



Intel® Ethernet NVM Update Tool

Quick Usage Guide for EFI

Ethernet Products Group (EPG)

January 2022

Revision 1.5
332160-006

Revision History

Revision	Date	Comments
1.5	January 28, 2022	Updates include the following: <ul style="list-style-type: none"> Added Section 5.2, "Command Line Parameters". Added Section 5.3, "Exit Codes". Added Section 5.4, "Fields Preserved During Reset to Default". Added Section 7.4, "Configuration File Rules and Restrictions".
1.4	November 19, 2020	Updates include the following: <ul style="list-style-type: none"> Updates to include Intel® Ethernet Controller X710-TM4/AT2 as part of the 700 Series. Minor formatting and pagination.
1.3	October 12, 2020	Updates include the following: <ul style="list-style-type: none"> Updates for Intel® Ethernet Controller E810.
1.2	November 19, 2018	Updates include the following: <ul style="list-style-type: none"> Updates for NVM version 6.80 for Intel® Ethernet Controller X710/XXV710/XL710. Updates for NVM version 2.00 for Intel® Ethernet Controller X550. Added Section 5.1, "Recovery Mode". Added Section 6.0, "Troubleshooting". Added Section 7.0, "Create/Edit nvupdate.cfg for Custom NVM Images".
1.1	February 10, 2016	Updates for Software Release 5 for i40e and EFI.
1.0	March 4, 2015	Initial public release.

Contents

1.0	Introduction	5
2.0	Update Both NVM and Driver at the Same Time	5
3.0	Obtaining New Images	6
4.0	Preparing for the Update	7
5.0	Running the NVM Update Tool	8
5.1	Recovery Mode	9
5.2	Command Line Parameters	10
5.3	Exit Codes	11
5.4	Fields Preserved During Reset to Default	13
6.0	Troubleshooting	14
6.1	Troubleshooting Using Debug Logs	14
7.0	Create/Edit <i>nvmupdate.cfg</i> for Custom NVM Images	15
7.1	Sample Configuration File Template	15
7.2	Device Block in the Configuration File	15
7.3	Steps to Create/Edit <i>nvmupdate.cfg</i> File to Update Custom NVM Image	16
7.3.1	On 800 Series Devices	16
7.3.2	On 700 Series and 500 Series Devices	17
7.4	Configuration File Rules and Restrictions	18
8.0	Summary	20



NOTE: *This page intentionally left blank.*

1.0 Introduction

This document demonstrates how to use the Intel® Ethernet NVM Update Tool (NVM Update Tool) to update both the Non-Volatile Memory (NVM) and drivers on Network Adapters based on the following devices:

Series	Devices
Intel® Ethernet 800 Series (800 Series)	<ul style="list-style-type: none">• Intel® Ethernet Controller E810-XXVAM2• Intel® Ethernet Controller E810-CAM1• Intel® Ethernet Controller E810-CAM2
Intel® Ethernet 700 Series (700 Series)	<ul style="list-style-type: none">• Intel® Ethernet Controller X710-AM2• Intel® Ethernet Controller X710-BM2• Intel® Ethernet Controller XXV710-AM1• Intel® Ethernet Controller XXV710-AM2• Intel® Ethernet Controller XL710-AM1• Intel® Ethernet Controller XL710-AM2• Intel® Ethernet Controller XL710-BM1• Intel® Ethernet Controller XL710-BM2• Intel® Ethernet Controller X710-AT2• Intel® Ethernet Controller X710-TM4
Intel® Ethernet 500 Series (500 Series)	<ul style="list-style-type: none">• Intel® Ethernet Controller X550-AT• Intel® Ethernet Controller X550-AT2• Intel® Ethernet Controller X550-BT2

This document is a guide to servicing NVM images, firmware, and drivers of the 800 Series, 700 Series, and 500 Series devices by customers and service technicians in the field.

Note: The information in this document is for experienced system administrators who are familiar with server, network, and data center concepts and technologies.

2.0 Update Both NVM and Driver at the Same Time

Keeping up with software changes, performance enhancements, or security updates requires the most current hardware drivers for supported systems. Previous updates to Intel network adapters were driver specific. With 800 Series, 700 Series, and 500 Series Network Adapters, both the firmware (device NVM image) and network drivers are field-serviceable, allowing the NVM image and network driver to be updated as a matched set. Updating the device image and driver together can increase key features including performance, manageability, media types, physical port counts, virtualization, offloads, remote boot options, VLAN support, teaming, and Receive Side Scaling.

Note: The EFI version of the NVM Update Tool allows the NVM update without updating to the most current *ice* and *i40e* drivers. It is recommended that the most current *ice* and *i40e* drivers be installed once the system has been booted into its running operating system.

