POST CARD ERROR CODE TABLE

User Manual

※ Quick Error Code Reference Table

To compare the error message shown on the Error Test Indicator with the Quick Error Code Reference Table, you have to make sure the system BIOS supplier and the BIOS version, then find out the corresponding trouble shooting tips, then resolve the problem.

The Error code is based on hex (0 , 1 , 2 .....9 , A , B , C , D , E , F , 10 , 11 , 12 .... 19 , 1A , 1B , 1C , 1D , 1E , 1F , 20...) rather than decimal arithmetic (0 , 1 , 2 ..... 8 , 9 , 10 , 11), so value "2C to 34" in Error code means value "2C, 2D, 2E, 2F, 30, 31, 32, 33, 34".

Quick Error Code Reference Table

Error caused by	AMI BIOS (c) 1990 HiFLEX BIOS	AMI BIOS (c) 1994 WinBIOS	Award	Phoenix	Quadtel AT BIOS	Touble shooting
CPU	01 ~ 02	01 ~ 02	01 ~ 02	01	02 ~ 04	Replace CPU (speed should be the same)
Math Coprocessor	9C ~ 9D	9C ~ 9D	45	3E	76 ~ 78	1. Replace CPU 2.Replace CPU if NPU is built in CPU
Memory	13 20 ~ 24	D1 ~ D3 20 40 ~ 58	C1 08 31 ~ 32	09 ~ 1F	1C ~ 20 28 ~ 2C 32,3A 60 ~ 68	1.Check and clean RAM Module 2.Replace RAM module
Dispaly Card	2C ~ 34	2C ~ 34	0D ~ 0E	2B ~ 33	46	1.Check & clean Display Card slot & golden finger 2. Replace Display Card
M/B error	Others DIY Code	Others DIY Code	Others DIY Code	Others DIY Code	Others DIY Code	Call Customer Service center for help or replace M/B

※ Error Code Message

In the appendix section you may find the Error Code message lists for AMI BIOS, PHOENIX BIOS, and AWARD BIOS, you may refer the error code shown on the Error Test Card to the appendix list or the BIOS POST Code list from your system supplier, and find out the defective parts.

Following is the brief description of the relationship between the Error Test Card, PC system, and Error code, and how to benefit from the Error code.

When system turns on, BIOS will initialize and verify every part of system, and test the function, every initialization, verification and test mean a small, separate program, and have its own error code. The error code will be exported prior to execute the small program, Error Test Card is the indicator which can immediately show the exported error code. So the error code shown on the card means system is executing the corresponding small program, if the initialization, verification or test fails, system will pause and the error code will keep showing on the card, you may find out the defective part by checking the meaning of the error code. Now we can conclude as bellows:

1. Error Test Card: show system self test status and the error code.
2. Error Code: dedicated code for every small program including initialization, verification and test, the combination of every small programs means Power On System Test.

Note 1: Error Test Card is compatible with every IBM compatible system, since there are too many BIOS supplier and the version changes soon, so if your system BIOS information can't be found here, please ask the BIOS supplier for free information.

Note 2: The Error code here may be called POST code, POST error code, PORT 80 code in your system manual.

※ BIOS POST Code Description

The POST (Power-On Self Test) routines are performed by the BIOS each time the system is reset (or power-on). The main purpose of this routines is to perform diagnostics tests upon the system components and to initialize parts of the system. It will put the POST code in port address 80H for each testing procedure process. The error test card can latch this value and show it on the 7-segment LED with "xx" format. Where the "xx" is a POST code. When system is bad before booting. And refer the lists of BIOS POST code to do the trouble-shooting.

All the BIOS POST code listings source of appendix are allowed by the BIOS makers (AMI,Award,Phoenix). These listings used reference only,the same brand BIOS maybe had little difference with each version update . User can also see the real POST code list of your computer in some user's manual of the motherboard.

※ AMI BIOS POST Test Code Listing

1. AMI WinBIOS/HiFLEX 101094 Checkpoint List

The following is a list of AMI WinBIOS/HiFLEX System BIOS POST Checkpiont Codes, which are displayed through hardware port address 80H each time a POST routine has been completed.

