VESA®

Video Electronics Standards Association

2150 North First Street, Suite 440
San Jose, CA 95131-2029

Phone: (408) 435-0333
FAX: (408) 435-8225

VESA BIOS Extension/ Flat Panel Interface (VBE/FP) Proposal

Version: 1.0
Revision: 0.32p
Revision Date: January 28, 1997

IMPORTANT NOTICE: This is a draft proposal from the VESA Software Standards Committee. This proposal has not yet been ratified by the VESA general membership and is subject to change.

Purpose

To standardize a common software interface to Flat Panel controllers, to simplify and encourage the development of applications that use the special abilities of most flat panel controller chips found in notebook, sub-notebook and laptop personal computers.

Summary

This proposal provides a set of functions that are accessed through the INT 10h and VESA VBE interfaces. These functions will provide an application with the capabilities of the underlying video controller and flat panel, the current state of these capabilities, and the ability to change this state.

Intellectual Property

Trademarks
All trademarks used in this document are property of their respective owners.

- VESA, VBE and VBE/FP are trademarks of Video Electronics Standards Association.
- IBM and VGA are trademarks of International Business Machines

Patents
VESA proposal and standards documents are adopted by the Video Electronics Standards Association without regard to whether their adoption may involve patents on articles, materials, or processes. Such adoption does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the proposal or standards document.

Support for this Specification

Clarifications and application notes to support this standard will be published as the need arises. To obtain the latest standard and support documentation, contact VESA.
If you have a product which incorporates VBE, you should ask the company that manufactured your product for assistance. If you are a display or controller manufacturer, VESA can assist you with any clarification you may require. All comments or reported errors should be submitted in writing to VESA using one of the following mechanisms:

World Wide Web: www.vesa.org
E-mail:techsupport@vesa.org
Fax: 408-435-8225
Voice: 408-435-0333
Mail to: Video Electronics Standards Association
Suite 440 - 2150 North First Street,
San Jose, CA 95131-2029
USA

Acknowledgments

This document was made possible by the joint efforts of the members of VESA Software Standards Committee. In particular, the following individuals and their companies contributed with time and knowledge.

VBE/FP Work group Leader
Kevin Gillett, S-MOS Systems, Inc.,
Work group Members
Brad Haakenson, Cirrus Logic, Inc.,
George Bystricky, S-MOS Systems, Inc.,
Bipin Biscuitwala, Chips and Technologies
Raluca Iovan, Phoenix Technologies
Mike Calyer, Phoenix Technologies
Rebecca Nolan,Neomagic Corp.
Joseph Casano, Chips and Technologies
Bill Hsu, Cirrus Logic, Inc.,
Ken Fuiks, Texas Instruments

Revision History

0.1p Initial proposal. December 22, 1993
0.11p First revision with revision marks. March 1, 1994
0.12p First revsion. March 1, 1994
0.13p Second revsion proposal. March 10, 1994
0.14p Second revision, March 20, 1994
0.15p Lots of changes May 11, 1994
0.16p Same as .15p without revision marks May 11, 1994
0.17p Added function 0 information, new headers in Functions 0,1,2,3
0.18p Same as .17p without revision marks July 26, 1994
0.19p Major cleanup, added comments to functions, removed function 7, proposed split up of function 04h?. Sept. 13, 1994
0.20p Same as .19p without revision marks.. Sept. 13, 1994
0.22p Changes as a result of Dec '95 meeting:
0.23p Addition of new committee members, removal of Latency in Function 01, addition of descriptions for OffScrnMemXXX in Function 01, improved example in Function 03, Limit descriptions in Functions 04,05, renumbering 0y to 06, 0? to 07, removal of software expansion in Function 07, complete removal of Function 0x. January 31, 1996
0.24p Same as .23p without revision marks. January 31, 1996
0.25p Improved introduction, terms and definitions, subfunction 00h (SupVbeSubFunc description), redefinition of subfunction 03h, clarifications in subfunctions 04/05h, addition of Questions and Answers section, general cleanup. February 5, 1996
0.26p Same as .25p without revision marks and updated TOC. February 5, 1996.
0.27p Wording changes to entire document (clarifications, removal of ambiguities, more definitions, change of register usage in subfunctions 6 and 7, added clarifications, changed subfunction 1 to pass in ES:DI., changed structure slightly in subfunction 0. March 27, 1996
0.28p Same as .27p without revisions and updated TOC. March 27, 1996
0.29p Clarifications, addition of Appendix B, start of work on Appendix A. June 5, 1996
0.30p Same as .29p without revisions and updated TOC. June 5, 1996
0.31p Clarifications and corrections, January 28, 1997
0.32p Same as .31p without revisions and with updated TOC, January 28, 1997

Table of Contents

1. Introduction and Scope

The purpose of this document is to describe the VESA BIOS Extension Subfunctions for Flat Panel Controllers (VBE/FP). These subfunctions are used to supplement the standard INT 10h BIOS functions and give an application developer access to the specialized hardware found in Flat Panel Controllers. The subfunctions allow applications to query the Flat Panel specific hardware and then allow for control of this hardware to improve display quality. VBE/FP is designed as a VBE 2.0 Module (see below).