The NVM Update Tool has a built-in integrity check that ensures only Intel-approved firmware updates on 800 Series, 700 Series, and 500 Series devices. Integrity validation of NVM updates is provided by a digital signature. NVM updates are validated prior to invalidating the old NVM configuration, so the old NVM and the configuration are still usable should the update fail.

Note: On 700 Series and 500 Series devices, updating to the most current NVM (with the NVM Update Package) and driver does not update the Option ROM. Intel recommends an Option ROM update after the NVM and driver are updated. Refer to the [User Guide for Intel® Ethernet Adapters](#) page for the most current Option ROM update process version.

Note: When running SR-IOV, it is recommended that all Virtual Functions be disabled prior to the NVM Update process.

3.0 Obtaining New Images

The [Intel Download Center](#) is Intel's repository for the latest software and drivers for Intel products. The NVM Update Packages for Windows, Linux, ESX, FreeBSD, and EFI/EFI2 are located at:

Network Adapter Series	Link
Intel® Ethernet Network Adapter E810 Series	https://downloadcenter.intel.com/download/29736
Intel® Ethernet Network Adapter 700 Series	https://downloadcenter.intel.com/download/24769
Intel® Ethernet Network Adapter X550 Series	https://downloadcenter.intel.com/download/28336

Use the Software/NVM matrix tables in following documents to ensure firmware image and driver compatibility. These documents are continuously maintained and always up-to-date:

Document	Link
<i>Intel® Ethernet Controller E810 Feature Support Matrix</i>	Doc ID: 630155
<i>Intel® Ethernet Controller X710-TM4/AT2 and V710-AT2 Feature Support Matrix</i>	Doc ID: 619407
<i>Intel® Ethernet Controller X710/XXV710/XL710 Feature Support Matrix</i>	Doc ID: 332191
<i>Intel® Ethernet Controller X550 Feature Support Matrix</i>	Doc ID: 335253

4.0 Preparing for the Update

Download the package of the NVM Update Tool, and perform the following steps:

1. Unzip the package contents to a flash drive or CD-ROM.
2. Insert the flash drive or CD-ROM into the system of the adapter to be updated.
3. Boot the system to the EFI shell.
4. Navigate to the correct location of the NUP tool executable and binary files as shown in [Figure 1](#).

```
UEFI Interactive Shell v2.2
EDK II
UEFI v2.70 (American Megatrends, 0x0005000E)
Mapping table
  FS1: Alias(s):HD1b65535a1::BLK6:
        PciRoot(0x0)/Pci(0x17,0x0)/Sata(0x1,0xFFFF,0x0)/HD(1,GPT,99C3FE3D-D4C6-45A9-999D-3827F50
EA1,0x800,0x100000)
  FS0: Alias(s):HD0f0b::BLK2:
        PciRoot(0x0)/Pci(0x14,0x0)/USB(0x5,0x0)/HD(1,MBR,0x15A5BE95,0x800,0x77F800)
  BLK0: Alias(s):
        PciRoot(0x0)/Pci(0x14,0x0)/USB(0x13,0x0)
  BLK8: Alias(s):
        PciRoot(0x0)/Pci(0x1C,0x4)/Pci(0x0,0x4)/USB(0x0,0x0)
  BLK9: Alias(s):
        PciRoot(0x0)/Pci(0x1C,0x4)/Pci(0x0,0x4)/USB(0x1,0x0)
  BLK3: Alias(s):
        PciRoot(0x0)/Pci(0x17,0x0)/Sata(0x0,0xFFFF,0x0)
  BLK5: Alias(s):
        PciRoot(0x0)/Pci(0x17,0x0)/Sata(0x1,0xFFFF,0x0)
  BLK4: Alias(s):
        PciRoot(0x0)/Pci(0x17,0x0)/Sata(0x0,0xFFFF,0x0)/HD(1,GPT,E33237BD-DD6D-4F76-AC2E-22CD227
EB4,0x800,0x2E938800)
  BLK7: Alias(s):
        PciRoot(0x0)/Pci(0x17,0x0)/Sata(0x1,0xFFFF,0x0)/HD(2,GPT,20C9D643-DCB9-4820-B115-727A206
432,0x100800,0x37D42800)
  BLK1: Alias(s):
        PciRoot(0x0)/Pci(0x14,0x0)/USB(0x5,0x0)
Press ESC in 1 seconds to skip startup.nsh or any other key to continue.
Shell> fs0:
FS0:\> cd \E810\EFI2x64
FS0:\E810\EFI2x64\> nvupdate64e.efi_
```

Figure 1. NUP Tool Executable and Binary Files

5.0 Running the NVM Update Tool

The NVM Update Tool runs from a Command-Line Interface (CLI). There are optional CLI attributes for specific tasks and are recommended for advanced users only. As CLI syntax, the NVM Update Tool can be scripted to run across large environments. An example of the update syntax is as follows:

```
nvmupdate64e.efi - command syntax
nvmupdate64e.efi -l fileoutput.txt - command with optional attribute
```

Note: For assistance with optional CLI attributes, contact your Intel Representative.