POST Code	AMI WinBIOS uncompress code checkpoints Description
C2	NMI is Disabled. Power on delay starting.
C5	Power on delay complete. Going to disable Cache if any.
C6	Calculating ROM BIOS checksum.
C7	ROM BIOS checksum passed. CMOS shutdown register test to be done next.
C8	CMOS shutdown register test done. CMOS checksum calculation to be done next.
CA	CMOS checksum calculation is done, CMOS Diag byte written. CMOS status register about to init for Date and Time.
CB	CMOS status register init done. Any initialization before keyboard BAT to be done next.
CD	BAT command to keyboard controller is to be issued.
CE	Keyboard controller BAT result verified. Any initialization after KB controller BAT to be done next.
CF	Initialization after KB controller BAT done. Keyboard command byte to be written next.
D1	Keyboard controller command byte is written. Going to check pressing of <INS> key during power-on.
D2	Checking for pressing of <INS> key during power-on done. Going to disable DMA and Interrupt controllers.
D3	DMA controller #1,#2, interrupt controller #1,#2 disabled. Chipset init/ auto memory detection about to begin.
D4	Chipset initialization/ auto memory detection over. To uncompress the RUNTIME code.
D5	RUNTIME code is uncompressed.
D0	Transfer control to uncompressed code in shadow ram at F000:FFF0.

Runtime code is uncompressed in F000 shadow ram

POST Code	AMI HiFLEX BIOS/WinBIOS runtime code Description
01	Processor register test about to start. and NMI to be disabled.
02	NMI is Disabled. Power on delay starting.
03	Power on delay complete. To check soft reset/power-on.
05	Soft reset/power-on determined. Going to disable Cache if any.
06	POST code to be uncompressed.
07	POST code is uncompressed. CPU init and CPU data area init to be done next
08	CPU and CPU data area init done.CMOS checksum calculation to be done next.
09	CMOS checksum calculation is done, CMOS Diag byte written. CMOS init to begin (If "Init CMOS in every boot" is set).
0A	CMOS initialization done (if any). CMOS status register about to init for Date and Time.
0B	CMOS status register init done. Any initialization before keyboard BAT to be done next.
0C	KB controller I/B free.Going to issue the BAT command to keyboard controller.
0D	BAT command to keyboard controller is issued. Going to verify the BAT command.
0E	Keyboard controller BAT result verified. Any initialization after KB controller BAT to be done next.
0F	Initialization after KB controller BAT done. Keyboard command byte to be written next.
10	Keyboard controller command byte is written. Going to issue Pin-23,24 blocking/unblocking command.
11	Pin-23,24 of keyboard controller is blocked/ unblocked. Going to check pressing of <INS> key during power-on.
12	Checking for pressing of <INS> key during power-on done. Going to disable DMA and Interrupt controllers.
13	DMA controller #1,#2, interrupt controller #1,#2 disabled. Video display is disabled and port-B is initialized. Chipset init about to begin.

15	Chipset initialization over. 3254 timer test about to start.
19	8254 timer test over. About to start memory refresh test.
1A	Memory Refresh line is toggling. Going to check 15 micro second ON/OFF time.
20	Memory Refresh period 30 micro second test complete. Base 64K memory to be initialized.
23	Base 64K memory initialized. Going to set BIOS stack and to do any setup before Interrupt vector init.
24	Setup required before interrupt vector initialization complete. Interrupt vector initialization about to begin.
25	Interrupt vector initialization done. Going to read Input port of 8042 for turbo switch (if any) and to clear password if post diag switch is on.
26	Input port of 8042 is read. Going to initialize global data for turbo switch.
27	Global data initialization for turbo switch is over. Any initialization before setting video mode to be done next.
28	Initialization before setting video mode is complete. Going for monochrome mode and color mode setting.
2A	Different BUSes init (system, static, output devices) to start if present. (Please see next section for details of different BUSes).
2B	About to give control for any setup required before optional video ROM check.
2C	Processing before video ROM control is done. About to look for optional video ROM and give control.
2D	Optional video ROM control is done. About to give control to do any processing after video ROM returns control.
2E	Return from processing after the video ROM control. If EGA/VGA not found then do display memory R/W test.
2F	EGA/VGA not found. Display memory R/W test about to begin.
30	Display memory R/W test passed. About to look for the retrace checking.
31	Display memory R/W test or retrace checking failed. About to do alternate Display memory R/W test.
32	Alternate Display memory R/W test passed. About to look for the alternate display retrace checking.
34	Video display checking over. Display mode to be set next.
37	Display mode set. Going to display the power on message.
38	Different BUSes init (input, IPL, general devices) to start if present. (Please see next section for details of different BUSes).
39	Display different BUSes initialization error messages. (Please see Appendix for details of different BUSes).
3A	New cursor position read and saved. Going to display the Hit <DEL> message.