1.1. VBE 2.0 Module

VBE 2.0 allows for plug in modules into the specification that can be optionally added if required. During the development of the VBE 2.0 specification, Subfunction 11h of the standard VBE function 4Fh was reserved for Flat Panel Controllers. These functions conform to the VBE 2.0 standard for Supplemental Specifications.

1.2. Required Functions

Implementations of VBE/FP must expose all the features supported by the hardware. Subfunction 00h and 01h must be supported as a minimum implementation. Subfunctions not supported by the VBE/FP implementation must return AL != 4Fh. Supported subfunctions must be reflected in the field SupVbeSubFunc (returned by Subfunction 00h) where the appropriate bit field must be set to 1. Subfunctions not supported by the current hardware configuration must return AH=02h.

1.3. Display Switching

Earlier proposals for the VBE/FP standard included a function for display switching. More recently, however, it has become commonplace for graphics controllers to support output to TVs and other devices, requiring a method for display switching which may or may not include flat panels as one of the display options. For this reason, display switching will be addressed in another standard.

1.4. Protected Mode

These subfunctions currently do not include a protected mode interface. Discussions are ongoing as to how to incorporate a protected mode interface and how to support System Management Interrupts (SMI). This functionality may be added in a later version of the specification.

2. Terms and Definitions

2.1. VESA BIOS Extensions (VBE)

VBE is a set of standards published by the Video Electronics Standards Association. VBE is in summary a set of Video BIOS functions (INT 10h) that supplement the standard defined Video BIOS. For more information on the VBE specification contact the VESA office.

2.2. Flat Panel Controller

Any video controller designed to control flat panel displays. These displays can be LCD, Plasma or EL in nature. Typically these controllers have extra features designed to improve the display quality of the panel. These features can include grey scale weighting algorithms, text expansion or a myriad of other features.

2.3. LCD

Liquid Crystal Displays come in several different types.

STN or TN - Super Twisted Nematic or Twisted Nematic panels are passively addressed and require external row and column drivers. They are typically one of two types: dual and single drive. Passive panels are relatively inexpensive in comparison to active panels. Monochrome STN panels dominated the early notebook/laptop designs, but around 1995 color STNs began to dominate the market.

Dual Drive - Dual drive (aka Dual Scan) panels are displays that are logically divided into an upper half and a lower half, such that a line in the upper half and a line in the lower half are simultaneously written over different data lines. That is to say that if a panel was 640x480 pixels, it would actually be made up of two separate panels of 640x240.

Single Drive - Single drive (aka Single Scan) panels are typically made up of one LCD element, i.e. a 640x480 panel is actually one piece of glass covering 640x480 pixels. The advantage to these panels is the price, as there is less circuitry required. The disadvantage to these panels is that they are typically of poorer quality when compared to an equivalent technology dual drive panel. LCD charge begins to fade as the addressing gets closer to the lower portion of the panel, this results in a degradation of contrast.

TFT - Thin Film Transistor LCD panels are actively addressed. The advantage to these panels is that they have an active switching element for each pixel or sub-pixel of the display, to provide electrical isolation and quick charging/discharging of the display elements; this gives the display superior quality due to lack of ghosting and fast refresh time for motion video applications.

2.5. System Management Interrupt (SMI)

System Management Interrupts are the result of CPUs adding power management circuitry. A high priority unmaskable interrupt occurs and SMM code (System Memory Management) swaps normal RAM out and swaps the normally hidden interrupt code in. The ideal use for this is an interrupt request to power down some logic. Until the interrupt occurs, the code to shut down the logic is hidden from normal applications, and when the interrupt occurs, the application code is swapped, and the interrupt code is executed in its place, eventually to be swapped out again.

SMIs can be used to do a variety of tasks, for example, they can be initiated by a function key being pressed to change the state of the inverse/normal circuitry.

2.6. Half Frame Buffers

Because flat panel graphics controllers write data sequentially, and a dual drive panel requires that the two panel halves be updated simultaneously, the half frame buffer is used to store the data for the first half of the panel so that when the controller writes to the second half, the first half can be written out of the buffer at the same time. Typically, the frame buffer memory resides in normal graphics memory and this area cannot be used for any other purpose.

2.7. Inverse Video

Inverse video is simply the result of negating the output to an LCD pixel. The resultant display is a negative image (black on white instead of white on black). Some applications look better with inversion on. Many graphics controller manufacturers allow separate controls the inversion of text and graphics modes.

