User Guide

for version 3.3

Contents

CONVENTIONS AND TERMS

Sans Serif indicates a **keyboard key**, a **dialog item**, or a **Windows menu item**. In addition to simple emphasis, *italic* indicates the introduction of an *important term* or *proper name*. *Italic* is also used for *program file names*.

```
Indented monospaced text indicates a command line
```

Real mode refers to a processor mode of operation in which 16-bit MS-DOS® programs run most efficiently, with direct access to hardware. In Microsoft® Windows®, DOS programs run in a *Virtual DOS Machine* (*VDM*) in *virtual-8086 mode,* a subset of the 32-bit *protected mode* in which Windows runs. Programs running in Windows share processor time and do not normally have direct access to system hardware.

Note that you cannot get to real mode directly from 32-bit Windows without rebooting. Windows 95 & 98 let you to start real-mode MS-DOS at boot time by pressing the **F8** key immediately before Windows starts, then selecting **Command Prompt Only** from the menu, and also let you reboot to real-mode MS-DOS from a shortcut, with custom config.sys and autoexec.bat settings. But running real-mode MS-DOS with other 32-bit Windows versions typically requires a bootable MS-DOS diskette or CD-ROM, or a dual-boot system.

MHz is short for *megahertz* (millions of cycles per second). *GHz* is short for *gigahertz* (billions of cycles per second). *THz* is short for *terahertz* (trillions of cycles per second).

A *level 1 cache,* also called a *primary cache*, is a temporary storage area for often-used code and data that the microprocessor can access far more quickly than system memory.

Mo'Slo is a contraction of 'more slow' and rhymes with *mojo*.

WHAT'S NEW SINCE VERSION 2?

Obvious enhancements include two new slowdown methods, 0.01% slowdown increments, level 1 processor cache disabling, five processor emulations, delayed slowdown and slowdown for a specified duration, convenient slowing of a command prompt, and a simplified keyboard speed control that is disabled by default to reduce memory footprint. Mo'Slo Deluxe can now be run as a *Terminate but Stay Resident* program. All of these are described in more detail later in this document.

In addition, interruption of system functions has been virtually eliminated to avoid low-speed lockups, and memory footprint has been *reduced* to a mere 672 bytes plus DOS environment when keyboard control is not enabled (keyboard control uses an additional 896 bytes).

COMPATIBILITY

Mo'Slo Deluxe v.3.3 can be used on systems with Intel-compatible processors 80386 and later, running MS-DOS 5.0 and later and/or Windows, as well as in DOS and Windows emulators on other platforms.

INSTALLATION

To install, run the installation program *moslodx.exe*—in either DOS or Windows—and follow instructions. The default installation is in the *c:\moslo* directory, which will be created if it does not exist. All examples in this document assume *c:\moslo\moslo.com* is the path and filename of Mo'Slo Deluxe.

ABOUT MO'SLO DELUXE

Mo'Slo Deluxe lets you temporarily slow the effective speed of your computer while another software program or batch file executes. Mo'Slo Deluxe saves the system state, slows the system to a specified speed, then runs the specified program or batch file. When the program or batch file exits, Mo'Slo Deluxe restores the system to its previous state and exits also.

Mo'Slo Deluxe can optionally be loaded into memory as a terminate but stay resident (TSR) program. In this mode, the program returns to the MS-DOS command prompt but leaves the slowdown engine in memory (see TERMINATE BUT STAY RESIDENT, page 7).

Mo'Slo Deluxe is an MS-DOS program that slows programs that run under MS-DOS. It does not slow Windows-only programs, and has no Windows graphical interface. Instructions for what to slow and how to slow it must be included in the command line that runs Mo'Slo Deluxe. This makes it useful in batch files and Windows shortcuts, where the same slowdown parameters must be used each time a particular program is run. Mo'Slo Deluxe can slow many MS-DOS programs in Windows, but it is most effective in real mode MS-DOS.

ABOUT COMMAND LINES

A *command line* is a line of text the operating system interprets to run a software program. It may include a *path name* and *filename*, plus any *parameter*s required by the program.

