

QP/M 2.7

Features and Facilities

September, 1985

Created by
MICROCode Consulting
www.microcodeconsulting.com

CP/M 2.2 is a registered trademark of Digital Research, Incorporated. Z-80 is a registered trademark of Zilog, Incorporated. NSC-800 is a trademark of National Semiconductor Corporation. WordStar is a trademark of MicroPro International.

Copyright © 1985, 2002 by MICROCode Consulting. 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 MICROCode Consulting.

DISCLAIMER

MICROCode Consulting 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, MICROCode Consulting reserves the right to revise this publication and to make changes from time to time in the content hereof without obligation of MICROCode Consulting to notify any person of such revision or changes.

TABLE OF CONTENTS

A.	INTRODUCTION	4
B.	LINE EDITING	6
C.	BUILT-IN COMMANDS.....	7
C-1.	Introduction.....	7
C-2.	DFD [drive][:].....	9
C-3.	DFLT.....	10
C-4.	DFU [n].....	11
C-5.	DIR [afn].....	12
C-6.	ERA afn	13
C-7.	GET addr ufn	14
C-8.	GO [params].....	15
C-9.	JUMP addr [params].....	16
C-10.	LIST ufn.....	17
C-11.	REN ufnnew=ufnold.....	18
C-12.	SAVE n ufn [option].....	19
C-13.	TIME.....	20
C-14.	TOF.....	21
C-15.	TYPE ufn [option]	22
C-16.	USER [n].....	23
D.	QP/M UTILITIES.....	24
D-1.	D (2k).....	25
D-2.	D (4k).....	26
D-3.	INIT (Included with QBIOS package only)	28
D-4.	QBACKUP	29
D-5.	QINSTALL.....	30
D-6.	QPATCH.....	31
D-7.	QPIP.....	32
D-8.	QSTAMP/QSTAMPX/QSTAMPV	35
D-9.	QSTAT.....	36
D-10.	QSUB.....	41
D-10.1	GENERAL COMMANDS	41
D-10.2	NESTING	42
D-10.3	VARIABLES	42
D-10.4	SUBSTITUTION.....	43
D-10.5	PRIMARY COMMANDS.....	44
D-10.6	LABELS and JUMPS	45
D-10.7	CONDITIONALS.....	45
D-10.8	MODIFYING VARIABLES	46
D-10.9	EXAMPLE.....	46
D-11.	SETxx (Generated by TDCNFG program).....	48
D-12.	SYSGEN.....	49
D-13.	TDCNFG.....	50
E.	QCP CAPABILITIES.....	51
E-1.	QP/M SEARCH CAPABILITY	51
E-2.	QCP COMMAND SUPPORT.....	53

A. INTRODUCTION

QP/M is an optimized single-user operating system that was created using CP/M's BIOS structure and BDOS function calls as a root. All of the features of CP/M 2.2 are present; additional capability has been added for the three new features: time/date stamping, user search, and drive search.

Included in this chapter is a description of the QP/M commands available at the user level. (Programmers will find the QP/M application level documentation in part III, QP/M Interface Guide.) This manual is not intended as a tutorial on QP/M. If you are a novice to this subject, it is strongly recommended that you purchase one of the several introductory books available on CP/M.

The QCP (CCP replacement) portion of QP/M is a user-to-QDOS (BDOS replacement) interface having a number of useful transient commands, which are described in section C. For example, changing the user area (USER), listing the contents of a file on the terminal (TYPE), and displaying the directory contents of a drive (DIR) are all transient capabilities.

A line entered at the terminal can be edited during entry; this is described in section B.

Utilities provided with QP/M (depending upon the package purchased) are described in section D.

During normal operation, QCP will display the drive, user, and the QCP prompt. For example, if you are located on drive A user 1, you would see

```
A1>
```

As delivered, your QCP is configured to suppress display of user number when it is 0. The prompt when you are on drive C, user 0, would be

```
C>
```

When QCP gives the prompt 'du>' or 'd>' and QSUB is not active, the system is waiting for a command from the user. Transient commands can be directly executed at any time. Commands resident on disk are executable if they have the file extension (filename.EXT) of 'COM'.

A command-search hierarchy is implemented within QDOS; the QCP takes advantage of this feature when a command is entered as follows:

- the user's command is checked against the QCP-resident (transient) commands and executed immediately if a match is found UNLESS the command is preceded by a drive designation (i.e. TIME will execute the transient command whereas B:TIME will not.)
- failing that, the current user number on the current disk is scanned for the COM file; the COM file is loaded and executed if found
- failing that, a default user number (usually 0) on the current disk is scanned for the COM file UNLESS default user search has been disabled; the COM file is loaded and executed if found
- finally, failing that, the default user number on the default disk (usually A:) is scanned for the COM file UNLESS default drive search has been disabled; the COM file is loaded and executed if found or an error message (COMMAND?, where COMMAND was the user's command name) is printed

A more detailed description of the QDOS search procedure is presented in section E.

B. LINE EDITING

Pressing the keys shown below will determine what action should take place at the QP/M ">" prompt:

<DELETE>/<RUBOUT> - BACKSPACE

<CTRL> + <C> - REBOOT SYSTEM (if done at beginning of line)

+ <E> - END OF LINE

+ <H> - BACKSPACE

+ <J> - LINE FEED

+ <M> - CARRIAGE RETURN

+ <R> - RETYPE LINE

+ <U> - DELETE LINE

+ <X> - BACKSPACE to beginning of line

C. BUILT-IN COMMANDS

C-1. Introduction

The built-in commands of QP/M are an enhanced/modified version of ZCPR 1.0. Although most of the code was re-written to accommodate QP/M features, some of it has not been changed. Special thanks go to Richard Conn, Frank Wancho, Keith Peterson, and Ron Fowler who have unknowingly inspired this complete replacement for CP/M 2.2.

QP/M uses the file syntax of "DU:FILENAME.EXT" or "DU;FILENAME.EXT". DU is an optional drive and/or user specifier which **MUST** be followed by either a colon (:) or a semi-colon (;). A drive is designated by a letter between A and P, allowing up to 16 drives to be selected (providing your system supports that many). User is designated by a number between 0 and 15 (optionally 31) allowing each drive to be partitioned into separate areas. If drive (user) is not specified, the current drive (user) is used as default.

A filename can be from one to eight characters, excluding special characters "< > . , ; : = ? * [] ", with no intervening spaces. An optional "extension" (up to three characters) can follow the filename, if preceded by a period (.). Only those files with an extension of ".COM" can be directly executed at the QP/M prompt (normally ">").

QP/M also recognizes "Wild Card" characters "?" and "*". These two symbols are used as follows:

? is used to match any character of a filename in that same position, including blanks.

Q???.COM would produce three matches if QCP.COM, QPIP.COM, and QSUB.COM files were on the referenced disk. QSTAT.COM would not be a match, as there are more characters in the filename than were specified by the "???".

Another example of this wild card would be to specify ?DATA???.?85. This would search for all filenames at least five characters long, with one character preceding "DATA", and extensions ending in "85".

* is used as a "shortcut" to typing "?" through the eighth character of the filename.

