Ctrl-P.

VER (Display version number)

Format: VER

Explanation: This displays the version of Concurrent DOS that you are using.

VERIFY

Format: VERIFY [ON|OFF]

Explanation: VERIFY ON causes Concurrent to check that data is written to the disk correctly after each write operation. This, however, causes your system to run slightly slower and the default value for VERIFY is OFF.

Entering VERIFY with no parameters shows whether VERIFY is currently ON or OFF.

VOL (Display volume label)

Format: VOL [d:]

Explanation: VOL displays the volume label of the specified drive, or of the default drive if no drive name is given.

If the disk in the drive has no label then the message

Volume in drive d has no label

will be issued.

WINDOW

Format: WINDOW command NUMBER=n parameter [,parameter...]

Explanation: The WINDOW command displays and changes window parameters and records the window and screen contents. Table 22 summarizes the command options of the WINDOW command. You can enter the command in its full or abbreviated form.

Several WINDOW command options require a window number; others require more information in the command line. Commas are used to separate window parameters. Take care in using commas and spaces properly when entering a WINDOW command as shown above and in the examples below.

The WINDOW and WMENU commands (see WMENU) provide the same window management tools. Use WINDOW commands when you need to manage a window with command-line entries. For example, use WINDOW commands in BATCH files to create window displays for your applications. Use WMENU (Window Menu) to create window displays interactively and manipulate window placement, size, and color with the arrow keys.

Table 22. WINDOW Command Summary

| Command | Abbr | Summary |
|---------|------|--|
| VIEW | V | Displays each window's current parameter values |
| TOP | T | Switches in the window specified |
| FULL | F | Displays the complete screen |
| WRITE | WR | Records the window or entire screen contents in a file |
| CHANGE | CH | Modifies a window's parameters |

Concurrent's windows are a mechanism for controlling four program displays on your screen. When you receive Concurrent, the four windows are displayed separately. Simultaneously press the Ctrl key and one of the numeric keypad numbers 1 through 4 to switch from one window to another.

You can also display two or more windows at the same time by manipulating the size and placement of the windows. The windows are stacked one on top of another on your screen somewhat like sheets of paper. When you start or reset Concurrent, window 1 is the topmost sheet; window 2 is underneath it; window 3 is the next, and window 4 is on the bottom. As you change windows, the sheets of paper are shuffled into a new order.

Each window's characteristics are determined by a set of parameters that can be changed with the WINDOW command. The window size, for example, is determined by parameters which define the number of rows and columns displayed. In a full screen display, each window has 24 rows and 80 columns. Row 1 runs along the top of the screen; row 24 runs along the bottom. (The status line at the bottom of the screen is on row 25 and is not considered part of a window.) Column 1 is the leftmost column, column 80 the rightmost.

Before reading the following detailed descriptions of WINDOW command options, read through the WMENU description and run through the exercise provided in the "Example" section of WMENU. WMENU provides a fast, easy method for understanding the concepts of windows.

VIEW

Format: WINDOW V[IEW]

Explanation: WINDOW VIEW lists the current parameter values for all four windows. Figure 2 illustrates the display. Table 23 defines the column headers. The values shown in the columns are representational and do not describe any particular window arrangement.

| Number | PRow | PCol | NRows | NCols | VRow | VCol | Tracking | Display | FGColor | BGColor |
|--------|------|------|-------|-------|------|------|----------|---------|---------|---------|
| 1 | 1 | 1 | 24 | 80 | 1 | 1 | Row | Color | White | Magenta |
| 2 | 10 | 10 | 12 | 40 | 1 | 1 | Row | B&W | Black | White |
| 3 | 12 | 30 | 11 | 63 | 5 | 7 | No | Color | Red | Cyan |
| 4 | 5 | 15 | 5 | 20 | 1 | 1 | Row | B&W | White | Black |

