



Customization Guide for NVIDIA Performance Control Panel

CONFIDENTIAL INFORMATION
DO NOT COPY

Version 1.6
November, 2008

PLATFORM PROCESSORS

Document Change History

Version	Date	Owner	Reason for Change
1.0	02/13/08	D. Hunt	Initial release
1.1	07/28/08	M. Peters	Added default CMOS section.
1.2	08/01/08	M. Peters	Added loading specific default profile across new users
1.3	08/08/08	M. Peters	Added section to disable device icons
1.4	08/11/08	M. Peters	Fixed typo in Aux fan control.
1.5	09/29/08	M. Peters	Update OEM default profile section with correct steps.
1.6	11/21/08	M. Peters	Added information to automatically agree to EULA for OEMs.

CONFIDENTIAL INFORMATION
DO NOT COPY

Table of Contents

1.0	NVSUOEM.INI.....	2
1.1	INI File Checksum	2
1.2	BIOS integration of the INI File.....	3
1.2.1	Table DESCRIPTION_HEADER format:	3
1.3	Voltage Settings	4
1.3.1	Linear Stepping Example.....	4
1.3.2	Non-linear Stepping Example.....	4
1.4	Memory Slot Definition.....	5
1.5	BIOS Fields.....	6
1.5.1	Setting up the Pages	6
1.5.2	Defining Field Values and CMOS Interface	7
1.5.2.1	Field Values.....	7
1.5.2.2	CMOS Interface	8
1.5.2.3	Defining Default Values	8
1.5.3	Establishing Field Dependencies	8
1.6	Changing Auxiliary Fan Label.....	8
1.7	LED Control	9
1.8	Disabling Device Settings icons	9
2.0	Registry Settings.....	11
2.1	Loading Default Profile to New Users.....	11
2.2	Agreeing to EULA for Pre-Install images	12



NVIDIA.

NVIDIA Corporation
2701 San Tomas Expressway
Santa Clara, CA 95050
www.nvidia.com

1.0 NVSUDOEM.INI

To enable the advanced features of the NVIDIA nTune, the OEM must provide a customized nvsuoem.ini file for their platform. The OEM can define voltage settings, levels of support, memory slot configuration, and the BIOS field information used on the “Dynamic BIOS Access” page.

The same nvsuoem.ini may be used for multiple platforms. Each set of platform information should start with the model name (ex. [NVK8-CRB]).

Note: The model name must match the name reported by the System BIOS in the NVIDIA nTune BIOS interface sub-function 25h. If this is not correct, the advanced functions will not be enabled.

1.1 INI File Checksum

To prevent a user from manipulating the data in the nvsuoem.ini file, a checksum has been added to the last 4-bytes of the file. This checksum must be removed and updated any time the OEM changes a setting in the INI file. The tool for appending the checksum to the file is WinWptTest.exe, which is described in a later section.

Note: If the checksum is missing or not valid, nTune will revert to the basic feature set and alert the user to upgrade the INI file or reinstall nTune.

1.2 BIOS integration of the INI File

If desirable, the System BIOS may integrate the INI file as part of its BIOS image. This allows the customization to be easily tied to a platform without the overhead of maintaining separate INI file install images.

1.2.1 Table DESCRIPTION_HEADER format:

Signature (4 bytes): OEMN
Length (4 bytes): Table length
Revision (1 byte): 01
Checksum (1byte)
OEMID (6 bytes): NVIDIA
OEM TableID (8 bytes): NTUNEOEM
OEM Revision (4 bytes): 1
Creator ID: ignored
Creator Revision: ignored

To integrate the file and header the System BIOS must do the following:

1. Zip the oem.ini file and included in System BIOS binary
2. During POST, SBIOS unzip oem.ini and inserts above table entry with unzipped oem.ini to RSDT table.

1.3 Voltage Settings

Voltage settings are defined for:

- **cpu_**: used for CPU voltage
- **mem_**: used for Memory voltage
- **agp_**: used to define AGP, PCI-E, or SPP core (for XE parts) voltage
- **core_**: used for Chipset core or MCP core (on XE parts) voltage
- **aux_**: used for Auxiliary voltage
- **htcpuspp_**: used for the HT link to the CPU voltage
- **htsppmcp_**: used for the HT link between the SPP and MCP voltage

The following examples will show setting of the agp_ voltage parameters.