2.8. NTSC and PAL

The National Television System Committee (NTSC) was formed to deal with the problem of sending color transmission over previously existing black and white television standards, in the predominant North American market. NTSC luminance calculations are the basis for the standard grayscaling algorithms in personal computers. The NTSC color model is called YIQ where Y is luminance, I is an orange-cyan axis and Q is a magenta-green axis. The combination of these factors give a colorrepresentation.that needs phase shifting via a hue control.
PAL used mostly in Europe is a system based on the NTSC standard but with some minor modifications. PAL uses the YUV color model which is similar to YIQ but phase shifted and not requiring a hue control.

2.9. OEM

Original Equipment Manufacturers in the flat panel market are typically system, notebook and laptop vendors. LCD panel vendors and graphics controller vendors typically sell to OEMs, who then package them together to make a product.

2.10. Backlight

Most flat panels are difficult to read without some sort of light filtering through the back of the display medium. For most LCD panels, the source of the light is a one or two fluorescent tube lighting system which shines through the LCD material to give the contrast. These backlights are usually controlled with an on/off level through the graphics controller, however some OEMs allow multiple voltage levels to be set through an external controller, which can be altered by programming an I/O port.

2.11. Contrast Control

Contrast control on LCD panels are typically done two ways, by changing the frame rate on STN panels, or by modifying the voltage levels to the biasing circuitry on the panel. Most graphics controllers do not have a method of varying this voltage, however many OEMs have implemented a software programmable interface to the voltage biasing circuitry.

2.12. CRT

A Cathode Ray Tube is the main display medium for televisions and computer monitors. The CRT is a vacuum tube with an electron gun which fires electrons at phosphor particles which when excited by the electron charge, glow. Color CRTs have 3 electron guns (one for each base color Red, Green and Blue) firing at phosphors which glow in color.

2.13. Simultaneous Display

Simultaneous Display is the term used to describe primary and secondary display devices displaying the same data simultaneously from a single controller. The display devices can be multiple CRT, flat panel, or TV devices, or a mixture of these devices.

3. Flat Panel Interface Functions

This chapter describes in detail each of the functions defined by the VBE/FP standard. VBE/FP functions are called using the INT 10h interrupt vector, passing arguments in the 80x86 registers. The INT 10h interrupt handler first determines if a VBE/FP function has been requested, and if so, processes that request; otherwise control is passed down the chain to the next INT 10h handler.

All VBE/FP functions are called with the AH register set to 4Fh and the AL register set to 11h. The AH register is used to determine that it is a VBE call, the AL register is used to determine that the Subfunction is a Flat Panel Extension.

For future revisions of the specification, VBE/FP assumes that other registers in a function call will be destroyed; the exception to this is SI,BP,DS,SS for obvious reasons and ES:DI when not used to return information. This assumption allows for the use of other currently undefined registers in eachfunction,if necessary.

Reserved values should always be set to the value zero (0).

3.1 VBE/FP Completion Codes

The AX register is used to indicate the completion status upon return from VBE functions. If VBE support for the specified function is available, the 4Fh value passed in the AH register on entry is returned in the AL register. If the VBE function completed successfully, 00h is returned in the AH register. Otherwise the AH register is set to indicate the nature of the failure.

VBE RETURN STATUS

AL == 4Fh:	Function is supported
AL != 4Fh:	Function is not supported
AH == 00h:	Function call successful
AH == 01h:	Function call failed
AH == 02h:	Function is not supported in the current hardware configuration
AH == 03h:	Function call invalid in current video mode

Note: Applications should treat any non-zero value in the AH register as a general failure condition as later versions of the VBE may define additional error codes.

3.2. Subfunction 00h - Return Flat Panel Extensions Support Information

This function is designed to allow application developers to get information about the Flat Panel Extensions. This function is required by VBE 2.0 as a Supplemental Specification. For more details on the decisions for this function, please refer to the VESA BIOS Extensions Standard version 2.0 Chapter 5.

Input:	AX	=	4F11h	VBE Flat Panel Extension request
	BL	=	00h	Return Flat Panel Extensions Support Information
	ES:DI	=		Pointer to 256 byte buffer in which to place SupVbeInfoBlock structure
Output:	AX	=		VBE Return Status

(Note: Currently undefined registers may be destroyed with the exception of SI,BP,DS and SS)

The information block has the following structure:

```
SupVbeInfoBlock struc
SupVbeSignature     db      'VBE/FP',0FFh       ; Supplemental VBE Signature
SupVbeVersion       dw      0100h               ; Supplemental VBE Version
SupVbeSubFunc       db      03h,7 dup (0)       ; Bitfield of supported subfunctions
OemSoftwareRev      dw      ?                   ; OEM Software revision
OemVendorNamePtr    dd      ?                   ; Pointer to Vendor Name String
OemProductNamePtr   dd      ?                   ; Pointer to Product Name String
OemProductRevPtr    dd      ?                   ; Pointer to Product Revision String
OemStringPtr        dd      ?                   ; Pointer to OEM String
Reserved            db      221 dup (?)         ; Reserved for description strings and future
                                                ;       expansion
SupVbeInfoBlock ends
```

Note: All data in this structure is subject to change by the VBE implementation when any VBE Subfunction 00h is called. Therefore it should not be used by the application to store data of any kind; if the information is required at a later time, a local copy of the data should be made.

