How to Use an INF to Override the Monitor EDID

November 5, 2008

Abstract

All display devices (monitors, projectors, and televisions), analog or digital, must supply Extended Display Identification Data (EDID), which is stored in the monitor’s EEPROM. Sometimes during manufacturing, incorrect or incomplete EDID might be flashed on the EEPROM. Such EDIDs can lead to problems such as setting incorrect display modes. This white paper describes how monitor manufacturers can avoid reflashing the monitor’s EEPROM by implementing a monitor INF that overrides part of, or the entire, EDID information in software. This paper describes how to implement such an INF.

This information applies for the following operating systems:
 Windows® 7
 Windows Server® 2008
 Windows Vista®

References and resources discussed here are listed at the end of this paper.

For the latest information, see:
 <http://www.microsoft.com/whdc/device/display/edid_over.mspx>

Disclaimer: This is a preliminary document and may be changed substantially prior to final commercial release of the software described herein.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, email address, logo, person, place or event is intended or should be inferred.

© 2008 Microsoft Corporation. All rights reserved.

Microsoft, Windows, Windows Server, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Document History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Change |  |  |  |
| November 5, 2008 | First publication |

Contents

[Introduction 3](#_Toc212365938)

[How to Update an EDID by Using an INF 3](#_Toc212365939)

[EDID Format 4](#_Toc212365940)

[How to Override EDID Information in an INF 4](#_Toc212365941)

[Implementation Requirements and Guidelines 5](#_Toc212365942)

[Windows Logo Program EDID Requirements 7](#_Toc212365943)

[Call to Action 8](#_Toc212365944)

[Resources 8](#_Toc212365945)

# Introduction

All monitors, analog or digital, must support Extended Display Identification Data (EDID), which contains information such as the monitor identifier, manufacturer data, hardware identifier, timing information, and so on. This data is stored in the monitor’s EEPROM in a format that is specified by the Video Electronics Standards Association (VESA).

Monitors provide the EDID to Windows® components, display drivers, and some user-mode applications. For example, during initialization the monitor driver queries the WDDM driver for brightness query interface and device driver interface (DDI) support, which is in the EDID. Incorrect or invalid EDID information on the monitor’s EEPROM can therefore lead to problems such as setting incorrect display modes.

There are two approaches to correcting EDIDs:

* The standard solution is to have the customer send the monitor back to the manufacturer, who reflashes the EEPROM with the correct EDID and returns the monitor to the customer.
* A better solution is for the manufacturer to implement an INF file that contains the correct EDID information and download it to the customer’s computer. Windows extracts the updated EDID information from the INF and provides it to components instead of the information from the EEPROM EDID, effectively overriding the EEPROM EDID.

This paper describes how to implement an INF to override a monitor’s EDID and briefly discusses the related Windows logo requirements.

In addition to replacing the EDID information as described in this white paper, a vendor can provide an override for the monitor name and the preferred display resolution. Such an override is frequently made available to end users through Windows Update or digital media in the shipped box. Such an override receives higher precedence than the EDID override mentioned in this paper. Guidelines for achieving this can be found in “Monitor INF File Sections” in the Windows Driver Kit (WDK).

# How to Update an EDID by Using an INF

To update an EDID by using an INF:

1. The monitor manufacturer implements an INF that contains the updated EDID information and downloads the file to the user’s computer. This can be done through Windows Update or by shipping a CD with the monitor.

2. The monitor class installer extracts the updated EDID information from the INF and stores the information as values under the following registry key:

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Enum\DISPLAY

 Each EDID override is stored under a separate key, for example:

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Enum\DISPLAY*\DELA007\
 5&1608c50f&0&10000090&01&20*\Device Parameters\EDID\_Override

3. The monitor driver checks the registry during initialization and uses any EDID information that is stored there instead of the corresponding information on EEPROM.

EDID information that has been added to the registry always takes precedence over EEPROM EDID.

4. Windows components and user-mode applications use the updated EDID information.

## EDID Format

EDID data is formatted as one or more 128-byte blocks:

* EDID version 1.0 through 1.2 consists of a single block of data, per the VESA specification.
* With EDID version 1.3 or enhanced EDID (E-EDID), manufacturers can specify one or more extension blocks in addition to the primary block.

Each block is numbered, starting with 0 for the initial block. To update EDID information, the manufacturer’s INF specifies the number of the block to be updated and provides 128 bytes of EDID data to replace the original block. The monitor driver obtains the updated data for the corrected blocks from the registry and uses the EEPROM data for the remaining blocks.

## How to Override EDID Information in an INF

To override an EDID, include an **AddReg** directive in the INF for each block that you want to override, in the following format:

HKR, EDID\_OVERRIDE, *BlockNumber*, *Byte 1, Byte 2, Byte 3, Byte 4*,...

The block number is followed by 128 hexadecimal integers that contain the binary EDID data.