1.3.1 Linear Stepping Example

This sets the minimum voltage for AGP to 1.5v.

```
agp_min = 1500000
```

This sets the maximum voltage for AGP to 1.7v.

```
agp_max = 1700000
```

This means the steps will increase between the min and max values at a rate of 0.1.

```
agp_increment_step=100000
```

1.3.2 Non-linear Stepping Example

This lists all the values for the specific voltage setting.

```
agp_list=1500000,1600000,1800000
```

This defines the number of values for the specific voltage setting.

```
agp_count=3
```

1.4 Memory Slot Definition

To ensure the DIMM slot numbering reported on the “Information” of NVIDIA nTune is accurate, the OEM may define the address map in the nvsuoem.ini file.

Example:

This defines the number of slots on the platform. In this example, a total of 2 slots have been defined.

```
NumberOfDimm=2
```

This defines the SMBus address location. In this example, the setting is for DIMM 0.

```
Dimm0Smbus =0x50
```

This defines what number the OEM uses for this slot. In this example, the setting is for DIMM 0.

```
Dimm0Slot=0
```

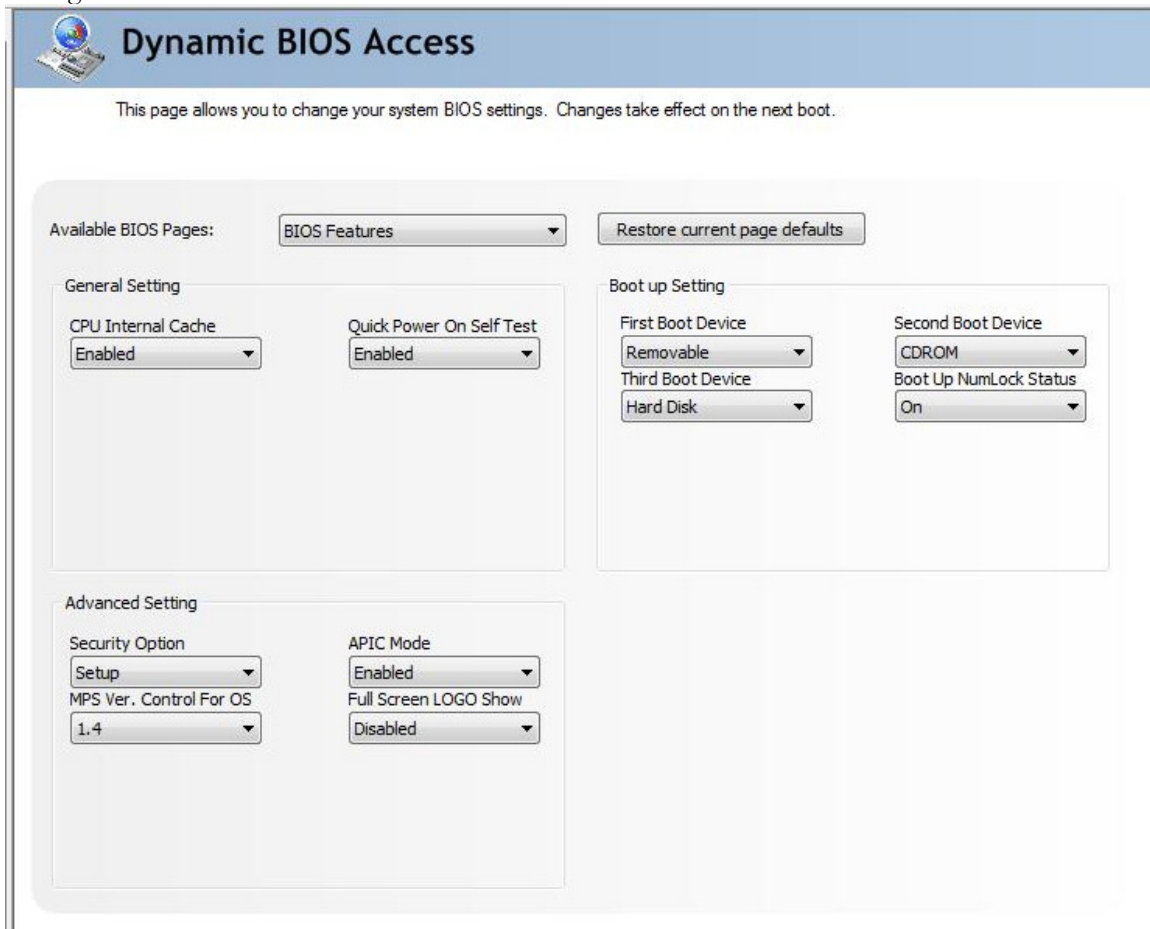
This defines the segment and is only used in a DIMM configuration of greater than 8 slots. In this example, the setting is for DIMM 0.

```
Dimm0SmbusSeg=0
```

