THE CMOS

By Mark E. Donaldson

INTRODUCTION AND OVERVIEW

A lot can happen with just the flip of a switch or the push of a button. You push a button on your PC, and in a few seconds up pops your operating system and you're ready to go. It all seems so simple, but if you stop to think about what has transpired in those few seconds, you're sure to be awed by the complexity of the process.

In the time between your turning on the machine and the appearance of the operating system, your system receives power; tests itself for possible problems; checks for a keyboard, hard disks, and most other hardware; and finds and loads your operating system. In order to do all of this successfully, your machine needs to know precisely how it is configured. This configuration information is stored in memory called CMOS.

CMOS (pronounced see-moss) stands for complementary metal-oxide semiconductor. This is a type of memory chip with very low power requirements, and in PCs it operates using small batteries. In PCs, CMOS is more specifically referred to as CMOS RAM. This is a tiny 64-byte region of memory that, thanks to the battery power, retains data when the PC is shut off.

The function of CMOS RAM is to store information your computer needs when it boots up, such as hard disk types, keyboard and display type, chip set, and even the time and date. If the battery that powers your CMOS RAM dies, all this information is lost, and your PC will boot with the default information that shipped with the motherboard. In most cases, this means you'll have no access to your hard disks until you supply CMOS with the necessary information. Without access to your hard disks, you won't be able to boot your operating system.

Fortunately, today's CMOS RAM is protected by nickel cadmium batteries, which the computer's power supply recharges. Even so, it's an extremely good idea to keep a copy of all the information stored in CMOS, in case disaster strikes.

The information stored in CMOS is required by your computer's Basic Input/Output System, or BIOS (pronounced bye-oss). Your PC contains several BIOSes--the video BIOS that interfaces your CPU and video card, for example, but the most fundamental is the system BIOS. The system BIOS is stored on a ROM (read-only memory) chip on the motherboard and is copied at boot time to a 64K segment of upper system RAM for faster system access (RAM is faster than ROM). The role of the system BIOS is to boot the system, recognize the hardware devices, and locate and launch the operating system. Once the operating system is loaded, the BIOS then works with it to enable access to the hardware devices.

If you're using DOS, access to the BIOS is usually direct, which means an application talks to the BIOS directly. If you're using applications written for Windows 95 or NT, however, access to the BIOS is controlled by the operating system, with applications programmed to make use of standard operating-system routines to access the BIOS. It's still possible for programmers to access the BIOS directly, but if they do so, they won't be able to display the Windows 95 logo on the product box.

POST

As soon as you turn on your machine, the BIOS runs a power-on self-test (POST), during which it completes a series of diagnostic tests and then checks for and initializes your system's hardware. You can watch this initialization on your screen:

First you'll see information about the video card, then information about the BIOS version itself, and then a test of your system's memory. Except for devices that contain their own BIOS information, for example, video cards and SCSI controllers, the BIOS initializes the devices according to the information stored in CMOS. The most obvious of these details concern the processor and hard disks, but a great many more are part of the boot sequence as well. You may not even notice the initialization checks on some components until the BIOS can't find them: If your keyboard isn't plugged in, for instance, or if your floppy disk drive is missing its interface cable, the BIOS will display an error message. After the POST is complete, the BIOS looks for and then loads the operating system.

CMOS SETUP PROGRAM

While the POST is in process, the BIOS gives you access to a small program, usually called Setup, that lets you examine and alter your CMOS settings. You'll see a message on your screen telling you to press a specific key (Del is common) or key combination to enter Setup, and you'll have a limited time to do so. Watch for this message, or simply hold down the appropriate key (if you know it) while the memory test is taking place. A screen will pop up with several menu options. Exactly what you'll see depends on which brand of BIOS your computer is running, but you'll probably see a menu called Standard CMOS Setup and another called Advanced CMOS Setup.

Be extremely careful when changing CMOS settings, as you can easily render your computer unbootable. Fortunately, to enter CMOS Setup, you don't need to load the operating system. Just turn on the machine and press the appropriate key combination. If your BIOS can't find the information it needs, it will inform you that you must enter CMOS Setup and make changes. Even better, today's BIOSes have auto-configuration features, which make CMOS Setup much easier than it was a few years ago.

When you exit CMOS Setup (procedures vary according to BIOS brand), you'll be asked whether you want to save changes. If you're not certain, don't save your changes. In either case, exiting will cause your machine to reboot, so you'll be able to tell immediately whether you've done something drastically wrong.