Run the tool like any CLI executable. An example of the EFI CLI version of the NVM Update Tool update and its output is shown in [Figure 2](#).

Note: A typical update takes several minutes to complete.

```
NVMUpdate version 1.35.33.4
Copyright (C) 2013 - 2020 Intel Corporation.

WARNING: To avoid damage to your device, do not stop the update or reboot or power off the system
ring this update.
Inventory in progress. Please wait [***[.....]]

Num Description                               Ver.(hex)  DevId S:B   Status
===
01) Intel(R) Ethernet Network Adapter 1.64(1.40) 1592 00:024 Update
      E810-C-Q2                               available

Options: Adapter Index List (comma-separated), [A]ll, e[X]it
Enter selection: a
Would you like to back up the NVM images? [Y]es/[N]o: n
Update in progress. This operation may take several minutes.
[**-.....]

Num Description                               Ver.(hex)  DevId S:B   Status
===
01) Intel(R) Ethernet Network Adapter 2.00(2.00) 1592 00:024 Update
      E810-C-Q2                               successful

Reboot is required to complete the update process.

Tool execution completed with the following status: All operations completed successfully.
Press any key to exit.
FS0:\E810\EFI\I2x64\> _
```

Figure 2. Example Update and Output

Note: The NVM update may require a two-step process depending on the initial image revision. Use the Software/NVM Compatibility table to verify the latest image versions.

When the flash image write completes, the tool asks for a reboot of the system to complete the update process and load the new firmware. After the reboot, boot the host into the intended operating system, and update the network adapter to the most current driver.

Note: When updating from early NVM images, the NVM Update Tool may indicate that a power-cycle of the system is necessary.

Note: The tool allows for updating one, multiple, or all of the installed adapters. For example, to update NVM firmware for two of three installed adapters, follow the syntax as shown in the example below. Enter selection 02,03 (separated by a commas).

```

Num Description                               Ver. DevId S:B      Status
=====
01) Intel(R) Ethernet Converged Network      1.147 1563 00:004 Update not available
Adapter X550-T2
02) Intel(R) Ethernet Network Adapter         5.81  158B 00:006 Update available
XXV710-2
03) Intel(R) Ethernet Converged Network      5.05  1583 00:131 Update available
Adapter XL710-Q2

Options: Adapter Index List (comma-separated), [A]ll, e[X]it
Enter selection:02,03
Would you like to back up the NVM images? [Y]es/[N]o: n
Update in progress. This operation may take several minutes.
[***+.....]
Reboot is required to complete the update process.

Tool execution completed with the following status: All operations completed successfully
Press any key to exit.

```

Figure 3. Example for Updating Multiple Adapters

Note: An NVM image downgrade process is available from Intel. Contact your Intel Representative for support of this process.

5.1 Recovery Mode

When using the NVM Update Tool, it is possible to get a status of "RECOVERY" or messages about Recovery Mode from the tool and/or Base Driver. If this occurs please refer to the [Recovery Mode for Intel® Ethernet Products Application Note](#) (Doc ID: 606286).

5.2 Command Line Parameters

When run with no parameters, the NVM Update Tool runs in Interactive mode.

Note: The **-i** and **-u** parameters are mutually exclusive. Do not use both parameters at the same time.

Table 1 lists and describes the parameters available when running the NVM Update Tool.

