DIGITAL
RESEARCH(R)

CP/M Plus™
Operating System
Command Summary

COPYRIGHT

Copyright (C) 1984 Digital Research Inc. All rights reserved. No part of this publication may be reproduced, transmitted, 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 INC. 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

From time to time changes are made in the filenames and in the files actually included on the distribution disk. This manual should not be construed as a representation or warranty that such files or facilities exist on the distribution disk or as part of the materials and programs distributed. Most distribution disks include a "README.DOC" file. This file explains variations from the manual which do constitute modification of the manual and the items included therewith. Be sure to read this file before using the software.

TRADEMARKS

CP/M and Digital Research and its logo are registered trademarks of Digital Research Inc. CP/m Plus, LINK-80, MAC, MP/M, PL/1-80, RMAC, SID, TEX, and XREF are trademarks of Digital Research Inc. Intel is a registered trademark of Intel Corporation. Microsoft is a registered trademark of icrosoft Corporation.

The CP/M Plus Operating System Command Summary was prepared using the Digital Research TEX™Text Formatter and printed in the United States of America.

****
* First Edition: March 1984 *
****

Table of Contents

How to Enter a CP/M Plus Command	1
CP/M Plus File Specifications	1
Command Summary Conventions	2
Control Characters	3
COPYSYS	5
DATE	6
DEVICE	7
DIR	10
DUMP	15
ED	16
ERASE	19
GENCOM	20
GET	22
HELP	24
HEXCOM	25
INITDIR	26
LIB	27
LINK	30
MAC	33
PATCH	35
PIP	36
PUT	39
RENAME	41
RMAC	43
SAVE	44

Table of Contents
(continued)

SET	46
SETDEF	51
SHOW	53
SID	55
SUBMIT	59
TYPE	61
USER	62
XREF	63

HOW TO ENTER A CP/M PLUS COMMAND

To give CP/M Plus a command, type a complete command line following the CP/M Plus system prompt, A>. A CP/M PLUS command line consists of a command, an optional command tall, and a RETURN or ENTER keystroke. The command is the name of a program to run. An optional command tail can consist of a drive letter followed by a colon, one or more file names, and some options. To complete the command you must press the RETURN or ENTER key.

A>**COMMAND** <RET>

CP/M PLUS FILE SPECIFICATIONS

CP/M Plus identifies every file by its complete name or tile specification. A file specification is any valid combination of the drive specification, f ilename, f iletype, and password, all separated by their appropriate delimiters. A drive letter must be followed by a colon. A filetype must be preceded by a period. A password must be preceded by a semicolon. The term filespec is an abbreviation for file specification.

This summary uses the following symbols to designate the parts of a filespec.

Symbol	Meaning
d:	Represents the optional drive specif ication, which can be any single alphabetic character in the range A through P, followed by a colon.
filename	Represents the required filename, which can be from 1 to 8 alphanumeric characters.

Symbol	Meaning
.typ	Represents the optional filetype, which can be from 0 to 3 alphanumeric characters; separated from the filename by a period. A period does not precede the filetype when the filetype is named alone in the text of this command summary.
;password	The optional password, which can be from 0 to 8 alphanumeric characters; separated from the filetype by a semicolon.

Valid combinations of the elements of a filespec are shown below:

- filename
- d:filename
- filename.typ
- d:filename.typ
- filename;password
- d:filename;password
- filename.typ;password
- d:filename.typ;password

COMMAND SUMMARY CONVENTIONS

The command summary alphabetically lists each CP/M Plus command using the following special symbols to define command syntax:

Symbol	meaning
{ }	optional item
|	'or' bar; separates choices when only one option can be used at a time
n	number
CTRL	control key
0	option or an option list
<RET>	RETURN or ENTER key

Symbol	Meaning
RW	Read/write
RO	Read/Only
SYS	System attribute; file does not appear when directory is displayed by DIR command
DIR	Directory attribute; file appears in response to DIR command
...	means the element can be repeated as many times as you want
*	wildcard; replaces all or part of a filename and/or filetype; must be last character in filename or filetype
?	wildcard; replaces any single character in the same position of a tilename and/or filetype
[]	type square brackets to enclose an option list

Certain CP/M Plus commands can be modified by options added to the command or file specification. Options are enclosed in square brackets. The right-hand square bracket is optional. Only one or two characters of the option word are necessary to specify an option. Generally, options can be grouped together, separated by commas or spaces in the square brackets. This does not apply to options that contradict each other. Sometimes the whole command tail is optional.

CONTROL CHARACTERS

The following list presents CP/M Plus control characters and their functions.

Control Character	Function
CTRL-A	Moves cursor one character to the left. Works only if your computer has bank-switched memory.
CTRL-B	Moves cursor from beginning to end of command line and back without affecting command.

CONTROL CHARACTERS (continued)

Control Character	Function
CTRL-C	Stops executing program when entered at the system prompt or after CTRL-S.
CTRL-E	Forces a physical return without sending command to CP/M Plus.
CTRL-F	Moves cursor one character to the right. Banked system only.
CTRL-G	Deletes character at current cursor position if in the middle of a line. Banked system only.
CTRL-H	Delete character to the left of cursor.
CTRL-I	Same as the TAB key.
CTRL-J	Moves cursor to the left of the command line and sends command CP/M Plus. Line feed; same effect as RETURN.
CTRL-K	Deletes character at cursor and all characters to the right.
CTRL-M	Same as RETURN.
CTRL-P	Echoes console output to the list device.
CTRL-Q	Restarts screen scrolling.
CTRL-R	Retypes the characters to the let t of the cursor on a new line; updates the command line buffer.
CTRL-S	Stops screen scrolling.
CTRL-U	Updates the command line but ter to contain the characters to the left of the cursor; deletes current line.
CTRL-W	Recalls previous command line if current line is empty; otherwise moves cursor to end of line. CTRL-J,-M,-R,-U and RETURN update the command line better for recall with CTRL-W. Banked system only.
CTRL-X	Deletes all characters to the left of the cursor.

COPYSYS

Syntax:

COPYSYS

Purpose:

COPYSYS copies the CP/M Plus system from a CP/M Plus system diskette to another diskette. The new diskette must have the same format as the original system diskette.

Examaple:

A>COPYSYS

DATE

Syntax:

DATE
DATE C
DATE CONTINUOUS
DATE time-specification
DATE SET

Purpose:

The DATE command lets you display and set the date and time of day.

