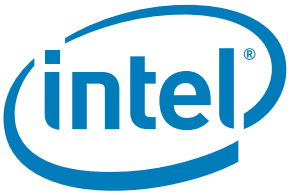
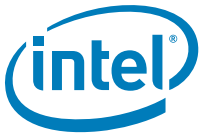


Intel® iSCSI Remote Boot Application Notes for BIOS Engineers

Intel® 10/100/1000 Mb/s and 10 GbE Ethernet Controllers

November 2010

322328-004
Revision 1.3

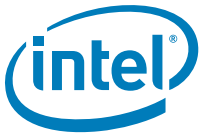


Contents

1.0 Introduction 5
2.0 Purpose 5
3.0 Intel® iSCSI Remote Boot Option ROM Image Options 5
4.0 PCI 3.0 Considerations 6
5.0 EEPROM Considerations 6
6.0 EEPROM Example 7
7.0 Memory and Image Storage Requirements 8
8.0 BIOS Operation 8
9.0 Miscellaneous Considerations 9
10.0 Causes of Intel® iSCSI Remote Boot Related EEPROM Corruption 9

Revision History

Date	Revision	Description
November 2010	1.3	Updated section 5.0.
September 2010	1.2	Updated section 5.0 and 7.0.
August 2010	1.1	Updated section 3.0, 5.0, and 10.0.
July 2009	1.0	Initial public release.



1.0 Introduction

The Intel® iSCSI Remote Boot Software is a personal computer option ROM designed to supply industry standard iSCSI compliant pre-boot services for x86 architecture computers using a legacy style BIOS. The agent can be stored on a Flash component on an add-in Network Interface Card (NIC), or it can be integrated into the BIOS. Configuration data for the Intel® iSCSI Remote Boot software is stored in the network device EEPROM.

The Intel® iSCSI Remote Boot Software is compliant to the PCI Firmware Specification R3.0, but can be used in a BIOS written to the PCI 2.x specification.

2.0 Purpose

This document can be used to assist BIOS developers in the process of integrating the Intel® iSCSI Remote Boot software into the BIOS and troubleshooting common problems. This document should be used in conjunction with the EEPROM / NVM reference material for your networking component.

This document does not address iSCSI remote boot in UEFI environments. Intel provides UEFI drivers that are used in conjunction with other UEFI software components to provide network capabilities in UEFI environments. Refer to www.uefi.org for information on UEFI.

3.0 Intel® iSCSI Remote Boot Option ROM Image Options

Creating an option ROM image that can be integrated into a BIOS requires the use of the utility program Bootutil.exe. This is a universal utility that can both update the option ROM image in the Flash memory on a NIC, or can be used to create an image for inclusion in a system BIOS on LOM designs. To use this utility, you must include the master image file `BootIMG.FLB` in the same folder.

To create a LOM image, use the following command line parameters:

```
Bootutil -di=iscsi -devid='device id value in hex' -lom
```

where `device id value in hex` means the PCI device ID of the Intel network device in hexadecimal format. This is in the form `0xyyyy` with `yyyy` representing the four character device ID.

Example:

```
bootutil -di=iscsi -devid=0x10a9 -lom
```

The resulting file name has an extension of `.LOM`. Various image types are supported by `bootutil`, but for a LOM implementation, only the `.LOM` file type is supported.

4.0 PCI 3.0 Considerations

If the BIOS reports itself as supporting PCI Firmware version 3.0 or higher, it must also set the BX CPU register to the final memory address of the option ROM memory area before calling the initialization code. This and the required input parameters are detailed in the PCI Firmware 3.0 (BIOS) specification. The Intel® iSCSI Remote Boot software has no method to validate this address, and therefore uses whatever value the BIOS passes in. For example, if the option ROM is loaded at 0xC900:0, then BX needs to be set to 0xC900 before calling the Intel® iSCSI Remote Boot software.

The Intel® iSCSI Remote Boot option ROM component is created with an integrated table of all supported PCI ID's. A PCI 3.0 BIOS supports all ID's listed in the table in the Intel® iSCSI Remote Boot firmware component.

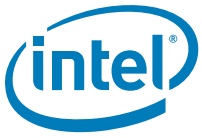
5.0 EEPROM Considerations

The adapter EEPROM image holds the Intel® iSCSI Remote Boot configuration data. Configuration data can be stored for up to two interfaces. See the reference documentation for your specific Ethernet component for details.

