WIKIPEDIA

VESA BIOS Extensions

VESA BIOS Extensions (VBE) is a VESA standard, currently at version 3, that defines the interface that can be used by software to access compliant video boards at high resolutions and bit depths. This is opposed to the "traditional" int 10h BIOS calls, which are limited to resolutions of 640×480 pixels with 16 color (4-bit) depth or less. VBE is made available through the video card's BIOS, which installs during boot up some interrupt vectors that point to itself.

Most newer cards implement the more capable VBE 3.0 standard. Older versions of VBE provide only a real mode interface, which cannot be used without a significant performance penalty from within protected mode operating systems. Consequently, the VBE standard has almost never been used for writing a video card's drivers; each vendor has thus had to invent a proprietary protocol for communicating with its own video card. Despite this, it is common that a driver thunk out to the real mode interrupt in order to initialize screen modes and gain direct access to a card's linear frame buffer, because these tasks would otherwise require handling many hundreds of proprietary variations that exist from card to card.

Contents

Standards

VESA BIOS Extensions (VBE core) 2.0 [November 1994]

This standard provides the primary functionality of the VESA BIOS Extensions. It allows applications to determine the capabilities of the graphics card and provides the ability to set the display modes that are found. VBE 2.0 adds some new features above the prior VBE 1.2 standard including linear framebuffer access and protected mode banking. Some of the VBE Core 2.0 features include:

Linear framebuffer access
: Enables direct framebuffer access in protected mode as one large area of memory instead of less efficient smaller chunks.

Protected mode banking
: Allows access to the framebuffer from protected mode without "thunking" down to real mode.

Super VGA page flipping
: Allows higher performance animation to provide for smooth animation for computer games and other high performance graphics programs.

Super VGA virtual screens
: Allows software to set up virtual display resolutions, larger than the actual displayed resolution, and smoothly scroll or pan around the larger image.

High Color and TrueColor modes
: Industry standard 16-bit and 24-bit graphics modes for resolutions from 320×200 up to 1,600×1,200.

VESA BIOS Extensions (VBE core) 3.0 [September 1998]

A superset of the VBE 2.0 standard. This standard adds refresh rate control, facilities for stereo glasses, improved multi-buffering and other functions to the VBE 2.0 standard.

Triple buffering
: Allows high speed applications to perform multi-buffering with less screen flickering and without having to wait for the graphics controller.

Refresh rate control using GTF timings
: This allows applications and operating system utilities to change the refresh rate in a standard way on all VBE 3.0 graphics controllers. Important for stereo applications, since when stereo is enabled, the user's effective refresh rate is cut in half.

Stereo page flipping
: When viewing an application using stereo glasses, software needs to page flip twice as often as normal, because it needs to generate separate images for each eye. This new feature allows stereo compatible software to display properly.

Hardware stereo sync
: Allows stereo software to determine if there is a connector for stereo glasses on the user's graphics card.

VBE/accelerator functions (VBE/AF) [August 1996]

VBE/AF provides a low-level, standard interface to common acceleration functions available on most hardware. Some of the functions defined in the standard are access to hardware cursors, Bit Block Transfers (Bit Blt), off screen sprites, hardware panning, drawing and other functions.

Supplemental specifications

Supplemental specifications provides device independent interface between application software and Super VGA hardware. Function numbers are assigned by VESA Software Standards Committee (SSC).

Power management extensions (PM)

DPMS is a hardware standard that allows graphics cards to communicate with DPMS-compliant monitors via a special signaling system that can be used with existing graphics controllers and monitor cables. This signaling system allows the graphics card to tell the monitor to go into a number of different power management or power saving states, which effectively allow the monitor to turn itself off when it is not in use.

Flat panel interface extensions (FP)

Allows access to special features in flat panel controllers.

Audio interface extensions (AI)

Provides standard to audio services.

Currently (version 1.00), the VBE/AI specification defines three device classes: WAVE, MIDI, and VOLUME. Device types not covered:

CDROM control
: which is covered by the Microsoft's CD-ROM Extensions.

Effects processors
: This class of device will be expanded in future version of the VBE/AI specification.

OEM extensions

Provides standard entry to vendor specific extensions.