Description of the **SupVbeInfoBlock** structure fields:

The **SupVbeSignature** field is filled with the ASCII characters 'VBE/FP'. The 7th character must be filled with FFh.

The **SupVbeVersion** is a BCD value which specifies what version of VBE/FP is implemented in the software. The higher byte specifies the major version number. The lower byte specifies the minor version number. The current value is: 0100h.

The **SupVbeSubFunc** is a bitfield that represents the subfunctions available in this implementation of VBE/FP. If the bit representing a particular subfunction is set to 1, then that subfunction is supported. Subfunction '00h' is represented by the LSB of the first byte, bit 1 corresponds to subfunction 01h and the other subfunctions follow suit. The LSB of byte 2 corresponds to subfunction 09h etc.. Only bits for subfunctions defined in the specification can be set to 1, other bits should be set to 0. The required function set for VBE/FP is 0000 0011b, the other bits must be set to 1 if the corresponding subfunctions are implemented. In addition, any subfunctions not supported should turn the corresponding bits to 0 and must return AL!=4Fh as a completion code.

The **OemSoftwareRev** field is a BCD value which specifies the OEM revision level of VBE/FP. The higher byte specifies the major version number. The lower byte specifies the minor version number. This field can be used to identify the OEM's software release, so that upgrades can be identified.

The **OemVendorNamePtr** is a Real Mode far pointer to a null-terminated OEM-defined string containing the name of the vendor who produced the display controller board product. Other supplemental functions may point to the same location.

The **OemProductNamePtr** is a Real Mode far pointer to a null-terminated OEM-defined string containing the product name of the display controller board. Other supplemental functions may point to the same location.

The **OemProductRevPtr** is a Real Mode far pointer to a null-terminated OEM-defined string containing the revision or manufacturing level of the display controller board product. This field can be used to determine which production revision of the display controller board is installed. Other supplemental functions may point to the same location.

The **OemStringPtr** is a Real Mode far pointer to a null-terminated OEM-defined string. This string may used to identify the graphics controller chip or OEM product family for hardware specific display drivers. There are no restrictions on the format of the string. This pointer may point into either ROM or RAM, depending on the specific implementation. Other supplemental functions may point to the same location.

3.3. Subfunction 01h - Return Flat Panel Information

This subfunction allows application software to determine information about the currently attached flat panel.

Input:	AX	=	4F11h	VBE Flat Panel Extension request
	BL	=	01h	Return Flat Panel Information
	ES:DI	=		Pointer to 32 byte buffer in which to place FPInfo structure
Output:	AX	=		VBE Return Status

(Note: Currently undefined registers may be destroyed with the exception of SI,BP,DS and SS)

```
FPInfo struc
HSize                   dw      ?       ; Horizontal Size in Pixels
VSize                   dw      ?       ; Vertical Size in Lines
FPType                  dw      ?       ; Technology (TSTN LCD, TFT LCD,
                                        ; Plasma, EL), Single/Dual, Color/Mono
RedBPP                  db      ?       ; Red Bits Per Primary
GreenBPP                db      ?       ; Green Bits Per Primary
BlueBPP                 db      ?       ; Blue Bits Per Primary
ReservedBPP             db      0       ; Reserved Bits Per Primary
RsvdOffScrnMemSize      dd      ?       ; Size in KB of Offscreen Memory required
                                        ; frame buffer
RsvdOffScrnMemPtr       dd      ?       ; Pointer to reserved offscreen memory
Reserved                db      14 dup(0)       ; remainder of FPInfo
FPInfo ends
```

The **HSize** field is an integer value representing the horizontal size of the flat panel in pixels. A typical 640x480 flat panel will have an HSize = 640 (280h).

The **VSize** field is an integer value representing the vertical size of the flat panel in pixels. A typical 640x480 flat panel will have an VSize = 480 (1E0h).

