



[\[H\]ard|Forum](http://hardforum.com/index.php) (<http://hardforum.com/index.php>)

- [Video Cards](http://hardforum.com/forumdisplay.php?f=3) (<http://hardforum.com/forumdisplay.php?f=3>)

- - [How VSync works, and why people loathe it](http://hardforum.com/showthread.php?t=928593) (<http://hardforum.com/showthread.php?t=928593>)

Arkalius

07-18-2005 12:50 AM

How VSync works, and why people loathe it

I recently learned that how I thought vsync worked was wrong, and now knowing the way it really does work, I think it would be worthwhile to make sure everyone here understands it.

What is VSync? VSync stands for Vertical Synchronization. The basic idea is that synchronizes your FPS with your monitor's refresh rate. The purpose is to eliminate something called "tearing". I will describe all these things here.

Every CRT monitor has a refresh rate. It's specified in Hz (Hertz, cycles per second). It is the number of times the monitor updates the display per second. Different monitors support different refresh rates at different resolutions. They range from 60Hz at the low end up to 100Hz and higher. Note that this isn't your FPS as your games report it. If your monitor is set at a specific refresh rate, it always updates the screen at that rate, even if nothing on it is changing. On an LCD, things work differently. Pixels on an LCD stay lit until they are told to change; they don't have to be refreshed. However, because of how VGA (and DVI) works, the LCD must still poll the video card at a certain rate for new frames. This is why LCD's still have a "refresh rate" even though they don't actually have to refresh.

I think everyone here understands FPS. It's how many frames the video card can draw per second. Higher is obviously better. However, during a fast paced game, your FPS rarely stays the same all the time. It moves around as the complexity of the image the video card has to draw changes based on what you are seeing. This is where tearing comes in.

Tearing is a phenomenon that gives a disjointed image. The idea is as if you took a photograph of something, then rotated your view maybe just 1 degree to the left and took a photograph of that, then cut the two pictures in half and taped the top half of one to the bottom half of the other. The images would be similar but there would be a notable difference in the top half from the bottom half. This is what is called tearing on a visual display. It doesn't always have to be cut right in the middle. It can be near the top or the bottom and the separation point can actually move up or down the screen, or seem to jump back and forth between two points.

Why does this happen? Lets take a specific example. Let's say your monitor is set to a refresh rate of 75Hz. You're playing your favorite game and you're getting 100FPS right now. That means that the mointor is updating itself 75 times per second, but the video card is updating the display 100 times per second, that's 33% faster than the mointor. So that means in the time between screen updates, the video card has drawn one frame and a third of another one. That third of the next frame will overwrite the top third of the previous frame and then get drawn on the screen. The video card then finishes the last 2 thirds of that frame, and renders the next 2 thirds of the next frame and then the screen updates again. As you can see this would cause this tearing effect as 2 out of every 3 times the screen updates, either the top third or bottom third is disjointed from the rest of the display. This won't really be noticeable if what is on the screen isn't changing much, but if you're looking around quickly or what not this effect will be very apparant.

Now this is where the common misconception comes in. Some people think that the solution to this problem is to simply create an FPS cap equal to the refresh rate. So long as the video card doesn't

go faster than 75 FPS, everything is fine, right? Wrong.

Before I explain why, let me talk about double-buffering. Double-buffering is a technique that mitigates the tearing problem somewhat, but not entirely. Basically you have a frame buffer and a back buffer. Whenever the monitor grabs a frame to refresh with, it pulls it from the frame buffer. The video card draws new frames in the back buffer, then copies it to the frame buffer when it's done. However the copy operation still takes time, so if the monitor refreshes in the middle of the copy operation, it will still have a torn image.

VSync solves this problem by creating a rule that says the back buffer can't copy to the frame buffer until right after the monitor refreshes. With a framerate higher than the refresh rate, this is fine. The back buffer is filled with a frame, the system waits, and after the refresh, the back buffer is copied to the frame buffer and a new frame is drawn in the back buffer, effectively capping your framerate at the refresh rate.

That's all well and good, but now let's look at a different example. Let's say you're playing the sequel to your favorite game, which has better graphics. You're at 75Hz refresh rate still, but now you're only getting 50FPS, 33% slower than the refresh rate. That means every time the monitor updates the screen, the video card draws 2/3 of the next frame. So let's track how this works. The monitor just refreshed, and frame 1 is copied into the frame buffer. 2/3 of frame 2 gets drawn in the back buffer, and the monitor refreshes again. It grabs frame 1 from the frame buffer for the first time. Now the video card finishes the last third of frame 2, but it has to wait, because it can't update until right after a refresh. The monitor refreshes, grabbing frame 1 the second time, and frame 2 is put in the frame buffer. The video card draws 2/3 of frame 3 in the back buffer, and a refresh happens, grabbing frame 2 for the first time. The last third of frame 3 is drawn, and again we must wait for the refresh, and when it happens, frame 2 is grabbed for the second time, and frame 3 is copied in. We went through 4 refresh cycles but only 2 frames were drawn. At a refresh rate of 75Hz, that means we'll see 37.5FPS. That's noticeably less than 50FPS which the video card is capable of. This happens because the video card is forced to waste time after finishing a frame in the back buffer as it can't copy it out and it has nowhere else to draw frames.