Examples:

A>DATE

Displays the current date and time.

A>DATE C

Displays the date and time continuously.

A>DATE 08/14/82 10:30:0

Sets the date and time.

A>DATE SET

Prompts for date and time entries.

DEVICE

Syntax:

```
DEVICE
DEVICE NAMES
DEVICE VALUES
DEVICE logical-de v {XON|NOXON|baud-rate},
DEVICE physical-dev {XON|NOXON|baud-rate}
DEVICE logical-dev=physical-dev {option}
        {,physical-dev {option},...}
DEVICE logical-dev = NULL
DEVICE CONSOLE {PAGE}
DEVICE CONSOLE [COLUMNS=n,LINES=n}
```

Purpose:

DEVICE displays current logical device assignments and physical device names.

Examples:

A>DEVICE

Displays the physical deltices and current assignments of the logical device in the system.

A>DEVICE NAMES

Lists the physical devices with a summary of the device characteristics.

A>DEVICE VALUES

Displays the current logical device assignments.

A>DEVICE CRT

Displays the attributes of the physical device CRT.

A>DEVICE CON

Displays the assignment of the logical device CON:

DEVICE (continued)

Purpose:

DEVICE assigns logical devices to peripheral devices attached to the computer, and sets the communications protocol and speed of a peripheral device. Note that the physical devices available can differ from system to system. Consult your hardware manufacturer 's documentation for valid device names. An explanation of the valid device options follows:

DEVICE Options:

Option	Explanation
XON	Sets the device to the XDN/XOFF communications protocol.
NOXON	Indicates no protocol; the computer sends data to the device whether or not the device is ready to receive it.
baud-rate	Is the speed of the device. T h e system accepts the following haud rates:

50	75	110	134
150	300	600	1200
1800	2400	3600	4800
7200	9600	19200	

Examples:

A>DEVICE CONOUT:=LPT,CRT

Assigns the system console output (CONOUT:) to the printer (LPT:) and the creen (CRT:).

A>DEVICE AUXIN:=CRT2 [XON,96001

Assigns the auxiliary logical input device (AUXIN:) to the physical device CRT2: using protocol XON/XOFF, and sets the transmission rate for the device at 9600.

A>DEVICE LST:=NULL

Disconnects the list output logical device (LST:).

A>DEVICE LPT [XON,9600]
Sets the XON/XOFF protocol for the physical device LPT: and sets the transmission speed at 9600.

DEVICE (continued)

Purpose:

DEVICE displays or sets the current console screen size.

Examples:

A>DEVICE CONSOLE [PAGE]

Displays the current console page width in columns and length in lines.

A>DEVICE CONSOLE [COLUMNS=40,LINES=16]

Sets the screen size to 40 columns and 16 lines.

DIR
Syntax:

DIR
DIR d:
DIR filespec

DIRS
DIRS d:
DIRS filespec

DIR d: options
DIR filespec,... filespec options

Purpose:

The DIR and DIRS commands display the names of files cataloged in the directory of an on-line disk. DIR lists the names of files in the current user number that have the Directory (DIR) attribute. DIRS lists the names of files in the current directory that have the system (SYS) attribute. DIR and DIRS accept the * and ? wildcards in the file specification.

The DIR command with options displays the names of files and the characteristics associated with the files.

DIR and DIRS are built-in utilities. DIR with options is a transient utility and must be loaded into memory from the disk.

Examples:

A>DIR

Displays all files in user 0 on drive A that have the Directory attribute.

A>DIR B:

Displays all DIR files in user 0 on drive B.

2A>DIR C:ZIPPY.DAT

Displays the name ZIPPY.DAT if the file is in user 2 on drive C.

4A>DIR *.BAS

Displays all DIR files with filetype BAS in user 4 on drive A.

DIR (continued)

B3>DIR X*.C?D

Displays all DIR files in user 3 on drive B whose f ilename begins with the letter X, and whose three-character filetype contains the first character C and last character D.

A>DIRS

Displays all files for user 0 on drive A that have the system (SYS) attribute.

A>DIRS *.COM

Displays all SYS files with filetype COM on drive A in user 0. A command (COM) file in user 0 with the system attribute can be accessed from any user number on that drive, and from any drive in the search chain (see SETDEF).

Purpose:

The DIR command with options is an enhanced version of the DIR built-in command and displays your files in a variety of ways. DIR can search for files on any or all drives, for any or all user numbers. One or two letters is sufficient to identify an option. You need not type the right square bracket.

DIR Options:

Option	Function
ATT	Displays the user-definable file attribute.
DATE	Displays date and time stamps of files.
DIR	Displays only files that have the DIR attribute.
DRIVE=ALL	Displays files on all on-line drives.
DRIVE=(A,B,C,...,P)	Displays files on the drives specified.

DI R (continued)

Option Function

DRIVE=D
Displays files on the dr ive specified by d.

EXCLUDE
Displays files that DO NOT MATCH the files specified in the command line.

FF Sends an initial form feed to the printer device if the printer has been activated by CTRL-P.

FULL
Shows the name, size, number of 128-byte records, and attributes of the files. If there is a directory label on the drive, DIR shows the password protection mode and the time stamps . If there is no directory label, DIR displays two file entries on a line, omitting the password and time stamp columns. The display is alphabetically sorted. (See SET for a description of f ile attributes, directory labels, passwords and protection modes.)

LENGTH=N
Displays n lines of printer output before inserting a table heading. n is a number between 5 and 65536.

MESSAGE
Displays the names of drives and user numbers DIR is searching.

NOPAGE Continuously scrolls information by on the screen.

NOSORT
Displays f iles in the order it finds them on the disk.

RO
Displays only the files that have the Read/Only attribute.

RW
Displays only the files that are set to Read/Write.

SIZE

Displays the filename and size in kilobytes (1024 bytes).

DIR (continued)

Option	Function

SYS
Displays only the f iles that have the SYS attribute.

USER=ALL
Displays all f iles in all user numbers for the default or specified drive.

USER=N Displays the f iles in the user number specified by n.

USER=(O,l,...,15)
Displays f iles under the user numbers specified.

Examples:

A>DIR C: [FULL]

Displays full set of characteristics for all files in user 0 on drive C.

A>DIR C: [DATE]

Lists the f iles on drive C and their dates.

A>DIR D: [RW,SYS]

Displays all files in user 0 on drive D with Read/Write and System attributes.