Referencing the above examples, Q*.COM would be equivalent to typing Q????????.COM and would find matches of QCP.COM, QSTAT.COM, and QSTAMP.COM. DATA*.8* would find matches in DATA111.855, DATA.854, or DATA-B.851

. is equivalent to ??????????.???, and will match all files.

The file reference conventions found below will be used throughout this section.

afn ambiguous file reference (could reference one or more filenames.)

ufn unambiguous file reference (can only reference one filename.)

params optional parameters specified by the user,
 following a transient command or program name.

Examples of params are:

```
DIR Q* .COM
WS C ;DOCFILE
QSTAT C7:QSTAMP $R/O
COMPARE A:DATA1 B:DATA1
```

All commands are executed at the QP/M prompt ">", whenever the <RETURN> or <LINE FEED> keys are pressed.

Optional parameters are surrounded by brackets '[]'; do not enter the brackets themselves. Lower-case characters are translated into upper case when commands are executed.

C-2. DFD [drive][:]

The DFD command sets the default drive to the drive specified; the drive letter may or may not be followed by a colon or semi-colon. For example, to set drive B as the default drive, you would enter

```
DFD B: (or DFD B or DFD B;)
```

Only drives A through O can be set as default drive. **All other values or omitting the drive name disables the drive search capability of QDOS until another DFD, a system call to Function 41 with a valid default drive, or a cold boot.** For example, the commands

```
DFD  
and  
DFD Z
```

disable the drive search feature of QP/M.

C-3. DFLT

DFLT prints the active default drive and user at the terminal. For example, if the default drive is A and default user is 0, typing DFLT would result in

A0

being displayed. If the default user is disabled, no user number is displayed. If the default drive is disabled, '@' is displayed in place of the drive letter.

Some examples are included below:

A4	<--- Default drive is A, default user is 4
@1	<--- Default drive disabled, default user is 1
C	<--- Default drive is C, default user disabled
@	<--- Both default drive and default user are disabled

C-4. DFU [n]

The DFU (Default User) command will change the Default User number to be applied for user searches in QP/M. This new Default User number remains active until the next DFU command is issued, QDOS function 42 is called with a new Default User number, or a cold boot. Only values between 0 and 14 (decimal) will set the default user area, although you can still access up to user area 31 with the USER command (see C-16). A 'blank' implies User 0. A value of 15 (0FH) disables the default user search within QDOS. Formats for DFU are

DFU 13 <-- Set Default User number to 13 decimal
DFU 15 <-- Disable Default User Search
DFU <-- Set Default User number to 0

C-5. DIR [afn]

The DIR (directory) command lists all matched files in the currently active user area on the specified (or default) drive. The "afn" is optional. As a special case, the command

```
DIR
```

lists all files in the currently active user area of the default disk. The command:

```
DIR *.*
```

is equivalent to the first command (DIR) as it specifies that all matching files should be listed. Valid commands can also include a drive designator as shown below

```
DIR Q????.COM    <-- Display all files Q.COM, Q?.COM, Q??.COM or Q???.COM
DIR *.DOC        <-- Display all files with extension DOC
DIR B;           <-- Display all files on drive B
DIR C:* .A?M     <-- Display all files with extension of A?M on drive C
DIR 7:           <-- Display all files in user area 7 of the current drive
DIR A11;* .      <-- Display all files with no extension on drive A, user 11.
```

C-6. ERA afn

The ERA (erase) function removes all specified files on the default (or specified) drive. All files in the directory matching that file specification will be listed on the console. If any erase attempt is made to a file that is R/O (read only), a message

QDOS error on d: File R/O

will appear and the file will not be erased (even though it was listed on the console). Whenever a wild-card (* or ?) is used, several files matching the (afn) designation may be displayed on the console. However, the above message occurs (and the erase aborts) if and when the first R/O file is encountered. Unfortunately, there is no way of knowing which files that were listed on the console were actually erased, and which were not. Only by listing the directory will you be sure.

The ERA command will remove both DIR and SYS files. If your QCP is configured as delivered, the DIR command will not show SYS files, but ERA will still remove them. Examples of valid commands are

```
ERA TMP . 1      <-- Erase TMP.1
ERA * .BAK      <-- Erase all files with extension .BAK
ERA B : TEST . * <-- Erase all files on drive B with name of TEST, and any extensions
ERA C 7 ; README <-- Erase file README (no extension) on drive C, user 7.
ERA * . *       <-- Erase all files
```

This last command will erase all files on a specified drive. As a final confirmation that you really want to erase all files, QCP will prompt

All?

Entering a character other than "y" or "Y" will abort the erase command.

C-7. GET addr ufn

The GET command loads the file 'ufn' from disk into memory starting at hexadecimal location 'addr'. GET employs the QDOS search path in finding the file. Valid commands are

```
GET 3000 PROG.UTL    <-- Load PROG.UTL into memory at 3000H
GET 118H TEMP.TST    <-- Load TEMP.TST into memory at 118H
```

If there is insufficient memory to load the file at the given address, the message

Full

will be displayed and the GET will not be completed.

C-8. GO [*params*]

The GO command is a specialized case of the JUMP command where the address is always the start of the TPA (usually 100H). GO is equivalent to "JUMP 100H". [*Params*] are the optional parameters described in section C-1.

C-9. JUMP addr [params]

The JUMP command "calls" the subroutine at the specified address 'addr'. 'Addr' is evaluated as a hexadecimal number regardless of a following "H" character as in other commands. The JUMP command actually performs a "CALL" so that the called routine may return to the QCP. Examples of JUMP commands are

```
JUMP 7000      <-- Jump to 7000H
JUMP 7000H     <-- Jump to 7000H
JUMP 0         <-- Jump to 0000H
JUMP           <-- Jump to 0000H
```


C-10. LIST ufn

The LIST command will print the specified file on the QP/M LST: (printer) device. Typing <CTRL+C> during listing will abort the command. Examples are

```
LIST QPM.DOC  
LIST B:HELLO.C
```

C-11. *REN ufnnew=ufnold*

The REN (rename) command will change the name of a file to new name; only unambiguous filenames are allowed. Rename takes the form

```
REN ufnnew=ufnold
```

where `ufnold` is the old filename and `ufnnew` is the new name. Examples are

```
REN PIP.COM=QPIP.COM    <-- Rename QPIP.COM to PIP.COM
REN 4;QDOC=QDOC.BAK    <-- Rename QDOC.BAK TO QDOC on user 4
```

If `ufnnew` already exists, the message

```
Delete File?
```

will be displayed. Pressing any key other than a "y" or "Y" will abort the rename.

If drive/user is specified in `ufnnew`, the rename will occur on that drive/user. A drive/user specifier is not necessary in `ufnold`; QP/M assumes that `ufnold` is on the same drive/user as `ufnnew`.

If a drive/user specified in `ufnold` is different than `ufnnew`, the rename will not occur.

C-12. SAVE n ufn [option]

The SAVE command copies 'n' pages (256-byte blocks) from the TPA (usually starting at 100H) into a file 'ufn'. 'N' is treated as a hexadecimal value if immediately followed by an "H", otherwise it is treated as a decimal value. The 'option' is the character "S" which specifies to copy 'n' SECTORS (128-byte blocks) rather than 'n' PAGES into the file.