Essentially this means that with double-buffered VSync, the framerate can only be equal to a discrete set of values equal to $\text{Refresh} / N$ where N is some positive integer. That means if you're talking about 60Hz refresh rate, the only framerates you can get are 60, 30, 20, 15, 12, 10, etc etc. You can see the big gap between 60 and 30 there. Any framerate between 60 and 30 your video card would normally put out would get dropped to 30.

Now maybe you can see why people loathe it. Let's go back to the original example. You're playing your favorite game at 75Hz refresh and 100FPS. You turn VSync on, and the game limits you to 75FPS. No problem, right? Fixed the tearing issue, it looks better. You get to an area that's particularly graphically intensive, an area that would drop your FPS down to about 60 without VSync. Now your card cannot do the 75FPS it was doing before, and since VSync is on, it has to do the next highest one on the list, which is 37.5FPS. So now your game which was running at 75FPS just halved its framerate to 37.5 instantly. Whether or not you find 37.5FPS smooth doesn't change the fact that the framerate just cut in half suddenly, which you would notice. This is what people hate about it.

If you're playing a game that has a framerate that routinely stays above your refresh rate, then VSync will generally be a good thing. However if it's a game that moves above and below it, then VSync can become annoying. Even worse, if the game plays at an FPS that is just below the refresh rate (say you get 65FPS most of the time on a refresh rate of 75Hz), the video card will have to settle for putting out much less FPS than it could (37.5FPS in that instance). This second example is where the perceived drop in performance comes in. It looks like VSync just killed your framerate. It did, technically, but it isn't because it's a graphically intensive operation. It's simply the way it works.

All hope is not lost however. There is a technique called triple-buffering that solves this VSync

problem. Lets go back to our 50FPS, 75Hz example. Frame 1 is in the frame buffer, and 2/3 of frame 2 are drawn in the back buffer. The refresh happens and frame 1 is grabbed for the first time. The last third of frame 2 are drawn in the back buffer, and the first third of frame 3 is drawn in the second back buffer (hence the term triple-buffering). The refresh happens, frame 1 is grabbed for the second time, and frame 2 is copied into the frame buffer and the first part of frame 3 into the back buffer. The last 2/3 of frame 3 are drawn in the back buffer, the refresh happens, frame 2 is grabbed for the first time, and frame 3 is copied to the frame buffer. The process starts over. This time we still got 2 frames, but in only 3 refresh cycles. That's 2/3 of the refresh rate, which is 50FPS, exactly what we would have gotten without it. Triple-buffering essentially gives the video card someplace to keep doing work while it waits to transfer the back buffer to the frame buffer, so it doesn't have to waste time. Unfortunately, triple-buffering isn't available in every game, and in fact it isn't too common. It also can cost a little performance to utilize, as it requires extra VRAM for the buffers, and time spent copying all of them around. However, triple-buffered VSync really is the key to the best experience as you eliminate tearing without the downsides of normal VSync (unless you consider the fact that your FPS is capped a downside... which is silly because you can't see an FPS higher than your refresh anyway).

I hope this was informative, and will help people understand the intricacies of VSync (and hopefully curb the "VSync, yes or no?" debates!). Generally, if triple buffering isn't available, you have to decide whether the discrete framerate limitations of VSync and the issues that can cause are worth the visual improvement of the elimination of tearing. It's a personal preference, and it's entirely up to you.

Brent_Justice

07-18-2005 01:04 AM

Did you write this or did you copy and paste it from somewhere else? Cause that is an absolutely perfect and easy to understand description of VSYNC.

In addition I would add that enabling Tripple Buffering solves the halving of the framerate problem with VSYNC enabled.

Arkalius

07-18-2005 01:08 AM

I wrote that myself. Can you elaborate on the triple buffering issue?

robberbaron

07-18-2005 01:23 AM

Great essay. I just googled a random part of it and can confirm that this guy wrote it out himself. Thanks for the info!

Brent_Justice

07-18-2005 01:25 AM

[QUOTE=Arkalius]I wrote that myself. Can you elaborate on the triple buffering issue?[/QUOTE]

That was great, I appreciate you writing this, I may sticky this or add it to the FAQ or something so that everyone can read it on this forum all the time. It is a very common concern and a topic that is brought up frequently.