Figure 2. Sample WINDOW VIEW Display

Table 23. WINDOW VIEW Parameter Definitions

| Parameter | Definition |
|-----------|---|
| Number | Window number for that row of values. |
| PRow | Physical row number of the window's top border. |
| PCol | Physical column number of the window's left border. |
| NRows | Number of rows in the window display. |
| NCols | Number of columns in the window display. |
| VRow | Row number of the virtual scroll point.* |
| VCol | Column number of the virtual scroll point.* |
| Tracking | Indicates that the window always contains the row the cursor is on (the ROW setting) or that the window does not necessarily contain the cursor (the NO setting). |
| Display | Indicates whether the window is displayed on a color or monochrome monitor. |
| FGColor | Color of the characters on the screen. |
| BGColor | Background color. |

* The virtual scroll point is an imaginary point in the upper-left corner of the window. The row and column numbers are the coordinates of the point with respect to the full screen program display. Concurrent uses these values to maintain the window position while the display underneath scrolls.

TOP

Format: WINDOW T[OP] N[UMBER]=n

Explanation: WINDOW TOP switches in the window number entered. This command performs the same operation as pressing Ctrl and one of the numeric keypad numbers 1, 2, 3, or 4.

A>WINDOW T N=4

The previous command switches in window 4.

FULL

Format: WINDOW F[ULL] N[UMBER]=n

Explanation: WINDOW FULL can be used to expand the designated window to the full 24 rows by 80 columns display. When you enter your current window as "n," the expansion is immediate. (In this case, the command has the same effect as pressing the Ctrl-Del key combination.) When you specify a switched-out window, the full-screen window is not shown until you switch it in.

A>WINDOW F N=3

This command expands the size of window 3 to the full 24 by 80 display. If window 3 is current, this result is immediate. If window 3 is switched-out, the current screen remains unchanged and the result is not apparent until window 3 is switched in.

WRITE

Format: WINDOW WR[ITE] N[UMBER]=n TYPE=W,FILE=filespec

Explanation: The WINDOW WRITE command records the screen display in the file designated. The complete display is recorded when you specify TYPE=C; only the window portion is saved when you specify TYPE=W. The file created is a standard text file that you can edit and print.

The WINDOW WRITE file specification must contain at least the filename. The file extension and disk reference are optional. Unless a disk reference is specified, the file is written to the current user number or subdirectory on the current disk.

A>WINDOW WR N=2 TYPE=W,FILE=SHADES

This command writes just the window portion of window 2 into the file SHADES.

A>WR N=2 TYPE=C,FILE=D:SASH.TXT

This command writes the entire screen display of window 2 into the file SASH.TXT on drive D.

CHANGE

Explanation: Use WINDOW CHANGE to modify a window's parameters. Table 24 lists the parameters and the values each can take. The letters shown in parenthesis are the abbreviations you can use for each parameter and value.

The WINDOW CHANGE command requires a window number. You can modify one parameter or all parameters from a single command line. If a parameter is not entered, it is not changed. Separate each parameter/value equation with a comma.

Table 24. WINDOW CHANGE Parameter Values

| PRow (PR) | PCol (PC) | NRows (NR) | NCols (NC) | Tracking (T) | Display (D) | Colors* |
|----------------------|----------------------|-----------------------|-----------------------|-------------------------|------------------------|--|
| 1-24 | 1-80 | 1-24 | 1-80 | ROW NO | COLOR B&W | BLACK (BLA) WHITE (W) RED (R) YELLOW (Y) CYAN (CY) MAGENTA (M) BLUE (BLU) GREEN (G) |

* The colors listed are available for the FGColor and BGColor parameters.