Display Data Channel (DDC)

The **Display Data Channel** or **DDC** is a digital connection between a computer display and a graphics adapter that allows the display to communicate its specifications to the adapter. The standard was created by VESA.

Serial Control Interface (SCI)

Provides hardware independent means for operating system and application to read and write data over I²C serial control interface.

VBE mode numbers

Although mode number is a 16-bit value, the optional VBE mode numbers are 14 bits wide. Bit 15 is used by VGA BIOS as a flag to clear or preserve display memory. VBE defined mode numbers as follows:

Bit	Meaning
0–8	Mode numbers. If bit 8 is 1, it is a VESA defined VBE mode.
9-10	Reserved for expansion. Must be set to 0.
11	Refresh rate control Select. If set to 1, use user specified CRTC values for refresh rate, otherwise use BIOS default refresh rate.
12–13	Reserved for VBE/AF. Must be set to 0.
14	Linear/Flat Frame Buffer Select. If set to 1, use linear frame buffer, otherwise use banked frame buffer.
15	Preserve Display Memory Select. If set to 1, preserve display memory, otherwise clear display memory.

Starting in VBE/Core 2.0, VESA no longer defines new VESA mode numbers and no longer requires a device to implement the old numbers. To properly detect information of a screen mode, use *Function 01h - Return VBE Mode Information*.

Mode 81FFh is a special video mode designed to preserve current memory contents and give access to the entire video memory.

Modes defined by VESA

Beginning with the VBE 2.0 standard, no new modes will be defined by VESA, and old modes are no longer mandatory. The use of defined modes should be considered deprecated: modern video cards may or may not use these mode numbers (even though most do for backward compatibility), and modern software should not use them. The correct way for software to discover available display modes is to obtain a list of modes (using "Function 00h - Return VBE Controller Information") and then to check each mode (using "Function 01h: Return VBE Mode Information") until it finds the mode/s it requires.

Graphics modes	320×200	640×400	640×480	800×600	1024×768	1280×1024
16-color palette				258 (0102h), 106 (6Ah)	260 (0104h)	262 (0106h)
256-color palette		256 (0100h)	257 (0101h)	259 (0103h)	261 (0105h)	263 (0107h)
15-bit (5:5:5)	269 (010Dh)		272 (0110h)	275 (0113h)	278 (0116h)	281 (0119h)
16-bit (5:6:5)	270 (010Eh)		273 (0111h)	276 (0114h)	279 (0117h)	282 (011Ah)
24-bit (8:8:8)	271 (010Fh)		274 (0112h)	277 (0115h)	280 (0118h)	283 (011Bh)

Modes 264–268 are text modes. 264 (0108h) is 80 columns × 60 rows (80×60), 265 (0109h) is 132×25, 266 (010Ah) is 132×43, 267 (010Bh) is 132×50 and 268 (010Ch) is 132×60.

Text modes	Columns	
Rows	**80**	**132**
25		265 (0109h)
43		266 (010Ah)
50		267 (010Bh)
60	264 (0108h)	268 (010Ch)

Other commonly available graphics modes

The table below combines the modes defined by VESA (the values denoted in black) along with modes commonly used, but which may not work on all graphics cards as they are not defined by any standard (denoted in red).

Graphics modes	320×200	640×400	640×480	800×500	800×600	896×672	1,024×640	1,024×768	1,152×720	1,280×1,024	1,440×900	1,600×1,200
16-color palette	[1]		[1]		258 (0102h), 106 (6Ah)			260 (0104h)		262 (0106h)		
256-color palette	[1]	256 (0100h)	257 (0101h)	367 (016Fh)	259 (0103h)	303 (012Fh)	362 (016Ah)	261 (0105h)	357 (0165h)	263 (0107h)	352 (0160h) [2]	284 (011Ch)
15-bit (5:5:5)	269 (010Dh)	289 (0121h)	272 (0110h)	368 (0170h)	275 (0113h)	304 (0130h)	363 (016Bh)	278 (0116h)	358 (0166h)	281 (0119h)	353 (0161h)	285 (011Dh)
16-bit (5:6:5)	270 (010Eh)	290 (0122h)	273 (0111h)	369 (0171h)	276 (0114h)	305 (0131h)	364 (016Ch)	279 (0117h)	359 (0167h)	282 (011Ah)	354 (0162h)	286 (011Eh)
24-bit (8:8:8)	271 (010Fh)	291 (0123h)	274 (0112h)	370 (0172h)	277 (0115h)	306 (0132h)	365 (016Dh)	280 (0118h)	360 (0168h)	283 (011Bh)	355 (0163h)	287 (011Fh)
32-bit (8:8:8) [3]		292 (0124h)	297 (0129h)	371 (0173h)	302 (012Eh)	307 (0133h)	366 (016Eh)	312 (0138h)	361 (0169h)	317 (013Dh)	356 (0164h)	322 (0142h) [4]