To elaborate on Triple Buffering with VSYNC first we need to understand how it works in normal double buffering mode. In double buffering 1 frame is written to the backbuffer and then flipped over to the front buffer when it is rendered and scanned out through the DAC.

When you have VSYNC enabled what is actually happening is that with double buffering the image is only copied to the front buffer when the electron gun is at the top line of the frame. If you have the refresh set at 60Hz for example and VSYNC is enabled and you are below or above that limit then the graphics card cannot draw the back buffer fast enough to complete the entire frame so the front buffer doesn't change until a complete frame is drawn, reducing the framerate in half.

When you force Triple Buffering with VSYNC enabled you are creating three framebuffers, 1 front buffer and 2 back buffers. In this situation both backbuffers are rendered even before one is flipped to the front buffer, so the graphics card can render one backbuffer and start on the second backbuffer even if the electron gun hasn't got back up to the top line of the frame yet.

So in effect there is no halving of the framerate. You get the benefit of VSYNC enabled (no tearing) and have no drawbacks with halving of the framerate.

air2k5

07-18-2005 01:25 AM

yea to top it off great info

Flak Monkey

07-18-2005 01:27 AM

Wow, that was awesome, I never knew the thing about framerates haveing to at specific ratios to the refresh rate. Thanks for the great guide.

Mister E

07-18-2005 01:28 AM

awesome info in here that's easy to assimilate. bravo. :)

AznAnarchy99

07-18-2005 01:29 AM

i never notice tearing. is there anyway a person can get some kind of screenshot?

Ducay

07-18-2005 01:31 AM

So if I have an LCD running @ 60Hz (yah yah I know)

It is beneficial to have Vsynch off I presume? I figure this way, the Vsynch wont be limiting the FPS to 60 and 30.....etc

Great read! (y)

Brent_Justice

07-18-2005 01:33 AM

[QUOTE=Ducay]So if I have an LCD running @ 60Hz (yah yah I know)

It is beneficial to have Vsynch off I presume? I figure this way, the Vsynch wont be limiting the FPS to 60 and 30.....etc

Great read! (y)[/QUOTE]

Yes, VSYNC disabled eliminates the halving of the framerate. But it produces tearing.

If you can force Triple Buffering in a game though then you can run with VSYNC enabled and not have tearing and not have the framerate halving.

Ducay

07-18-2005 01:36 AM

Even with an LCD? Isn't tearing a CRT only problem? Considering the pixels on an LCD aren't being refreshed, shouldn't this make Vsync and 3x buffering non-applicable?

robberbaron

07-18-2005 01:37 AM

[QUOTE=Ducay]Even with an LCD? Isn't tearing a CRT only problem? Considering the pixels on an LCD aren't being refreshed, shouldn't this make Vsync and 3x buffering non-applicable?[/QUOTE]

It's still a matter of the video card rendering the video incorrectly, if I understand correctly.

Brent_Justice

07-18-2005 01:38 AM

[QUOTE=Ducay]Even with an LCD? Isn't tearing a CRT only problem? Considering the pixels on an LCD aren't being refreshed, shouldn't this make Vsync and 3x buffering non-applicable?[/QUOTE]

Tearing is still a problem on LCD's as well, yes, they don't refresh, but tell Windows that :D Notice they do run at 60Hz in Windows. VSYNC is a part of games and the OS.

Arkalius

07-18-2005 01:39 AM

Some people are more sensitive to tearing than others. Chances are you've gotten used to it so you don't notice it. It's usually more noticeable if your FPS is higher than your refresh rate. When it's lower, there's still tearing but it's harder to see. If you're used to it, it's hard to really point it out. The only way you can really learn to see it if you're used to it is play the same sequence of a game with VSync both on and off, then you'll probably see the difference.

Screenshots won't show tearing. Tearing is a side effect of the refresh rate, and therefore is the monitor's fault technically. Screenshots always draw a single complete frame.

The following is speculative on my part: LCD's don't show tearing for an FPS less than the max refresh rate. The refresh rate on an LCD is simply the maximum rate it can update itself. Unlike CRT's they don't have to constantly update the screen. They only update when new data is available, however they can only update so fast. If the framerate is lower than the refresh rate, you're fine, however if it gets higher, it can produce tearing. Remember this is speculation on my part...

banned_user

07-18-2005 01:39 AM

Thanks a ton. Tearing has always distracted me more than a lack of AA or AF has. I should have researched it myself, but I didn't think it was that involved.

Ducay

07-18-2005 01:39 AM

I never understood the refresh rate settings for LCDs.

Its my understanding that the LCD will change its pixels when needed at the rate of FPS or its technical maximum rate.

EDIT: question pretty much answered while I was typing this

Brent_Justice

07-18-2005 01:42 AM

[QUOTE=Ducay]I never understood the refresh rate settings for LCDs.