VRow and VCol were left out of Table 24 intentionally. You do not need to modify these parameters except under the most extraordinary circumstances. If you want to experiment, their parameters range from 1 to 24 and 1 to 80 respectively.

```
A>WINDOW CH N=4 FG=CY,BG=BLU,T=NO
```

This command modifies the window 4 display so that the foreground color is cyan (light blue), the background color is dark blue, and cursor tracking is turned off.

```
A>WINDOW CH N=1 PR=1,PC=1,NR=24,NC=80,VR=1,VC=1,TR=ROW,FG=WH,BG=BLA
A>WINDOW CH N=2 PR=1,PC=1,NR=24,NC=80,VR=1,VC=1,TR=ROW,FG=WH,BG=BLA
A>WINDOW CH N=3 PR=1,PC=1,NR=24,NC=80,VR=1,VC=1,TR=ROW,FG=WH,BG=BLA
A>WINDOW CH N=4 PR=1,PC=1,NR=24,NC=80,VR=1,VC=1,TR=ROW,FG=WH,BG=BLA
```

This set of commands returns all windows to the full-screen size with white characters on a black background. It might be worthwhile for you to create a BATCH file with these commands to clear your screen of existing windows and return to the default setting.

WMENU

Format: WMENU

Explanation: WMENU (Window Menu) provides interactive window manipulation. Once WMENU is loaded, the cursor control keys on the numeric keypad select parameters and change values. As you manipulate the values, the changes are reflected on the screen.

A significant feature of the WMENU command is that it creates a BATCH file of WINDOW commands for you. When you have settled on a display, select the WRITE command in the WMENU Command Menu and then the WSETUP.BAT option. This records the WINDOW commands for all four windows in the file WSETUP.BAT. When you run this file, the windows are set up exactly as you configured them with WMENU.

Invoke WMENU by entering WMENU and pressing the Enter key. The message

```
Window Manager installed
```

is displayed, WIN appears on the status line, and the command-line prompt is returned to the screen. As long as WIN is shown on the status line, you can use the WMENU facilities.

To manipulate the windows, press the Ctrl key and + key (on the numeric keypad) simultaneously. This replaces the status line at the bottom of your screen with the WMENU Command Menu and enables the numeric keypad keys for WMENU use.

Although you can manipulate windows while other programs are running, this can cause conflicts over the status line display. Consequently, set up your windows with WMENU before you run your applications.

The remainder of the WMENU explanation describes the rules of WMENU operation and the command menu options. Table 30 summarizes the commands for quick reference. The example at the end of this description is a hands-on exercise that shows you how to create a window display. If you have never used the WMENU or the WINDOW commands, run through this exercise before proceeding with the rest of the WMENU description.

Table 25. WMENU Command Summary

| Command | Description |
|----------|--|
| PLACE | Moves the window's top and left borders. |
| SIZE | Moves the window's right and bottom borders. |
| SCROLL | Moves the window around to expose different portions of the program display. |
| TRACKING | Turns cursor tracking on (ROW) or of (NO). |
| DISPLAY | Selects a COLOR or monochrome (B&W) monitor as the display device. Note that you can have both connected to your computer. |
| COLOR | Selects foreground and background colors for each window display. |
| WRITE | Saves the WINDOW commands in WSETUP.BAT, the complete program display in CONSOLEn.TXT, or the window contents in WINDOWn.TXT. In both cases, n is replaced by the window number. |
| ABORT | Terminates the WMENU command. |

WMENU Rules of Operation

Load WMENU and press the Ctrl++ key combination to display the command menu on the status line. N=n: (where n is replaced by a number) at the beginning of the line indicates the current window number. You change windows by pressing Ctrl and one of the keypad numbers 1, 2, 3, or 4. You do not need to leave WMENU to change windows.

Once you modify a window, you can switch to full screen and back at any time by pressing Ctrl-Del.

Select a command from the menu by moving the white box over it with any of the arrow keys or the spacebar, and pressing Enter. The status line display is replaced by the options for the command you selected.

To exit the WMENU Command Menu and return to the command-line prompt, press the Esc key. You can also use the Esc key to return to the previous menu.

Note: Except for the ABORT and WRITE commands, you can press Enter or Esc to exit the other command options and return to the WMENU command menu.