Several areas of the EEPROM are used for the Intel® iSCSI Remote Boot configuration. Words 0x30 through 0x3E are reserved for pre-boot firmware use, and word 0x3F contains the checksum for words 0x0 through 0x3E. Words 0x30 - 0x3E are primarily used to store pre-boot software configuration settings.

The Intel® iSCSI Remote Boot option ROM uses several words in this shared range. Word 0x33 bit 4 must be set to 1b, as this indicates the interface is capable of running Intel® iSCSI Remote Boot. When the option ROM executes, it checks this bit and, if set to 0b, it immediately exits and returns control to the BIOS. Word 0x33 bits 0 and 1 are used to indicate that PXE is enabled. For PXE to run, both of these bits should be equal to 1b. For LOM implementations only, it is permissible to have both iSCSI boot and PXE bits in the EEPROM set to indicate the device is capable of running both, although only one option ROM can actually be active on any one port at any time.

Word 0x3D (in GbE controllers) is a software pointer to the EEPROM address where the iSCSI boot block begins. Word 0x19 is used for this purpose in 10 GbE controllers (82598, 82599). The boot block is where the iSCSI client configuration data is stored. The first word is always the Intel® iSCSI Remote Boot signature; 0x5369 (equal to iS in ASCII). The second word is the boot block length. This is usually 0x0384 for single-port controllers, 0x05E0 for dual-port controllers, and 0x0A98 for quad-port controllers. The remaining fields store the configuration data. The NVM images supplied for your network controller should, by default, have the proper pointer and boot block in place.



Word 0x30 (for single port controllers, or the first port of multi-port controllers), word 0x34 (for the second port of multi-port controllers), word 0x38 (for the third port of four-port controllers), and word 0x3A (for the fourth port of four-port controllers) use bits 0:2 to store port-specific functional configuration data. These bits are set by user configuration utilities. These include the pre-boot configuration utility built into the Intel® iSCSI Remote Boot option ROM and accessed during POST by the Ctrl-D key combination, as well as software supplied by Intel for various operating system environments. These bits reflect the following settings:

Word 0x30 (Port 0), 0x34 (port 1, second port of a 2- or 4-port controller), 0x38 (port 2, third port of a 4-port controller), and word 0x3A (port 3, fourth port of a 4-port controller) bits [1:0] indicate Intel® iSCSI Remote Boot port priority or PXE status for that port.

- 000b: PXE enabled on this port
- 001b: Port disabled for PXE, iSCSI boot and FCoE
- 010b: This port is the iSCSI Boot primary port
- 011b: This port is the iSCSI Boot secondary port
- 100b: This port enabled for FCoE
- 101b: Reserved, treated as port disabled
- 110b: Reserved, treated as port disabled
- 111b: Reserved, treated as port disabled

Note: Bit 5, formerly used to indicate iSCSI enable / disable, is no longer valid and is not checked by software.

6.0 EEPROM Example

Note: Single port GbE, iSCSI boot and PXE enabled, and iSCSI boot primary port

```
Addr   0/8  1/9  2/A  3/B  4/C  5/D  6/E  7/F

0000 | 8888 8888 8887 0800 FFFF 0053 FFFF FFFF

0008 | FFFF FFFF 10C3 0000 8086 10DE 8086 0000

0010 | 0D01 0000 0000 A705 3020 0A00 0000 8D07

0018 | 0684 2B00 4043 0000 10DE BAAD 10DE 10DF

0020 | BAAD BAAD BAAD BAAD 0000 0000 0000 0000

0028 | 0000 0000 0000 0000 0000 0000 0000 0000

0030 | 0102 4000 1228 4013 FFFF FFFF FFFF FFFF

0038 | FFFF FFFF FFFF FFFF FFFF 0100 FFFF FFFF

0040 | 6020 001F 0002 0013 8000 001D 00FF 0016

0048 | 9922 0018 2011 0017 DDDD 0018 2012 0017

0050 | 8000 001D 0000 001F FFFF FFFF FFFF FFFF

0058 | FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
```

(addresses 0060 - through 00F7)

```
00F8 | FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF

0100 | 5369 0384 0001 0000 0000 0000 0000 0000

0108 | 0000 0000 0000 0000 0000 0000 0000 0000
```

(Remaining addresses of EEPROM)

- **0x30** = port 1 config, set to iSCSI boot primary
- **0x33** = iSCSI boot capability true (bit 4 = 1b) and PXE capability true (bits [1:0] =1b)
- **0x3D** = pointer to the iSCSI boot block starting address; in this example, 0x100
- **0x100** = beginning address of iSCSI boot block

7.0 Memory and Image Storage Requirements

The option ROM image takes up approximately 122 KB of Flash memory space, and requires this amount of free RAM space to load and start execution. This size estimation is approximate and subject to change with different revisions of the firmware. The option ROM requires additioinal conventional RAM durng execution.