Assuming you wanted to save a file located from 100H to 1000H, there are four ways to accomplish this

```
SAVE 15 QPATCH.COM    <-- Saves 15 decimal pages
SAVE FH QPATCH.COM    <-- Saves 0F hexadecimal pages
SAVE 30 QPATCH.COM S  <-- Saves 30 decimal sectors
SAVE 1EH QPATCH.COM S <-- Saves 1E hexadecimal sectors
```

If 'ufn' already exists, the message

Delete File?

will be displayed. Typing a "y" or "Y" will erase the current file on disk and replace it with the memory saved. Pressing any key other than a "y" or "Y" will abort the SAVE.

If the SAVE command requires saving more memory than the specified disk space remaining, the message

Full

will appear and the SAVE will not occur.

C-13. TIME

The TIME command displays the current date and time (in 24-hour format) on the console device. The format is

```
TIME
```

at which the QCP returns

```
dd-mmm-yy hh:mm:ss
```

For example, QCP might return

```
01-Nov-85 17:31:20
```

If the QDOS has not been provided with a time-date vector, QCP will print

```
No clock
```

C-14. TOF

The TOF command sends a top-of-form character to the QP/M LST: device. As distributed, this is the form-feed character (ASCII decimal 12), but can be changed via QINSTALL to be any user specified character.

C-15. TYPE ufn [option]

The TYPE command will display the file on the console device. 'Option' is the character "P" which changes the default setting of the PAGING flag. As distributed, PAGING is TRUE which means that the display will pause whenever the screen is filled during the TYPE command; display resumes when any key is pressed. The default value of the PAGING flag as well as the length of the screen can be changed with QINSTALL. Examples of TYPE commands are

TYPE QNOTES.DOC	<-- Lists QNOTES.DOC on the console, pausing whenever the screen is filled
TYPE B:QNOTES.DOC P	<-- Lists QNOTES.DOC from drive B on the console without pausing
TYPE D13;QNOTES.OLD	<-- Lists QNOTES.OLD from drive D, user 13 on the console

Note: Pressing <CTRL+C> while TYPE is displaying or pausing will abort the command.

C-16. USER [n]

The USER command will change the currently active user number to the value of 'n'. 'N' defaults to a decimal value but will be treated as a hexadecimal value if immediately followed by an "H". If 'n' is omitted, it is assumed to be 0 (zero). The USER number remains active until the next USER command, cold-boot, or QDOS function 32 is executed. Valid USER commands are

```

USER 1      <-- Sets user number to 1 decimal
USER 12     <-- Sets user number to 12 decimal
USER 0DH   <-- Sets user number to 13 decimal
USER       <-- Sets user number to 0

```

Valid user numbers 'n' range from 0 to 15 (optionally 31) decimal.

An abbreviated method can also be used to change the active user number. By simply specifying the user number, followed by a colon, you can go directly to that user area without having to type USER first. Any of these commands will change the current user number:

```

USER 5
5:
5;
B5;

```

D. QP/M UTILITIES

A description of all QP/M utilities is presented in this section. Many of these utilities can be customized to suit your needs. Refer to Part IV (Supplements), Section B, for detailed information.

D-1. D (2k)

The D(irectory) utility functions similar to the DIR transient command, but includes much more information. The 2k version of D.COM is written at the "front" of your disk (just past the directory area) by the QSTAMP utility, for fastest operation.

All files on the current or specified drive/user area are listed in alphanumeric order, and individual file sizes (in kBytes) are displayed. After the last file is displayed, D will count the directory entries, total the file sizes, show maximum disk capacity, and compute remaining disk space.

Either the 2k or 4k version of D.COM will accept an "option" which consists of a "\$" (dollar sign) followed by the option letter(s). The 2k "options" are:

- \$T - Show file time/date information
- \$L - List matching contents of .LBR library files
- \$R - Reset system before examining the directory
- \$S - Show system files (files having SYS flag set)

Examples of valid commands are:

- D \$T <-- Display all files and show time/date info
- D Q.* \$TL <-- Display all files of filename Q with time/date info
and show any matching .LBR directories
- D B: ?80.* \$T \$S <-- Display all files on drive B of name ?80.???
including SYStem files and show time/date information

In actuality, the "options" toggle the related flags in the D.COM programs. Each option you enter changes the specific flag from FALSE [TRUE] to TRUE [FALSE]. Entering the same option twice as in

- D \$TT
- D \$S \$S

toggles the flag twice and hence has no effect.

As delivered, both versions of D.COM have \$S set to TRUE and all other option flags set to FALSE; thus, a directory listing will show system files but will not reset disk system, show time/date information, or list contents of .LBR files.

QPATCH allows you to change the default settings of \$S, \$L, \$R, or \$T within either D.COM (2k or 4k).

D-2. D (4k)

The 4k version of D.COM (written by QSTAMPX) has all of the same features of the 2k version, plus these new capabilities:

1. Search for files created, updated, or backed-up on, before, after, or between specific dates.
2. Search all user areas.
3. Search all drives.
4. Search for all files that have not been backed up.

All 4k D.COM options are listed below. (The 2k options, \$T, \$L, \$R, \$S, are toggled; the unique 4k options must be entered every time you wish to use them.)

\$T - Show file time/date information
 \$L - List matching contents of .LBR library files
 \$R - Reset system before examining the directory
 \$S - Show system files (files having SYS flag set)
 \$A - Search all user areas
 \$D - Search all logged drives
 \$B - Use "backup" date instead of "updated" date
 \$N - Show files that have not been backed up
 \$C - Use "created" date instead of "updated" date
 \$U - Use "updated" date (default)

The format of the 4k D.COM command line is:

D [du:filename.ext] [\$options(s)] [begin d/t][;end d/t]

where du:filename.ext is a file specifier (with drive, user and extent optional)
 \$option(s) are one or more options
 begin d/t is the beginning date/time range
 end d/t is the ending date/time range

The command line must have the parameters in the above order (i.e. \$option(s) MUST occur before dates). All parameters are optional.

Date searches are done as follows:

[begin date/time][;end date/time]

The format of 'begin date/time' and 'end date/time' is as follows:

m/d/y [h:m]

The time is optional and is set to 00:00 if omitted in 'begin date/time'; if omitted in 'end date/time', 23:59 is used.

If 'begin date' is omitted, a date of 00/00/00 is used, and time defaults to 00:00. If 'end date'

is omitted, a date of 99/99/99 and a time of 23:59 is used. Search ranges are inclusive of the date(s) and time(s) referenced.

Examples of valid commands are:

- D \$T\$N <-- Display all files that have not been backed up, showing time/date information
- D Q*.* 2/1/85 <-- Display all files which start with "Q", that were updated on or after 2/1/85.
- D B;*.* \$TN \$C ;3/1/85 <-- Display all files on drive B that were created on or before 3/1/85, and have not been backed up. Also show time/date created.
- D *.* \$DA 7/1/85 8:00;7/1/85 12:00 <-- Display all files in all user areas and all drives that were updated between 8 A.M. and noon on 7/1/85.

Another 4k D.COM directory program is generated via QSTAMPV that prints the directory vertically sorted rather than horizontally sorted. In all other ways, it is identical to the 4k D.COM utility described above.