PLACE

The PLACE command moves the top and left window borders. Use the Down- and Right-arrow keys to move the window down and to the right. As you move the window, notice that the next window is revealed underneath. PLACE does not decrease the size of the window; it only moves portions of the right side and the bottom offscreen.

PLACE changes the values of the PRow and PCol parameters. The current values of these parameters appear in the status line.

SIZE

Use the SIZE command to decrease the size of the window. Use the arrow keys to move the righthand and bottom window borders in towards the center of the screen. The lefthand and top borders are not affected by the SIZE command.

When you first press the Left- and Up-arrow keys, you may not see anything happening on the screen. This is because the borders are offscreen because of a previous PLACE command.

SIZE changes the values of the NRows and NCols parameters. The current values of these parameters appear in the status line.

SCROLL

SCROLL uses the arrow keys to move the window backward, forward, up, and down over the program display. Thus, you can adjust your view of the program's display to show just the area you want.

If you select ROW with the TRACKING command, the SCROLL command can only move the window sideways. This is because ROW tracking locks the window onto the row containing the cursor. If you want to expose a portion of the window that does not include the cursor row, you must select the NO tracking option.

SCROLL changes the values of the VRows and VCols parameters. The current values of these parameters are displayed on the status line.

TRACKING

Use the TRACKING command to enable and disable cursor tracking. The two TRACKING options are described as follows:

- * ROW (the default value): The window is locked over the row the cursor is on. If the window includes column 1 when ROW is selected, the command-line prompt is always shown.
- * NO: The window is locked over a position that does not include the cursor row. Select this option when you want to use SCROLL to view the program display away from the cursor location.

When you select the TRACKING command, the status line display shows the current setting in a white box. Use the arrow keys or spacebar to change options.

When the option you want is highlighted, press the Esc key to return to the WMENU Command Menu.

Use DISPLAY to indicate whether your window's monitor is color or monochrome (B & W).

When you select the DISPLAY command, the status line indicates the current setting. Use the arrow keys or spacebar to switch between the COLOR or B&W option. When the option you want is highlighted, press Esc to return to the WMENU Command Menu.

Note: Some computer manufacturers make a monochrome monitor appear as though it were a color monitor. Consequently, the DISPLAY command indicates COLOR rather than B&W. Do not be concerned if the option named is not reflected on your screen; performance is not affected.

COLOR

Use the COLOR command to select the window's foreground and background colors. When you select the COLOR command, the status line shows the following two color parameters:

- * FG: foreground color parameter
- * BG: background color parameter

You press the Right- or Left-arrow key to switch your selection from one to the other. The white box highlights the current selection.

The current color for each parameter is always displayed on your window. You change colors by pressing the Up- or Down-arrow key or the spacebar. As you change colors, the display changes. The following colors are available for both background and foreground parameters:

| | |
|--------|---------|
| Black | White |
| Green | Cyan |
| Blue | Red |
| Yellow | Magenta |

When you have selected both the foreground and the background colors, press Esc to return to the WMENU Command Menu.

The window display on a monochrome monitor can be affected by the color selection. For example, you can select between standard white letters on a black background or inverse video with black letters on a white background. On monochrome monitors that come up as COLOR when you select the DISPLAY command, you might have some other screen options as well. You can try changing the foreground and background colors just to see what effect it has and if it is useful to your applications.

WRITE

Use the WRITE command to record the current windows' parameters, the window contents, or the complete screen display.

When you select the WRITE command, WMENU displays the following on the status line:

```
WINDOW WRITE N=n Type=t,File=filename.ext
```

You can select one of three values for Type and File. Table 26 lists your options.

Table 26. WMENU WRITE Command Options

| Type | File | Description |
|------|--------------|---|
| S | WSETUP.BAT | Records the current parameters of all windows |
| W | WINDOWn.TXT | Records the contents of window n |
| C | CONSOLEn.TXT | Records the full program display |

Use an arrow key to change and display the options. Press Enter to select an option and write the file. After the file is written, you are returned to the WMENU Command Menu.