The **FPType** describes the type of the flat panel as follows:

D0	= 0 Monochrome
	= 1 Color
D1	= 0 Single panel construction
	= 1 Dual (Split) panel construction
D2-D7	= 0000 00 STN (Passive Matrix)
	= 0000 01 TFT (Active Matrix)
	= 0000 10 Other LCD
	= 0000 11 EL

= 0001 00 Plasma
D8-D15 = Reserved

The **RedBPP**, **GreenBPP**, and **BlueBPP** represent the color (mono) capabilites of the panel. The number of Bits Per Primary (BPP) is a description of the color depth supported by the panel. The GreenBPP field should be used for monochrome panels such that an 8-bit per pixel monochrome panel should be described as RedBPP:GreenBPP:BlueBPP = 0:8:0. Note that this field describes the capability of the panel and not the color palette or frame rate control capabilities of the graphics controller.

The **ReservedBPP** is a reserved field indicating BitsPerPrimary in the event that color depths increase beyond 24-bit. This is the same method that VBE 1.2 uses to reserve extra information for color depth. This field shall be reserved as 0 until needed.

The Rsvd **OffScrnMemSize** is the size of the unavailable Off Screen Memory that is required for flat panel related issues such as dual drive panel frame buffers. This size should be subtracted from the Total Memory installed that is reported by VBE Core functions.

The Rsvd **OffScrnMemPtr** is a pointer to the location of the unavailable memory used for the full or half frame buffers.

The remainder of the FPInfo structure is **Reserved** for future use.

3.4. Subfunction 02h - Return/Select Inverse Video

This subfunction allows an application to check/set the current state of the screen inverse. To best describe this, screen inversion is the ability to display black text/graphics on a white background, normal is white text/graphics on a black background.

Input:	AX	=	4F11h	VBE Flat Panel Extension request
	BL	=	02h	Return/Select Screen Polarity
	BH	=	00h	Return request
Output:	AX	=		VBE Return Status
	BL	=		Current Polarity State
				B0=Text modes inverted
				B1=Graphics modes inverted
				B2-B7=Reserved, must be 0
	BH	=		Available Polarity Settings
				B0=Text inverse available
				B1=Graphics inverse available
				B2=Text and Graphics inverse must be the same
				B3-B7 Reserved, must be 0

Input:	AX	=	4F11h	VBE Flat Panel Extension request
	BL	=	02h	Return/Select Screen Polarity
	BH	=	01h	Select request
	CL	=		Active Polarity to set
				B0=Invert text modes
				B1=Invert graphics modes
				B2-B7=Reserved
Output:	AX	=		VBE Return Status

(Note: Currently undefined registers may be destroyed with the exception of SI,BP,DS,SS,ES and DI)

Note: If B2 of "available settings" is set to 1 indicating that text and graphics must be set the same and this is not done on the "set" command, the function will turn inverse "off". The application must test the available settings prior to using "set" command and must set the bits 0 and 1 accordingly.

3.5. Subfunction 03h - Return/Select Flat Panel Shading Options

This subfunction allows an end user the ability to select between a number of different shading options supported by the current video controller. This becomes most useful when displaying color modes on a monochrome flat panel. Examples of such shading may be: NTSC; green gun only; or various combinations of R,G,B weighting. The above shading examples are not required by this specification, this subfunction simply supplies the end user a means of selecting from the OEM supplied options.

An application that may use this function could be a control panel which after querying VBE/FP for the number of options, gives the user a cyclic method to select one of the different gray scale methods. Because gray scale methods are subjective and different OEMs will supply different options, the user would only need to know which option they prefer for their implementation. To keep this function as flexible as possible it was decided that assigning specific names to options should not be done. (This allows OEM1 to supply different options from OEM2.)

Input:	AX	=	4F11h	VBE Flat Panel Extension request
	BL	=	03h	Return/Select Shading Options
	BH	=	00h	Get number of shading options request
Output:	AX	=		VBE Return Status
	CL	=		Number of shading options
	CH	=		Current shading option
Input:	AX	=	4F11h	VBE Flat Panel Extension request
	BL	=	03h	Return/Select Shading Options
	BH	=	01h	Select option request
	CH	=		Shading option number to set
Output:	AX	=		VBE Return Status

(Note: Currently undefined registers may be destroyed with the exception of SI,BP,DS,SS,ES and DI)

Note: This subfunction may only be supported on monochrome panels

3.6. Subfunction 04h - Return/Select Flat Panel Contrast

This subfunction allows an application the ability to select flat panel contrast levels. This function should be implemented in the event that the OEM has provided a software programmable interface for adjusting the voltage to the biasing circuitry on the panel (which in effect changes the contrast).This is not a frame rate control. Frame rate adjustments will be dealt with in other VESA standards.