D-3. INIT (Included with QBIOS package only)

The INIT(ialize) utility is used to make a "blank" or unformatted disk compatible with your system. You must specify which drive is being initialized (A-P). The INIT command also works to effectively erase all data on existing formatted disks, so use caution not to INIT unless you are sure.

You will be asked for the PHYSICAL drive to be initialized, regardless of logical drive assignment, so again use extra care.

D-4. QBACKUP

QBACKUP is used to make a backup (identical copy) of an entire disk, or to copy only those files which have been updated since the last time QBACKUP was used. The latter feature is especially useful when you have a full disk of files to backup, yet only a few have been updated.

Both source and destination disks must be properly time/date stamped for QBACKUP to work (see QSTAMP).

You will be given the option to delete "old" files that exist on the backup disk, but not on the source disk. This is recommended when you have a dedicated backup disk for a specific source disk, as you no longer have to compare directories side-by-side to weed out the files previously deleted from the source.

QBACKUP will transfer files sequentially, starting at the beginning of the sorted directory (as you would see when running D.COM). During transfer, if there is inadequate space to write a new file to the destination disk, QBACKUP will give an error message and return to the beginning of the program. You can then use a new backup disk to finish the process, or exit QBACKUP and delete unwanted files from the original backup disk.

QBACKUP transfers files safely in a similar manner to QPIP (the "old" file is not deleted from the destination disk until the "new" file is written and verified). This means that there must be adequate space on the backup disk to hold the source files, PLUS enough free space equal to the largest file that requires transfer. As an example, assume that three files require transfer as follows:

Filename	Size
-----	----
CALCDATA.001	15k
PROPOSAL	32k
D.COM	4k

If these are all old files whose sizes didn't change, the destination disk would require 32k (largest file to transfer). If less than 32k is available, QBACKUP will print a message like:

```
There is not enough room on destination drive for backup.
32k is required, but only 5k is available.
```

```
Do you want QBACKUP to delete the (old) destination file
BEFORE transferring the (new) source file? (Y/N):
```

This will cause QBACKUP to delete the current destination file FIRST (thereby making more space available), then transferring the updated file from the source. QBACKUP leaves the choice to you, as there is a risk (though minimal) should the source file not transfer successfully.

D-5. QINSTALL

QINSTALL is used for either first-time installation of QP/M or modifying the QCP and QDOS default parameters. Refer to part I, QP/M Installation Guide, for further details.

D-6. QPATCH

QPATCH is used to modify the default settings of D.COM (either the 2k or 4k versions), and QBACKUP.COM. This utility is entirely self-prompting.

Experienced users who wish to change the "Restore default settings" values of QPATCH should refer to section B of the QP/M Supplement.

D-7. QPIP

QPIP is the Peripheral Interchange Program for QP/M users. While QPIP is fully compatible with all CP/M PIP options (plus a few new ones), QPIP will ONLY work while running under QP/M.

QPIP allows you to copy single files, or concatenate (combine) several files into one. QPIP also lets you specify the device names (TTY:, CRT:, PTR:, etc.) to which the files should be copied (or printed).

QPIP can be entered as a command by itself, or may contain it's own parameters:

```
A>QPIP
A>QPIP HAPPY=SAD
```

When entered as in the first example, QPIP gives a prompt of "*", and will execute subsequent commands read from the console, until a <CTRL+C> or empty line <RETURN> is encountered. At the "*" prompt, "R" will reset all drives, and "^" (circumflex) will reset all drives and repeat the prior command. In the second example (QPIP HAPPY=SAD), QP/M loads the QPIP program, then reads the parameters directly from the console (no further prompting is given). When QPIP finishes, it returns you to QP/M.

In all QPIP commands, the destination is entered first, then an "equal" sign (=), followed by the source(s). The following rules also apply:

- A drive designator is permitted on either the source and/or each destination (default is current drive)

```
A>QPIP A:QSTAMP.NEW=B:QSTAMP.NEW
```

- In addition, QPIP allows you to specify the user area that should be read from/copied to (default is current user area):

```
B>QPIP B2:D.COM=A0:D.COM
```

Multiple source files may be used to combine several files into one destination file, but each source file must be separated a comma (.). In addition, these files should be ASCII files as QPIP ends the transfer upon reading an EOF character (ASCII decimal value of 26).

```
B>QPIP BIG1=SMALL1 , SMALL2 , SMALL3
```

If either the destination or source filename is omitted, QPIP will use the one filename for both read and write; the examples below produce identical results

```
B2>QPIP B:=A:QINSTALL.COM
```

```
B2>QPIP B:QINSTALL.COM=A:
```


QPIP will always copy actual file dates (update and create), but will not set the archive bit (see QSTAT). If you desire to use the current time/date instead, specify the transfer command with a [C] option. If you want the transferred files to carry a time and date other than the original or current values, you will have to first transfer the files, then use QSTAT with an option of [\$CRE], [\$UPD], or [\$BAK].

Wild cards (see 2k D.COM documentation) may be used to specify groups of files; each file transferred will be listed sequentially on the console

```
A1>QPIP A:=B:Q*.COM
```

Physical or logical device names (see QSTAT) may be used

```
A>QPIP LST:=SOURCE1
B>QPIP DUMP7=UR1:
```

QPIP also has some special device names

INP: This is a special QPIP input source that gets data one character at a time, by repetitive CALLs to 103H; data is returned at 109H **

OUT: This is a special QPIP destination that CALLs location 106H with the data in register C, to transmit one character at a time **

PRN: This is identical to LST:, except that tabs are expanded at every eighth character position, lines will be numbered, and an initial page eject is sent before transmission begins, and every sixty lines thereafter

** The QPIP program must be patched for this function

QPIP performs a special function when writing to a .HEX file from an external peripheral device. When reading from a properly formed hex file and an error occurs, QPIP will give an error message. The program halts while the peripheral device is "backed up". A <RETURN> will tell QPIP to re-read that same section again, or continue from that point forward. If the source device is RDR:, QPIP will allow you to type <CTRL+Z> to signify end-of-file.

QPIP has several optional parameters which can be specified at the end of the command line by enclosing the option in [brackets]. These parameters are:

B *Block mode transfer:* Buffer all input data until a <CTRL+S> (ASCII X-OFF character) is received from the source device. This allows continuous reading from tape, for example. After the X-OFF is received, QPIP clears the buffer and returns for more input data. An error message will tell of buffer overflow.

C *Use current time/date when transferring files.*

Dn *Delete characters extending past column n.*

E *Echo all transfers to the console.*

F *Filter (remove) all imbedded form feeds from the file.* P can be used simultaneously to insert new form feeds.

Gn *Get file from User number n* (QPIP will do this automatically if the drive and user number are specified in the command line)

H *Hex data transfer* (see above)

I *Ignore ":00" records while transferring an Intel file.* I sets H parameter automatically.

L *Translate all upper case characters to lower case.*

N *Add line numbers, starting at "1:".* Leading zeroes are suppressed, unless "N2" is specified. N2 also inserts a tab after the number, and the tab is expanded if the T option is also used.

O *Object (non-ASCII) file transfer.* Ignore normal QP/M end of file.