The Standard CMOS Setup lets you configure time and date, floppy and hard disk drive types, keyboard installation, and primary display type. This is the starting point for all PC BIOSes, because without a keyboard you can't do anything, without a display type you can't see anything, and without a disk drive you can't run anything. All other CMOS settings are supplemental to these few.

The CMOS Time and Date setting is self-explanatory. Use the adjustment keys (usually listed at the bottom of the screen) to change the time and date. DOS and Windows provide a means of making these changes from within the operating system, DOS's TIME and DATE commands and Windows' Date/Time utility in Control Panel, so it's not necessary to adjust the setting in CMOS. But if your computer is running slow or fast (typically you'll be off a minute or two each month, these ain't Swiss timepieces!), simply adjust it at this point.

The keyboard setting is also obvious. Either you have one installed (the CMOS default) or you don't. If you choose Not Installed as the option, the BIOS won't check for it during POST and, as a result, won't display an error message. You might want to do this if you suspect a faulty keyboard is preventing your computer from booting into the operating system properly, or if you're running a software demo or a Web kiosk station, allowing either no input or mouse input only.

THE CMOS

By Mark E. Donaldson

Your BIOS needs to know how to **display information** on initial boot-up. CMOS Setup gives you a choice among VGA (the standard display type) and older display types such as EGA (16-color) and possibly CGA (4-color) and MDA (text-only). Your particular BIOS might offer more. As long as you have a VGA card installed (and it's pretty much impossible to buy a non-VGA card these days), select VGA. On today's machines, VGA is the default.

CMOS Setup gives you two options for **floppy disk drives**: Drive A and Drive B. Most computers today ship with only one floppy disk drive, but you can add another if you wish. Using the appropriate keys (typically PgUp and PgDn), you can cycle through the various floppy disk drive types: 360K (5.25-inch), 720K (3.5-inch), 1.2MB (5.25-inch), 1.44MB (3.5-inch), and 2.88MB (3.5-inch). The most common is 1.44MB. More recently introduced drives of 100MB and more are not supported directly in BIOS but depend instead on a separate driver.

Hard disk configuration is by far the most complex task in CMOS Setup. Fortunately, BIOSes appearing within the last couple of years usually contain built-in configuration utilities that recognize and configure hard disks automatically, but with older systems it's still necessary to perform the configuration manually, configuring each hard disk individually.

Most of today's systems contain an EIDE (enhanced integrated drive electronics) drive controller, which can handle up to four disks (including CD-ROM drives). If you have an older IDE controller or a controller even older than IDE, you're restricted to two hard disks (and no CD-ROM drive). On your CMOS Setup, IDE might be called ATA or AT-bus, and EIDE might be labeled Fast-ATA.

Unless you bought identical hard disks, it's entirely likely that each disk in your system must be configured with different parameters. To configure them, you need information that can be provided only by the disk manufacturer. If you can't locate that information on the disk itself, it might be in the disk's manual. If, like most people, you have no idea what happened to the manual, then you'll have to call the disk manufacturer. But even if your system is working fine, you should enter CMOS Setup, write down the information about your hard disks, and store this information in a safe place in case anything goes wrong. Taping a sheet to the computer itself isn't a bad idea.

Note that if you have SCSI hard disks, their configuration depends on the SCSI controller, not your BIOS, and you don't configure them with CMOS Setup.

Each hard disk can be given a specific number denoting its type. On older systems, one of the preconfigured numbers (you'll likely have 46 of these) might be applicable. In most systems today, however, the disk number is 47, which means you must provide the parameters yourself. These parameters include the following, and again, if your CMOS Setup displays them, write them down for safekeeping:

Cylinders: This value shows the number of cylinders on your hard disk.

Heads: This value represents the number of heads on your hard disk.

Sectors: This value shows the number of sectors per hard disk track.

Size: This value refers to the size of your hard disk, expressed in megabytes (MB). Don't worry if this number is at slight variance with the size of the disk shown on the box in which it was shipped or in the manual. You will still have access to all the data storage space.

Write Precompensation (WPCom): On IDE, EIDE, or SCSI hard disks, it's a meaningless value, because IDE/EIDE drives contain the necessary information themselves, and SCSI disks are handled by their controllers. On pre-IDE disks, however, the BIOS had to compensate for the difference in the density of data on the disks' inner tracks versus the outer tracks, and the WPCom value accomplished that. Usually, the number is either 0 or 65535.