If you have selected the WRITE command and do not want to write a file, press the Esc key to return to the WMENU Command Menu.

The file is always written the current directory (DOS media) on the current drive. You cannot change any of the file specifications for the three types of WRITE operations.

The WRITE command replaces any existing WSETUP.BAT, WINDOWn.TXT, or CONSOLEn.TXT file. If you want to keep the existing versions, rename them before you select the WRITE command.

The WSETUP.BAT file contains the series of WINDOW commands that configure windows 1 through 4 to the configuration they had when WSETUP.BAT was created. Whenever you want your windows to assume these characteristics, just enter WSETUP.

The Type W and C options are useful to save the information in the program display in a text file. Later, you can edit or print the file for any number of uses.

ABORT

Once you have created and saved your windows, there is no need to leave WMENU running. When you select ABORT, the following message appears:

(Press [Enter] to abort WMENU.
[ESC] to go back to previous menu)

Press Enter to remove WMENU from memory. The command-line prompt appears. You cannot use WMENU again until you enter the WMENU command, press the Enter key, and press the Ctrl-+ combination.

Example:

The following example is an exercise in using WMENU to manipulate windows. The steps listed below create a unique window and write this configuration in BATCH file form to the current disk. Note that you can switch between windows and configure them all before selecting the WRITE command.

Several commands used in this exercise anticipate the presence of a color monitor. If you do not have one, proceed as though you do. The change-color steps do not harm a monochrome console; by using these commands you see how your console is affected. (Not all computers and monochrome monitors respond in the same way.)

This exercise is not intended to generate a meaningful window display. Rather, it is intended to teach you how to create your own windows and to save them for use with your application programs.

Start Concurrent and run through the windows by pressing Ctrl-2, 3, and 4 to see if there are any start up programs running. You might want to terminate these programs but it is not necessary. Return to window 1 when done and enter

A>WMENU

The WMENU installation message is displayed and you are ready to proceed.

1. Press Ctrl-+ to display the WMENU Command Menu on the status line.
2. Press the right-arrow key until COLOR is highlighted. Select it by pressing the Enter key. The foreground and background color options appear. Use the Left- or Right-arrow key to switch from the FG or BG color. Select the BG color. Press the Up- or Down-arrow key, or spacebar to change the background color to Cyan (light blue). Notice the color on your screen changes with each new choice. Use the Left-arrow key to switch to the FG color. Use the Up- or Down-arrow key or spacebar to change the foreground color to Magenta. Press Enter to select these two colors and return to the WMENU Command Menu.
3. Press the Left-arrow key until SIZE is highlighted. Press Enter. Press the Up-arrow key 12 times until NRows=12. Press the Left-arrow key 40 times until NCols=40. Press Enter.
4. Press the Left-arrow key once to highlight PLACE. Press Enter. Press the Right-arrow key 39 times until PCol=40. Press Enter. You now have in the upper-right corner of the screen a window that is 1/4 the size of the full screen. The characters are Magenta on a Cyan background. Notice how window 1 overlays window 2.
5. Press the Right-arrow key until TRACKING is highlighted. Press Enter. Use the Left- or Right-arrow key to switch from ROW to NO. (Cursor tracking is now disabled.) Press Enter.
6. Press the Left-arrow key once so that SCROLL is highlighted. Press Enter. Press the Right-arrow key 33 times until VCol=34. Note that as you SCROLL over the screen to the right, the screen characters appear to move to the left off the window. To bring the command-line prompt back into view, press the Left-arrow key 34 times. Press Enter.
7. Press the Left- or Right-arrow key to highlight ABORT. Select it by pressing Enter. Press Enter again to exit WMENU and return the system status line and command-line prompt to your window. You can now enter Concurrent commands.

To make a small window full screen, press Ctrl-Del. Press Ctrl-Del again to return to the smaller window.

XCOPY

Format: XCOPY [/H] filespec [filespec] [options]

Explanation: XCOPY selectively copies groups of files. It can be used to copy whole subdirectories.