3B	Hit <DEL> message displayed. Virtual mode memory test about to start.
40	Going to prepare the descriptor tables.
42	Descriptor tables prepared. Going to enter in virtual mode for memory test.
43	Entered in the virtual mode. Going to enable interrupts for diagnostics mode.
44	Interrupts enabled (if diagnostics switch is on). Going to initialize data to check memory wrap around at 0:0.
45	Data initialized. Going to check for memory wrap around at 0:0 and finding the total system memory size.
46	Memory wrap around test done. Memory size calculation over. About to go for writing patterns to test memory.
47	Pattern to be tested written in extended memory. Going to write patterns in base 640k memory.
48	Patterns written in base memory. Going to findout amount of memory below 1M memory.
49	Amount of memory below 1M found and verified. Going to findout amount of memory above 1M memory.
4B	Amount of memory above 1M found and verified. Check for soft reset and going to clear memory below 1M for soft reset. (If power on, go to check point# 4Eh).
4C	Memory below 1M cleared. (SOFT RESET) Going to clear memory above 1M.
4D	Memory above 1M cleared. (SOFT RESET) Going to save the memory size. (Goto check point# 52h).
4E	Memory test started.(NOT SOFT RESET) About to display the first 64k memory size.
4F	Memory size display started. This will be updated during memory test. Going for sequential and random memory test.
50	Memory testing/initialization below 1M complete. Going to adjust displayed memory size for relocation/ shadow.
51	Memory size display adjusted due to relocation/ shadow. Memory test above 1M to follow.
52	Memory testing/initialization above 1M complete. Going to save memory size information.
53	Memory size information is saved. CPU registers are saved. Going to enter in real mode.
54	Shutdown successful, CPU in real mode. Going to disable gate A20 line and disable parity/NMI.
57	A20 address line, parity/NMI disable successful. Going to adjust memory size depending on relocation/shadow.
58	Memory size adjusted for relocation/shadow. Going to clear Hit <DEL>

	message
59	Hit <DEL> message cleared. <WAIT...> message displayed. About to start DMA and interrupt controller test.
60	DMA page register test passed. To do DMA#1 base register test.
62	DMA#1 base register test passed. To do DMA#2 base register test.
65	DMA#2 base register test passed. To program DMA unit 1 and 2.
68	DMA unit 1 and 2 programming over. To initialize 8259 interrupt controller.
67	8259 initialization over.
7F	Extended NMI sources enabling is in progress.
80	Keyboard test started. clearing output buffer, checking for stuck key, About to issue keyboard reset command.
81	Keyboard reset error/stuck key found. About to issue keyboard controller interface test command.
82	Keyboard controller interface test over. About to write command byte and init circular buffer.
83	Command byte written, Global data init done. About to check for lock-key.
84	Lock-key checking over. About to check for memory size mismatch with CMOS.
85	Memory size check done. About to display soft error and check for password or bypass setup.
86	Password checked. About to do programming before setup.
87	Programming before setup complete. Going to uncompress SETUP code and execute CMOS setup.
88	Returned from CMOS setup program and screen is cleared. About to do programming after setup.
89	Programming after setup complete. Going to display power on screen message.
8B	First screen message displayed. <WAIT...> message displayed. About to do Video BIOS shadow.
8C	Video BIOS shadow successful. Setup options programming after CMOS setup about to start.
8D	Setup options are programmed, mouse check and init to be done next.
8E	Mouse check and initialization complete.Going for hard disk controller reset.
8F	Hard disk controller reset done. Floppy setup to be done next.
91	Floppy setup complete. Hard disk setup to be done next.
94	Hard disk setup complete. To set base and extended memory size.
95	Memory size adjusted due to mouse support.Init of different BUSes optional ROMs from C800 to start. (Please see next section for details of different BUSes).

96	Going to do any init before C800 optional ROM control.
97	Any init before C800 optional ROM control is over. Optional ROM check and control will be done next.
98	Optional ROM control is done. About to give control to do any required processing after optional ROM returns control.
99	Any initialization required after optional ROM test over. Going to setup timer data area and printer base address.
9A	Return after setting timer and printer base address. Going to set the RS-232 base address.
9B	Returned after RS-232 base address. Going to do any initialization before Coprocessor test.
9C	Required Initialization before Coprocessor is over. Going to initialize the Coprocessor next.
9D	Coprocesor initialized.Going to do any initialization after Coprocessor test.
9E	Initialization after Coprocessor test is complete. Going to check extd keyboard, keyboard ID and num-lock.
9F	Extd keyboard check is done, ID flag set, num-lock on/off. Keyboard ID command to be issued.
A0	Keyboard ID command issued. Keyboard ID flag to be reset.
A1	Keyboard ID flag reset. Cache memory test to follow.
A2	Cache memory test over. Going to display any soft errors.
A3	Soft error display complete. Going to set keyboard typematic rate.
A4	Keyboard typematic rate set. To program memory wait states.
A5	Memory wait states programming over. Going to clear the screen and enable parity/NMI.
A7	NMI and parity enabled. Going to do any initialization required before giving control to optional ROM at E000.
A8	Initialization before E000 ROM control over.E000 ROM to get control next.
A9	Returned from E000 ROM control. Going to do any initialization required after E000 optional ROM control.
AA	Initialization after E000 optional ROM control is over. Going to display the system configuration.
B0	System configuration is displayed.
B1	Going to copy any code to specific area.
00	Copying of code to specific area done. Going to give control to INT-19 boot loader.