Table 1. Parameter Syntax

Parameter	Description
-a <path>	Specify a path for all file operations. The path is applied to all operations (such as locating the configuration file and NVM images, as well as where the log file and results file are generated). Any path specified in the configuration file is appended to the -a path. This parameter is used only with the -i or -u parameters.
-b	Save a backup copy of the current NVM image(s). The NVM Update Tool creates a subdirectory using the NIC(s) MAC Address as the directory name and stores the backup files there. Use this parameter only with the -u parameter.
-c <file_name>	Specify the name of the configuration file. This is a text file that contains descriptions of networking devices and firmware versions for those devices. The syntax of the configuration file can be found in Section 7.0 . If the -c is not specified, <i>nvmupdate.cfg</i> is used. If you do not specify a configuration file, <i>nvmupdate.cfg</i> must be present in the same directory as the NVM Update Tool for Interactive Mode to function correctly.
-f	Force the NVM update. The tool skips binary verification and applies the update even if the image version is the same.
-h or -?	Help Display command line usage help.
-i	Inventory mode. Lists the devices in the system and indicates the status of each device.
-k <hash_string>	Generate a SHA hash to compare against an internally-computed hash of the configuration file. If the hashes do not match, the NVM Update Tool aborts the procedure and reports an error.
-l <file_name>	Specify the name of the log file. This is a text file that contains a history of the NVM Update Tool's execution, including the success or failure status for each operation, and what adapters and ORMs were discovered. The log file is overwritten each time the NVM Update Tool is executed. If no file name is specified, the log messages are displayed on the system console.
-location <SS:BBB>	Specify a device for this instance of the tool to update. SS is the PCI segment of the desired device. BBB is the PCI bus of the desired device. You cannot run multiple instances of the tool on a specific device. The -location parameter must be used with the -i or -u parameter.
-m <MAC_Address>	Update only the device with the specified MAC Address. This is the LAN MAC Address, not the SAN or AltMAC Address. Note: The device must have an entry in the configuration file.
-o <file_name>	Specify the name and path of the results file. This is an XML file that contains the inventory/update results. If no file name is specified, the inventory/update results are displayed on the system console. Like the log file, the results file is overwritten each time the NVM Update Tool is executed.
-optinminsrev	Sets the minimum security revision of the NVM. In Update Mode, the -optinminsrev parameter sets the minimum security revision of the NVM that is allowed to be installed on all devices defined in the configuration file, even if MINSREV is set to FALSE or is missing for a device. If -optinminsrev is not specified, only devices that have MINSREV set to TRUE in the configuration file will have their minimum security revision increased. In Inventory Mode, displays the device's SRev and MinSRev values in the results XML file. The -optinminsrev parameter must be used with the -c parameter.

Table 1. Parameter Syntax [continued]

Parameter	Description
-p	Keep existing Option ROM image. Suppress update of the OROM.
-r	Rollback mode. Must be used with the -m parameter. You must have previously created a backup with the -b parameter for a restore point. The NVM Update Tool blocks rollback if the restore point NVM version is too old or does not contain critical updates.
-rd	Reset to default. Reset user settings to default values during update.
-RecoveryDevices	Recovery devices. Must be used only with the -i or -u parameters. If you specify -i -RecoveryDevices , only devices in recovery mode are displayed. If you specify -u -RecoveryDevices , only devices in recovery mode are updated.
-s	Silent mode. Specifying this option suppresses all output to the display. The NVM Update Tool output is limited to the results file and the log file.
-sv	Skip verification. Skip image verification.
-u	Update mode. Updates the devices in the system. You must provide a configuration file to use update mode.
-UpdateDevices	Update devices. Must be used only with the -i or -u parameters. Ignores devices in recovery mode during update. If you specify -i -UpdateDevices , devices in recovery mode are not displayed. If you specify -u -UpdateDevices , devices in recovery mode are not updated.
-v	Version Display the version of the QV SDK and Ethernet driver installed in the system (base driver for Linux, Solaris, FreeBSD, ESXi, and QV driver for Windows).

5.3 Exit Codes

Upon exit, when possible, the NVM Update Tool reports an overall status code to indicate the results of the operation. In general, a non-zero return code indicates an error occurred during processing.

Table 2 lists and describes the exit codes.

Table 2. Exit Codes

Value	Description
0	All operations completed successfully.
1	General tool execution error.
2	The configuration file could not be opened/read, or a syntax error was discovered in the file.
3	The inventory process failed.
4	A file error occurred when accessing the results file.
5	Bad command line parameter.
6	An error occurred when updating a firmware module.
7	A file error occurred when creating/writing the log file.
8	An error occurred accessing the device.

Table 2. Exit Codes [continued]

