



PhoenixBIOS 4.0 Release 6.0

POST Tasks and Beep Codes

When you turn on or reset an IBM-compatible PC, the BIOS first performs a number of tasks, called the **Power-On-Self-Test (POST)**. These tasks test and initialize the hardware and then boot the Operating System from the hard disk.

At the beginning of each POST task, the BIOS outputs the **test-point error code** to I/O port 80h. Programmers and technicians use this code during trouble shooting to establish at what point the system failed and what routine was being performed. Some motherboards are equipped with a seven-segment LED display that displays the current value of port 80h. For production boards which do not contain the LED display, you can purchase an installable "Port 80h" card that performs the same function. If the BIOS detects a terminal error condition, it issues a terminal-error beep code (See following), attempts to display the error code on upper left corner of the screen and on the port 80h LED display, and halts POST. It attempts repeatedly to write the error to the screen. This attempt may "hash" some CGA displays.

If the system hangs before the BIOS can process the error, the value displayed at the port 80h is the last test performed. In this case, the screen does not display the error code.

Copyright © 1997 by Phoenix Technologies Ltd.
All Rights Reserved
Phoenix Technologies Ltd.
Desktop Division
135 Technology Drive
Irvine, CA 92618
<http://www.phoenix.com>

Terminal POST Errors

There are several POST routines that require success to finish POST. If they fail, they issue a **POST Terminal Error** and shut down the system. Before shutting down the system, the error handler issues a beep code signifying the test point error, writes the error to port 80h, attempts to initialize the video, and writes the error in the upper left corner of the screen (using both mono and color adapters).

The routine derives the beep code from the test point error as follows:

1. The 8-bit error code is broken down to four 2-bit groups.
2. Each group is made one-based (1 through 4) by adding 1.
3. Short beeps are generated for the number in each group.

Example:

Testpoint 16h = 00 01 01 10 = 1-2-2-3 beeps

POST Task Routines

The following is a list of the Test Point codes written to port 80h at the start of each routine, the beep codes issued for terminal errors, and a description of the POST routine. Unless otherwise noted, these codes are valid for PhoenixBIOS 4.0 Release 6.0.

NOTE: The following routines are sorted by their test point numbers assigned in the BIOS code. Their actual order as executed during POST can be quite different.

Code	Beeps	POST Routine Description
02h		Verify Real Mode
03h		Disable Non-Maskable Interrupt (NMI)
04h		Get CPU type
06h		Initialize system hardware
08h		Initialize chipset with initial POST values
09h		Set IN POST flag
0Ah		Initialize CPU registers
0Bh		Enable CPU cache
0Ch		Initialize caches to initial POST values
0Eh		Initialize I/O component

Code	Beeps	POST Routine Description
0Fh		Initialize the local bus IDE
10h		Initialize Power Management
11h		Load alternate registers with initial POST values
12h		Restore CPU control word during warm boot
13h		Initialize PCI Bus Mastering devices
14h		Initialize keyboard controller
16h	1-2-2-3	BIOS ROM checksum
17h		Initialize cache before memory autosize
18h		8254 timer initialization
1Ah		8237 DMA controller initialization
1Ch		Reset Programmable Interrupt Controller
20h	1-3-1-1	Test DRAM refresh
22h	1-3-1-3	Test 8742 Keyboard Controller
24h		Set ES segment register to 4 GB
26h		Enable A20 line
28h		Autosize DRAM
29h		Initialize POST Memory Manager
2Ah		Clear 512 KB base RAM
2Ch	1-3-4-1	RAM failure on address line xxxx*
2Eh	1-3-4-3	RAM failure on data bits xxxx* of low byte of memory bus
2Fh		Enable cache before system BIOS shadow
30h	1-4-1-1	RAM failure on data bits xxxx* of high byte of memory bus
32h		Test CPU bus-clock frequency
33h		Initialize Phoenix Dispatch Manager
36h		Warm start shut down
38h		Shadow system BIOS ROM
3Ah		Autosize cache
3Ch		Advanced configuration of chipset registers
3Dh		Load alternate registers with CMOS values
42h		Initialize interrupt vectors
45h		POST device initialization
46h	2-1-2-3	Check ROM copyright notice
48h		Check video configuration against CMOS
49h		Initialize PCI bus and devices
4Ah		Initialize all video adapters in system
4Bh		QuietBoot start (optional)
4Ch		Shadow video BIOS ROM