3A>DIR [USER=ALL, DRIVE=ALL)

Displays all the files in all user numbers (0-15) in all on-line drives.

B6>DIR [EXCLUDE] *.DAT

Lists all the files on drive B in user 6 that do not have a filetype of DAT.

3B>DIR [SIZE] *.PLI *.CON *.ASH

Displays all the tiles of type PLI, COM, and ASM in user 3 on drive B in size display format.

DIR (continued)

A>DIR (DRIVB=ALL USER=ALL] TESTFILE.BOB

DIR displays the filename TESTFILE.BOB if it is found on any drive in any user number.

A>DIR [SIZE,RW] D:

DIR lists each Read/Write file that resides on Drive D, with its size in kilobytes, Note that D: is equivalent to D:*.

DUMP

Syntax:

DUMP filespec

Purpose:

DUMP displays the contents of a file in and ASCII format.

Example:

A>DUMP ABC.TEX

ED

Syntax:

ED
ED input-filespec
ED input-filespec {d: | output-filespec}

Purpose:

Character file editor. To redirect or rename the new version of the file specify the destination drive or destination filespec.

ED Command Summary:

Command Action

nA
append n lines from original file to memory buffer

OA
append file until buffer is one half full

#A
append file until buffer is full (or end of file)

B, -B
move CP to the beginning (B) or bottom (-B) of buffer

nC, -nC
move CP n characters forward (C) or back (-C) through buffer

nD, -nD
delete n characters before (-D) or from (D) the CP

E save new file and return to CP/M Plus

Fstring-Z
find character string

H
save new file, reedit, use new file as original file

I<RET>
enter insert mode

Istring^Z
insert string at CP

ED (continued)

Command	Action
Jsearch-str^Zins_str^Zdel_to_str	juxtapose strings
nK, -nK	delete (kill) n lines from the CP
nL, -nL, OL	move CP n lines
rMcommands	execute commands n times
n, -n	move CP n lines and display that line
n:	move to line n
:ncommand	execute command through line n
Nstring^Z	extended find string
O	return to original file
nP, -nP	move CP 23 lines forward and display 23 lines at console
Q	abandon new file, return to CP/M Plus
R^Z	read X$$$$$$$.LIB file into buffer
Rfilespec^z	read filespec into buffer
Sdelete-string^Zinsert-string	substitute string
nT, -nT, OT	type n lines
U, -u	uppercase translation
V, -V	line numbering on/off
OV	display free buffer space

ED (continued)

Command Action

nW
write n lines to new file

OW
write until buffer is half empty

nX
write or append n lines to X$$$$$$$.LIB

nxfilespec-z
write n lines to filespec; append if previous xcommand applied to same

0x^Z file

delete file X$$$$$$$.LIB

Oxfilespec^Z
delete filespec

nZ
wait n seconds

Note: CP points to the current character being referenced in the edit buffer. Use CTRL-Z to separate multiple commands on the same line. Your screen displays ^Z.

ERASE

Syntax:

ERASE
ERASE filespec
ERASE filespec [CONFIRM]

Purpose:

The ERASE command removes one or more files from the directory of a disk. Wildcard characters are accepted in the filespec. Directory and data space are automatically reclaimed for later use by another file. The ERASE command can be abbreviated to ERA.

[CONFIRM] option informs the system to prompt for verification before erasing each file that matches the filespec. CONFIRM can be abbreviated to C.

Examples:

A>ERASE X.PAS

Removes the file X.PAS from the disk in drive A.

A>ERA *.PRN

ERASE *.PRN (Y/N) ?Y

All files with the filetype PRN are removed from the disk in drive A.

B>ERA A:MY*.* [CONFIR14]

Each file on drive A with a filename that begins with MY is displayed with a question mark for confirmation. Type Y to erase the file displayed, N to keep the file.

A>ERA B:*.*

ERASE B:*.* (Y/N) ?Y

All files on drive B are removed from the disk.

GENCOM

Syntax:

GENCOM COM-Eilespec RSX-filespec... RSX-Eilespec
{[LOADER | SCB=(Qffset,value)]}
GENCOM RSX-filespec ... RSX-filespec
{[NULL | SCB=(Offset,value)l}
GENCOM filename
GENCOM filename [SCB=(offset,value)]

Purpose:

The GENCOM command attaches RSX files to a COM file, or creates a dummy COM file containing only RSXS. It can also restore a previously GENCOMed file to the original COM file without the header and RSXS, add or replace RSXs in already GENCOMed f iles, and attach header records to COM files without RSXS.

GENCOM Options:

Option Function

LOADER
Sets a flag to keep the program loader active.

NULL
Indicates that only RSX files are specified. GENCOM creates a dummy COM file for the RSX files. The output COM filename is taken from the filename of the first RSX- filespec.

SCB=(offset,value)
Sets the System Control Block from the program by using the hex values specified by (offset,value).

Examples:

A>GENCOM HYPROG PROGI PROG2

Generates a new COM file MYPROG.CDM with attached RSX's PROG1 and PROG2.

GENCOM (continued)

A>GENCOM PROG1 PROG2 [NULL]

Creates a COM file PROGI.CDM with RSX's PROGI and PROG2.

A>GENCON HYPROG

GENCOM takes MYPROG.COM, strips off the header and deletes all attached RSX's to restore it to its original COM format.

A>GENCO14 MYPROG PROGI PROG2

GENCOM looks at the already-GENCOMed file MYPROG.COM to see if PROG1.RSX and PROG2.RSX are already attached RSX files in the module. If either one is already attached, GENCOM replaces it with the new RSX module. Otherwise, GENCOM appends the specified RSX files to the COM file.

GET

Syntax:

GET {CONSOLE INPUT FROM} FILE filespec options

GET {CONSOLE INPUT FROM} CONSOLE

Purpose:

GET directs the system to take console input from a file for the next system command or user program entered at the console.

Console input is taken from a file until the program terminates. If the file is exhausted before program input is terminated, the program looks for subsequent input from the console. If the program terminates before exhausting all its input, the system reverts back to the console for console input.

GET Options:

Option	Function
ECHO	Specifies that input is echoed to the console. This is the default option.
NO ECHO	Specifies that file input is not echoed to the console. The program output and the system prompts are not affected by this option and are still echoed to the console.
SYSTEM	Spec if ies that the system immediately go to the specified file for console input. The system reverts to the console for input when it reaches the end of file. You can redirect the system to the console for console input with the GET CONSOLE INPUT FROM CONSOLE command as a command line in the input file.