Microsoft operating systems identify file types by their filename *extension*: the last period and subsequent letters in the name. Extensions *.com* and *.exe* identify program files. Extension *.bat* identifies a *batch file*, which is a text file consisting of a sequence of commands. In Windows Explorer and My Computer, these file name extensions are hidden by default (see UNHIDING FILENAME EXTENSIONS IN WINDOWS, page 8).

The path name is the complete description of a file's location, including the drive and directories. The backslash (\) is the *path separator*, separating drive, directories, and the file name. Directories are special files that contain the names and locations of other files and directories. Directories appear as *folders* in Windows Explorer and My Computer.

The behavior of Mo'Slo Deluxe is controlled by optional parameters called *switches*. A switch is a forward slash (/) followed by one or more letters and/or a number. The switches can be in any order, but they must occur between the Mo'Slo filename and the filename of the program you want to slow. Spaces between switches are optional, but spaces *must* be used between switches and path names.

To use Mo'Slo Deluxe to slow a program or batch file, the original command used to execute the program is used as a *parameter*, or *argument*, to Mo'Slo Deluxe. For example, here is a hypothetical command line to load an old program:

```
c:\progs\old\pcxtprog.exe
```

And here is a command line you might use to slow the program with Mo'Slo Deluxe:

```
c:\moslo\moslo.com /20 /m1 /d5 c:\progs\old\pcxtprog.exe
```

| DRIVE | DIRECTORY (FOLDER) | FILENAME (WITH EXTENSION) | SWITCHES (OPTIONS) | DRIVE | DIRECTORIES (FOLDERS) | FILENAME (WITH EXTENSION) |

The command line above tells the operating system to execute *c:\moslo\moslo.com*. The subsequent items on the command line are parameters for *moslo.com*.

Files and folders may be named with uppercase and lowercase letters in Windows, but command processing is *case-insensitive*. On a command line, *C:\MoSlo\MOSLO.COM* is equivalent to *c:\moslo\moslo.com*. Do not worry about capitalization in command lines.

TELLING MO'SLO DELUXE WHAT TO SLOW

Here is a command to slow a popular DOS communications program:

```
c:\moslo\moslo.com /p5 pcplus.exe
```

The command processor will load *moslo.com*, located on the *C* drive in the *moslo* folder, passing the rest of the command line to *moslo.com* as parameters. The */p5* switch tells moslo.com to slow the system to emulate a 166 MHz Pentium processor. *Pcplus.exe* is the filename of the program file that will be slowed.

Note that there is no path name preceding *pcplus.exe*. This command line assumes that the current default drive and directory is the same as that in which *pcplus.exe* is located. Since many programs and batch files look for other files in the default drive and directory, it's a good idea to change the default drive and directory to the location of the program file to be slowed before slowing it.

For example, the following sequence of commands changes the current drive to *D*, changes the current directory to *\games\mars* and then runs *mars1.exe* at the 80486 processor emulation speed:

```
d:
cd \games\mars
c:\moslo\moslo.com /p4 mars1.exe
```

Use of executable and batch file name extensions *.com*, *.exe*, and *.bat* is optional. When an extension is not given, Mo'Slo Deluxe will first look for a file with an extension of *.com*, then *.exe*, then *.bat*.

Parameters for Slowed Programs

Some programs and batch files require their own parameters. Mo'Slo Deluxe will pass parameters to a slowed program or batch file. Simply append the parameters to the tail of the command line as you normally would.

Slowing a Command Prompt

The *command prompt* switch is a forward slash followed by the letter 'c'. Use this switch in place of a path name to slow a command prompt (MS-DOS prompt). The following example slows a command prompt using default emulation speed and slowdown method:

```
c:\moslo\moslo.com /c
```

Any DOS program you run from the slowed prompt will also run slowed.

TELLING MO'SLO DELUXE HOW MUCH TO SLOW

The speed at which a slowed program or batch file is to be run can be specified as *percent-of-system-speed* or a *processor emulation* or both. If no speed is specified, Mo'Slo Deluxe approximates the speed of a 4.77 MHz 8088 processor.