Input:	AX	=	4F11h	VBE Flat Panel Extension request
	BL	=	04h	Return/Select Flat Panel Contrast
	BH	=	00h	Return Range Request
Output:	AX	=		VBE Return Status
	CH	=		Upper Limit
	CL	=		Current Flat Panel Contrast

Input:	AX	=	4F11h	VBE Flat Panel Extension request
	BL	=	04h	Return/Select Flat Panel Contrast
	BH	=	01h	Select Request
	CL	=		Flat Panel Contrast to Set
Output:	AX	=		VBE Return Status

(Note: Currently undefined registers may be destroyed with the exception of SI,BP,DS,SS,ES and DI)

The Lower Limit value is always zero (0). An application can assume that if the Upper Limit is returned as 7, that there are 8 options (0-7). If the VBE/FP implementation returns a value of 1, there are 2 options (0-1). If this function is supported, the Upper Limit must be >= 1.

A call with a number out of the range will result in the failure of this call with an error code of AH=01h.

3.7. Subfunction 05h - Return/Select Flat Panel Brightness

This subfunction allows an application the ability to select flat panel brightness levels. This function should be implemented in the event that the OEM offers a software programmable interface for adjusting the voltage to the backlight.

Input:	AX	=	4F11h	VBE Flat Panel Extension request
	BL	=	05h	Return/Select Flat Panel Brightness
	BH	=	00h	Return Range Request
Output:	AX	=		VBE Return Status
	CH	=		Upper Limit
	CL	=		Current Flat Panel Brightness

Input:	AX	=	4F11h	VBE Flat Panel Extension request
	BL	=	05h	Return/Select Flat Panel Brightness
	BH	=	01h	Select Request
	CL	=		Flat Panel Brightness to Set
Output:	AX	=		VBE Return Status

(Note: Currently undefined registers may be destroyed with the exception of SI,BP,DS,SS,ES and DI)

Note: 0 = turn backlight OFF
maximum value = turn backlight on full

The Lower Limit value is always zero (0). An application can assume that if the Upper Limit is returned as 7, that there are 8 options (0-7). If the VBE/FP implementation returns a value of 1, there are 2 options (0-1). Most flat panel controllers offer at least two levels of programming, on and off, however, the OEM may have external circuitry to vary the backlight voltage to more than these two extreme levels. If this function is supported the Upper Limit must be >= 1.

A call with a number out of the range will result in the failure of this call with an error code of AH=01h.

3.8. Subfunction 06h - Return/Select Vertical and Horizontal Positioning

This subfunction allows an application the ability to position the displayed portion of a mode. The need for this subfunction arises from the fact that today's flat panel devices are of fixed dimensions, and the displayed portions of some modes are smaller than these fixed dimensions. With the ability to position the displayed portion of a mode, an application can produce a better look.

This subfunction represents a global hardware setting. Other factors within the current operation of the controller may limit whether or not this setting actually occurs (e.g. a controller may not allow centering of a 350 line image but will center a 400 line image. The result is that the hardware request for centering is actually set, however its effectiveness may be mode dependent).

VBE/FP Implementations of this function should return actually hardware values that have been set, not mode dependent information.

Input:	AX	=	4F11h	VESA VBE Flat Panel Extension request
	BL	=	06h	Return/Select Vertical and Horizontal Positioning
	BH	=	00h	Return request
Output:	AX	=		VESA VBE Return Status
	BL	=		Available Horizontal Position Settings B0=Left B1=Center B2=Right B3-B7=Reserved, must be 0
	BH	=		Available Vertical Position Settings B0=Top B1=Center B2=Bottom B3-B7=Reserved, must be 0
	CL	=		Current Horizontal Position 0=Left 1=Center 2=Right all other values are reserved
	CH	=		Current Vertical Position 0=Top 1=Center 2=Bottom all other values are reserved

Input:	AX	=	4F11h	VESA VBE Flat Panel Extension request
	BL	=	06h	Return/Select Vertical and Horizontal Positioning

BH	=	01h	Select request
CH	=		Vertical Position to set 0=Top 1=Center 2=Bottom all other values are reserved
CL	=		Horizontal Position to set 0=Left 1=Center 2=Right all other values are reserved

Output: AX = VESA VBE Return Status

(Note: Currently undefined registers may be destroyed with the exception of SI,BP,DS,SS,ES and DI)

Note: If Vertical Positioning is not available, Current Vertical Position shall return zero (BL=00h), Available Vertical Position Settings shall return zero (BH=00h), and any attempt to set a position shall be ignored. If Horizontal Positioning is not available, Current Horizontal Position shall return zero (CL=00h), Available Horizontal Position Settings shall return zero (CH=00h), and any attempt to set a position shall be ignored.

3.9 Subfunction 07h - Return/Select Vertical and Horizontal Expansion

This subfunction allows an application the ability to expand the displayed portion of a mode. The need for this subfunction arises from the fact that today's flat panel devices are of fixed dimensions, and the displayed portions of some modes are smaller than these fixed dimensions. With the ability to expand the displayed portion of a mode, an application can produce a better look.

