///DOS////////////////

# A User's Guide to the Secrets of AUTOEXEC.BAT

AN INTRODUCTION TO CONFIG.SYS'S PARTNER IN PC CONFIGURATION—WITH SOME LESS-THAN-COMMON OPTIMIZATION TIPS. **BY JEFF PROSISE**

IN APRIL AND May, we looked at the 15 CONFIG.SYS directives that DOS 5.0 supports and explained how to use them to customize your system's setup. This month, we'll discuss some tips and techniques for using AUTOEXEC.BAT—the file that DOS looks for every time you boot up.

## The Role of AUTOEXEC.BAT

In most respects, AUTOEXEC.BAT is like any other batch file—it contains a list of DOS commands that are executed line by line in the form of a script. You can even run it by entering AUTOEXEC at the DOS command line. But what sets it apart from other batch files is that DOS looks for it in the root directory of the startup drive every time you start your PC. If found, AUTOEXEC.BAT is executed automatically. This makes it an ideal place to store commands that you want executed every time you start up.

Any command that is valid on the command line or in a batch file is also valid in AUTOEXEC.BAT, but certain commands occur so often in AUTOEXEC.BAT files and so infrequently elsewhere that they are associated with automatic execution at boot-up. For example, nearly every AUTOEXEC.BAT file contains a PATH command that establishes a list of directories that DOS will search for executable files. (When you type an external command, DOS searches for a file of that name with the extension .COM, .EXE, or .BAT first in the current directory, and then in the PATH directories. Therefore, you can execute programs stored in PATH directories no matter what drive and directory are current.) For example, in order to execute any command from your DOS, WordPerfect, or utilities directory, enter a line such as the following in AUTOEXEC.BAT:

```
PATH C:\DOS;C:\WP51;C:\UTIL
```

When setting up your own PATH, always include drive letters with the directory names. Otherwise, the search path will be invalid if you switch to another drive.

A problem that users encounter frequently with the PATH command stems from the 127-character limit that DOS places on commands. If a PATH statement is longer than 127 characters, extra characters are ignored. One way to overcome this limitation is to use the SUBST command to create 2-letter drive aliases for PATH directories. In the previous example, you could shorten the PATH statement from 27 characters to 15 by substituting the following commands after including a LASTDRIVE=O statement in CONFIG.SYS:

```
SUBST M: C:\DOS
SUBST N: C:\WP51
SUBST O: C:\UTIL
PATH M:\;N:\;O:
```

Another common AUTOEXEC.BAT command is PROMPT, which determines the appearance of the command prompt. The default command prompt for hard disk users is C>. You can spruce this up with commands such as PROMPT $p$g, which reconfigures the prompt to show the current drive and directory, or you can get fancy with color (if DOS's ANSI.SYS driver is installed as a Device in CONFIG.SYS). For example, to display the current drive and directory in

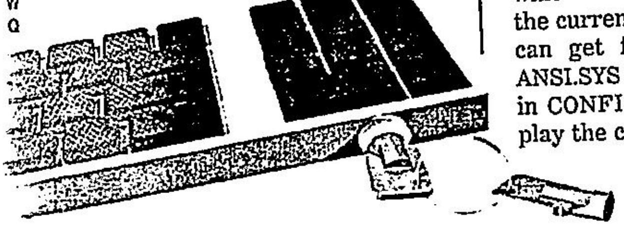
## /// DOS Helper ///////////////

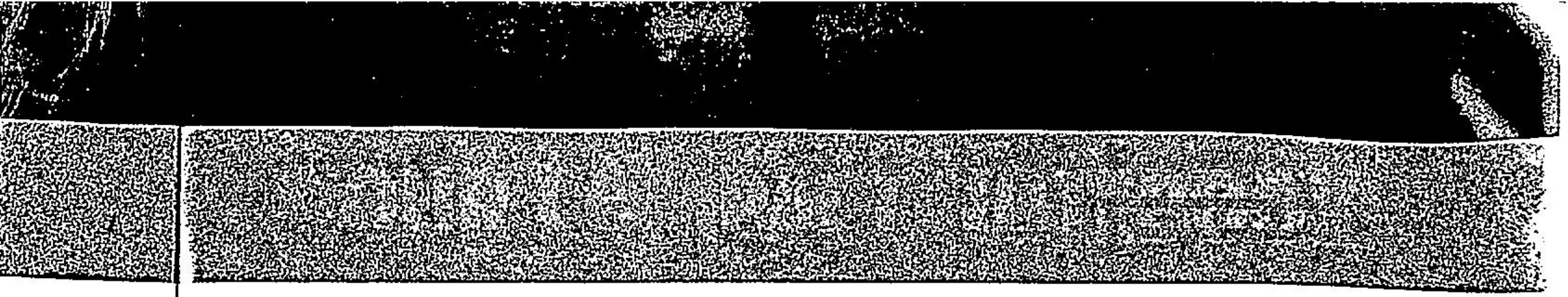
### NUMLOCK.COM and CAPSLOCK.COM

AUTOEXEC.BAT is a great place to stash commands for turning Num Lock and Caps Lock on or off, provided you have utilities to toggle these keyboard states. Such free utilities abound, but you can create your own with this simple DEBUG script:

Name this text file LOCK.SCR, and use the command DEBUG < LOCK.SCR to create two programs—NUMLOCK.COM and CAPSLOCK.COM—that will toggle the Num Lock and Caps Lock states, respectively. Follow either command with a plus sign to turn the locks on, or a minus sign to turn them off (separate with only one space). For example, NUMLOCK - turns Num Lock off, while CAPSLOCK + turns Caps Lock on.

```
N NUMLOCK.COM
E Ø1ØØ B8 4Ø ØØ 8E CØ B3 2Ø 8Ø
E Ø1Ø8 3E 82 ØØ 2D 74 Ø6 26 Ø8
E Ø11Ø 1E 17 ØØ C3 F6 D3 26 2Ø
E Ø118 1E 17 ØØ C3
RCX
1C
W
N CAPSLOCK.COM
E Ø1Ø6 4Ø
W
Q
```





3AT first in
then in the
'e, you can
'ATH direc-
and direc-
le, in order
from your
s directory,
llowing in

L

PATH, al-
ith the di-
he search
itch to an-

unter fre-
and stems
that DOS
TH state-
cters, ex-
ne way to
) use the
tter drive
1 the pre-
orten the
acters to
ing com-
RIVE=O

EC.BAT
ermines
prompt.
for hard
e this up
'T $p$g,
to show
, or you
DOS's
Device
to dis-
tory in

white on red, use the command

```
PROMPT $e[1;37;41m$p$g$e[0;37;40m
```

AUTOEXEC.BAT is a great place to install TSRs such as DOSKEY and to execute configuration commands such as MODE CON: RATE=32 DELAY=2, which increases the keyboard's typematic repeat rate. Here's a tip: Preface all external commands in AUTOEXEC.BAT with full pathnames. For example, the lines

```
C:\DOS\DOSKEY
C:\DOS\MODE CON: RATE=32 DELAY=2
```

execute faster than the simpler

```
DOSKEY
MODE CON: RATE=32 DELAY=2
```

### Protecting AUTOEXEC.BAT

Most installation programs modify AUTOEXEC.BAT only after asking permission. However, some make secret changes, sometimes with disastrous results. Here are four tips for protecting AUTOEXEC.BAT from such programs:

1. At a command line, type ATTRIB +R AUTOEXEC.BAT to make AUTOEXEC.BAT a read-only file. This prevents any program from editing the file—but it also prevents you from editing. You'll have to remove the read-only attribute before editing (type ATTRIB -R AUTOEXEC.BAT), and then replace it when you're done.

2. Place all startup commands in a batch file called STARTUP.BAT, and make AUTOEXEC.BAT a one-line file that invokes it. Even if a program alters AUTOEXEC.BAT, the commands in STARTUP.BAT will be unaffected.

3. Keep a backup copy of AUTOEXEC.BAT in a remote directory or on a floppy disk. Then, if an installation program modifies the original, you still have the backup.

4. If DOS suddenly begins to ignore your PATH or show a default prompt, check CONFIG.SYS for a SHELL= statement that loads COMMAND.COM. Any such command should end in a /P switch. Without this switch, COMMAND.COM ignores AUTOEXEC.BAT—and worse, leaves your PC susceptible to crashes. ◪

## /// Hot Tips //////////////////

### Remove All Unwanted Files at Once

Programs tend to clutter your hard disk with files you don't need, usually with .BAK, .$$$, and .TMP extensions that take up precious hard disk space. These files can be spread across many drives and directories, which makes seeking them out for deletion prohibitively difficult. This command uses the QBasic program KILLBAK.BAS, which in turn uses two DIR switches, /B and /S, present only in DOS 5.0.

Enter the following listing in a text editor, but adjust the assumptions in the first section to match your system. Save it in your utilities or batch directory.

```
'Assumptions. These will vary depending on your system
DRIVE$ = "C" 'Start with this Drive
TOPDRIVE$ = "E" 'Last real drive in system
TEMPFILE$ = "F:\TEMPFILE.$$$" 'F is the system's RAM disk
BATFILE$ = "F:\KILLBAK.BAT"
BAD$(1) = "*.BAK"
BAD$(2) = "*.$??"
BAD$(3) = "*.TMP"
BADSUM = 3 'Total number of file extensions destined for deletion
'Program starts here
OPEN BATFILE$ FOR OUTPUT AS #2
DO
FOR I% = 1 TO BADSUM
SHELL "DIR " + DRIVE$ + ":\" + BAD$(I%) + " /S /B > " + TEMPFILE$
OPEN TEMPFILE$ FOR INPUT AS #1
WHILE NOT EOF(1)
LINE INPUT #1, L$
PRINT #2, "DEL " + L$
WEND
CLOSE #1
NEXT I%
DRIVE$ = CHR$(ASC(DRIVE$) + 1)
LOOP UNTIL DRIVE$ > TOPDRIVE$
PRINT #2, "DEL " + TEMPFILE$
CLOSE
SHELL BATFILE$
KILL BATFILE$
SYSTEM
END
```

The program creates a batch file that deletes files with the specified extensions—a process that works fastest if you set the TEMPFILE$ and BATFILE$ variables as files in a RAM drive. When you're ready to run the file, just invoke the BASIC interpreter and load the file. At a DOS prompt, enter QBASIC /RUN KILLBAK to delete all relevant files except read-only files.

**ED QUILLEN**

### Dangerous Stop Sign

You wouldn't expect one character to cause you to lose half your spreadsheet, yet the end-of-file (EOF) mark could do just that. E-mail downloads, exported files, and other working files often contain the EOF character, ASCII 026, which is usually shown as ^Z or a right-pointing arrow. Usually, it's at the end of the file where you'd expect it to be, but if you merge files, those little arrows could be anywhere

*CONTINUED ON PAGE 308*