Percent-of-system-speed

You specify percent-of-system-speed with a forward slash followed by a number (without the percent sign). You can specify a percentage from 0.01% through 99.99%. For example, when the following command is executed on a system with a 1 GHz processor:

```
c:\moslo\moslo.com /3.3 c:\rd\rdprog.com
```

rdprog.com will execute as if the processor frequency were 3.3% of 1,000 MHz: 33 MHz.

Processor Emulation

Processor emulation approximates the speed of a specific processor, regardless of the speed of the current system. You specify processor emulation with a forward slash, followed by the letter 'p', followed by a number from 1–5. Mo'Slo Deluxe tests current system speed and applies a combination of slowdown speed, slowdown method, and cache disabling to approximate a specific processor. Here are the relationships:

EMULATION SWITCH	PROCESSOR	SLOWDOWN METHOD	CACHE DISABLED?
/p1	4.77 MHz 8088	2	yes*
/p2	~6 MHz 80286	2	yes*
/p3	~25 MHz 80386	2	yes*
/p4	~48 MHz 80486	2	no
/p5	166 MHz Pentium	1	no

in real mode only

With this example, Mo'Slo Deluxe will attempt to emulate a 25 MHz 80386 processor:

```
c:\moslo\moslo.com /p3 ultima.com
```

Emulated processor speeds are necessarily very loose approximations, because the same code may execute at very different speeds on processors of different types running at the same frequency. If you are trying to emulate an 80386 processor at 25 MHz, for example, you may need to fine-tune slowdown speed with a percent-of-system-speed argument in addition to the processor emulation switch. Mo'Slo Deluxe displays emulated speed before loading the slowed program, making it easier to estimate an approximate percent when the emulated speed is too fast or slow. If you feel that the /p3 switch runs your program twice as fast as you'd like, and you note that the emulated speed is 8% of normal, try the /p3 switch with the /4 switch. Example:

```
c:\moslo\moslo.com /p3/4 ultima.com
```

The /p1 and /p5 switches are typically more accurate for specific purposes. The /p1 switch emulates an 8088 IBM-PC for the purpose of running classic old animated games. The /p5 switch emulates a 166MHz Pentium processor for the purpose of loading and running early 1990's programs that suffer from run time errors 200 and R6003 at load time.

DISABLING THE LEVEL 1 CACHE

Intel-compatible processors since the 80486 rely on intelligent caching of code and data to achieve higher processing speeds. Caching stores oft-used code and data in high-speed memory close to the processor where it can be accessed much more quickly. This can be problematic for old DOS programs that rely on predictable execution speed throughout the program. Mo'Slo Deluxe disables the level 1 cache when you use processor emulations 1–3 (8088–80386) in real mode. In addition to slowing animation more smoothly, disabling the cache can overcome problems with some old programs that communicate with remote hardware through the RS232 serial port. You can force cache disabling with any slowdown method or processor emulation by using the no-cache switch, which is a forward slash followed by the letter n. Example, no-cache with method 1 at 0.48% system speed:

```
c:\moslo\moslo.com /n /.48 sprog.exe
```

Note that the cache can be disabled in <u>real mode only</u>. You cannot disable the cache when Windows or a protected mode memory manager such as EMM386 is running.

SLOWDOWN METHODS

Mo'Slo Deluxe implements three slowdown methods, each using a different system timer. Method 1 uses the programmable interval timer (PIT) with 55 ms slowdown intervals. Method 2 uses the realtime clock (RTC) with 7 ms intervals. Method 3 uses the advanced programmable interrupt controller (APIC) with <1 ms intervals. Each method has advantages and shortcomings. Method 1 is used by default when neither method nor processor emulation are specified, and also with processor emulation 5. Method 1 works to some degree in all DOS-compatible environments, but the 55 ms interval can cause jerky animation at low percentages. Method 2—used by default with processor emulations 1 through 4—provides smooth animation, but cannot overcome load-time divide errors common in programs developed in the early 1990s. Method 3 is smooth and effective for text *and* graphical programs. Since the APIC timer is not used by old DOS programs, there is no chance of conflict. Method 3 would be the default method were it not for two shortcomings: (1) it cannot be used in protected mode and therefore will not work in Windows or while EMM386 expanded memory manager is active, and (2) the APIC is not supported with processors prior to Intel Pentium 3 and AMD Athlon.