Value	Description
10	Package hash compare failed. Check integrity of the configuration file or correctness of the supplied hash.
12	The EEPROM MAP file could not be opened/read, or a syntax error was discovered in the file.
14	Reboot required to apply OROM size pointers (reboot to complete update).
15	Another instance of the NVM Update Tool is already running.
18	An error occurred during reset.
19	Device not found.
20	Communication with base driver failed. Please verify that the base driver is present.
21	Unsupported NVM image discovered. Please upgrade to the latest version of the NVM Update Tool.
22	The image backup process failed.
23	The requested image cannot be applied over the existing NVM content. Please download the most recent update package and retry the update.
24	Cannot restore manufacturing data.
25	Update stopped due to Rollback Revision mismatch. The NVM Update Tool blocks rollback if the restore point NVM version is too old or does not contain critical updates.
26	The selected adapter cannot be updated due to strict MMIO memory settings in the kernel. Set the <i>iomem</i> kernel parameter to "relaxed" and reboot the system before running this utility again. Consult the utility documentation for more information.
30	MINSREV update failed. The minimum security revision was not set for the device.
31	Image prerequisite check failed.
35	The tool cannot find the correct preservation rules in the NVM on the device. User settings cannot be preserved with this update. To update the NVM, overwrite the user settings by using the -rd switch.
36	Parallel execution of nvmupdate for this device is not allowed. Either there is already an instance of nvmupdate running on this device or the device does not support the -location parameter.
37	Requested image cannot be applied over existing NVM content. Please download the most recent update package and retry update.
38	NVM update functionality is not supported on this device using this driver. Please download the latest driver from SourceForge.
39	NVM update functionality in recovery mode is not supported on this device using this driver. Please download the latest driver from SourceForge.
40	Individual updates are not supported on this device using the kernel driver. Please download the latest driver from SourceForge.
50	Please perform the indicated reset action and run the NVM Update Tool again to complete the update.
51	Update available for one or more adapters ¹ .

1. EFI versions of this tool might report an incorrect error code when no adapter is installed. This is due to a known limitation in the UDK2015 UEFI Development Kit (UDK) build environment.

5.4 Fields Preserved During Reset to Default

Table 3 provides a list of fields whose values are preserved when using the **-rd** parameter during update. See Table 1 for more detail on the **-rd** parameter.

Table 3. Preserved Fields

Series	Preserved Fields
Intel® Ethernet 700 Series	<ul style="list-style-type: none"> • Auto-Generated GLPCI_SERL registers • Auto-Generated GLPCI_SERH registers • PRTGL_SAL/H registers • Auto-Generated PRTMAC_HSEC_CTL_TX_SA_PART1 registers • Auto-Generated PRTMAC_HSEC_CTL_TX_SA_PART2 registers • Auto-Generated PRTMAC_HSEC_CTL_RX_PAUSE_DA_UCAST_PART1 registers • Auto-Generated PRTMAC_HSEC_CTL_RX_PAUSE_DA_UCAST_PART2 registers • PBA • SAN MAC address • MNG MAC Address Pointer • PF MAC Address Pointer
Intel® Ethernet 500 Series	<ul style="list-style-type: none"> • PCIe Serial MAC address • MAC address Port 0 • MAC address Port 1 • PBA • Alternate SAN MAC address • SAN MAC address • Alternate MAC address

6.0 Troubleshooting

- Update to the most current base driver prior to running the NVM Update Tool to ensure the newest features of the NVM image can be installed.
- Refer to the “NVM and Software Compatibility” section in each of the following documents:

Document	Link
Intel® Ethernet Controller E810 Feature Support Matrix	Doc ID: 630155
Intel® Ethernet Controller X710-TM4/AT2 and V710-AT2 Feature Support Matrix	Doc ID: 619407
Intel® Ethernet Controller X710/XXV710/XL710 Feature Support Matrix	Doc ID: 332191
Intel® Ethernet Controller X550 Feature Support Matrix	Doc ID: 335253

The “Software/NVM Compatibility” tables indicate the set of NVM images and Intel® Ethernet Controller software releases that go together. Intel recommends that you update the NVM and Software driver to compatible versions.

The “NVM Transition Support” tables indicate the version of NVM from which the NVM Update Tool allows updates.

- In case of a security issue, the security revision might be incremented and then an NVM update to an older NVM with a lower security revision might not be allowed.
- The NVM version for the X550 is NOT shown in the GUI. Only the EEtrack ID is displayed. If you run **nvmupdate** with **-i**, the version is displayed.

6.1 Troubleshooting Using Debug Logs

1. Use the following command to get the log file if there is any error seen.

```
nvmupdate64e -l nvmupdate.log
```

This is a text file that contains history of the NVM Update Tool's execution, including the success or failure status for each operation, and what adapters and ORMs were discovered. After running this command, the tool creates the *nvmupdate.log* file under the same folder as *nvmupdate.cfg*. The log file is overwritten each time the NVM Update Tool is executed.

2. Use following command to get a little more information on what is in the system by using **nvmupdate** with an inventory mode.

```
nvmupdate64e -i -l inv.log
```

This provides more details about the adapters in the system to help narrow down the debug scope.