Logical Block Addressing (LBA): This is an addressing scheme that's at the heart of EIDE disk control. LBA assigns a number to each sector on the hard disk, a more efficient way to address hard disks than the older system of cylinder, head, and sector (CHS) addressing. LBA has always been used for SCSI disks, but IDE disks started to use this as an alternative addressing method in 1994. This happened to be when hard disks larger than 504MB (the maximum for CHS BIOSes) started to appear, but LBA is not necessary for large hard disks to function. What is necessary is a BIOS that can perform CHS-to-CHS translation (to map the cylinders, heads, and sectors properly), a scheme known as ECHS or XCHS. LBA is one means of allowing the BIOS to communicate with a hard disk and performing the necessary translation, and in effect it has become the standard. LBA can support disk sizes up to 128GB.

Usually it's not necessary to configure the LBA settings manually for hard disks, because BIOSes today handle the setup through auto-configuration. Because older BIOSes do not support LBA, however, manufacturers of large hard disks include a floppy disk containing a driver that allows you to use the entire hard disk.

That's it for standard CMOS settings. Next issue, we'll head into the advanced settings and beyond, including information about how to optimize your system for best possible performance.

The CMOS setup on your computer may differ from those that we discuss here. Different BIOS makers offer different options, and their CMOS setup programs group these options into various categories. For the most part, however, the similarities will outweigh the differences. If you find an option not listed here, use your CMOS setup program's help menu to see what the option does.

Be aware that changing your CMOS settings can prevent your computer from starting up properly. If your machine does not boot, try entering CMOS again, usually it will get this far, and select the default settings (or whatever name that option bears). If that fails, try powering down the computer, then holding down the Del key while you power it back up. This should bypass the BIOS options and allow your machine to start. If that fails, call your dealer. The real trick is to change only one option at a time, then save the CMOS settings and reboot the machine to see whether your system behaves any differently.

These options are typically subsumed under a setup area labeled Advanced Options, or something similar. On your computer, they might be split into two or more categories. Note that in the recommendation for each feature, the terms On and Off are usually used. On some BIOSes, On and Off correspond to Enabled and Disabled, respectively.

Above 1MB Memory Test: The BIOS checks system memory for errors or malfunctions. It checks the first megabyte automatically, but it will check all your memory if you tell it to. This seems like a good idea, except that DOS and Windows offer their own verification through system drivers, so the BIOS check is

unnecessary, especially since it slows boot time by several seconds. Turn it off for better boot-up speed, but toggle it on if you're finding small, inexplicable errors in your applications. (This option is called Quick Boot in some BIOSes, in which case the recommendation should read On). Recommendation: Off.

Memory-Test Tick Sound: This is the clicking sound that accompanies the memory count at boot time. Turn it off if it annoys you. Leave it on if you want an audible confirmation that your system memory is working well. Recommendation: Off.

System-Boot Number Lock: By default, most PCs boot with the Num Lock key turned on. If you don't need this (and many users don't), you can toggle it off from CMOS setup. Recommendation: User preference.

Wait for F1 on Error: By default, if your system encounters a nonfatal error (one that will still allow the operating system to start) at boot time, everything will halt until you press the F1 key. Toggle this feature off if you want to start a machine that doesn't have a keyboard, or if you just
want to take your chances. Recommendation: On.

Typematic Rate: The typematic rate, expressed in characters per second, is the rate at which a key repeats (how many times it prints to the screen in a second) if you hold it down. You can set the rate to suit your preference, the choice is simply Fast or Slow on some BIOSes, but usually your system default will suffice. The typematic rate can also be set through the Control Panel in Windows. Recommendation: System default.

Typematic Rate Delay: This is the length of time between when you hold a key down and when it will begin to auto-repeat. If you have heavy fingers or a disability that makes it difficult to lift your fingers quickly off the keys, set this to a higher value. The typical value is 500 nanoseconds or so, but you should experiment with it. This delay can also be set through the Windows Control Panel. Recommendation: System default.

Numeric Processor Test: If your processor contains a math coprocessor, this setting should be toggled on. All 486DX and faster processors have math coprocessors built in, so on most systems this option will automatically be enabled. If you have a 486SX or slower processor and you add a math coprocessor, be sure to toggle the setting on. Recommendation: Depends on processor.

