

# How The BIOS Works

A typical BIOS (Basic Input/Output System) ROM chip contains three types of code: startup, test, and run-time services. The startup and test make up the POST (power on self test) portion of the BIOS' duties.

This diagram gives a high-level overview of typical POST functions, presented in a brief,

step-by-step walk-through format rather than listing the hundreds of individual POST commands. Keep in mind that different BIOSes carry out POST operations in different orders, so our example may not precisely follow the procedure your PC's BIOS uses.

After you turn the power on, the CPU reads the first 16 bytes of instructions contained in the BIOS and initiates the POST process, which consists of initialization and test routines.

**CPU CMOS**  
During the POST, the BIOS verifies the system data it is collecting with that stored in the CMOS (complementary metal-oxide semiconductor).

**DMA**  
BIOS initializes the DMA (Direct Memory Access) controller.

**RAM & DRAM**  
The BIOS initializes and tests system DRAM (dynamic RAM) and RAM.

**Timer**  
BIOS initializes the system timers.

**CPU Cache**  
External and CPU memory caches are now enabled.

**Shadow RAM**  
The BIOS expands RAM into Shadow RAM, a portion of system memory set aside for storing BIOS commands until they are executed.

**Option ROMs**  
The BIOS now searches for secondary ROMs, (BIOSes located on or within various components such as video cards) and if need be, updates itself with the capabilities of new hardware.

**Power Management**  
BIOS initializes the Power Management hardware capabilities.

**Basic & Extended RAM**  
The BIOS tests the basic and extended RAM. The basic RAM consists of the first 640KB, and the extended RAM consists of everything above 640KB.

**Video Adapter, Keyboard, Date & Time**  
The BIOS now sets up hardware interrupts, initializes the video capabilities of the system, and sets the system date and time by querying the RTC (real-time clock) and CMOS. At this point, the POST operations are complete and the system is ready.

**Registers & Controllers**  
System initialization begins. The CPU, following the BIOS' instructions, initializes system hardware, including the CPU registers, the CPU memory caches, local bus IDE (Integrated Drive Electronics), and the keyboard controller.



**BIOS**  
The BIOS above is a small chip found on every motherboard. It is much smaller than the up-close version we depict on these two pages.

**Bootable Diskette**  
The BIOS searches for a bootable disk, beginning with the PC's floppy diskette drive (often labeled A:) and moving on to the primary hard drive (usually C:). Once it finds one, it passes control to a loader program, which loads the operating system. In the MS-DOS environment, for example, the loader program initiates the Io.sys and Sysinit files, which in turn load the MS-DOS kernel and eventually start the familiar Config.sys, Command.com, and Autoexec.bat files. Later Windows versions, of course, implement the boot sequence differently. For example, Windows 95's Io.sys file incorporates the old MSDOS Io.sys, Mmsdos.sys, Config.sys, and Autoexec.bat into one 32-bit file.