The BIOS might load and execute the option ROM image in the Upper Memory Area (UMB), but if the BIOS is compliant to the PCI Firmware 3.0 specification, the BIOS might load the image in conventional memory and set the final runtime address in the UMB in the BX register. When the option ROM successfully connects to an iSCSI target disk after initialization, the image size in the UMB shrinks to <50 KB for runtime.

During initialization, the option ROM additionally allocates approximately 22 KB of conventional memory using the PMM service. If the allocation fails, control is passed back to the BIOS. This memory is released after initialization. For the 82563, 82564, and the 82580 additional memory is allocated for initialization purposes.

The option ROM requires 25 KB in EBDA during initialization and runtime. The EBDA base address is fixed before starting the initialization.

8.0 BIOS Operation

The BIOS calls the BCV entry point in the Intel® iSCSI Remote Boot Option ROM PnP header after the initialization so that INT13h ISR is registered in the vector address.

The BCV entry pointer is not visible until the iSCSI target disk is found in initialization.

The BIOS takes the iSCSI target disk as BCV/INT13h device to BIOS IPL. The BIOS handles the iSCSI disk the same as a typical BCV/INT13h device.

9.0 Miscellaneous Considerations

- The BIOS can use the device ID list pointed to in the PCI data structure so that one image can support various NICs.
- The class/type settings are in the Intel® iSCSI Remote Boot Option ROM headers
- The PCI data structure is set to Ethernet adapter
- The PnP header is set to BCV/INT13h device
- If the image is loaded in conventional memory, it returns to the BIOS from the segment although the runtime is in UMB. The segment is no longer used after the initialization and the BIOS might need to verify the runtime code in UMB for further actions

10.0 Causes of Intel® iSCSI Remote Boot Related EEPROM Corruption

- VSCC Not Set Correctly — The ICH8, ICH9, and ICH10 use a single Flash component that is shared between the BIOS and other firmware components, such as ASF or AMT. If the network MAC in these components is also used (in conjunction with the 82566, 82567, 82562V, 82577, or 82578 PHY), the EEPROM image for the network interface is also stored in its own partition called the gigabit region. The Intel® iSCSI Remote Boot option ROM accesses this region just as it would the dedicated EEPROM on a NIC.

  To enable more flexibility in selecting Flash components, the ICH8 stores the codes needed to operate the shared Flash component in a reserved area. These are referred to as the Vendor Specific Component Capabilities (VSCC). The ICH9, ICH10, and the Intel® 5 Series Express Chipset add additional flexibility by the use of two of these areas, referred to as the Upper Vendor Specific Component Capabilities (UVSCC) and Lower Vendor Specific Component Capabilities (LVSCC).

  The values entered in the VSCC must be correct for the Flash component type plus other configurable attributes (block sizes, etc.). These values are set by the BIOS and are not controlled by the Intel® iSCSI Remote Boot option ROM, but must be used by the option ROM to read and write to the Flash. If corruption occurs to the NVM as a result of an Intel® iSCSI Remote Boot write operation, it is possible that the VSCC values are not set correctly. Please refer to the SPI Programming Guide for the ICH8, ICH9, ICH10, or the Intel® 5 Series Express Chipset as needed in your application.
- Block Allocation Error — When using a shared Flash, such as used with the ICH8, ICH9, ICH10 or the Intel® 5 Series Express Chipset, the entire EEPROM image area, including the Intel® iSCSI Remote Boot configuration words, must be duplicated to allow for both an active and an inactive copy of the EEPROM image. This enables the EEPROM image update code to erase and then program the inactive area, mark it active, and mark the previously active copy as inactive. If there are not enough blocks of Flash memory available to maintain both copies, when the boot agent or software tools write to the Flash it might cause the EEPROM image to become corrupted and render the LAN device unusable. Note that each area must be at least 4 KB in size and at least as large as the Flash device sector erase size. For example, if the Flash has a 64 KB erase size, you'll have to allocate at least 128 KB since when the update process erases the inactive area, that's the smallest size it can erase.

Note: This page intentiionally left blank.