CONFIDENTIAL INFORMATION
DO NOT COPY

1.5 BIOS Fields

The OEM may elect to provide a thorough access to the BIOS from within Windows. This interface is built from the definitions in the nvsuoem.ini file. The OEM may define up to 8 pages consisting of up to 4 groups. Each group may have up to 8 fields. This equates to a possible 256 BIOS settings.



1.5.1 Setting up the Pages

This defines the total number of pages (limited to 8) that can be access on the "Access BIOS" page.

TotalTab=5

This will define the name for page 0.

T0Name=BIOS Features

This will define the number of groups for page 0.

T0GroupNo=3

This will define the name for group 2.

```
T0Group2Name=Advanced Setting
```

This will define the number of fields for group 2.

```
T0G2CmosFieldNo=5
```

This will define the name for field 1 of group 2 on page 0.

```
T0G2Cmos1Name=APIC Mode
```

This will indicate to show the “Restore Current page defaults” button on each page.

```
EnabledDBARestoreDefaults=1
```

1.5.2 Defining Field Values and CMOS Interface

1.5.2.1 Field Values

Example 1:

This would be an example of typical usage.

```
T1G0Cmos0Value=Optimal|Aggressive|Expert
```

In this example, for field 0 of group 0 of page 1; the user will see the following settings in the pull-down list: Optimal, Aggressive, and Expert

Note: The | symbol is used to separate the values.

Note: The value of each selection is assigned sequentially and linearly. This means “Optimal” = 0x0, “Aggressive” = 0x1, and “Expert” = 0x2.

Example 2:

There may be cases in which it is desirable to skip certain values, but maintain a specific list order.

```
T1G0Cmos0Value=Optimal|Aggressive|N/A|N/A|Expert
```

In this example, the pull-down list will still display: Optimal, Aggressive, and Expert; however, the values will be assigned as follows: “Optimal” = 0x0, “Aggressive” = 0x1, and “Expert” = 0x4.

Example 3:

Due to the length of some lists, it may be necessary to continue the values on the next line. This can be accomplished by using the “|” at the end of the first line to notify NVIDIA nTune to continue with the next. In this context the next line is viewed as a continuation of the previous, and the value assignment follows the rules.

```
T1G0Cmos1Value=Optimal|200.0|202.0|203.0|204.0|205.3|206.0|207.0|208.0|209.0|
```

```
T1G0Cmos1Value0=211.0|212.0|213.0|214.0|215.0|216.0|217.0|218.0|219.0|220.0|
```

```
T1G0Cmos1Value1=221.0|222.0|223.0|224.0|225.0|226.0|227.0|228.0|229.0|230.0
```

1.5.2.2 CMOS Interface

Defines the CMOS location for field 2 of group 1 on page 0 as 0x34

```
T0G1Cmos2Location=0x34
```

Defines the bitmask used for field 2 of group 1 on page 0 as 0x01

```
T0G1Cmos2BitMask=0x01
```

1.5.2.3 Defining Default Values

Define the CMOS default value for field 2 of group 1 on page 0 as a value of 1

```
T0G1Cmos2DefaultValue=0x00000001
```

Bit 31 must be 0 for default CMOS to indicate supported.

Note: Only available in Performance 6.2 and higher

1.5.3 Establishing Field Dependencies

On occasion, it may be necessary to have a specific field locked until another field is set to a specific value. An example is the locking of memory timing fields unless “Expert” mode is selected.

Example:

Defines the control field for unlocking field 1 of group 0 on page 1.

```
T1G0Cmos1Control=T1G0Cmos0
```

Defines the control value for unlocking field 1 of group 0 on page 1.

```
T1G0Cmos1CtrlValue=Expert
```

1.6 Changing Auxiliary Fan Label

There are two auxiliary fan controls available to the ODM/OEM. These controls may be used for other fans or to represent some other cooling control. To change the label used for these sliders, add the following to the INI entry for the platform (Limited to 8 characters).