3. Use following command(s) to get a superset of debug logs.

First set following environment variables before **nvmupdate** execution. For debugging purposes, it is necessary to set these flags:

```
export NUL_DEBUGLOG=1
export QV_DEBUG_LOG=0xFFFFFFFF
```

Now the log generated using the following command is much more detailed.

```
nvmupdate64e -l nvmupdate.lo
```

If you continue to have issues, contact Intel support with all these log files.

7.0 Create/Edit *nvmupdate.cfg* for Custom NVM Images

The goal of this section is to assist Intel Ethernet users to create/edit the *nvmupdate.cfg* file for their custom NVM images. For the 800 Series, 700 Series, and 500 Series Network Adapters, this allows the use of NVM Update utility to update custom NVM images that are not included in the NVM Updated packages posted by Intel.

7.1 Sample Configuration File Template

The following is an example of a configuration file with one device block:

```
=====
CURRENT FAMILY: 1.0.0
CONFIG VERSION: 1.20.0

; NIC device
BEGIN DEVICE
DEVICENAME: E810_CQDA2_O_SEC_FW
VENDOR: 8086
DEVICE: 1592
SUBVENDOR: 8086
SUBDEVICE: 0002
NVM IMAGE: E810_CQDA2_O_SEC_FW_1p4p1p13_NVM_2p0_PLDMoMCTP_80003D96_signed_pldm_fixed.bin
EEPID: 80003D96
SKIP NETLIST: FALSE
IMAGE DOWNGRADE: TRUE
RESET TYPE: REBOOT
CURRENT GFID: 0157-1590
ORIGINAL GFID: 0157-1590
REVISION: 02
; REPLACES: 80003D96
END DEVICE
=====
```

7.2 Device Block in the Configuration File

A device block in configuration file lists out following information:

- **CONFIG VERSION** — Version of syntax for the configuration file.
- **DEVICENAME** — Device name currently in use. For example, Intel X550 Adapter, etc.
- **VENDOR** — PCI vendor ID 8086 identifies Intel as the manufacturer of the device.
- **DEVICE** — Device ID. Device IDs for supported retail Intel Ethernet Adapters can be found in here:
<https://www.intel.com/content/www/us/en/support/articles/000005612/network-and-i-o/ethernet-products.html>
- **SUBVENDOR** — Sub-vendor ID in hexadecimal format. This is optional when EEPID is used.
 - On 800 Series NIC entries, this is a mandatory field.
 - On 700 Series and 500 Series devices, this is optional when EEPID is used.
- **SUBDEVICE** — Sub-device ID in hexadecimal format. This is optional when EEPID is used.
 - On 800 Series NIC entries, this is a mandatory field.
 - On 700 Series and 500 Series devices, this is optional when EEPID is used.
- **NVM IMAGE** — NVM Image binary file name with which to update.

- **OROM IMAGE** — OROM Image file name.
- **EEPID** — ETrack ID of NVM Image with which to update.
- **REPLACES** — ETrack ID of NVM Image that with which to replace. Multiple ETrack IDs can be entered, separated by spaces.
 - On 800 Series NIC entries, the tool compares 4-part ID, which makes this field optional.
- **RESET TYPE** — Specifies whether reboot/power cycle is required to complete the NVM update process.
- **REVISION** — Revision number in hexadecimal format.
 - On 800 Series NIC devices, this field differentiates between B0 and C0 devices.
 - This is optional on 700 Series and 500 Series devices.
- **CURRENT GFID** — On 800 Series devices, the value consists of Intel IANA and Silicon default Device ID. If this field is not present but **ORIGINAL GFID** is, the tool uses **ORIGINAL GFID** value for both cases. This is important to reduce the time required for update to complete.
- **ORIGINAL GFID** — On 800 Series devices, the value consists of Intel IANA and Silicon default Device ID. If this field is not present but **CURRENT GFID** is, the tool uses **CURRENT GFID** value for both cases. This is important to reduce the time required for update to complete.

7.3 Steps to Create/Edit *nvmupdate.cfg* File to Update Custom NVM Image

7.3.1 On 800 Series Devices

1. Include the custom NVM Image binary file under the same folder as the *nvmupdate.cfg* file and *nvmupdate64e* executable file.
2. Open the *nvmupdate.cfg* file as text file.
3. Copy and paste one of the device blocks (or use the example in [Section 7.1](#)) and update following information for custom NVM image update.
 - a. Custom NVM Image binary file name in **NVM IMAGE** field.
 - b. ETrack ID (in the **EEPID** field) of the custom NVM Image listed in **NVM IMAGE** field. This is the image that the device will be update to.
 - c. ETrack ID (in **REPLACES** field) of NVM Image that an update will be allowed from applies only for LOM designs, and is not needed for NIC devices.