GET(continued)

Examples:

A>GET FILE XINPUT

A>MYPROG

Tells the system to activate the GET utility. Because SYSTEM is not specified, the system reads the next input line from the console and executes MYPROG. if MYPROG program requires console input, it is taken from the file XINPUT. When MYPROG terminates, the system reverts back to the console for console input.

A>GET FILE XIN2 [SYSTEM]

Immediately directs the system to get subsequent console input from file XIN2 because it includes the SYSTEM option. The system reverts back to the console for console input when it reaches the end of file in XIN2. Or XIN2 can redirect the system back to the console if it contains a GET CONSOLE command.

A>GET CONSOLE

Tells the system to get console input from the console. This command can be used in a file (previously specified in a GET FILE command), which is already being read by the system for console input. It is used to redirect the console input back to the console before the end of file is reached.

HELP

Syntax:

HELP
HELP topic
HELP topic subtopic
HELP topic [NOPAGE]
HELP topic subtopicl...subtopic8

HELP>topic
HELP>.subtopic

Purpose:

HELP displays a list of to-pics and provides summarized information for CP/M Plus commands.

Typing HELP topic displays information about that topic. Typing HELP topic subtopic displays information about that subtopics

One or two letters is enough to identify the topics. After HELP displays information for your topic, it displays the special prompt HELP> on your screen, followed by a list of subtopics.

- Enter ? to display list of main topics.
- Enter a period and subtopic name to access subtopics.
- Enter a period to redisplay what you just read.
- Press RETURN to return to the CP/M Plus system prompt.
- [NOPAGE) option disables the 24 lines per page console display.
- Press any key to exit a display and return to the HELP> prompt.

Examples:

A>HELP
A>HELP DATE
A>HELP DIR OPTIONS
A>HELP>.OPTIONS
HELP>SET
HELP>SET PASSWORD
HELP>.PASSWORD
HELP>.
HELP><RET>

HEXCOM

Syntax:

HEXCOM filename

Purpose:

The HEXCOM Command generates a command file (filetype COM) from a HEX input file. it names the output tile with the same filename as the input file but with filetype COM. HEXCOM always looks for a file with filetype HEX.

Example:

A>HEXCOM B:PROGRAM

Generates a command file PROGRAM.COM from the input hex file PROGRAM.HEX on drive B:.

INITDIR

Syntax:

INITDIR d:

Purpose:

The INITDIR command initializes a disk directory to allow date and time stamping of files on that disk. INITDIR can also recover time/date directory space.

Example:

A>INITDIR C:

INITDIR WILL ACTIVATE TIME STAMPS FOR SPECIFIED DRIVE.
Do you want to re-format the directory C: (Y/N)?Y

LIB

Syntax:

LIB fiiespec options
LIB filespec toptionsl=filespec <modifier>
f,filespec<modifier> ...

Purpose:

A library is a f ile that contains a collection of object modules. Use the LIB utility to create libraries, and to append, replace, select, or delete modules from an existing library. Use LIB to obtain information about the contents of library files.

LIB creates and maintains library files that contain object modules in Microsoft REL file format. These modules are produced by the Digital Research relocatable macro-assembler program, RMAC, or other language translator that produces modules in Microsoft REL file format.

You can use LINK-80 to link the object modules contained in a library to other object files. LINK-80 automatically selects from the library only those modules needed by the program being linked, and then forms an executable file with a filetype of Com.

LIB Options:

Option	Function
I	The INDEX option creates an indexed library file of type IRL. LINK-80 searches faster on indexed libraries than on nonindexed libraries.
M	The MODULE option displays module names.
P	The PUBLICS option displays module names and the public variables for the new library file.
D	The DUMP option displays the contents of object modules in ASCII form.

LIB (continued)

Use modif iers in the command line to instruct LIB to delete, replace, or select modules in a library f ile. Angle brackets enclose the modules to be deleted or replaced. Parentheses enclose the modules to be selected.

LIB Modifiers:

Modifier	Meaning
Delete	<module=>
Replace	<module=filename.REL> If module name and filename are the same, you can use the following shorthand;
	<filename>
Select	(modFIRST-modLAST,modl...modN)

Examples:

A>LIB TEST4[P]

Displays all modules and publics in TEST4.REL.

A>LIB TEST5[P]=FILE1,FILE2

Creates TEST5.REL from FILEL.REL and FILE2.REL and displays all modules and publics in TEST5.REL.

A>LIB TEST=TESTI(MOD1,MOD4),TEST2(CI-C4, C6)

Creates a library file TEST.REL from modules in two source files. TEST1.REL contributes MOD1 and MOD4. LIB extracts modules Cl, C4, and all the modules located between them, as well as module C6 from TEST2.REL.

A>LIB FILE2=FILE3<MODA=>

Creates FILE2. REL from FILE3. REL, omitting MODA, which is a module in FILE3.REL.

A>LIB FILE6=FILE5<MODA=FILEB.REL>

Creates FILE6.REL from FILE5.REL; FILEB.REL replaces MODA.

LIB (continued)

A>LIB FILE6=FILE5<THISNA14E>

Module THISNAME is in FILE5.REL. When LIB creates FILE6.REL from FILE5.REL, the file THISNAME.REL replaces the similarly named module THISNAME.

A>LIB FILE1[I]=B:FILE2(PLOTS,FIND, SEARCH-DISPLAY)

Creates FILEL.IRL on drive A from the selected modules PLOTS, FIND, and modules SEARCH through the module DISPLAY, in FILE2.REL on drive B.

LINK

Syntax:

LINK filespec toptions)
LINK filespec toptionsl,...Jilespec [options!
LINK filespec joptionsj=tilespec [options],...

Purpose:

LINK combines relocatable object modules such as those produced by RMAC and PL/I- 80 into a COM file ready for execution. Relocatable files can contain external references and publics. Reiocatable files can reference modules in library files. LINK searches the library f iles and includes the referenced modules in the output file. See the Programmer 's Utilities Guide for the CP/Ms Family of Operating Systems for a complete description of LINK-80.

Use LINK ootion switches to control execution parameters. Link options follow the file specifications and are enclosed within square brackets. Multiple switches are separated by commas.