The system BIOS gives control to the different BUSes at following checkpoints to do

various tasks on the different BUSes.

POST code	AMI HiFLEX BIOS/WinBIOS Checkpoints of the BUSes
2A	Different BUSes init (system, static, output devices) to start if present.
38	Different BUSes init (input, IPL, general devices) to start if present.
39	Display different BUSes initialization error messages.
95	Init of different BUSes optional ROMs from C800 to start.

While control is inside the different BUS routines, additional checkpoints are output to port 80h as WORD to identify the routines under execution. These are WORD checkpoints, the LOW BYTE of checkpoint is the system BIOS checkpoint from where the control is passed to the different BUS routines and the HIGH BYTE of checkpoint is the indication of which routine is being executed in the different BUSes. THe details of HIGH BYTE of these checkpoints are as follows:

HIGH BYTE XY

the upper nibble 'X' indicates the function# is being executed. 'X' can be from 0 to 7.

0 = func#0, disable all devices on the BUS concerned.
1 = func#1, static devices init on the BUS concerned.
2 = func#2, output device init on the BUS concerned.
3 = func#3, input device init on the BUS concerned.
4 = func#4, IPL device init on the BUS concerned.
5 = func#5, general device init on the BUS concerned.
6 = func#6, error reporting for the BUS concerned.
7 = func#7, add-on ROM init for all BUSes.

the lower nibble 'Y' indicates the BUS on which the different routines are being executed. 'Y' can be from 0 to 5.

0 = Generic DIM (Device Initialization Manager).
1 = On-board System devices.
2 = ISA devices.
3 = EISA devices.
4 = ISA PnP devices.
5 = PCI devices.

※ AWARD BIOS POST Test Code Listing

POST Code	Award POST Routine Description
C0	Turn off chipset cache.
01	Test processor flag register.
02	Test all processor registers except SS, SP and BP with pattern FF and 00.
03	Initialize Chips (RTC,8254,8237,8259) ,Reset math coprocessor,Clear CMOS shutdown byte and page register
04	Test DRAM refresh
05	Blank video,keyboard controller initialization.
07	Test CMOS interface and battery status
BE	Initialize chipset with power on BIOS defaults.
C1	Memory presence test. (OEM Specific-Test to size on-board memory)
C5	Early Shadow.(OEM Specific-Early Shadow enable for fast boot)
C6	Cache presence test. (Extenal cache size detection)
08	Setup low memory (base 64K memory test).
09	Early cahe initilization (Cyrix CPU initialization ,Cache Initialization.)
0A	Setup interrupt vector table
0B	Test CMOS RAM checksum ,load default value if test is bad.
0C	Initialize keyboard (detect keyboard type and set NUM_LOCK status)
0D	Initialize and detect video adapter interface
0E	Test video memory, write sign-on message to screen. Setup shawdow RAM - Enable shadow according to setup.
0F	Test DMA controller 0
10	Test DMA controller 1
11	Test DMA page registers (74612)
14	Test Timer 0 counter 2
15	Test 8259-1 interrupt mask register (port 21H)
16	Test 8259-2 interrupt mask register (port A1H)
17	Test stuck 8259's interrupt bits
18	Test 8259 interrupt functionality
19	Test stuck NMI bits (Verify NMI can be cleared)
1A	Display CPU clock
1F	Set EISA mode.
20	Enable Slot 0 (Initialize slot 0 - system board)
21-2F	Enable Slot 1-15 (Initialize slot 1 to 15)
30	Size base memory and extended memory
31	Test base memory and extended memory
32	Test EISA extended memory. Skip in ISA mode.
3C	Setup enabled.
3D	Initialize and setup mouse.