This subfunction represents a global hardware setting. Other factors within the current operation of the controller may limit whether or not this setting actually occurs (e.g. a controller may not allow expansion of a 14 point font, however will expand a 16 point font. The result is that the hardware request for expansion is actually set, however its effectiveness may be mode dependent).

VBE/FP Implementations of this function should return actually hardware values that have been set, not mode dependent information.

Input:	AX	=	4F11h	VESA VBE Flat Panel Extension request
	BL	=	07h	Return/Select Vertical and Horizontal Expansion
	BH	=	00h	Return request
Output:	AX	=		VESA VBE Return Status
	BL	=		Available Horizontal Expansion Settings B0=Text Expansion 0 = Not Available, 1= Available B1=Graphics Expansion 0 = Not Available, 1= Available B2-B7=Reserved, must be 0
	BH	=		Available Vertical Expansion Settings B0=Text Expansion 0 = Not Available, 1= Available B1=Graphics Expansion 0 = Not Available, 1= Available B2-B7=Reserved, must be 0
	CL	=		Current Horizontal Expansion B0=Text Expansion 0 = Disabled, 1 = Enabled B1=Graphics Expansion 0 = Disabled, 1 = Enabled B2-B7=Reserved, must be 0
	CH	=		Current Vertical Expansion B0=Text Expansion 0 = Disabled, 1 = Enabled B1=Graphics Expansion 0 = Disabled, 1 = Enabled

B2-B7=Reserved, must be 0

Input:	AX	=	4F11h	VESA VBE Flat Panel Extension request
	BL	=	07h	Return/Select Vertical and Horizontal Expansion
	BH	=	01h	Select request
	CH	=		Vertical Expansion
				B0=Text Expansion
				0 = Disabled, 1 = Enabled
				B1=Graphics Expansion
				0 = Disabled, 1 = Enabled
				B2-B7=Reserved, must be 0
	CL	=		Horizontal Expansion
				B0=Text Expansion
				0 = Disabled, 1 = Enabled
				B1=Graphics Expansion
				0 = Disabled, 1 = Enabled
				B2-B7=Reserved, must be 0
Output:	AX	=		VBE Return Status

(Note: Currently undefined registers may be destroyed with the exception of SI,BP,DS,SS,ES and DI)

Note: If Vertical Expansion is not available, Current Vertical Expansion shall return zero (BL=00h), Available Vertical Expansion Settings shall return zero (BH=00h), and any attempt to set vertical expansion shall be ignored. If Horizontal Expansion is not available, Current Horizontal Expansion shall return zero (BL=00h), Available Horizontal Expansion Settings shall return zero (BH=00h), and any attempt to set horizontal expansion shall be ignored.

Appendix A: BIOS Implementation Notes

The goal for VBE/FP is to give application developers the ability to at least query the flat panel environment that exists. As a result, the minimum implementation required, is to implement subfunctions 0 and 1. All other functions are optional. It is a requirement that the SupVbeSubFunc of the SupVBEInfoBlock represent all supported functions correctly.

For example, if subfunctions 0,1,3 and 4 are supported an no others, SubVbeSubFunc will contain the values 00011011, 00000000, 00000000, 000000000, 00000000, 00000000, 00000000, 00000000.

All non supported subfunctions will return AL!=4Fh.

Appendix B: Application Development Notes

Subfunction 00h
All data in the SupVbeInfoBlock structure, including the reserved bytes, is subject to change by the VBE/FP BIOS when this subfunction is called. It should never be used by an application to store data of any kind. If the buffer is going to be reused, any data that will be needed later needs to be copied elsewhere or subfunction 00h needs to be called again.

Subfunction 01h
This subfunction details the physical capabilities of the panel, but does not necessarily reflect the capabilities of the graphics controller that is driving the panel. Most controllers use dithering and scaling to provide the appearance of different resolutions and palette sizes. To obtain this information, programs need to use the interfaces provided by the VESA VBE Core Functions or the VBE/AF specifications.
The RsvdOffScrnMemPtr and RsvdOffscrnMemSize fields define an area of memory that should never be changed by application programs. This memory can be used directly by the graphics controller for a number of purposes. The most common use is to contain half a frame buffer for computers that use a dual/split STN panel (see the FPType field in subfunction 01h). For this type of panel, the graphics controller generally uses some of the off screen memory to hold an image of one-half of the panel as it is drawn, so that both halves can then be written simultaneously. While this memory is only needed when the panel is active, it should never be used by application programs. Most notebook computers have hot keys that will switch between displays at the user's discretion. If an application is started on the CRT and so decides that it can use the reserved memory, everything may work fine (if the controller is not using the memory for something else) until the user hits the button to switch to the panel. At that point, results will range from a corrupted display to the program crashing. For these reasons (and many others) stay away from the reserved memory of the graphics controller even if VBE 2.0 says that the memory is available.