Weitek Coprocessor: Some 386 motherboards shipped with Weitek math coprocessors. These are faster than most coprocessors, because they use system RAM, and that requires an allocation of address space. Toggle the setting on only if you have this coprocessor. Recommendation: Off.

Floppy Disk Drive Seek at Boot: By default, most systems will look for floppy disk drives when they boot. That's because operating systems used to load from floppies, and the system had to recognize the drive before it could be loaded. Floppy disk drives are rarely used to boot computers anymore, so there's little need for your system to perform this search. Recommendation: Off.

System Boot Sequence: This setting tells the BIOS on which drive to find the operating system. The standard sequence used to be A:, C:, this meant that the BIOS would check the floppy disk drive first. Then, if a floppy disk wasn't present, it would look to the primary hard disk for the operating system. This sequence is the reason your boot-up halts if you leave a nonbootable floppy in the drive. If you're sure you won't need to boot from floppies, choose the sequence C:, A: instead; this speeds the boot

sequence, because the floppy disk drive isn't checked. This is not recommended for users who don't know anything about CMOS setup, however; the sequence C:, A: often hangs after a failure to find the operating system on C:, which means the user can't just slip a bootable floppy into the drive and keep going. You have to reboot. Newer BIOSes let you specify a CD-ROM drive in the boot sequence as well, for systems (such as Windows NT Server 4.0) that ship with their operating systems on bootable compact disks. Recommendation: C:, A: for experienced users.

External/Internal Cache Memory: The external memory (or L2 cache memory) setting refers to the high-speed RAM that speeds data transfer between the processor and main memory. All current systems ship with L2 cache memory, and this setting should always be enabled. If you notice that your machine is lethargic, check to make sure that the L2 cache memory setting hasn't automatically switched itself off, as may happen during failed memory checks at start-up. The internal cache memory, or L1 cache, is located on the processor itself and should be turned on if your machine has a 486 or faster processor. These settings are listed separately on some BIOSes.

On recent BIOSes, you can also specify that each cache use the WriteBack or WriteThru caching algorithm; the former is the default, and the latter (which is slightly faster) should be used only if you are unafraid of a possible (but unlikely) loss of data should the PC's power go down. Recommendation: On for cache, WriteThru for type.

Hit DEL Message Display: Each BIOS offers its own key combination for entering the CMOS setup utility during the boot sequence. This option lets you enable or disable the message that appears on-screen telling the user what key to press. Toggle it off if you don't want users even knowing that CMOS setup is possible. Recommendation: On.

Turbo Switch: If your PC has a turbo switch, allowing you to switch from the current blazing speed of your processor to the very slow speed of the first PCs (8 MHz) for the sake of compatibility with old software, you can choose to have the BIOS control the switch's function. Recommendation: On.

Shadowing refers to the copying of ROM-based memory from the BIOS or from peripheral devices (such as video cards or network cards) into the same addresses in main RAM. In some cases, you can specify that the shadow be cacheable (in other words, able to be copied into main RAM and accessible via the L2 cache). Since RAM is about three times as fast as ROM, successful shadowing improves system performance, and the even faster cache RAM further aids performance. But you have to experiment carefully with shadowing to avoid conflicts with hardware trying to use the same memory addresses as the shadow. For this reason, you should change these settings one at a time, rebooting each time to see that everything still works properly. If there's a problem, reenter your CMOS setup program and reverse the last adjustment, then reboot and try another. Generally speaking, shadowing on PCs works well only with DOS and Windows systems; it doesn't work with PC-based Unix systems such as Linux.

These options usually have three settings. Disabled means that shadowing will not occur and there's no chance of address conflicts. This is the safest setting, but it does nothing to enhance performance. Shadow enables shadowing into main RAM but does not make this address cacheable. Cache both shadows the ROM into main RAM and allows it to be cached. By default, all shadowing is disabled and must be toggled on through the CMOS setup utility.

System BIOS Shadow Cacheable (also called System ROM Shadow F000): This option lets you copy the F0000h address segment of system memory (that of the BIOS itself) from slower ROM to faster L2 cache RAM. This provides the fastest possible access by the CPU to the BIOS. Recommendation: On.