Code	Beeps	POST Routine Description
4Eh		Display BIOS copyright notice
50h		Display CPU type and speed
51h		Initialize EISA board
52h		Test keyboard
54h		Set key click if enabled
58h	2-2-3-1	Test for unexpected interrupts
59h		Initialize POST display service
5Ah		Display prompt "Press F2 to enter SETUP"
5Bh		Disable CPU cache
5Ch		Test RAM between 512 and 640 KB
60h		Test extended memory
62h		Test extended memory address lines
64h		Jump to UserPatch1
66h		Configure advanced cache registers
67h		Initialize Multi Processor APIC
68h		Enable external and CPU caches
69h		Setup System Management Mode (SMM) area
6Ah		Display external L2 cache size
6Bh		Load custom defaults (optional)
6Ch		Display shadow-area message
6Eh		Display possible high address for UMB recovery
70h		Display error messages
72h		Check for configuration errors
76h		Check for keyboard errors
7Ch		Set up hardware interrupt vectors
7Eh		Initialize coprocessor if present
80h		Disable onboard Super I/O ports and IRQs
81h		Late POST device initialization
82h		Detect and install external RS232 ports
83h		Configure non-MCD IDE controllers
84h		Detect and install external parallel ports
85h		Initialize PC-compatible PnP ISA devices
86h		Re-initialize onboard I/O ports.
87h		Configure Motheboard Configurable Devices (optional)
88h		Initialize BIOS Data Area
89h		Enable Non-Maskable Interrupts (NMIs)
8Ah		Initialize Extended BIOS Data Area
8Bh		Test and initialize PS/2 mouse
8Ch		Initialize floppy controller

Code	Beeps	POST Routine Description
8Fh		Determine number of ATA drives (optional)
90h		Initialize hard-disk controllers
91h		Initialize local-bus hard-disk controllers
92h		Jump to UserPatch2
93h		Build MPTABLE for multi-processor boards
95h		Install CD ROM for boot
96h		Clear huge ES segment register
97h		Fixup Multi Processor table
98h	1-2	Search for option ROMs. One long, two short beeps on checksum failure
99h		Check for SMART Drive (optional)
9Ah		Shadow option ROMs
9Ch		Set up Power Management
9Dh		Initialize security engine (optional)
9Eh		Enable hardware interrupts
9Fh		Determine number of ATA and SCSI drives
A0h		Set time of day
A2h		Check key lock
A4h		Initialize Typematic rate
A8h		Erase F2 prompt
AAh		Scan for F2 key stroke
ACh		Enter SETUP
A Eh		Clear Boot flag
B0h		Check for errors
B2h		POST done - prepare to boot operating system
B4h	1	One short beep before boot
B5h		Terminate QuietBoot (optional)
B6h		Check password (optional)
B9h		Prepare Boot
BAh		Initialize DMI parameters
BBh		Initialize PnP Option ROMs
BCh		Clear parity checkers
BDh		Display MultiBoot menu
BEh		Clear screen (optional)
BFh		Check virus and backup reminders
C0h		Try to boot with INT 19
C1h		Initialize POST Error Manager (PEM)
C2h		Initialize error logging
C3h		Initialize error display function

Code	Beeps	POST Routine Description
C4h		Initialize system error handler
C5h		PnPnd dual CMOS (optional)
C6h		Initialize notebook docking (optional)
C7h		Initialize notebook docking late
C8h		Force check (optional)
C9h		Extended checksum (optional)
D2h		Unknown interrupt

Code	Beeps	For Boot Block in Flash ROM
E0h		Initialize the chipset
E1h		Initialize the bridge
E2h		Initialize the CPU
E3h		Initialize system timer
E4h		Initialize system I/O
E5h		Check force recovery boot
E6h		Checksum BIOS ROM
E7h		Go to BIOS
E8h		Set Huge Segment
E9h		Initialize Multi Processor
EAh		Initialize OEM special code
EBh		Initialize PIC and DMA
ECh		Initialize Memory type
EDh		Initialize Memory size
EEh		Shadow Boot Block
EFh		System memory test
F0h		Initialize interrupt vectors
F1h		Initialize Run Time Clock
F2h		Initialize video
F3h		Initialize System Management Mode
F4h	1	Output one beep before boot
F5h		Boot to Mini DOS
F6h		Clear Huge Segment
F7h		Boot to Full DOS

Continued...

* If the BIOS detects error 2C, 2E, or 30 (base 512K RAM error), it displays an additional word-bitmap (**xxxx**) indicating the address line or bits that failed. For example, "2C 0002" means address line 1 (bit one set) has failed. "2E 1020" means data bits 12 and 5 (bits 12 and 5 set) have failed in the lower 16 bits. Note that error 30 cannot occur on 386SX systems because they have a 16 rather than 32-bit bus. The BIOS also sends the bitmap to the port-80 LED display. It first displays the check point code, followed by a delay, the high-order byte, another delay, and then the low-order byte of the error. It repeats this sequence continuously.