Note: When editing the *nvmupdate.cfg* file, if there is a need to have ETrack ID in the **REPLACES** field, you must ensure that this ETrack ID and the **EEPID** field are the same type of image and are both created for the device that is being updated.

- d. **DEVICE, VENDOR, SUBDEVICE, and SUBVENDOR** must be set correctly to match the device required to update.

The remaining fields (including **CURRENT FAMILY, CONFIG VERSION, DEVICENAME, and RESET TYPE**) can typically be left as is.

4. Run the *nvmupdate64e* executable file.

Following is an example of Device block in the configuration file. This includes the minimum fields required for an update. The fields listed in red should be updated:


```
=====
CURRENT FAMILY: 1.0.0
CONFIG VERSION: 1.20.0

; NIC device
BEGIN DEVICE
  DEVICENAME: E810_CQDA2_O_SEC_FW
  VENDOR: 8086
  DEVICE: 1592
  SUBVENDOR: 8086
  SUBDEVICE: 0002
  NVM IMAGE: nvmImage.bin [Include the NVM Image File name to be updated with]
  EEPROMID: 80003FFF [Mention ETrack ID of NVM Image that need to be updated with]
  SKIP NETLIST: FALSE
  IMAGE DOWNGRADE: TRUE
  RESET TYPE: REBOOT
  CURRENT GFID: 0157-1590
  ORIGINAL GFID: 0157-1590
  REVISION: 02
; REPLACES: 80003D96 [Optional when 4 part ID is used above, Multiple Etrack IDs
                      can be entered separated with space]
END DEVICE
=====
```

7.3.2 On 700 Series and 500 Series Devices

1. Include the custom NVM Image binary file under the same folder as the *nvmupdate.cfg* file and *nvmupdate64e* executable file.
2. Open the *nvmupdate.cfg* file as text file.
3. Copy and paste one of the device blocks (or use the example in [Section 7.1](#)) and update following information for custom NVM image update.
 - a. Custom NVM Image binary file name in **NVM IMAGE** field.
 - b. ETrack ID (in the **EEPROMID** field) of the custom NVM Image listed in **NVM IMAGE** field. This is the image that the device will be update to.
 - c. ETrack ID (in **REPLACES** field) of NVM Image that an update will be allowed from.

Note: When editing the *nvmupdate.cfg* file, you must ensure the ETrack ID in the **REPLACES** field and the **EEPROMID** field are the same type of image and are both created for the device that is being updated. For example, in the 700 Series, a device with CFG_ID 2.4 should be updated with an image with CFG_ID 2.4.

- d. **DEVICE**, **VENDOR**, **SUBDEVICE**, and **SUBVENDOR** must be set correctly to match the device required to update. **SUBDEVICE** and **SUBVENDOR** are optional when **EEPROMID** is used.

OROM update can be skipped by including the line `SKIP OROM: TRUE`. Use **bootutil** to update the OROM if necessary. The remaining fields, including **CURRENT FAMILY**, **CONFIG VERSION**, **DEVICENAME**, and **RESET TYPE**, can typically be left as is.

4. Run the *nvmupdate64e* executable file.

Following is an example of Device block in the configuration file. This includes the minimum fields required for an update. The fields listed in red should be updated:

```
=====
CURRENT FAMILY: 12.1.1
CONFIG VERSION: 1.7.0

BEGIN DEVICE
DEVICENAME: Intel x540 Adapter
VENDOR: 8086
DEVICE: 10C9
NVM IMAGE: nvmImage.bin [Include the NVM Image File name to be updated with]
SKIP OROM: TRUE
EEPID: 800007A9 [Mention ETrack ID of NVM Image that need to be updated with]
REPLACES: 80000692 [Looks for Etrack ID of NVM Image that need to be replaced,
Multiple Etrack IDs can be entered separated with space]

RESET TYPE: POWER
END DEVICE
=====
```