Pn *Add page ejects, starting at the front of the file, and every n lines thereafter.* Default is every 60 lines. If F is also used, form feed suppression precedes the page ejects.

Qs^z *Quit copying from source when string s found.* (string terminated by a <CTRL+Z>)

R *Read system files*

Ss^z *Start copying from source when string s found.* (string terminated by <CTRL+Z>)

Tn *Extend tabs <CTRL+I> to every nth column during transfer.*

U *Translate lower case to upper case.*

V *Verify correct transfer by rereading a disk file immediately after it is written.*

NOTE: Verify is ALWAYS done by QPIP when transferring disk files.

W *Write over R/O files without prompting for Y/N* (see example below).

Z *Zero the parity bit on input for each ASCII character.*

Unless the [W] option is used, whenever QPIP is asked to write over a R/O file, the user will be asked for confirmation in this format:

File QINSTALL.COM R/O, Delete? (Y/N):

Responding with a "Y" will overwrite the old file with the new data. Otherwise, QPIP returns a response of:

** NOT DELETED **

D-8. QSTAMP/QSTAMPX/QSTAMPV

QSTAMP/QSTAMPX/QSTAMPV are the QP/M disk time/date stamping utilities. Since time/date stamping is independent of the system tracks, they can be blank, or contain a CP/M or QP/M system. However, before time/date stamping will occur, you **MUST** run QSTAMP/X/V to place all necessary time/date parameters on each disk.

Two things must be present before QP/M will time/date stamp a disk:

- QP/M must be the running system
- The disk itself must be time/date capable (QSTAMP must have already been completed on this disk)

Running QSTAMP on a disk with CP/M system tracks **WILL** make a disk time/date capable, but your system **MUST** be running under QP/M, as CP/M alone **CANNOT** update the time/date information. This also means that if you "swap" disks, then do a warm boot <CTRL+C>, the disk that you warm-booted **MUST** have QP/M system tracks.

QSTAMP will automatically reset the referenced drive before attempting to time/date stamp. This is useful when you have multiple disks to stamp. Note that QP/M uses the **CURRENT** time/date when initially stamping a disk.

QSTAMP also automatically writes a D.COM (2k directory utility) file to the very first file sectors on the disk. This ensures fastest access to the directory. The (optional) QSTAMPX utility writes the expanded (4k) D.COM file, while QSTAMPV writes a special 4k D.COM program which displays the directory in vertical format.

If an attempt is made to time/date stamp a disk that has already been made time/date capable, QSTAMP will display a message to that effect and no stamping will be done. This prevents accidental destruction of existing time/date information.

If you wish to install your own directory program, refer to Section B of the QP/M Supplement.

D-9. QSTAT

QSTAT has three main functions: examine general statistical information about a file; alter file status and time/date information; and examine/alter the current physical-to-logical device assignments. QSTAT, like QPIP, will only work with QP/M.

To examine the specifications of a file, enter:

```
A>QSTAT QPATCH.COM
```

If a file match is found, QSTAT will display the specifications of the file in the following format:

```
Recs Bytes Ext Acc Bak du:Filename
    25   4k  1  R/O Yes A0:(QPATCH.COM)
```

Recs shows the number of 128-byte records

Bytes shows the number of kilobytes allocated

Ext shows the number of directory extensions

Acc shows R/O or R/W access mode

Bak shows whether file has been backed up ("Yes" = backed up; blank = not backed up)

Files in () designate SYS files

A single \$ option (\$T) may be entered to display time/date information as well:

```
A>QSTAT HELLO.COM $T
```

If a match is found, QSTAT will display:

```
Recs Bytes Ext Acc Bak du:Filename
          Created          Updated          Backup
    25   12k  1  R/O   A1:HELLO.COM
          01-Feb-85 12:00   02-Feb-85 11:45   02-Feb-85 16:15
```

QSTAT will also accept a \$\$ option (with or without \$T) to show virtual size:

```
B>QSTAT QSTAT.COM $$
```

```
Size Recs Bytes Ext Acc Bak du:Filename
    16   16   4k  1  R/O Yes A2:QSTAT.COM
```

As a special case, QSTAT may be entered as a command by itself:

```
A>QSTAT
```

This will cause QSTAT to calculate the available storage on all active drives and print status as follows:

```
Drive A is R/O, with 18k space remaining.
Drive B is R/W, with 490k space remaining.
```

Drive and user number may be used to status a file on any drive, in any user area, regardless of current or default drive/user area:

```
A5>QSTAT B1 :QPIP.COM
```

Wild cards can be used on the command line to specify ambiguous file names, such as:

```
B>QSTAT *.* (run QSTAT for all files on current drive and current user area)
```

```
A>QSTAT B3 ;Q*.COM (QSTAT all files on B3; matching Q.COM)
```

A drive may be set to R/O (read only) or R/W (read/write) by entering:

```
A>QSTAT B:=R/O
```

```
A3>QSTAT A:=R/W
```

QSTAT will give a message such as:

```
Drive B is now R/O.
```

The first example will set drive B: to R/O. The second example will set drive A: to R/W (effectively resetting it). Note that you cannot set individual user areas to R/O or R/W, only the entire drive. A drive set to R/O will remain R/O until the next warm or cold start, or drive reset is performed. If an attempt is made to write to a drive that is set R/O, QP/M will return the error message:

```
QDOS error on B: R/O
```

Individual files may be set to R/O or R/W (read/write), SYS or DIR (non-sys) using the "\$" options listed below:

```
$R/O (make file read/only)
$R/W (make file read/write)
$SYS (make file SYS)
$DIR (make file non-SYS)
```

NOTE: Multiple \$xxx options can be used if they are separated by single spaces. If two options on the QSTAT command line conflict, QSTAT will only use the latter of the two.

If QP/M tries to write to a file that is marked R/O, an error message will be returned, such as:

QDOS error on B: File R/O

QSTAT also permits you to change QP/M-unique parameters:

\$CRE (change create date/time)
\$UPD (change update date/time)
\$BAK (change backup date/time)

The format for all time/date changes is as follows:

\$CRE=mm/dd/yy [hh:mm]

A check is made for month range 1-12, day range 1-31, year range 0-99, hours 0-23, and minutes 0-59. The hours and minutes are optional and default to 00:00.

As an example of the preceding paragraphs, you could enter:

```
A>QSTAT QBACKUP.COM $UPD=2/1/85 12:00
```

This would change the update date of QBACKUP.COM to 01-Feb-85 12:00.

The backup status (archive bit) can be set or reset as follows:

```
A>QSTAT D.COM $NEW  
B>QSTAT D.COM $OLD
```

The \$OLD option tells QP/M to set the file to "backup" status (i.e., treat as if QBACKUP has already been done). \$NEW resets the archive bit as if the file is "new" and has never been backed up with QBACKUP.