The first file specification on the command line is the drive, path and name of the files to be copied. Wildcard filenames are allowed. The second file specification is the destination drive and path to which the files will be copied. You can also rename files while copying them by specifying a new file name for the destination file.

The options are:

- /A Copy only those files that have the archive bit set. The attribute of the source file is not changed. (See the ATTRIB command for an explanation of the archive bit and other attributes.)
- /D:date
Copy only those files whose date is the same or later than the date specified. The format of the date depends on the current country code (see the DATE command).
- /E Create subdirectories on the destination even if they end up being empty after all the copying is complete.
- /H Include system files. (Files with the system attribute are explained in the ATTRIB command description.) The default is for system files to be ignored.
- /P or /C
XCOPY prompts you as each file is copied.
- /R Overwrite read-only files.
- /S Copy the files in the source directory and files in all the subdirectories below it.
- /V Verify that the data written to the destination is a true copy of data in the source.
- /W Wait for a diskette to be inserted before beginning to search for source files and (if necessary) wait again before the files are written to the destination.

Examples: Suppose you have files in a directory called C:\wordproc which has two subdirectories \wordproc\jenny and \wordproc\olivia. You want to copy all the files that have the file extension .LET from these directories on drive C onto a diskette in the A drive. You also want the appropriate subdirectories to be created on the diskette in A. The command is:

```
C>XCOPY wordproc *.let A: /E /S
```

The following example copies all the files that are marked for archiving from a diskette in drive A to another diskette in drive B. (Note: the archive attribute is set by Concurrent whenever a file is written to and then closed.) You will be prompted to insert a diskette in drive A before the copying operation starts. The archive attribute will be turned off at the source file to show that the file has been copied.

```
A>XCOPY *.* B: /M /W
```

XDEL

Format: XDEL [/H] filespec [/D] [/N] [/P] [/S]

Explanation: XDEL is a powerful tool for deleting files in subdirectories and removing empty directories.

The options are:

- /H Displays help information.
- /D Removes empty subdirectories.
- /N Deletes all the specified files without asking further permission.
- /P Prompts you before deleting each file.
- /S Deletes files in subdirectories derived from the current directory.

Example: The following command will delete all the files with the .OLD extension from the current directory and all the subdirectories below it.

```
A>XDEL *.old /S
  path: a:\
  file: *.doc
warning: FILES IN SUB-DIRECTORIES MAY BE DELETED
Is this what you wish to do (Y/N) ? Y
a:\first.doc
a:\sub1\mem1.doc
a:\sub2\mem2.doc
a:\sub3\letter3.doc

A>
```

XDIR

Format: XDIR [+|-ADHRS] [filespec] [/B] [/C] [/L] [/P] [/S] [/T] [/W]

Explanation: XDIR is an expanded version of the the DIR command, offering extra options for displaying the contents of directories and subdirectories.

The options are:

+|- [A] [D] [H] [R] [S]

You can specify to include (+) or exclude (-) files with a certain attribute. The default is for all the files to be displayed, including system files. See the ATTRIB command for details of the different attributes.

- /B Produces a "brief" directory display, only showing filenames and paths.
- /C Computes and displays a hash code for each file. A hash code is a four-digit hexadecimal number calculated from the bytes within the file. It can be used to verify that two files are the same.
- /L Produces a "long" display format. This is the default option.
- /P Pauses after each screenful of display.
- /R Reverses the sort order - either the alphabetical order by filename (the default) or the chronological order, set by the /T option below.
- /S Displays the files in subdirectories derived from the current directory.
- /T Sorts the directory by date and time, rather than by filename.
- /W Displays the directory in a wide, abbreviated format. This is useful if your directory display would otherwise be too long for the screen.