Manufacturers must update only those EDID blocks that are incorrect. The system obtains the remaining blocks from EEPROM. The following example shows the relevant sections of an INF that updates EDID blocks 0, 4, and 5. The monitor driver obtains blocks 1 - 3 and any extension blocks that follow block 5 from EEPROM:

[ABC.*DDInstall*.HW]

ABC.AddReg

...

[ABC.AddReg]

HKR, EDID\_OVERRIDE, 0, 1, 00, FF, ..., 3B

HKR, EDID\_OVERRIDE, 4, 1, 1F, 3E, ..., 4E

HKR, EDID\_OVERRIDE, 5, 1, 24, 5C, ..., 2D

...

For more information on INFs in general and **AddReg** and *DDInstall* in particular, see “Creating an INF File” in the WDK.

# Implementation Requirements and Guidelines

The following example shows a complete generic INF file for overriding monitor EDID information for Windows Vista® and later versions. The example is an edited version of a WDK sample—monsamp.inf—which is located at WinDDK\*BuildNumber*\src\video\monitors\inf\monsamp.inf. For an explanation of the numbered comments, see the notes after the following example:

[Version]

signature="$WINDOWS NT$"

Class=Monitor

ClassGuid={4D36E96E-E325-11CE-BFC1-08002BE10318}

Provider="MS\_EDID\_OVERRIDE"

DriverVer=04/18/2006, 1.0.0.0

**;[1]**

;CatalogFile=Sample.cat

[DestinationDirs]

DefaultDestDir=23

[SourceDisksNames]

1=%SourceDisksNames%

**;[2]**

[SourceDisksFiles]

;profile1.icm=1

[Manufacturer]

%MS\_EDID\_OVERRIDE%=MS\_EDID\_OVERRIDE,NTx86,NTamd64

**;[3]**

[MS\_EDID\_OVERRIDE.NTx86]

%MS\_EDID\_OVERRIDE-1%=MS\_EDID\_OVERRIDE-1.Install, MONITOR\MON1234

[MS\_EDID\_OVERRIDE.NTamd64]

%MS\_EDID\_OVERRIDE-1%=MS\_EDID\_OVERRIDE-1.Install.NTamd64, MONITOR\MON1234

[MS\_EDID\_OVERRIDE-1.Install.NTx86]

DelReg=DEL\_CURRENT\_REG

AddReg=MS\_EDID\_OVERRIDE-1.AddReg, 1024, 1280, DPMS

CopyFiles=MS\_EDID\_OVERRIDE-1.CopyFiles

[MS\_EDID\_OVERRIDE-1.Install.NTamd64]

DelReg=DEL\_CURRENT\_REG

AddReg=MS\_EDID\_OVERRIDE-1.AddReg, 1024, 1280, DPMS

CopyFiles=MS\_EDID\_OVERRIDE-1.CopyFiles

[MS\_EDID\_OVERRIDE-1.Install.NTx86.HW]

AddReg=MS\_EDID\_OVERRIDE-1\_AddReg

[MS\_EDID\_OVERRIDE-1.Install.NTamd64.HW]

AddReg=MS\_EDID\_OVERRIDE-1\_AddReg

**;[4]**

[MS\_EDID\_OVERRIDE-1\_AddReg]

HKR,EDID\_OVERRIDE,"0",0x01,0x00,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x00,0x35,0xEE,0x34,0x12,0x01,0x00,0x00,0x00,0x0A,0x0E,0x01,0x03,0x68,0x22,0x1B,0x78,0xEA,0xAE,0xA5,0xA6,0x54,0x4C,0x99,0x26,0x14,0x50,0x54,0xA5,0x4B,0x00,0x71,0x4F,0x81,0x80,0xA9,0x40,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x30,0x2A,0x00,0x98,0x51,0x00,0x2A,0x40,0x30,0x70,0x13,0x00,0x52,0x0E,0x11,0x00,0x00,0x1E,0x00,0x00,0x00,0xFF,0x00,0x41,0x42,0x30,0x30,0x30,0x30,0x30,0x30,0x30,0x30,0x30,0x31,0x0A,0x00,0x00,0x00,0xFC,0x00,0x4D,0x53,0x20,0x31,0x32,0x33,0x34,0x0A,0x0A,0x0A,0x0A,0x0A,0x0A,0x00,0x00,0x00,0xFD,0x00,0x38,0x4C,0x1F,0x50,0x12,0x00,0x0A,0x20,0x20,0x20,0x20,0x20,0x20,0x00,0xDB

[DEL\_CURRENT\_REG]

HKR,MODES

HKR,,MaxResolution

HKR,,DPMS

HKR,,ICMProfile

**;[5]**

[1024]

HKR,,MaxResolution,,"1024,768"

[1280]

HKR,,MaxResolution,,"1280,1024"

[DPMS]

HKR,,DPMS,,1

[MS\_EDID\_OVERRIDE-1.AddReg]