1. Modes available via the traditional 10h BIOS call

- 352 (0160h) also appears to select 1,280×800 (8-bit) for various laptops' displays
- 32-bit is really (8:8:8:8), but the final 8-bit number is an "empty" alpha channel. It is otherwise equal to 24-bit color. Many GPUs use 32-bit color mode instead of 24-bit mode merely for faster video memory access through 32-bit memory alignment.
- 322 (0142h) is 1,400×1,050

Linux video mode numbers

The Linux kernel allows the user to select the VESA mode at boot time by passing a code in memory to the kernel. The LILO boot loader passes this code based on a "vga" parameter in its configuration file. It takes the form 'vga=XXX', where XXX is the decimal value, or 'vga=0xHHH', where HHH is the hexadecimal value. However, the 'vga=' boot loader parameter does not directly accept VESA video mode numbers; rather, the Linux video mode number is the VESA number plus 512 (in the case of the decimal representation) or plus 0x200 (in the case of the hexadecimal representation). For example, the defined VESA value of 257 (0x101), representing 640x480 and 256 colours, has an equivalent Linux video mode value of 769 (0x301).

See the kernel documentation in Documentation/svga.txt (https://www.kernel.org/doc/Documentation/svga.txt) and Documentation/fb/vesafb.txt (https://www.kernel.org/doc/Documentation/fb/vesafb.txt).

As indicated earlier, the VESA standard defines a limited set of modes; in particular, none above 1,280×1,024 are covered and, instead, their implementation is completely optional for graphics adapter manufacturers. As vendors are free to utilize whatever additional values they please, this means that, in the table below, the modes denoted in red (and expressed in decimal form) **may not apply to your graphics adapter**[1]

	320 × 200	640 × 400	640 × 480	800 × 500	800 × 600	896 × 672	1,024 × 640	1,024 × 768	1,152 × 720	1,280 × 1,024	1,400 × 1,050	1,440 × 900	1,600 × 1,200	1,920 × 1,200
16 colors					770			772		774				
256 colors		768	769	879	771	815	874	773	869	775	835	864	796	893
15-bit (5:5:5)	781	801	784	880	787	816	875	790	870	793		865	797	
16-bit (5:6:5)	782	802	785	881	788	817	876	791	871	794	837	866	798	
24-bit (8:8:8)	783	803	786	882	789	818	877	792	872	795	838	867	799	
32-bit (8:8:8)[1]		804	809	883	814	819	878	824	873	829		868	834	

1: 32-bit is really (8:8:8:8), but the final 8-bit number is an "empty" alpha channel. It is otherwise equal to 24-bit color. Many GPUs use 32-bit color mode instead of 24-bit mode merely for faster video memory access through 32-bit memory alignment.

vga=864 [352 (0160h)] also appears to select 1,280×800 (8-bit) for various laptops' displays.
vga=834 [322 (0142h)] is 1,400×1,050

Alternative method

hwinfo is the hardware detection tool used in SuSE Linux and in some other GNU/Linux distributions.[1] To use hwinfo to get the actual mode number that you need to pass as a parameter to the kernel:

```
# hwinfo --framebuffer
```

The command should be run as root. Pick the number corresponding to the desired resolution. The modes reported by hwinfo are in hexadecimal. Use them with the '0x' prefix or convert them to decimal.