LINK-80 Options:

Option	Function
A	Additional memory; reduces buffer space and writes temporary data to disk.
B	BIOS link in Danked CP/M Plus system. Aligns data segment on page boundary. Puts length of code segment in header. Defaults to SPR filetype.
Dhbhh	Data origin; sets memory origin for common and data area.
Gn	Go; set start address to label n.
Lbhhh	Load; change defauit load address of module to hhhn. Default 0100H.
Mhhhh	Memory size; Define free memory requirements for MP/M modules.
ML	No listing of symbol taoie at console.
MR	No symbol table file.

LINK (continued)

Option	Function
OC	Output COM command file. Default.
OP	Output PRL page relocatable f ile for execution under MP/M in relocatable segment.
OR	Output RSP resident system process file for execution under MP/M.
Os	Output SPR system page relocatable file for execution under MP/M.
Phhhh	Program origin; changes default program origin address to hhhh. Default is 0100H.
Q	Lists symbols with leading question mark.
Ss	Search preceding file as a library.
$Cd	estination of console messages. d an be X (console), Y (printer), or Z (zero output) . Default is X.
$Id	Source of intermediate files; d is disk drive A-P. Default is current drive.
$Ld	Source of library files; d is disk drive A-P. Default is current drive.
$Od	Destination of object file; d can be Z or disk drive A-P. Default is to same drive as first file in the LINK-80 command.
$Sd	Destination of symbol file; d can be Y or Z or disk drive A-P. Default is to same drive as first file in LINK-80 command.

LINK (continued)

Examples:

A>LINK B:KYFILECNR]

LINK-80 on drive A uses as input MYFILE.REL on drive B and produces the executable machine code file MYFILE.COM on drive B. The ENRI option specifies no symbol table file.

A>LINK al,m2.m3

LINK-80 combines the separately compiled files ml, m2, and m3, resolves their external references, and produces the executable machine code file ml.COM.

A>LINK m=xl,n2,n3

LINK-80 combines the separately compiled f iles ml, m2, and m3 and produces the executable machine code file M.COM.

A>LINK MYFILE,FILE5[sl

The [s] option tells LINK-80 to search FILE5 as 'a library. LINK-80 combines MYFILE.REL with the referenced subroutines contained in FILE5.REL on the default drive A and produces MYFILE.COM on drive A.

MAC

Syntax:

MAC filename I$optionsi

Purpose:

MAC, the CP/M Plus macro assembler, reads assembly language statements from a file of type ASM, assembles the statements, and produces three output f iles with the input filename and filetypes of HEX, PRN, and SYM. Filename.HEX contains Intel hexadecimal format object code. Filename.PRN contains an annotated source listing that you can print or examine at the console. Filename.SYM contains a sorted list of symbols defined in the program.

Use options to direct the input and output of MAC. Use a letter with the option to indicate the source and destination drives, and console, printer, or zero output. Valid drive names are A through 0. X, P, and Z specify console, printer, and zero output, respectively.

MM Options:

Option	Function
A	source drive for ASM file (A-0)
H	destination drive for HEX file (A-0, Z)
L	source drive for macrolibrary LIB files called by the MACLIB statement.
p	destination drive for PRN file (A-0, X, P, Z)
S	destination drive for SYM file
+L	lists input lines read from macrolibrary LIB files
-L	suppresses listing (default)
+M	lists all macro lines because they are processed during assembly

MAC (continued)

MAC Options:

Option	Function
-M	suppresses all macro lines because they are read during assembly
*M	lists only hex generated by macro expansions
+Q	lists all LOCAL symbols in the symbol list
-Q	suppresses all LOCAL symbols in the symbol list (default)
+S	appends symbol file to print file
-S	suppresses creation of symbol file
+1	produces a pass 1 listing for macro debugging in PRN file
-1	suppress listing on pass I (default)

Examples:

A>MAC SAMPLE

MAC assembles the file SAMPLE.ASM.

A>MAC SAMPLE $PB AA HB SX -M

In this example, the option list directs the PRN file to drive B:, obtains the ASM file from drive A:, directs the HEX file to drive B:, the SYM file to the console, and suppresses all macro lines during assembly.

PATCH

Syntax:

PATCH filenamef.typj n

Purpose:

The PATCH command displays or installs patch number n to the CP/M Plus system or command f iles. The patch number n must be between 1 and 32 inclusive.

Example:

A>PATCH SHOW 2

Patches the SHOW.COM system file with patch number 2.

PIP

Syntax:

Destination = Source

PIP
PIP d:I[Gn]@=filespecloptionsl
PIP filespeci[Gnll=fil'especloptionslf,
PIP filespeci[Gnllldevice=filespectoptionsfidevice

Purpose:

The file copy program PIP copies files, combines files, and transfers files between disks, printers, consoles, or other devices attached to your computer. The first filespec is the destination. The second filespec is the source. Use two or more source filespecs separated by commas to combine two or more files into one file. lo] is any combination of the available options. The [Gn] option in the destination filespec tells PIP to copy your file to that user number. PIP with no command tail displays an * prompt and awaits your series of commands, entered and processed one line at a time. The source or destination can optionally be any CP/M Plus logical device.

Examples;

COPY A FILE FROM ONE DISK TO ANOTHER

A>PIP b:=a:draft.txt
A>PIP b:draft.txt = a:
B3>PIP myfile.dat=A:[G9]
A9>PIP B:[G3]=myfile.dat

COPY A FILE AND RENAME IT

A5>PIP newdraft.txt=oldraft.txt
C8>PIP b:newdraft.txt=a:oldraft.txt

COPY MULTIPLE FILES

A>PIP b:=draft.*
A>PIP b:=*.*
B>PIP b:=c:.*.*
C>PIP a:=*.com[wr]
B>PIP a:[g3l=c:*.*

PIP (continued)

COMBINE MULTIPLE FILES
A>PIP b:new.dat=filel.dat,file2.dat

COPY, RENAME AND PLACE IN USER 1
A>PIP newdraft.txt[gl]=oldraft.txt

COPY, RENAME AND GET FROM USER 1
A>PIP newdraft.txt=oldraft.txt[gll

COPY TO/FROM LOGICAL DEVICES
A>PIP b:funfile.sue=con:
A>PIP lst:=con:
A>PIP lst:=b:draft.txt[t8]
A>PIP prn:=b:draft.txt

PIP Options:

Option	Function
A	Archive. Copy only files that have been changed since the last copy.
C	Conf irm. PIP prompts for confirmation before each file copy.
Dn	Delete any characters past column n.
E	Echo transfer to console.
F	Filter form-feeds from source data.
Gn	Get from or go to user n.
H	Test for valid Hex format.
I	Ignore :00 Hex data records and test for valid Hex format.
L	Translate uppercase to lowercase.
N	Number output lines
0	0bject file transfer , ^Z ignored.
Pn	Set page length to n. (Default n=60).