Video ROM Shadow C000: The BIOS area between memory addresses C0000h and C8000h, which is reserved for the EGA/VGA BIOS, can be shadowed and made cacheable with this option toggled on. Recommendation: On (but test to make sure your video works).

Depending on your BIOS, you may be offered several other shadowing opportunities. These will come under headings such as C400 Shadow 16K, D800 Shadow, Adapter ROM Shadow E800, and so on. By default, these are disabled, and you should check with your peripheral manufacturer to see whether shadowing them is a good idea. D000, for example, is a typical setting for network cards, and shadowing this area makes a lot of sense in a busy LAN. But it makes no sense at all if shadowing the memory area will result in more than one device competing for the same address. Often, experimenting--one setting at a time, is the only way to determine success.

Although **power management** on desktop computers is nowhere near as efficient as it is on notebooks, most recent PC BIOSes allow you to adjust certain power settings. The primary idea behind this is to cut power to specific devices when they're not in use. A secondary purpose is to "wake up" the machine when other devices are activated. You can set your fax machine so that an incoming fax call powers on the computer, allowing the fax modem to receive the call.

At this stage in PC development, each BIOS (and motherboard) has its own unique power management options. Here, we'll cover some of the choices seen on the majority of BIOSes. The labels here come from the American Megatrends BIOS (AMIBIOS).

Full-On-to-Standby Time-Out and Standby-to-Suspend Time-Out: Computers capable of power management typically have three modes: Full-On, Standby, and Suspend. Full-On means that all devices are powered on at all times, whereas Suspend means that only a minimal amount of power is reaching the system, just enough to stop it from shutting down completely. A system in Standby mode powers back up more quickly than a system in Suspend mode. You can set the length of time the machine can sit inactive before switching to a lower-powered mode. Generally, you can set each such switch in 1-minute increments up to a maximum of 15 minutes, although various BIOSes offer various settings. Setting both time-outs to 10 minutes, for example, means that after 10 minutes of system inactivity, the system will switch from Full-On to Standby mode; then, after another 10 minutes, it will switch to Suspend mode. In each case, pressing any key will switch the machine back to Full-On mode. Recommendation: 15 minutes for both (experiment for your own preferences).

Hard Disk Power-Down Mode and Hard Disk Time-Out: Because decreasing the power to the hard disk results in such a long delay when powering back up (you can wait several seconds for a hard disk to respond to the first request from the operating system), controlling power to the hard disk is separate from controlling power to the computer as a whole. You can set the hard disk to power down to Standby or Suspend mode when inactivity reaches the number of minutes specified in the hard disk time-out option. In other words, it's possible for the computer as a whole to switch to Suspend mode while the hard disk remains in Standby mode. Such a setting would result in a faster return to full power. Recommendation: Standby, at the maximum time-out setting.

IRQ 3 throughIRQ 15: Each IRQ setting can be monitored individually to determine when an event occurs that requires it. When the event occurs while the computer is in Standby or Suspend mode, an event on the monitored IRQ will switch the computer to Full-On mode automatically. An example would be a fax modem set at IRQ 3 that receives an incoming fax call while the computer is in Suspend mode. Without this option, the computer would ne ed to be set to Full-On at all times in order to receive faxes. Recommendation: Off, except for specific devices as required.

Depending on your BIOS, you'll have a range of possible settings in addition to those discussed here. PCI-based motherboards typically offer an entire section on PCI settings, including enabling or disabling PCI Burst Mode (for performance boosts) and settings for an off-board PCI IDE card (one not found on the motherboard itself). You may also be able to set the BIOS to be aware of plug-and-play capability, in order to assist your operating system (Windows 95 in particular) in recognizing peripheral cards. Other peripheral settings are handled in the CMOS setup program either through the Advanced CMOS Settings menu or a separate Peripheral Setup menu. These include Parallel Port Mode (you have four choices: normal, bidirectional, Enhanced Parallel Port, and Extended Capabilities Port, with the choice depending mostly on your printer) and toggling on or off the floppy disk drive and serial port controllers on the motherboard. Finally, the CMOS setup utility may offer options for establishing system passwords; use these if you want to ensure that you're the only person who can boot the machine.

The trick to using the CMOS setup program is to change one setting at a time until everything is working optimally, and then to write down all the settings in case you have to set them from scratch later. Most of the time, your system will arrive from the dealer with perfectly functional CMOS settings already in place, but if you want full control over your system's performance, you owe it to yourself to master your CMOS setup program.