Methods 2 and 3 halt processor operation for part of each slowdown interval when running in real mode, reducing power consumption and heat.

Specify slowdown method with a forward slash, followed by the letter 'm', followed by the number 1, 2 or 3. Example, method 2 at default speed:

```
c:\moslo\moslo.com /m2 pinball.com
```

The method switch overrides the default method of a processor emulation. This example specifies method 3 with 166 MHz Pentium processor emulation:

```
c:\moslo\moslo.com /m3/p5 pcplus.exe
```

DELAYING SLOWDOWN AND LIMITING DURATION

Delay can be used when a program needs slowdown to run but not to load. Duration can be used when a program needs slowdown to load but can run at full speed.

Specify delayed slowdown, in seconds, with a forward slash, followed by the letter 'd', followed by a number from 1–14. Specify duration of slowdown, in seconds, with a forward slash, followed by the letters 'du', followed by a number from 1–14. Duration begins when slowdown begins, i.e. after any delay has expired. For example, the following command:

```
c:\moslo\moslo.com /d7/du14 c:\cb\cbstat.exe
```

will begin slowdown 7 seconds after cbstat.exe loads, and stop slowdown after another 14 seconds.

DISABLING THE COM PORT FIFO BUFFER

Old programs that communicate with external hardware through the RS232 serial port—such as radio servicing software and machine control applications—sometimes require that the FIFO buffer built into modern UART chips be disabled. You can tell Mo'Slo Deluxe to disable FIFO buffering for a specific COM port with a forward slash, followed by the letter 'f', followed by a number from 1–4. For example, '/f1' in the following command:

```
c:\moslo\moslo.com /f1/p3 c:\motorola\rdprog.com
```

disables the FIFO buffer on COM1 while Mo'Slo Deluxe is running.

KEYBOARD CONTROL AND QUICKLOAD

The *keyboard control* switch (*/k*) enables on-the-fly slowdown speed adjustment and on/off toggling of slowdown. For example, to run radmonk.exe with 80386 processor emulation and keyboard control enabled:

```
c:\moslo\moslo.com /k/p3 radmonk.exe
```

The *quickload* switch (*/q*) enables keyboard control and loads the specified program with slowdown toggled off. This is useful for programs that load very slowly when slowed. With quickload, you can load such programs at full speed. When the program is loaded, toggle slowdown on using the keyboard. Note that there is no need to use the keyboard control switch when you use the quickload switch.

To use keyboard control:

Press **Ctrl** + **Alt** + right **Shift** to toggle slowdown on/off.
Press **Ctrl** + **Alt** + '-' to reduce %-of-normal speed.
Press **Ctrl** + **Alt** + '=' to increase %-of-normal speed.

Feedback appears at the upper left of the display.

It is possible for a slowed program to disable keyboard control. For example, if you slow the Windows NT/2000/XP/Vista command processor, keyboard control is disabled at the command prompt. However, control is re-enabled when a DOS program is run from the prompt.

TERMINATE BUT STAY RESIDENT

The *terminate but stay resident* switch (*/r*) tells Mo'Slo Deluxe to initialize slowdown code, then exit back to DOS, leaving slowdown functionality resident in memory. Slowdown remains resident until it is removed by running Mo'Slo Deluxe with the *remove resident* switch (*/rr*), or until the system is shut down or rebooted. All slowdown parameters are valid except the command prompt switch and filename, which do not raise errors but are ignored. Example, loading as TSR with keyboard enabled, 80386 emulation:

```
c:\moslo\moslo.com /p3/k/r
```

If an upper memory manager such as EMM386.EXE is installed, Mo'Slo, Deluxe can be loaded in upper memory with the loadhigh (LH) command (refer to your MS-DOS documentation for details).

The resident code can be removed from memory provided that no other program has hooked the same system interrupts. On removal, the resident code restores system state and releases all of its memory for use by other programs. The command to remove resident code is:

```
c:\moslo\moslo.com /rr
```

ABOUT WINDOWS SHORTCUTS

A shortcut is a small data file that tells Windows where another file is located and how to open it. Shortcuts are also called *links*. A shortcut can refer to any file type recognized by Windows, including web pages, but they most commonly refer to program files on your computer. There are typically many shortcuts on the Windows desktop, displayed as icons. Program names in Start\Programs are also likely to be shortcuts, not actual executable files.