PIP (continued)

PIP Options:

Option	Function
QS^Z	Quit copying from source at string s.
R	Read files that have been set to SYStem.
Ss^Z	Start copying from the source at the string s.
Tn	Expand tabs to n spaces.
U	Translate lowercase to uppercase.
V	Verify that data has been written correctly.
W	Write over Read/Only files without console query.
Z	Zero the parity bit.

All options except C, Gn, K, 0, R, V, and W force an ASCII file transfer, character by character, terminated by a ^Z.

PUT

Syntax:

PUT CONSOLE {OUTPUT TO} FILE filespec {option}
PUT PRINTER {OUTPUT TO} FILE filespec {option}
PUT CONSOLE {OUTPUT TO} CONSOLE
PUT PRINTER {OUTPUT TO} PRINTER

Purpose:

PUT puts console or printer output to a file for the next command entered at the console, until the program terminates. Then console output reverts to the console. Printer output is directed to a file until the program terminates. Then printer output is put back to the printer.

PUT with the SYSTEM option directs all subsequent console/printer output to the specified file. This option terminates when you enter the PUT CONSOLE or PUT PRINTER command.

PUT Options;

Option	Function
ECHO	Specifies that output is echoed to the console. This is the default option when you direct console output to a file.
NO ECHO	Specifies that file output is not echoed to the console. NO ECHO is the default for the PUT PRINTER command.
FILTER	Specifies filtering of control characters, which means that control characters are translated to printable characters. For example, an ESCape character is translated to ^[.
NO FILTER	Means that PUT does not translate control characters. This is the default option.

PUT (continued)

PUT Options:

Option	Function
SYSTEM	Specifies that system output and program output are written to the file specified by filespec. Output is written to the file until a subsequent PUT CONSOLE command redirects console output back to the console.

Examples:

A>PUT CONSOLE OUTPUT TO FILE XOUT [ECHO]

Directs console output to file XOUT with the output echoed to the console.

A>PUT PRINTER OUTPUT TO FILE XOUT

A>MYPROG

Directs the printer output of program MYPROG to file XOUT. The output is not echoed to the printer.

A>PUT PRINTER OUTPUT TO FILE XOUT2 [ECHO,SYSTEMI

Directs all printer output to file XOUT2 as well as to the printer (with ECHO option), and the PUT is in effect until you enter a PUT PRINTER OUTPUT TO PRINTER command.

A>PUT CONSOLE OUTPUT TO CONSOLE

Directs console output back to the console.

A>PUT PRINTER OUTPUT TO PRINTER

Directs printer output back to the printer.

RENAME

Syntax:

RENAME
RENAME new-filespec=old-filespec

Purpose:

RENAME iets you change the name of a file in the directory of a disk. To change several filenames in one command use the * or ? wildcards in the file specifications. You can abbreviate the RENAME command to REN. REN prompts you for input.

Examples:

A>RENAME NEWFILE.BA.S=OLDFILE.BAS

The file OLDFILE.BAS changes to NEWFILE.BAS on drive A.

A>RENAME

The system prompts for the following filespecs:

```
Enter New Name:X.PRN
Enter Old Name:Y.PRN
Y       PRN=X          PRN
A>
```

File X.PRN is renamed to Y.PRN on drive A.

B>REN A:PRINTS.NEW=PRINCE.NEW

The file PRINCE.NEW on drive A changes to PRINTS.NEW on drive A.

A>RENAME S*.TEX=A*.TEX

The above command renames all the files matching A*.TEX to files with filenames S*.TEX.

RENAME (continued)

A>REN B:NEWLIST=B:OLDLIST

The f ile OLDLIST changes to NEWLIST on drive B. Because the second drive specifier, B:, is implied by the first one, it is unnecessary in this example. The command line above has the same ef f ect as the following:

A>REN B:NEWLIST=OLDLIST

or

A>REN NEWLIST=B:OLDLIST

RMAC

Syntax:

RMAC filespec toptions)

Purpose:

RMAC, a relocatable macro assembler, assembles ASM files into REL files that you can link to create COM files.

RMAC options specify the destination of the output files. Replace d with the destination drive letter for the output files.

RMAC Options (d=output option parameter):

Rd drive for REL file (A-0, Z)
Sd drive for SYM file (A-0, X, P, Z)
Pd drive for PRN file (A-0, X, P, Z)

The d parameter can have the following values:

A-0 specifies drive A-0
X means output to the console
P means output to the printer
Z means zero output

Example:

A>RMAC TEST $PX SB RB

Assembles the file TEST.ASM from drive A, sends the listing file (TEST.PRN) to the console, puts the symbol file (TEST.SYM) on drive B and puts the relocatable object file (TEST.REL) on drive B.

SAVE

Syntax:

SAVE

Purpose:

SAVE copies the contents of memory to a file. To use SAVE, first issue the SAVE command, then run your program which reads a file into memory. Your program exits to the SAVE utility which prompts you for a filespec to which it copies the contents of memory, and the beginning and ending address of the memory to be SAVED.

Example:

A>SAVE

Activates the SAVE utility. Now enter the name of the program which loads a file into memory.

A>SID dump.con

Next, execute the program.

#g0

When the program exits, SAVE intercepts the return to the system and prompts the user for the filespec and the bounds of memory to be SAVED.

SAVE Ver 3.0

SAVE(continued)

Enter file press RETURN t o exit):dump2.com

If file DUMP2.COM exists already, the system asks:

Delete dump2.com? Y

Then the system asks for the bounds of memory to be saved:

Beginning hex address: 100

Ending hex address: 400

The contents of memory from IOOH (Hexadecimal) to 400H is copied to file DUMP2.COM.

SET

Syntax:

SET [options]
SET d: [options]
SET filespec [options]
SET [option = modifier]
SET filespec [option = modifier]

Purpose:

SET initiates password protection and time stamping of files. It also sets ` the file and drive attributes Read/Write, Read/Only, DIR and SYS. It lets you label a disk and password protect the label. To enable time stamping of files, you must first run INITDIR to format the disk directory.

Examples:

SET Disk Label operations:

A>SET [NAME-DISKIOOI

Labels the disk on the default drive as DISK100.

A>SET [PASSWORD-SECRET]

Assigns SECRET to the disk label.