Examples: The following example displays all the read-only files in the current directory and the associated subdirectories:

```
A>XDIR +R *.* /S
```

The following example displays all the files that match the specification MEMO*. * on the current directory, and sorts them into reverse chronological order, so that the most recent are at the top of the list:

```
A>XDIR memo*. * /T /R
```

INDEX

%n, 17
8087, 6
8087 command, 2, 7
[], 1
\
 1, 68, 77

(A)

Abort WMENU, 141
Archive attribute, 10
ASCII, 34
Assign Func keys command, 64
Assign password, 80
Asterisk, 66
AUX command, 2, 11
Auxiliary port, 11

(B)

Background 122
Background color 137, 139, 142
Backslash key, 1, 68, 77
BACKUP command, 2, 12
BANK, 15
 command, 2, 15
BAT, 16
BATCH,
 command, 2, 16
 file, 16
 variables, 17
Baud rate, 104, 113
BG, 139
Boot disk 53
BREAK command, 2, 23
Built-in command, 6

(C)

CARDFILE command, 3, 24
CARDFILE.CMD, 24
CCONFIG.SYS, 41, 101
CCPM.SYS,
 modify, 108, 114
Change
 Directory command, 25
 window switching keys, 63,
 65

CHDIR, 25
 command 3, 25
 changing current directories,
 25
 displaying the current path,
 27
 floating drives, 26
CHKDSK command, 3, 28
CHSET command, 3, 29
Clock calibration, 123
CLS command, 3, 32
CMD, 3
 header field descriptions, 29
Colors, 132, 135, 137, 139
Column number, 133
COM, 2, 4, 7, 83
Command
 header, 29
 Header SET, 29
Command summary
 Concurrent, 2
 WINDOW, 131
 WMENU, 137
Communications protocol, 105,
106
CON:, 36
Concurrent command summary,
2
COPIES=n, 89
COPY
 command, 3, 33
 CON:, 36
 device names, 33
 options, 34
 use with passwords, 33
Copy
 diskette, 46, 47, 49
COPYMENU command, 3, 37
COUNTRY command, 41
CPM command 3, 38
CREATE option
 directory, 77, 143
 DOS partition 54

(D)

DATE, 40
 command, 3, 40
Date and time stamping, 40,
123, 125
Decrease memory, 74, 76
Default memory allocation, 74,
76, 114, 115
DELETE command, 42, 52
DELQ command, 3, 42
Dev., 1
DIR,
 attribute, 10
 command, 3, 43
 options, 44
Directory, 43
 label, 73
 path, 26, 27
Dirname 2
Dirpath 2
Disk format, 46
DISKCOMP command, 3, 45
DISKCOPY command, 3, 46
Diskette
 drive parameters, 111
DREDIX command, 3, 48
Drive
 speed, 111
Drive M, 110, 111
Drives
 floating drives 26, 121
DSKMAINT command, 3, 49
DSP, 88

(E)

ECHO subcommand, 18
EDITMENU command, 3, 50
Environment strings, 101
 in BATCH files, 17
ERAQ command, 3, 51
ERASE
 command, 3, 52
 file, 52
EXE, 2, 4, 7, 83

(F)

Fd, 2
FDISK command 3, 53
FG, 139
File Allocation Table (FAT), 28
File Manager, 2, 58
Filename, 1
Files
 BATCH, 16
 concatenate, 35
 copy, 33, 143
 display, 126, 127
 erase, 51, 52, 145
 print, 87
 rename, 95
 verify copy, 34, 35
Filespec, 1
FIND command, 3, 57
Floating drives, 2, 26, 121
FM command, 3, 58
FOR subcommand, 19
Foreground color, 137, 139, 142
FORMAT, 89
 command, 3, 59
 diskette, 49, 59
 hard disk, 53
Formfeeds, 89
FORMSIZE=n, 89
Full screen, 131, 134, 137
FUNCTION command, 3, 62
Function key
 assignments, 3, 62

(G)

GOTO subcommand, 19

(H)

HELP
 command, 3, 70
HELP files
 CREATE option, 71
 customize, 71
 EXTRACT option, 71
HELP.HLP, 70, 71

(I)