QSTAT allows you to change the physical-to-logical device assignment. The four logical devices are:

CON: Console device

RDR: Paper tape reader

PUN: Paper tape punch

LST: Output list device

These logical devices may be assigned to any of four physical devices as listed below:

<u>Logical</u>	<u>Physical</u>	<u>Description</u>
CON:	TTY:	Teletype device (slow speed console)
	CRT:	High speed console
	BAT:	Batch processing (console is current RDR:, and all output goes to current LST: device)
	UC1:	User-defined console
RDR:	TTY:	Teletype device (slow speed console)
	PTR:	Paper tape (high speed) reader
	UR1:	User-defined reader #1
	UR2:	User-defined reader #2
PUN:	TTY:	Teletype device (slow speed console)
	PTP:	Paper tape punch (high speed punch)
	UP1:	User-defined punch #1
	UP2:	User-defined punch #2
LST:	TTY:	Teletype device (slow speed console)
	CRT:	High speed console
	LPT:	Line printer
	UL1:	User-defined list device #1

To view the current physical device assignments, type:

```
B>QSTAT DEV:
```

Which might return:

```
CON: is CRT:
RDR: is UR1:
PUN: is PTP:
LST: is TTY:
```

To view the possible assignments which can be made, or to view the various QSTAT commands available, type:

```
A>QSTAT VAL:
```

Which will return:

```
Temp R/O Disk: d:=R/O
Temp R/W Disk: d:=R/W
Set indicators: du:name.typ $R/O $R/W $SYS $DIR $OLD $NEW
Change time/date: du:name.typ $CRE=mm/dd/yy hh:mm $UPD=... $BAK=...
User Status: USR:
Disk Status: DSK:
IObyte Assignment:
  CON: = TTY: CRT: BAT: UC1:
  RDR: = TTY: PTR: UR1: UR2:
  PUN: = TTY: PTP: UP1: UP2:
  LST: = TTY: CRT: LPT: UL1:
```

To change an assignment, you would type:

```
A>QSTAT logical device:=physical device:
```

To check the status of all logged-in drives, you would type:

```
A>QSTAT DSK:
```

This would return the drive characteristics in this format:

```
A: Drive Characteristics
5392: 128-byte Record Capacity
674: Kilobyte Drive Capacity
128: 32-byte Directory Entries
128: Checked Directory Entries
128: Records/ Extent
16: Records/ Block
72: Sectors/ Track
2: Reserved Tracks
```

To check active user areas (those which contain files), you would type:

```
B>QSTAT USR:
```

This would produce a display similar to:

```
Active User: 0
Active Files: 0 1 31
```


D-10. QSUB

QSUB is the QP/M utility used to execute a batch program. A batch program allows the computer to process commands and operate as if the user were entering commands from the console (usually a keyboard).

The QSUB batch utility consists of commands, one per line, which are executed sequentially, as if typed at the keyboard. All of the QCP transient commands are available (e.g. DIR, TYPE, etc.) as well as a number of special QSUB commands. Once the QSUB utility has been invoked, it assumes control of the system until the batch program is completed or an abort occurs. QSUB loads itself into memory directly below the QCP. If any other utility currently occupies this space, QSUB will print a message and abort. Since it is memory-resident, applications programs running under QSUB will have approximately 5k (5120 bytes) less execution space. However, the speed increase gained by having batch commands resident in memory versus reading them from disk is well worth the tradeoff.

The format of the QSUB command is

```
d>QSUB <QSUB file> param#1 param#2 ... <return>
```

where <QSUB file> is a special command file with an extension of ".QSB"
param#1, etc. are optional parameters entered on the command line.

Parameters entered on the command line are passed to QSUB as internal "variables" designated @1, @2, etc., up through @8. These variables correspond to the matching param# (i.e. @2 is set to the value of param#2). If the parameter is missing from the command line, QSUB sets the variable to a "null" value (@i = ""). There are two command line restrictions:

- (1) the length of any parameter cannot exceed 16 characters and
- (2) only 8 parameters maximum are allowed.

D-10.1 GENERAL COMMANDS

QSUB commands can be any commands that are normally entered from the keyboard. QSUB retains control of the console input at all times, although keyboard characters are processed within the @GET, @PAUSE, and @XSUB0 functions. In addition to the normal QCP transient commands (which are allowed in QSUB), QSUB offers a number of powerful commands to assist the programmer. All special QSUB commands are preceded by the "at" sign (@) which usually occurs at the beginning of a line.

QSUB operates by fetching a character from the QSUB file whenever one is requested by the system. However, all console status checks test the actual console. In this case, if a character is entered via the keyboard, it will be the next character that the system will receive. (A good example of this is when the D.COM utility pauses after the screen is full; since D.COM is checking the console status, the system will wait for a keyboard character even if a QSUB utility is active. After pressing any key, QSUB resumes control of the system.)

D-10.2 NESTING

QSUB programs may call other QSUB utilities, up to a level of 8 deep. All variables and settings within QSUB are restored to their previous values upon return from another QSUB utility. Although variables are initialized upon entry into another QSUB utility, the previous display setting (show/hide) remains intact.

D-10.3 VARIABLES

There are 16 variables in QSUB: **@1..@8** and **@V1..@V8**. The variables **@1** through **@8** are set from the command line of QSUB upon execution, whether activated from the console or from another QSUB program. The remaining user variables, defined as **@V1** through **@V8**, are initialized to null strings ("") upon execution of QSUB. NOTE: The variables are ALWAYS initialized upon entry into a QSUB program, regardless of whether QSUB was executed from the console or another QSUB program. When a QSUB batch program invokes another QSUB program, all variables are saved on a disk file and later recovered before the calling QSUB batch program resumes.

As an example, here are two QSUB programs, one of which is called by the other:

TEST.QSB

```
@LET @V2 = "Sample 1"
QSUB TEST1
@LET @V1 = @V2
```

TEST1.QSB

```
@LET @V2 = "No sample"
DIR
@EXIT
```

@V2 is set in TEST followed by control being transferred to TEST1. Upon entering TEST1, **@V2** becomes a null string; the first statement changes **@V2** to another value. When TEST1 completes, control is returned to TEST restoring the previous values of all variables. Consequently, the final statement of TEST will set **@V1** to "Sample 1".

A variable consists of an ASCII string which contains a maximum of 16 characters. Longer strings may be used as long as the maximum string area is not exceeded and with the understanding that a long string precludes use of succeeding variables. To help in understanding this, a memory map of variable storage is shown below:

<u>Variable Relative</u>	<u>Start Location</u>
@1	0
@2	16
@3	32
@4	48
@5	64
@6	80
@7	96
@8	112
@V1	128
@V2	144
@V3	160
@V4	176
@V5	192
@V6	208
@V7	224
@V8	240
<End of string space>	256

For example, if @2 = "This is a 30 byte long string.", then the @3 variable would be rendered useless even though there are still 2 bytes free.

D-10.4 SUBSTITUTION

In non-QSUB commands, each variable encountered is substituted by its ASCII string. For example, if @4 = "PROG.BAS", then

```
MBASIC @4
```

will actually execute

```
MBASIC PROG.BAS.
```

Only variables are substituted. If you need to use two strings as one substitution parameter, you must first set a free variable to the strings, then use that variable. For example, let's assume you really wanted to use drive C to get the file PROG.BAS.

```
MBASIC C:@4
```

is valid. However, you could have also used

```
@LET @V8 = "C:" & @4
MBASIC @V8
```

to achieve the same result. The latter is more powerful for applications that require it.

D-10.5 PRIMARY COMMANDS

Included here is a list of commands which are immediately executed and have no effect on the value of the variables.