A>SET [PASSWORD-<RET>]

Nullifies the existing password.

SET Password Operations:

SET [PROTECT=ON]
SET [PROTECT=OFF)
SET filespec [PASSWORD=password]
SET filespec [PROTECT=READ]
SET filespec [PROTECT=WRITE]
SET filespec [PROTECT=DELETE)
SET filespec [PROTECT=NONE]
SET filespec [attribute-options]

SET(continued)

Password Protection Modifiers:

Modifier	Protection
READ	The password is required for reading, copying writing, deleting or renaming the file.
WRITE	The password is required for writing, deleting or renaming the file. You do not need a password to read the file.
DELETE	The password is only required for deleting or renaming the file .You do not need a password to read or modify the file.
NONE	No password exists for the file. If a password exists, this modifier can be used to delete the password.

SET File Attribute Options:

Option	Function
RO	Sets the file attribute to Read/Only.
RW	Sets the file attribute to Read/Write.
SYS	Sets the file attribute to SYS.
DIR	Sets the file attribute to DIR.
ARCHIVE=OFF	Means that the file has not been backed up (archived) .
ARCHIVE=ON	Means that the file has been backed up (archived) . The Archive attribute can be turned on by SET or by PIP when copying a group of files with the PIP [A] option. SHOW and DIR display the Archive option.
FL--ONIIOFF	Turns on or off the user- definable file attribute Fl.
F2=ONIOFF	Turns on or off the user- definable file attribute F2.

SET (continued)

SET File Attribute Options:

Option	Function
F3=ON|OFF	Turns on or off the user- definable file attribute F3.
F4=ON|OFF	Turns on or off the user- definable file attribute F4.

Examples:

A>SET [PROTECT=ON]

Turns on password protection for all the files on the disk. You must turn on password protection before you can assign passwords to files.

A>SET EPROTECT=OFF]

Disables password protection for the files on your disk.

A>SET MYFILE.TEX [PASSWORD=MYFIL]

MYFIL is the password assigned to file MYFILE.TEX.

B>SET *.TEX [PASSWORD-SECRET, PROTECT=WRITE]

Assigns the password SECRET to all the TEX files on drive B. Each TEX file is given a WRITE protect mode to prevent unauthorized editing.

A>SET MYFILE.TEX [RO SYS]

Sets MYFILE.TEX to Read-Only and SYStem.

SET Default password operation:

A>SET [DEFAULT=passwordl

Instructs the system to use a default password if you do not enter a password for a password-protected file.

SET (continued)

SET Time-stamp Operations:

Syntax:

SET {d:} [CREATE=ON|OFF]
SET {d:} [ACCESS=ON|OFF]
SET {d:} [UPDATE=ON|OFF]

Purpose:

The above SET commands allow YOU to keep a record of the time and date of file creation and upditer or of the last access update of your files.

Time and Date Stamp options:

Option	Function
[CREATE=ON]	Turns on CREATE time stamps on the disk in the default or specified drive. To record the creation time of a file, the CREATE option must be turned on before the file is created.
[ACCESS=ON]	Turns on ACCESS time stamps on the disk in the default or specified drive. ACCESS and CREATE options are mutually exclusive; only one can be in effect at a time. If you turn on the ACCESS time stamp on a disk that previously had CREATE time stamp, the CREATE time stamp is automatically turned off.
[UPDATE=ON]	Turns on UPDATE time stamps on the disk in the default or specified drive. UPDATE time stamps record the time the file was last modified.

SET (continued)

Examples:

```
A>SET [ACCESS=ON]
A>SET [CREATE=ON,UPDATE=ONI
```

SET Drive Operations:

Syntax:

```
SET {d:} [RO]
SET {d:} [RW]
```

Example:

```
A>SET B: [RO]
```

Sets drive B to Read/Only.

SETDEF

Syntax:

SETDEF
SETDEF [TEMPORARY=d:l
SETDEF d:i,d:l,d:i,d:ill
SETDEF [ORDER= (typ t,typl)]
SETDEF [DISPLAY I NO DISPLAY)
SETDEP [PAGE I NOPAGE]

Purpose:

SETDEF allows the user to display or define up to four drives for the program search order, the drive for temporary files, and the filetype search order. The SETDEF definitions affect only the loading of programs and/or execution of SUBMIT (SUB) files. SETDEF turns on/off the system Display and Console Page modes. When on, the system displays the location and name of programs loaded or SUBmit files executed, and stops after displaying one full console screen of information.

Examples:

A > SETDI&F

Displays current SETDEF parameters.

A>SETDRF [T RARY=C:J

Sets disk drive C as the drive to be used for temporary files.

A>SETDEF C:,*

Tells the system to search for a program on drive C, then, if not found, search for it on the default drive.

A>SETDEF [ORDER=(SUB,COM)l

Instructs the system to search for a SUB file to execute. If no SUB file is found, search for a COM file.

SETDEF (continued)

A>SETDEP [DISPLAY]

Turns on the system display mode. The system now displays the name and location of prog.rams loaded or submit files execute

A>SETDEP [NO DISPLAY]

Turns off the system Display mode.

SHOW

Syntax:

SHOW
SHOW d:
SHOW d: (SPACE]
SHOW d: (LABEL]
SHOW d: [USERS]
SHOW d: (DIR]
SHOW d: [DRIVE]

Purpose:

The SHOW command displays the following disk drive information:

o access mode and the amount of free disk space
o disk label
o current user number
o number of files for each user number on the disk
o number of free directory entries for the disk
o drive characteristics

Examples:

A>SHOW

A>SHOW [SPACE]

Instructs the system to display access mode and amount of space left on logged-in drives.

A>SHOW B:

Shows access mode for drive B and amount of space left on drive B.

A>SHOW B:[LABEL]

Displays label information for drive B.

SHOW (continued)

A>SHOW [USERS]

Displays the current user number and all the users on drive A and the corresponding number of files assigned to them.

A>SHOW C:[DIR]

Displays the number of free directory entries on drive C.

A>SHOW [DRIVE]

Displays the drive characteristics of drive A.

SID

Syntax:

SID [pgm-filespecl],{sym-filespec}

Purpose:

The SID symbolic debugger allows you to monitor and test programs developed for the 8080 microprocessor. SID supports real-time breakpoints, fully monitored execution, symbolic disassembly, assembly, and memory display and fill functions. SID can dynamically load SID utility programs to provide traceback and histogram facilities.