Subfunction 02h
This subfunction is primarily intended for computers with monochrome panels. On color panels, reversing graphics is generally not desirable and so the graphics controller and/or the VBE/FP BIOS may not implement this function.
If bit 2 of the available settings is set to 1, text and graphics must be set to the same state. If this is not done properly on a set command, the function will turn inverse 'off'. The application must test the available settings prior to using the 'set' command and must set bits 0 and 1 accordingly.

Subfunction 03h
This subfunction is primarily intended for computers with monochrome panels.
It would be almost impossible to define a call that could define all of the different shading adjustment options that are available on different flat panel graphics controllers. For this reason, this function defines a generic interface that only specifies the number of available shading options and provides an interface to set each one. The application program that uses this call does not need to know what the different options do, or what they are called. It should just provide an interface to select method one, method two, etc.

Subfunction 04h
This subfunction will be implemented only if the computer manufacturer has provided a software programmable interface for adjusting the voltage to the biasing circuitry on the panel.
The lower limit value is always zero (0). An application can assume that if the upper limit is returned as 7, that there are 8 options (0 - 7). If the VBE/FP BIOS returns a value of 1, there are 2 options (0 - 1). If this function is supported, the upper limit will always be greater than or equal to one.
A call with an out of range number in CL will result in the failure of this call with an error code of AH=01h.

Subfunction 05h
This subfunction will be implemented only if the computer manufacturer has provided a software programmable interface for adjusting the voltage level for the backlight.
The lower limit is always zero (0). An application can assume that if the upper limit is returned as 7, that there are 8 levels (0 - 7). If the VBE/FP BIOS returns a value of 1, there are 2 options (0 - 1). Most flat panel graphics controller offer at least two levels of programming, on and off. However, the computer manufacturer may have external circuitry to vary the backlight voltage within this range. If this function is supported, the upper limit will always be greater than or equal to one.
A call with an out of range number in CL will result in the failure of this call with an error code of AH=01h.

Subfunction 06h
This subfunction sets and returns the global hardware settings for the flat panel graphics controller. Since the actual display capabilities will vary from mode to mode, these settings will not always reflect what is visible on the flat panel. For instance some graphics controllers do not allow centering on a 350 line mode, but will center a 400 line mode. If centering is enabled when in the 350 line mode, the hardware is set, but the change will not be visible until a mode is selected that can take advantage of the configuration.
The state returned by this subfunction will always reflect the global hardware settings, not the mode-dependent information.

If vertical positioning is not available, the returned Current Vertical Position field will contain a zero (CH = 00h), Available Vertical Position Settings will contain zero (BH=00h), and any attempt to set vertical position will be ignored. Horizontal positioning will behave in the same manner.

Subfunction 07h

This subfunction sets and returns the global hardware settings for the flat panel graphics controller. Since the actual display capabilities will vary from mode to mode, these settings will not always reflect what is visible on the flat panel. For instance a graphics controller might not allow expansion of a 14 pixel font, but will expand a 16 pixel high font. If expansion is enabled while using a video mode with a 14 pixel font, the hardware is set, but the change will not be visible until a mode is selected that can take advantage of the configuration.

The state returned by this subfunction will always reflect the global hardware settings, not the mode-dependent information.

If vertical expansion is not available, the returned Current Vertical Expansion field will contain a zero (CH = 00h), Available Vertical Expansion Settings will contain zero (BH=00h), and any attempt to set vertical expansion will be ignored. Horizontal expansion will behave in the same manner.

Appendix C: Related Documents

VESA BIOS Extensions (VBE) Core Functions Standard version 2.0

Appendix D: Questions and Answers

This appendix documents some of the major decisions and tradeoffs that were made by the VESA Software Standards Committee in developing this standard.

Q. Why did VESA develop the VBE/FP Standard?

A. VESA developed VBE/FP to allow application developers and system manufacturer's ways to simply access some of the most common enhancements in the flat panel controllers on the market today.

System manufacturers can now write a common interface to their applets (applications) which they ship with their systems; and application developers who make laptop/notebook enhancement utilities, or diagnostic programs can now inquire as to the capabilities of the given implementation, and potentially take advantage of them

Q. Why is there no display switching within this standard?

A. VBE/FP was originally intended to include display switching capabilities (i.e. switch from LCD to CRT or to simultaneous display), but because of the advances in the graphics controller designs and the abilities to incorporate other display devices such as NTSC output, it was decided to remove the display switching capabilities to another subfunction group. At this time, discussions of using subfunctions AL=12h or adding to VBE/DDC are ongoing.

Q. Why is power management not part of the standard?

A. VBE/PM has been designed to handle the power management requirements of flat panel controllers as well as for standard graphics output devices, and it was deemed redundant to duplicate those functions within VBE/FP.