IF subcommand, 20
 Increase memory, 74, 76, 114, 115
 Internal clock, 40, 123

(J)

Job queue, 88, 90

(L)

LABEL command, 3, 73
 Left margin, 89
 LIMSIZE command, 3, 74
 LOADSYS command, 4, 75
 LPT1:, 33
 LST:, 33

(M)

MARGIN=n, 89
 MDISK, 110
 Memory, 74, 76
 available, 117
 banked, 120
 disk, 110
 drive M, 110
 free, 120
 MEMSIZE, 4, 76, 114
 MKDIR command 3, 77
 MORE command 4, 79
 Multiuser support, 112

(N)

NOFEED, 89
 NUL:, 33
 NUMBER, 89

(O)

Optional items, 1

(P)

PAGESIZE=n, 89
 Paper length, 89
 Parity, 105, 106, 113
 Partition, 54

PASSWORD command, 4, 80
 Password protection, 80, 81
 Path, 27, 126
 CHDIR, 25
 display current directory, 27
 display the current path, 27
 PATH command, 4, 82
 PAUSE subcommand, 21
 PIFED, 4, 83
 PFK file, 65, 67
 PgDn, 67
 PgUp, 67
 Port 0 protocol 104
 Port 1 protocol 104
 Port configuration 104
 PRINT command, 88
 PRINTER command, 4, 86
 Printer numbers, 4, 86
 PRINTER=n, 89
 PRINTMGR command 4, 87
 PRN:, 33
 Program, stop, 119
 PROMPT command, 4, 92

(R)

REBOOT command, 4, 94
 REM subcommand, 22
 Remove
 Directory command, 97
 empty subdirectory, 145
 RENAME command, 95
 Rename file, 95
 REPLACE command, 4, 96
 RMDIR command, 4, 97
 R/O, 10
 Row number, 135
 RUNMENU command, 98
 R/W, 10

(S)

Save function keys in file, 65
 SCEPTEP command, 4, 99
 Scrolling, 115, 133, 137, 138
 Sectors, 45, 46
 Serial ports, 11, 103, 112
 SET command, 4, 101
 SETPORT
 command, 4, 103
 parameters, 105

SETUP command, 4, 107
SHARED SETTING ON, 29
SHIFT subcommand, 22
SHOW command, 4, 117
SPL, 88
START command, 88
STARThnn.BAT file, 123
STATUS command, 90
Stop a program, 119
Stop bits, 105, 106, 113
STOP command, 4, 119
Subdirectories, 43, 126
 floating drives, 26, 121
 how to make current, 27
 remove, 97, 145
Subname, 2
SUBST command, 4, 121
SUSPEND
 command, 4, 122
 SETTING ON, 29
Switch windows, 65
SYS,
 attribute, 10
System space, 109

(T)

Tabbing, 89
TABS=n, 89
Terminal port, 103
TERMINATE command, 87
Terminate program, 120
TIME command, 4, 40, 123
TOUCH command, 4, 125
Tracking, 139
TREE command, 5, 126
TYPE command, 5, 127

(U)

USER subcommand, 38
User number, 38

(V)

Variables
 with batch files, 17
VER command, 5, 128
Version number, 128

VERIFY command, 5, 129
Verify
 copy, 34
 diskette, 35
Virtual scroll point, 133
VOL command, 5, 130
Volume label, 73, 130

(W)

Wildcards, 33, 42, 50, 51
WINDOW
 change parameters, 135, 138
 change size, 137, 138
 command, 5, 131
 command summary, 131
 display parameters, 131
 FULL command, 134
 full screen, 134
 manipulate, 136
 move, 138
 parameters, 135
 save contents, 140, 141
 switch in/out, 133
 switching keys, 65
WINDOW VIEW display, 132
WMENU, 146
 command, 5, 136
 command summary, 137
Word length, 105, 106, 113
WSETUP.BAT, 136, 140

(X)

XCOPY command, 5, 143
XDEL command, 5, 145
XDIR command, 5, 146