Modes available in Parallels

The VESA BIOS emulation in the Parallels virtual machine has a different set of non-standard VESA modes. As of build 3214, vbetest reveals these modes:

	640 × 400	640 × 480	720 × 480	800 × 500	800 × 600	896 × 672	1,024 × 640	1,024 × 768	1,152 × 720	1,280 × 1,024	1,440 × 900	1,600 × 1,200
256 color palette	256 (0100h)	257 (0101h)	367 (016Fh)	364 (016Ch)	259 (0103h)	297 (0129h)	358 (0166h)	261 (0105h)	355 (0163h)	263 (0107h)	352 (0160h)	284 (011Ch)
15-bit (5:5:5)		272 (0110h)			275 (0113h)			278 (0116h)		281 (0119h)		285 (011Dh)
16-bit (5:6:5)	289 (0121h)	273 (0111h)	368 (0170h)	365 (016Dh)	276 (0114h)	298 (012Ah)	359 (0167h)	279 (0117h)	356 (0164h)	282 (011Ah)	353 (0161h)	286 (011Eh)
24-bit (8:8:8)	290 (0122h)	274 (0112h)	369 (0171h)	366 (016Eh)	277 (0115h)	299 (012Bh)	360 (0168h)	280 (0118h)	357 (0165h)	283 (011Bh)	354 (0162h)	287 (011Fh)

See also

- nouveau (software)

References

- VESA Super VGA BIOS Extension 1.0 (https://web.archive.org/web/20131211151957/http://www.gemixtes.de/super-vga-bios-extension-standard-vs891001/) (Standard # VS891001) 1 October 1989
- VESA BIOS Extension 1.2 (https://web.archive.org/web/20090114055246/http://docs.ruudkoot.nl/vesasp12.txt)
- VESA BIOS Extension 1.2 (mirror) (http://griwes.info/vbe/12.txt)
- VESA BIOS Extension 1.2 (Wayback Machine mirror) (https://web.archive.org/web/20090114055246/http://docs.ruudkoot.nl/vesasp12.txt)
- VESA BIOS Extension 2.0 (https://web.archive.org/web/20081211174813/http://docs.ruudkoot.nl/vbe20.txt)
- VESA BIOS Extension 2.0 (mirror) (http://griwes.info/vbe/20.txt)
- VESA BIOS Extension 2.0 (Wayback Machine mirror) (https://web.archive.org/web/20100305015947/http://docs.ruudkoot.nl/vbe20.txt)
- VESA BIOS Extension 3.0 (http://www.petesqbsite.com/sections/tutorials/tuts/vbe3.pdf)

External links

- Dr. Dobb's Examining the VESA VBE 2.0 Specification (http://www.ddj.com/architect/184409592)
- How To Use Super VGA (VESA 1.x Non-Linear) (https://web.archive.org/web/20120328134352/http://www.opferman.net/Text/svga.txt)
- Scitech Software (https://web.archive.org/web/20060613000649/http://www.scitechsoft.com/) Creators of the useful UniVBE and Scitech Display Doctor (https://web.archive.org/web/20061020161810/http://www.scitechsoft.com/sdd_win.html) VESA VBE enhancement software. Made freely available in recent years.
- SuperVGA/VESA programmer's notes (http://www.faqs.org/faqs/pc-hardware-faq/supervga-programming/)
- List of VESA VBE 2.0/3.0 implementing chipsets (https://web.archive.org/web/20090112210259/http://www.xenxys.info/index.php?title=VESA)
- Capture VBE mode info vbespy source package (http://www.phoronix.net/downloads/vbespy.tar.bz2)
- How to use vbespy source package (https://www.phoronix.com/scan.php?page=article&item=803&num=1)
- vbetool (http://www.codon.org.uk/~mjg59/vbetool/) - an application for executing video card BIOS code
- VESA BIOS Extension Serial Control Interface Standard (https://web.archive.org/web/*/http://www.vesa.org/Public/VBE/vbesci10-2w.pdf) at the Wayback Machine (archive index)
- VESA BIOS Extension/Accelerator Functions (VBE/AF) (https://web.archive.org/web/*/http://www.vesa.org/public/VBE/VBE-AF07.pdf) at the Wayback Machine (archive index)
- VESA BIOS interface (http://pdos.csail.mit.edu/6.828/2004/readings/hardware/vgadoc/VESA.TXT)

1. hwinfo source code on GitHub (https://github.com/openSUSE/hwinfo)