- @!s** Displays the string **s** on the console. A string can be either a **simple string** or a **complex string**. A **simple string** is either
- a variable (**@1 .. @8**, **@V1 .. @V8**)
 - a sequence of ASCII characters surrounded by either single (') or double (") quotes.
 - a constant. QSUB has four constants it recognizes:
 - @?U** returns a two character string equal to the current user (e.g. "00" through "31").
 - @?DU** returns a two character string equal to the current default user (e.g. "00" through "14"). A null string is returned if default user search is disabled.
 - @?D** returns a single character equal to the current drive (e.g. "A" through "P").
 - @?DD** returns a single character equal to the current default drive (e.g. "A" through "O"). A null string is returned if default drive search is disabled.

A **complex string** is two or more **simple strings** that are concatenated together with an ampersand (&). The string **s** may contain two special characters that QSUB recognizes: % and /, neither of which are displayed. When QSUB encounters a % in a string, it sends 30 return-line feeds to the console output simulating a clear screen command. A / is treated as a single return-line feed. Note that the @! does not automatically add a return-line feed sequence to the end of each line; the user must do it explicitly.

For example, let **@V4 = ":CALEN.COM"**, user be 3, and default drive be "A". The commands

```
@! "%Good morning! (Current user " & @U & ")"
@! '/File is ' & @DD & @V4
```

would cause QSUB to send 30 return-line feeds, and print

```
Good morning! (Current user 03)
File is A:CALEN.COM
```

The characters % and / are only treated specially in @! and @GET.

@ABORT Causes immediate termination of the QSUB utility. Regardless of how many levels of QSUB are active, **@ABORT** will return control of the operating system to the user.

@EXIT If the current QSUB program was executed from another QSUB program, **@EXIT** returns control to the calling QSUB. If not, **@EXIT** returns control of the operating system to the user.

NOTE: QSUB will automatically execute a **@EXIT** after the last statement of the currently running QSUB program has been completed.

@PAUSE Prints the message

"Press any key to continue (CTRL+C to abort) --- "

and waits for a console character. The character CTRL+C (03H) will immediately terminate the QSUB utility through the **@ABORT** command. Other characters will be discarded and QSUB execution resumed.

@DISP0 Disables the console output device, "hiding" all console output. This is useful for executing programs where you do not want to display the activity. This command stays in effect until the execution of a **@DISP1**, **@EXIT**, or **@ABORT**.

@DISP1 Enables the console output device. This has the opposite effect of the above command.

@XSUB0 With the exception of **@PAUSE** and **@GET**, QSUB normally fetches all characters from the batch file. **@XSUB0** disables this feature (i.e. characters are fetched from the console) until a warm boot occurs directly via "^C" (CTRL+C) or indirectly via program termination.

@@ This special sequence is used if the "at" sign is required by an applications program. Whenever two "at" signs appear together, it is passed to the executing program as a single "at" sign and no QSUB command check is made.

D-10.6 LABELS and JUMPS

@:label Defines a **label** for use within the current QSUB batch file. **Label** is a string from 1 to 16 characters; the two restrictions are that each character must be either an alphabetic or numeric character (A-Z, 0-9) and that a label must occur on a line by itself within the current QSUB utility.

@GOTO label Transfers control of the QSUB to the matching **label** described above. If **label** is not found, QSUB will print a message and abort.

D-10.7 CONDITIONALS

@IF(cs) cmd If the conditional statement **cs** is found to be TRUE, QSUB will execute the command **cmd**. If not, QSUB will resume execution with the next line in the batch file. **Cmd** is any group of characters which will be processed by the current program or it may be a QSUB command. **Cs** is a conditional statement of the following form

<simple string><conditional><simple string>

where

<simple string> is any variable or a group of ASCII characters surrounded by either single- or double-quotes.

<conditional> is any one of the six conditionals:

< less than	> greater than
= equal to	<> not equal to
<= less than or equal to	>= greater than or equal to

Since QSUB does not perform any case conversion ("a" to "A", for example), it is recommended that all filenames be in upper case when used in conditional statements.

NOTE: No intervening spaces are allowed between "IF" and "(".

@FIL?(s) cmd Looks for the file specified by the string **s**. String **s** may be either a simple or complex string as described in **@!** and is of the form "d:name.ext" with drive and extension optional. If the file is found, the command **cmd** will be executed; if not, QSUB will resume execution on the next line.

NOTE: No intervening spaces are allowed between "FIL?" and "(".

D-10.8 MODIFYING VARIABLES

@LET @LET is used to change the value of any variable to another value. The statement is always of the form

@LET <var> = <string>

where <string> can be either a simple or complex string as described in **@!**.

@GET(s) <var> Prints the string **s** at the terminal and waits for user input from the console. After the user has entered a valid 16 character or less string, QSUB sets the value of <var>, which is any of the 16 variables, to the string entered. This is the alternative to the **@LET** statement as it allows the user to change the value of a variable during execution. QSUB treats the two special characters (% and /) in **s** the same as it does in **@!**.

NOTE: No intervening spaces are allowed between "GET" and "(".

D-10.9 EXAMPLE

```
@! "% File Copy Utility/"
@:TOP
@GET("/Enter the source filename (d:name.ext): ") @V1
@IF(@V1 = "") @EXIT
@FIL?(@V1) @GOTO OKFIL
@! "File " & @V1 & " does not exist./"
@GOTO TOP
@:OKFIL
@GET("/Enter the destination filename (d:name.ext): ") @V2
@FIL?(@V2) @GOTO EXISTS
@:COPY
QPIP @V2=@V1
@GOTO TOP
@:EXISTS
@! "Destination " & @V2 & " exists./"
@GET(" Overwrite destination file? (Y/N): ") @3
@IF(@3 <> "Y") @GOTO TOP
ERA @V2
@GOTO COPY
```

Let's assume that the user wishes to move WS.COM on drive C to drive A and EXAMPLE.TXT from drive D to drive C renamed to EXAMPLE. EXAMPLE already exists on drive C. Comments are shown with <-- on the right hand side.

```
d>QSUB COPY
```

File Copy Utility

```
Enter the source filename (d:name.ext): C:WD.COM<-- misspelled
File C:WD.COM does not exist.<-- Whoops! Try again
```

```
Enter the source filename (d:name.ext): C:WS.COM<-- Source name
Enter the destination filename (d:name.ext): A:WS.COM
```

```
QPIP A:WS.COM=C:WS.COM<-- First copy being done
```

```
Enter the source filename (d:name.ext): D:EXAMPLE.TXT
                                     ^-- User enters name
Enter the destination filename (d:name.ext): C:EXAMPLE.TXT
Destination C:EXAMPLE.TXT exists.<-- Message
Overwrite destination file? (Y/N): Y<-- User still wants to copy
ERA C:EXAMPLE.TXT<-- Erase destination file
QPIP C:EXAMPLE.TXT=D:EXAMPLE.TXT<-- Do the copy
```

```
Enter the source filename (d:name.ext): <-- Return by itself exits COPY
```

D-11. SETxx (Generated by TDCNFG program)

The SETxx program is executed on cold-boot to set the date and time. (QBIOS modules are written so that the autoboot command is still available.) This program will ask you for the date and time (in 24-hour format) before allowing you to proceed into QP/M. If you use a software clock, it is IMPERATIVE that this program be the first one that is executed. Otherwise, the time/date information that QP/M uses will not be correct.