CREATING A DESKTOP SHORTCUT

To create a desktop shortcut, right-click on the desktop to pop up a menu, highlight **New** to pop up another menu, then click **Shortcut** to open the Shortcut Wizard. In the box labeled **Command line** or **Type the location**…, enter the path and filename of the program or batch file you want to run slowed. If you are unsure of the path or name, click the **Browse** button to find and select the file. When you have the filename in the text box, click **Next**. Enter a name for the shortcut (this is the name that will appear under the icon on the desktop). Then, if there is a **Next** button, click it and select an icon, then click **Finish**. Otherwise, just click **Finish**.

A shortcut to the program to be slowed will now appear on the desktop. You must then edit the shortcut properties to run the program slowed.

ADDING SLOWDOWN TO SHORTCUT PROPERTIES

To edit shortcut properties, right-click the shortcut's icon or text . When a menu appears, click **Properties** to display the shortcut's properties box. Click the **Program** tab. In the box labeled **Cmd line**, delete the drive letter and path, leaving only the name of the file. Then, to the left of the file name, type *c:\moslo\moslo.com*, followed by any parameters, followed by a space. Leave the box labeled **Working** as is.

As an example, assume you want to edit a shortcut to *c:\games\old\oldprog.exe* in order to slow the program to 4.77 MHz 8088 emulation. The **Cmd line** initially reads:

```
c:\games\old\oldprog.exe
```

Delete everything to the left of *oldprog.exe* and insert *moslo.com*'s path and filename plus processor emulation parameter. The **Cmd line** now reads:

```
c:\moslo\moslo.com /p1 oldprog.exe
```

The **Working** box reads:

```
c:\games\old
```

Uncheck the **Close on Exit** box and click **Apply**. Leave the properties box open and double-click the shortcut on the desktop to test slowdown. If necessary, add or adjust parameters in the **Cmd line** as necessary, clicking **Apply** after each change. When the shortcut is working as desired, check the **Close on Exit** box and click **OK**.

NOTE: In NT-based Windows, if a shortcut refers to a batch file there will be a different properties box, with a **Shortcut** tab instead of a **Program** tab, and with text boxes labeled **Target** and **Start in** instead of **Cmd line** and **Working**. In this case, add the *moslo.com* path, filename and parameters to the **Target**, then click **OK**. The icon will change, and you can then access DOS shortcut properties.

There are other settings in the shortcut properties that can affect slowdown. More consistent slowdown can sometimes be achieved by setting the **Idle sensitivity** slider under the **Misc** tab all the way to **Low**. You may also find that some programs that would normally display in a window work better **Full screen**.

LIMITATIONS OF WINDOWS MS-DOS EMULATION

Mo'Slo Deluxe can be used to slow MS-DOS programs in Windows, but it is most effective in real mode MS-DOS. To understand why this is so, consider that a 16-bit DOS program running in Windows shares processor time with 32-bit processes. The system rapidly switches back and forth between 32-bit protected mode and 16-bit virtual-8086 mode. Add to that the emulation of system interrupts, where both 32-bit and 16-bit service routines might be invoked hundreds of times per second, and you have a lot of processor time devoted to tasks other than the execution of the DOS program. Consequently, program execution in a Windows VDM can be erratic, especially in NT-based Windows.

UNHIDING FILE EXTENSIONS IN WINDOWS

Windows Explorer (My Computer) hides DOS file extensions by default. To show file extensions, double-click **My Computer**, click **Tools** (or **View** if there is no **Tools** item), then **Folder Options…** (or **Options…** in earlier Windows versions), and uncheck the box with a label that begins with **Hide file extensions…** or **Hide MS-DOS file extensions…** (wording is slightly different in different Windows versions).

TIPS & TROUBLESHOOTING

Slowdown that appears to be insufficient may actually be too much. Below an ideal level of speed reduction the system may appear to speed up erratically.