7.4 Configuration File Rules and Restrictions

- You cannot update the EEPROM and the NVM in the same session. Only one should be specified in the configuration file.
- You can update the EEPROM and the OROM in the same session. The EEPROM will be updated first, followed by the OROM.
- You can update the NVM and the OROM in the same session. The NVM will be updated first, followed by the OROM.
- Inventory mode does not consider potential EEPROM/NVM image changes if a sequential OROM update is requested (e.g., an additional reboot might be required).
- On multi-port adapters, due to there being only one EEPROM/flash that is shared by all ports, only port 0 will be updated (or the port with the MAC Address specified with the **-m** command line parameter).
- Blank lines or lines beginning with a semicolon (comments) are ignored.
- Unrecognized keywords in the configuration file will result in the NVM Update Tool exiting abnormally and reporting a syntax error (exit code=2). Run the NVM Update Tool again, with the Enable Logging command line parameter (**-l**), to see what specific line is in error.
- An OROM update is allowed only if the specified OROM component set matches combo rules (the vendor/device pairs defined in the *.flb* file).
- An OROM downgrade is allowed only if a value of TRUE is assigned to the **OROM DOWNGRADE** key.
- If the configuration file specifies the **OROM IMAGE** field, but does not specify any OROM components (PXE, EFI, etc.), the device's current set of OROM components is used as the component list.
- An OROM update preserves the current configuration of components, or resets the configuration to the default if new settings do not allow for configuration preservation (e.g., when combo rules differ).
- ETrackID of EEPROM or NVM and OROM components version values specified in the configuration file are compared against values stored in *.eep/.bin* or *.flb* files. If a mismatch is found, the inventory/update is aborted.

- If the ETrackID of EEPROM or NVM and OROM components version values are not specified in the configuration file, values from *.eep/.bin* or *.flb* are taken without verification.
- EEP/NVM/OROM update is always allowed if running image versions cannot be specified or are invalid.
- EEPROM/NVM downgrade is allowed only when the **REPLACES** key is specified with an active ETrackID value.
- A candidate match based on sub-IDs requires that both **SUBVENDOR** and **SUBDEVICE** fields are present in the configuration file.
- If the **REPLACES** field is not present, you must include the **SUBVENDOR** and **SUBDEVICE** fields in the configuration file. **SUBVENDOR** and **SUBDEVICE** are optional if **REPLACES** is present.
- The NVM Update Tool will not report errors for devices that are not specified in the config file.
- If recovery is requested but there are no devices in recovery mode, the NVM Update Tool will return success.

8.0 Summary

Updating the NVM and network driver can increase performance, manageability, and reliability of the 800 Series, 700 Series, and 500 Series Network Adapters. The update process has a built-in integrity feature to ensure that only Intel-approved firmware code is able to be updated after manufacturing. This procedure is performed each time an attempt is made to update one of the protected modules.

Intel Customer Support Services offers a broad selection of technical and customer support programs. For more information, contact your local Intel representative. Service and availability may vary by country.

For more information on the 800 Series Network Adapter family, go to the following links:

- [Intel® Ethernet 800 Series Network Adapters](#)
- [Downloads for Intel® Ethernet Network Adapter E810 Series](#)
- [Intel® Ethernet 800 Series Controllers](#)
- [Intel® Ethernet Controller E810 Technical Library](#)
- [Downloads for Intel® Ethernet Controller 800 Series](#)

For more information on the 700 Series Network Adapter family, go to the following links:

- [Intel® Ethernet 700 Series Network Adapters](#)
- [Downloads for Intel® Ethernet Network Adapter X710 Series](#)
- [Downloads for Intel® Ethernet Network Adapter XXV710 Series](#)
- [Downloads for Intel® Ethernet Server Adapter XL710 Series](#)
- [Intel® Ethernet 700 Series Controllers](#)
- [Intel® Ethernet Controller X710, 10 GbE Technical Library](#)
- [Intel® Ethernet Controller XXV710, 25 GbE Technical Library](#)
- [Intel® Ethernet Controller XL710, 40 GbE Technical Library](#)
- [Intel® Ethernet Controller X710-TM4/AT2 Technical Library](#)
- [Downloads for Intel® Ethernet 700 Series Controllers](#)

For more information on the 500 Series Network Adapter family, go to the following links:

- [Intel® Ethernet 500 Series Network Adapters](#)
- [Downloads for Intel® Ethernet Converged Network Adapter X550 Series](#)
- [Intel® Ethernet 500 Series Controllers](#)
- [Intel® Ethernet Controller X550 Series Technical Library](#)
- [Downloads for Intel® Ethernet 500 Series Controllers](#)

NOTE: *This page intentionally left blank.*



LEGAL

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

This document (and any related software) is Intel copyrighted material, and your use is governed by the express license under which it is provided to you. Unless the license provides otherwise, you may not use, modify, copy, publish, distribute, disclose or transmit this document (and related materials) without Intel's prior written permission. This document (and related materials) is provided as is, with no express or implied warranties, other than those that are expressly stated in the license.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

The products and services described may contain defects or errors which may cause deviations from published specifications.

Copies of documents referenced in this document may be obtained by visiting www.intel.com/design/literature.htm.

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

Other names and brands may be claimed as the property of others.

© 2015-2022 Intel Corporation.