You will be asked to set the date and time as shown:

(24-hour format)

Enter DATE and TIME: mm/dd/yy hh:mm:ss Display? (Y/N/R)

Type <Y> to display the time in the top right-hand corner of your screen. <N> will suppress the display. <R> allows you to re-enter the date and time.

SETxx will check for the following values:

mm between 01-12

dd between 01-31

yy between 00-99

hh between 00-23

mm between 00-59

ss between 00-59

SETxx will check for valid dates (e.g. only 30 days in September), except for February 29, which can be entered for any year. Line editing using backspace and delete is allowed, ESCape restarts SETxx, and a <CTRL>+<C> will abort SETxx.

D-12. SYSGEN

The SYSGEN.COM utility allows you to make an initialized disk "bootable" by writing QP/M to the system tracks. You will first be prompted to load QP/M into memory by reading the system tracks from an existing bootable disk. Once the QP/M system image is in memory, you can write system tracks to one or more disks in succession.

D-13. TDCNFG

TDCNFG is a utility for configuring almost any computer with a clock initialization program. On some systems, TDCNFG may also be able to install a clock driver (software clock).

TDCNFG recognizes a number of systems with built-in software and hardware clocks (such as Kaypro and Xerox). If you have a system that TDCNFG cannot configure, please contact MICROCode Consulting for technical assistance.

E. QCP CAPABILITIES

E-1. QP/M SEARCH CAPABILITY

The first and most basic thing to learn about QP/M is the order in which it searches for a COM file for execution of a file when a system call is made to Function 15 Open File. Under the CP/M 2.2 CCP, if the specified COM file command was not found on the current drive in the current user area, the CCP aborted with an error message. QP/M continues searching from this point a maximum of two more levels. This command hierarchy search was outlined in Section A, and is described here in further detail. (The search within Function 15 is described in part III.)

a. CHECK TRANSIENT COMMAND LIST

If the command is of the form 'COMMAND' and NOT 'd:COMMAND', the QCP-resident (transient) command list is searched for a match. If a match is found, the transient command is immediately processed. If a match is not found, QP/M continues to the next step.

(Note that the 'd:COMMAND' form is good for executing a command 'COM' file which has the same name as a transient command such as SAVE.COM or DIR.COM.)

b. CHECK CURRENT DRIVE/CURRENT USER AREA

The current drive and current user area is scanned for COMMAND.COM. If found, the command is loaded into memory starting at 100H. If not found, QDOS continues.

c. CHECK CURRENT DRIVE/COMMON USER AREA

If Default User Search has been disabled, QDOS skips to step d. Otherwise, a check is made to see if the current user number matches the default user number. If they are different, QDOS searches the directory of the default user area. If COMMAND.COM is found, it is loaded as described above and executed. If not found, QDOS proceeds to the next step.

(Default User number may be initially set on cold boot, entered manually via the DFU command, or changed by Function 42, Get/Set DFU.)

d. CHECK COMMON DRIVE/COMMON USER AREA

QDOS has now determined that COMMAND.COM cannot be accessed from the current drive, nor the common or current user areas. If default drive search is disabled, QDOS ends the search. Otherwise, QDOS checks to see if the current drive is equal to the default drive. If not, QDOS temporarily logs into the Default Drive and searches the Default User Area for COMMAND.COM. If found, it is loaded as described above and executed. If the file is not found or default drive search has been disabled, QCP prints the command name as an error

message (COMMAND?) and returns to command input mode, aborting the QSUB if COMMAND came from it.

In all cases of the search above, if and when COMMAND.COM is found and loaded into memory, QDOS returns to the original disk drive and user number.

To illustrate this command hierarchy search, consider the following examples:

Example 1: Default User is 0, Default Drive is A

```
B10> <-- User is on Drive B:, User Number 10
B10>ASM TEST.BBZ <-- User wishes to assemble TEST.BBZ in
Drive B:, User 10 <-- At this point, QDOS looks on B:/10 for ASM.COM, fails,
                    looks on B:/0, fails, and finally looks on A:/0; it
                    finds ASM.COM here and goes back to B:/10 for the file
                    TEST.BBZ
```

Example 2: Default User is 0, Default Drive is A

```
B10> <-- User is on Drive B:, User Number 10
B10>DFU 5 <-- User Selects User 5 as default
B10>ASM TEST.BBZ <-- As above
                    <-- At this point, QDOS looks on B:/10 for ASM.COM, fails, looks on B:/5, fails, and
                    finally looks on A:/5; it fails here also and prints ASM? as an error message
```

Example 3: Default User is 0, Default Drive is A

```
B10> <-- User is on Drive B:, User Number 10
B10>DFU 5 <-- User Selects User 5 as default
B10>DFD B: <-- User Selects Drive B as default
B10>ASM TEST.BBZ <-- As above
                    <-- At this point, QDOS looks on B:/10 for ASM.COM, fails, look on B:/5, fails and prints
                    ASM? as an error message
```

Example 4: Default User is 0, Default drive is A

```
B10> <-- User is on Drive B:, User Number 10
B10>DFU 15 <-- User Disables user search capability
B10>ASM TEST.AAZ <-- As above
                    <-- At this point, QDOS looks on B:/10 for ASM.COM, fails, looks on A:/10, fails, and
                    prints error message of ASM?
```

E-2. QCP COMMAND SUPPORT

QCP operates at ZCPR Command Level 3. This command level allows the current program in memory to place a command line into QCP. Once control is given to QCP, the current program releases control and the command is executed exactly as though it had been typed by the user at his console terminal.

In order for a program in memory to utilize the Command Level 3 facility, the program **MUST** do the following:

A. Locate the QCP. Since the QCP is ALWAYS 2K bytes in size and located directly under QDOS, the program can locate the QCP by examining the QDOS entry page address at location 0007H and subtracting 8 from this number (8 pages = 2K bytes). The resulting number is the base page address of QCP.

B. Store the command line in QCPBUF and the character count in QBUFF. The following information may help:

	ORG QCPLOC	;Base Address of QCP
	JMP QCP	;Enter QCP and Execute Default Cmd
	JMP QCP1	;Enter QCP and Don't Execute
MBUFF:	DB BUFLen	;Size of QCPBUF in bytes
QBUFF:	DS 1	;Number of Bytes in Command Line
QCPBUF:	DS BUFLen	;Buffer for Command Line
	DS 1	;Buffer for Ending 0 (set by QCP)
QCPPTR:	DS 2	;Address of QCPBUF (set by QCP)

C. Obtain the User/Disk Flag. Location 0004H contains this number, but the user may select a flag of his choice. This flag is one byte long, and the high-order nibble (4 bits) contains the user number and the low-order nibble contains the disk number from which to process the command. The User/Disk Flag must be passed to QCP in the C Register.

D. When ready, transfer control to QCP to process the command by JMPing to the base address of QCP. The first JMP in the JMP table given above is at this address. At this time, QCP will log in the user and disk in the User/Disk Flag and process the Command Level 3 Command Line.