HKR,"MODES\1024,768",Mode1,,"31.0-94.0,55.0-160.0,+,+"

HKR,"MODES\1280,1024",Mode1,,"31.0-94.0,55.0-160.0,+,+"

**;[6]**

[MS\_EDID\_OVERRIDE-1.CopyFiles]

;PROFILE1.ICM

[Strings]

MonitorClassName="Monitor"

SourceDisksNames="MS\_EDID\_OVERRIDE Monitor EDID Override Installation Disk"

MS\_EDID\_OVERRIDE="MS\_EDID\_OVERRIDE"

MS\_EDID\_OVERRIDE-1="MS EDID Override"

**Notes:**

1. Released versions of this INF should be accompanied by a signed catalog (.cat) file. To do this, remove the comment character from the following directive and change Sample.cat to the name of your signed catalog file.

2. To copy a monitor profile, remove the comment from the following directive. You must also add a corresponding **CopyFiles** section, as discussed in note 6.

3. Change the hardware ID from MON1234 to that of the monitor that is associated with this INF.

4. Replace the following example EDID directive with one or more EDID directives that contain the corrected information for each block that you want to override.

5. Use the following **AddReg** directives to define default settings for standard resolutions.

6. To copy a monitor profile, remove the comment character from the following directive.

# Windows Logo Program EDID Requirements

The Windows Logo Program requires display devices to provide a well-formed EDID. The specific requirement is Display-0065. The following list summarizes the requirements at the time this paper was published. You should check the latest version of the Windows logo specification for the current requirements.

#### Display - 0065: Display device implements the EDID data structure

**Effective Date:** 01-Jun-2006
**Expiration Date:** None
**Status:** Approved
**Required for:**
 **Basic:** Yes
 **Premium:** Yes

The monitor must transmit an EDID that contains:

* A unique manufacturer ID.
* A unique manufacturer name.
* A unique product code ID.
* All the required fields as defined by the VESA Enhanced Extended Display Identification Data Standard specification, Release A, Section 3.
* For analog CRTs, the EDID must indicate at least one VESA mode at 75 Hz or higher for each supported resolution.

All monitors must support E-EDID by implementing an EDID 1.3 or later data structure that sets the preferred -mode bit for all monitor types. The EDID must:

* Include timing data for the preferred display mode in Timing #1:

For an LCD or other fixed-format display, the preferred display mode is the panel’s native mode.

For other display types, the preferred display mode is the optimal display mode. It must be based on the size and capabilities of the device and meet the requirements for refresh rates that were defined earlier in this paper.

* Implement monitor descriptors.

The descriptors must include the monitor name and—for continuous frequency monitors—the monitor range limits. If the monitor does not support continuous frequencies, monitor range limits should not be exposed.

All-in-one systems—including mobile systems—must provide an EDID in one of the following ways:

* The LCD panel provides the EDID, much like an externally attached monitor.
* If the LCD panel does not provide an EDID, the WDDM miniport driver provides the EDID by using one of the following approaches:

The WDDM miniport driver defines the EDID itself.

The WDDM miniport driver retrieves the EDID from the system BIOS by executing the ACPI **\_DDC** method on the child device that is associated with the internal panel.

**Design and Implementation Note:** The ACPI specification defines the method that is used to obtain the EDID from the BIOS so that it has functionality the same as that specified in ACPI 2.0b, Appendix B, or later.

No hardware requirements specifically are related to overriding EEPROM EDIDs, so supplying an EDID override in the monitor INF is completely optional. In particular, including EDID information in the monitor INF is not necessary if the monitor’s EEPROM provides the correct information.

# Call to Action

Monitor manufacturers:

If your display devices contain incorrect EDID data in the EEPROM, supply the correct information in your monitor INF. This approach significantly reduces support costs compared to shipping display devices back to the factory to reflash the EEPROM.

**Note:** Monitor manufacturers must update the monitor INFs through Windows Update.

Graphics hardware vendors:

Design your Windows Display Driver Model (WDDM) drivers to take advantage of EDID overrides.

System builders:

Build systems that work with many different display devices and switchboxes.

# Resources

If you have additional questions that are not addressed by this document, send e‑mail to directx@microsoft.com. The following links provide further information on EDIDs and related topics.

Windows Driver Kit

Monitor INF File Sections

<http://msdn.microsoft.com/en-us/library/ms801139.aspx>

Creating an INF File

<http://msdn2.microsoft.com/en-us/library/ms790220.aspx>

WHDC Web site

Display and Graphics – Architecture and Driver Support Resources

<http://www.microsoft.com/whdc/device/display/default.mspx>

Windows Logo Program Requirements Suite, Version 3.0

<http://www.microsoft.com/whdc/winlogo/hwrequirements.mspx>

Video Electronics Standards Association (VESA)

<http://www.vesa.org>

VESA Standards Summaries

<http://www.vesa.org/Standards/summaries.htm>