Example:

Change the label for the “Aux 1” fan slider.

```
Aux1FanLabel=HDD Fan
```

Change the label for the “Aux 2” fan slider.

```
Aux2FanLabel=PWR Fan
```

Note: Available in nTune 5.5 and higher

1.7 LED Control

This control is for a non-ESA LED control implementation.

The OEM/ODM may add an LED control page to allow the user to manage the chassis LEDs. The following may be controlled: LED color, intensity, and effects. Also, these controls may be setup into different zones.

Examples:

Defines the feature as enabled(1) or disabled(0)

```
LedIntensity=0
```

Defines the number of active zones (1-4)

```
LedZoneCount=1
```

Defines the number of LED color variations (0-Fh)

```
LedColorCount=7
```

Defines the: [color name] | [value passed through WPT} and the [R][G][B] value for the UI. This list should match the total count defined by LedColorCount.

```
LedColor0=Ruby|1 255 0 0
```

```
LedColor1=Emerald|2 80 200 120
```

```
LedColor2=Amber|3 255 191 0
```

```
LedColor*=Sapphire|* 8 37 103
```

Defines the “Off” color: [value passed through WPT} and the [R][G][B] value for the UI.

```
LedColorOff=0 227 229 236
```

1.8 Disabling Device Settings icons

Each of the Device Settings buttons can be disabled from appearing through the ini file.

- DisableCpuTab – Hide the CPU icon button in Device settings

PROPRIETARY INFORMATION
DO NOT DISTRIBUTE

- DisableMemTab – Hide the Mem icon button in Device settings
- DisableMbTab – Hide the Motherboard icon button in Device settings
- DisableGpuTab – Hide the GPU icon button in Device settings
- DisableESATabs – Hide all ESA Device icon buttons in Device settings
- DisableESACHasisTab – Hide the Chassis icon button in Device settings
- DisableESAPsTab – Hide the Power Supply icon button in Device settings
- DisableESAWcTab – Hide the Water Cooler icon button in Device settings

2.0 Registry Settings

2.1 Loading Default Profile to New Users

The ability to load a specific profile as the default profile to all new users was added to Performance 6.3 or later using registry keys. To use this feature, OEMs need to perform the following steps:

1. Open Registry using regedit or similar program.
2. Click on HKEY_USERS tree.
3. Under File menu, choose "Load Hive ...". For the file to load choose the NTUser.dat under C:\Users\Default (Vista) or C:\Documents and Settings\Default User (XP) and click Open. A dialog will pop up to ask for a name so type in DefaultUser and click OK.
4. Now, under HKEY_USERS, there is one more entry called DefaultUser. Create a string value named "NvSuCurOemDfltProfile" with the value set to the default profile (for example "C:\windows\oem.nsu") under HKEY_USERS\DefaultUser\Software\NVIDIA Corporation\Ntune.
5. Click on DefaultUser and Under File menu, choose "Unload Hive...".

When a new user is created, this registry value is copied to current user registry.

When Performance is launched for first time, it will:

1. Check the registry value and if the profile is valid, copy it to user profile directory.
2. Load this profile.
3. Set the profile to load when windows starts and added it to profile rule control.
4. Notify nTuneService.

Note: Only available in Performance 6.3 and higher

2.2 Agreeing to EULA for Pre-Install images

This option allows OEMs to avoid the EULA message to the end user.

1. Open Registry using regedit or similar program.
2. Click on HKEY_USERS tree.
3. Under File menu, choose "Load Hive ...". For the file to load choose the NTUser.dat under C:\Users\Default (Vista) or C:\Documents and Settings\Default User (XP) and click Open. A dialog will pop up to ask for a name so type in DefaultUser and click OK.
4. Now, under HKEY_USERS, there is one more entry called DefaultUser. Create a string value named "NvSysEulaAgreee" with the value set to "Yes" under HKEY_USERS\DefaultUser\Software\NVIDIA Corporation\nTune.
5. Click on DefaultUser and Under File menu, choose "Unload Hive...".

Information furnished is believed to be accurate and reliable. However, NVIDIA Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. NVIDIA Corporation products are not authorized for use as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

Trademarks

NVIDIA and the NVIDIA logo are registered trademarks and nForce is a trademark of NVIDIA Corporation
Microsoft, Windows and the Windows logo are registered trademarks of Microsoft Corporation

Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

Copyright NVIDIA Corporation 2008

CONFIDENTIAL INFORMATION
DO NOT COPY