SID Commands:

Command	Meaning
As	Enter assembly language statements. s is the start address.
Cs{b{,d}}	Call to memory location from SID. s is the called address; b is the value of the BC register pair; d is the value of the DE register pair.
D{W}{s}{,f}	Display memory in hex and ASCII. W is a 16-bit word format, s is the start address, and f is the finish address.
Epgm-filespec {,sym-filespec}	Load program and symbol table for execution.
E*sym-filespec	Load a symbol table file.
Fs,f,d	Fill memory with constant value. s is the start address, f is the finish address, and d is an 8-bit data item.
G{p}{,a{,b}}	Begin Execution. p is a start address; a is a temporary breakpoint.

SID (continued)

SID Commands:

Command	Meaning
H	Displays all symbols with addresses in Hex.
H.a	Displays hex, decimal, and ASCII values of a where a is a symbolic expression.
Ha,b	Computes hex sum and difference of a and b where a and b are symbolic expressions.
Icomand tail	Input CCP command line.
L{-s}{,f}	List 8080 mnemonic instructions. s is the start address, and f is the finish address.
Ms,h,d	Move Memory Block. s is the start address, h is the high address of the block, and d is the destination start address.
P{p{,c}}	Pass point set, reset, and display. p is a permanent breakpoint address; c is initial value of pass counter.
Rfilespec{,d}	Read Code/Symbols. d is an offset to each address.
S{W}s	Set Memory Values. s is address where value is sent, and W is 16-bit word.
T{n{,c}}	Trace Program Execution. n is the number of program steps, and c is the utility entry address.
T{W}{n{,c}}	Trace without Call. W instructs SID not to trace subroutines, n is the number of program steps, and c is the utility entry address.

SID (continued)

SID Commands:

Command	Meaning
U{W}{n{,c}}	Monitor Execution without Trace. n is the number of program steps, c is the utility entry address, and W instructs SID not to trace subroutines.
V	Display the value of the next available location in memory (NEXT) , the next location after the largest f ile read in (MSZE) , the current value of the Program counter (PC) , and the address of the end-of-available memory (END).
Wfilespec,s,f	Write the contents of a contiguous block of memory to filespec. t is finish address.
X{f}{r}	Examine/alter CPU state. f is flag bit C, Z, M, E or I; r is register A, B, D, H, S or P.

Examples:

A>SID

CP/M Plus loads SID from drive A into memory. SID displays the # prompt when it is ready to accept commands.

A>B:SID SAKPLE.HEX

CP/M Plus loads SID and the program file SAMPLE.HEX into memory from drive B.

SID Utilities:

SID utilities, HIST.UTL and TRACE.UTL, are special programs that operate with SID to provide additional debugging facilities. The mechanisms for system initialization, data collection, and data display are described in the Symbolic Instruction Debugger Productivity Tool Reference Manual for the CP/M-80 Family of Operating Systems.

SID (continued)

The HIST utility creates a histogram (bar graph) showing the relative frequency of execution of code within selected program segments of the test program. The HIST utility allows you to monitor those sections of code that execute most frequently.

The TRACE utility obtains a backtrace of the instructions that led to a particular breakpoint address in a program under test. You can collect the addresses of up to 256 instructions between pass points in U or T modes.

SUBMIT

Syntax:

SUBMIT
SUBMIT filespec
SUBMIT filespec argument ... argument

Purpose:

The SUBMIT command lets you execute a group (batch) of commands from a SUBmit file (a file with filetype of SUB).

SUB files:

The SUB file can contain the following types of lines:

- any valid CP/M Plus command
- any valid CP/M Plus command with SUBMIT parameters ($O-$9)
- any data input line
- any program input line with parameters ($0 to $9)

The command line cannot exceed 135 characters.

The following lines illustrate the variety of lines which can be entered in a SUB file:

```
DIR
DIR *.BAK
MAC $1 $$$4
PIP LST:=$l.PRN[T$2 $3 $5]
DIR *.ASM
PIP
<B:=*.ASM
<CON:=DUMP.ASM

DIR B:
```

SUBMIT (continued)

Examples:

A>SUBMIT

SUBMIT prompts you for the name of the SUB file and any arguments.

A>SUBMIT SUBA

SUBMIT executes the commands found in the SUBA.SUB file.

A>SUBMIT AA ZZ SZ

SUBMIT executes the commands in AA.SUB, replacing all occurrences of $1 with the argument ZZ and all occurrences of $2 with SZ.

The PROFILE.SUB Start-up File:

Every time you power up or reset your computer, CP/M Plus looks for a special SUBMITfile named PROFILE.SUB to execute. If the file does not exist, CP/M Plus resumes normal operation. If the PROFILE.SUB file exists, the system executes the commands in the file. This file is convenient to use if you regularly execute a set of commands before you do your regular session on the computer.

TYPE

Syntax:

TYPE
TYPE filespec
TYPE filespec [PAGE]
TYPE filespec [NOPAGE]

Purpose:

The TYPE command displays the contents of an ASCII character file on your screen.

TYPE Options:

Option	Function
[PAGE]	Causes the console listing to be displayed in paged mode; that is, stop automatically after listing n lines of text, where n normally defaults to 24 lines per page.
[NOPAGE]	Turns off Console Page Mode and continuously displays a typed file on the screen.

Examples:

A>TYPE MYPROG.PLI

Displays the contents of the file MYPROG.PLI on your screen.

A>TYPE B:THISFILE [PAGE]

Displays the contents of the file THISFILE from drive B on your screen 24 lines at a time.

USER

Syntax:

USER
USER n

Purpose:

The USER command sets the current user number. The disk directory can be divided into distinct groups according to a User Number. User numbers range from 0 through 15.

Examples:

```
A>USER
Enter User#:5
5A>
```

The current user number is now 5 on drive A.

```
A>USER 3
3A>
```

This command changes the current user number to 3.

XREF

Syntax:

XREF {d:} filename {$P}

Purpose:

XREF provides a cross-reference summary of variable usage in a program. XREF requires the PRN and SYM files produced by MAC or RMAC for input to the program. The SYM and PRN f iles must have the same filename as the filename in the XREF command tail. XREF outputs a file of type XRF.

Examples:

A>XREF B:MYPROG

XREF operates on the files MYPROG. SYM and MYPROG. PRN on drive 3: , producing the file B:MYPROG.XRF.

A>XREF B:MYPROG $P

The $P option directs the output to the printer.