3E	Setup cache controller
BF	Chipset initialization (progran chipset registers with setup values)
40	Display virus protest disable or enable
41	Initialize floppy drive and controller
42	Initialize hard disk drive and controller
43	Detect & Initialize the serial, parallel and game ports.
4E	Check manufacture POST loop.
4F	Security check
50	Write CMOS (write all CMOS values back to RAM ans clear screen)
51	Pre-boot enable (enable parity check, NMI, cache)
52	Initialize option ROMs (address range from 0C8000H to 0EFFFFH)
53	Initialize time value in 40h: BIOS area
60	Setup virus protect according to setup
61	Set system speed for boot
62	Setup NumLock status according to setup
63	Boot attempt (set low stack, boot via INT 19H)
B0	Spurious (if interrup occurs in protected mode)
B1	Unclained NMI (if NMI occurs,Press F1 to disable NMI,F2 reboot)
E1-EF	Setup pages (E1 - page 1 , E2 - page 2 ... ,etc)
FF	Boot the operating system

※ Pheonix BIOS POST Test Code Listing

POST Code	Pheonix POST Rountine Description
02	Verify Real Mode
04	Get CPU type
06	Initialize system hardware
08	Initialize chipset register with initial POST value
09	Set in POST flag
0A	Initialize CPU register
0C	Initialize cache to initial POST values
0E	Initialize I/O CPU register
0F	Initialize the local bus IDE
10	Initialize Power Managment
11	Load alternate registers with Initial POST values
12	Jump to UserPatch0
14	Initialize keyboard controller
16	BIOS ROM checksum
18	8254 timer initialization
1A	8237 DMA controller initialization
20	Test DRAM refresh
22	Test 8042 keyboard controller
24	Set ES segment register to 4 GB
28	Autosize DRAM
2A	Clear 512K base RAM
2C	Test 512K base address lines
2E	Test 512K base memory
32	Test CPU bus-clock frequency
34	Test CMOS RAM
35	Initialize alternate chipset registers
37	Reinitialize the chipset
38	Shawdow system BIOS ROM
39	Reinitialize the cache
3A	Autosize cache
3C	Configure advanced chipset registers
3D	Load alternate registers with CMOS values
40	Set initial CPU speed
42	Initialize interrupt vectors
44	Initialize BIOS interrupts
46	Check ROM copyright notice
47	Initialize manager for PCI Option ROMs
48	Check video configuration against CMOS

49	Initialize PCI bus and devices
4A	Initialize all video adapters in system
4C	Shadow video BIOS ROM
4E	Display copyright notice
50	Display CPU type and speed
51	Initialize EISA board
52	Test keyboard
54	Set key click if enabled
56	Enable keyboard
58	Test for unexpected interrupts
5A	Display prompt "Press F2 to enter SETUP"
5C	Test RAM between 512 to 640K
60	Test extended memory
62	Test extended memory address lines
64	Jump to UserPatch1
66	Configure advanced cache registers
68	Enable external and CPU cache
6A	Display external cache size
6C	Display shadow message
6E	Display non-disposable segments
70	Display error messages
72	Check for configuration errors
74	Test real-time clock
76	Check for keyboard errors
7C	Setup hardware interrupt vectors
7E	Test corprocessor if presents
80	Disable onboard I/O ports
82	Detect and install external RS-232 ports
84	Detect and install external parallel ports
86	Re-initialize onboard I/O ports
88	Initialize BIOS Data Area
8A	Initialize Extended BIOS Data Area
8C	Initialize floppy controller
90	Initialize hard-disk controller
91	Initialize local-bus hard-disk controller
92	Jump to UserPatch2
93	Built MPTABLE for multi-processor boards
94	Disable A20 address line
96	Clear huge ES segment register
98	Search for option ROMs
9A	Shadow option ROMs

9C	Set up Power Management
9E	Enable hardware interrupts
A0	Set time of day
A2	Check key lock
A4	Initialize typematic rate
A8	Ease F2 prompt
AA	Scan for F2 stroke
AC	Enter SETUP
AE	Clear in-POST flag
B2	POST done - prepare to boot operating system
B4	One beep
B6	Check password(option)
B8	Clear global descriptor table
BC	Clear parity checkers
BE	Clear screen(option)
BF	Clear virus and backup remainders
C0	Try to boot with INT 19
D0	Interrupt handler error
D2	Unknown interrupt error
D4	Pending interrupt error
D6	Initialize option ROM error
D8	Shutdown error
DA	Extended BLOCK Move
DC	Shutdown 10 error

The following are for boot block in Flash ROM

POST Code	Pheonix POST Rountine Description
E2	Initialize the chipset
E3	Initialize refresh counter
E4	Check for forced Flash
E5	Check H/W status ROM
E6	BIOS ROM is OK
E7	Do a complete RAM test
E8	Do OEM initialization
E9	Initialize interrupt controller
EA	Read in the bootstrap code
EB	Initialize all vectors
EC	Boot the Flash program
ED	Initialize the boot device
EE	Boot code was read OK