When slowing an old DOS program to PC/XT or AT speeds, use of Delay or Quickload can speed load time, especially if there is a graphical introduction. With Quickload you can toggle slowdown off before exiting the program, avoiding a delay as the program is unloaded while the system is in a slowed state. (This tip assumes the slowed program does not take complete control of the keyboard.)

When slowing a program while EMM386 is running in 'AUTO' mode, or when slowing a program with its own protected mode memory manager, it is possible to use method 3 or cache disabling without an error message. But when the program switches to protected mode it may crash the system. If the program uses EMM386 when available but runs OK without it, you can turn off EMM386 with the command "`emm386 off`" or you can comment out the lines in your CONFIG.SYS file that load HIMEM.SYS and EMM386. Otherwise you must avoid using processor emulations 1-3 and/or method 3 with these programs.

If a program appears to lock up when loading full screen in Windows, press **Ctrl** + **Alt** + **Del** and end the task. Try again using less slowdown.

No software-based slowdown can be guaranteed to slow all programs effectively, but we'll try to find solutions when problems are known, so please let us know if an old program is not slowed satisfactorily. Of course, we really appreciate being informed of programs that *are* slowed satisfactorily, and of the parameters that give best results!

TECHNICAL SUPPORT

If you need help, please visit the Mo'Slo website at http://moslo.info and check out the SUPPORT, FAQ and FEEDBACK sections. If you don't find a solution, email support@hpaa.com, with the information requested in the SUPPORT section. If you are more comfortable using the telephone, you can usually reach a support person at +1 (818) 884-7157 between 9am and 5pm Pacific Time, Monday through Friday.

It is financially unfeasible for us to return support calls, so if you get voice mail when you call, please leave a message with a brief description of the problem and a time when you plan to call again and we'll try to have someone available and prepared to help. To ensure you reach a person on the first call, pre-arrange a call time via email (support@hpaa.com). Please include the word "Mo'Slo" in the subject of your email.

END USER SOFTWARE LICENSE AGREEMENT

The Mo'Slo Deluxe software application (the Software) is owned by David Perrell and is protected by United States copyright laws and related international treaty provisions. License to use the Software is granted under the following terms and conditions:

1. You may use one copy of the Software on a single terminal or workstation of a single computer so long as all of the terms and conditions of this agreement are complied with.

2. The Software and its documentation are provided as a single executable archive or install program (the Archive). You may extract copies of the Software and documentation from the Archive onto a permanent storage device (e.g., hard disk) of one computer, you may print a copy of the documentation, and you may store the Archive on magnetic or optical media for archival purposes. You may not otherwise copy or distribute the Software or documentation.

3. You may not make, or have made by another, any modification to the Software or its accompanying documentation. You may not adapt, translate, reverse engineer, disassemble, or create derivative works based on the Software.

4. You may transfer the Software to another party PROVIDED (a) you transfer the Software and all accompanying documentation in the same form in which it was transferred to you; (b) you retain no copies of the Archive, the Software or the documentation; and (c) the recipient agrees to the terms of this License Agreement.

5. If the Software is an upgrade, you may use the Software only as a replacement for a previous version of the Software. When you determine that the Software upgrade is a suitable replacement you must destroy all copies of the previous version. If you receive a Software upgrade after you have transferred a previous version, you must either transfer the upgrade to the current possessor of the previous version or destroy it.

Download and/or installation of the Software implies agreement. If you do not agree to comply with these terms and conditions then do not download or install the Software. Prior to downloading the software you can request a refund from the place of purchase.

DISCLAIMER OF WARRANTY

Mo'Slo Deluxe is provided without warranty of any kind, express or implied, and the user assumes the entire risk of using it. The author will not be liable for any incidental, consequential, indirect or similar damages resulting from the use of this software. In no event will the author's liability ever exceed the price paid for the license to use the software, regardless of the nature of the claim.

If license to use this software has been transferred to you from another licensee, please visit our website and register as the new licensee. The address of the registration page is:

> http://www.hpaa.com/moslo/register

Registering will enable us to provide you with software updates and information.

Note that we de do not share customer data with anyone without a court order.

division of
Hearn/Perrell Art Associates
23022 Hatteras Street
Woodland Hills, CA 91367
+1 (818) 884-7151 (main)
+1 (818) 884-7157 (support/fax)