Its my understanding that the LCD will change its pixels when needed at the rate of FPS or its technical maximum rate.

EDIT: question pretty much answered while I was typing this[/QUOTE]

The nature of games, video cards, the API and the OS is VSYNC still. While LCD's don't refresh, according to the game and Windows, they do.

Dark_Seph

07-18-2005 04:29 AM

[QUOTE=banned_user]Thanks a ton. Tearing has always distracted me more than a lack of AA or AF has.[/QUOTE] Same here, I can live without AA or AF but when I see it flash by, O_O.

Very good write up.

RandysWay

07-18-2005 04:47 AM

Very nice thread. Looks like you took your time writing this one.

And, it's...

[SIZE=4]STICKY WORTHY! [/SIZE] :D

kmeson

07-18-2005 06:14 AM

I will perform one correction.

[QUOTE=Arkalius]If your monitor is set at a specific refresh rate, it always updates the screen at that rate, even if nothing on it is changing. Note that this is not the case on LCD displays. LCDs maintain a static display on each pixel until it needs to change.
[/QUOTE]

LCD's must still sample the image held in vram at a particular rate. This rate is typically 60hz, and

is part of the DVI [URL=http://www.ddwg.org/lib/dvi_10.pdf]specification[/URL]. Sampling occurs regardless of content change, that is to say change in the contents of vram does not trigger an update of display.

IMO a clearer explanation of tearing follows:

Imagine a horizontal line moving down your display. It starts at the top, moves down to the bottom, and repeats. A complete traversal from top to bottom is called a frame. A number of these are completed per second and this is call the refresh rate, e.g. 60hz.

As the line moves from pixel row to pixel row, the content under the line are refreshed; the row of pixels corresponding to the line's current vertical position are displayed from vram. If you could change the content of vram faster than the rate at which a frame is complete you could display a number of different partial images. This is tearing.

The same happens both vertically and horizontally. Instead of a line imagine now a point. It moves left to right, top to bottom; starting at the top-left and ending at the bottom-right. Once complete it repeats. Again like the line above, the pixel under the point is display from vram.

Some history...

Originating on the C64 and Amiga a graphical effect known as "Copper Bars" utilized precise timing to update pixels as they were being refreshed. First were horizontal bars because time was easier. Then came vertical bars where timing is much harder. Typically a number of "copper bars" were displayed depicting a sinusoidal pattern.

The benefits of using this display effects are mostly lost today due to high color resolutions. For example updating the display palette faster than refresh rate allows for the simulation of greater color depth. However this technique is still used in handheld consoles today.

Anyway that's probably more than most care for. G-nigh

Blue Falcon

07-18-2005 06:20 AM

Tearing is actually much more noticable on an LCD because of the LCD's lack of physical refresh rate. Since the pixels on an LCD are always 'on' and not flickering on/off at 60,75,85,100, etc times a second, even the slightest bit of image tearing is noticeable.

Some people are more prone to noticing it than others, but stick someone on a HL2 map with lots of vertical-type textures, and they'll notice the tearing right away. :)

Oh and the first post of this thread is GREAT. There are soooo many people who don't 'get' the whole VSync thing. :p

Blue Falcon

07-18-2005 06:24 AM

[QUOTE=kmeson]Originating on the C64 and Amiga a graphical effect known as "Copper Bars" utilized precise timing to update pixels as they were being refreshed. First were horizontal bars because time was easier. Then came vertical bars where timing is much harder. Typically a number of "copper bars" were displayed depicting a sinusoidal pattern.[/QUOTE]

:p Nice to see someone who remembers the same shit that I do.

Oh the lovely days of intro/demo coding on the ol' C64. She was such a sexy little beast.

And the 1541 was so much fun to whack people with when it wasn't being used to read floppies.
:p

DandyBear

07-18-2005 08:07 AM

Most people that read and post in [H]ard's forums do not upgrade thier monitors. They do however go out and spend money on the fastest processors and graphics cards. I argue that a good quality, speedy monitor is more important than having the latest graphics card. What good is having the GTX or SLI even if your monitor only updates the screen a fixed number of times? All those extra frames per second are not viewable anyway. Visual Tearing is becoming increasingly more apparent as the gap between our monitors and video cards lengthens.

burninggrave101

07-18-2005 08:33 AM

Triple buffering still isn't available for nVidia cards on D3D is it? And even with ATI cards you can only enable triple buffering if the game supports it in D3D.

FlatLine84

07-18-2005 08:49 AM

The my question is this, I have a Dell 2001fp, I had always been under the impression that vsync would make no difference on an LCD, but I have texture tearing in Source, when vsync is disabled, no texture tearing when it's enabled. This has baffled me.

burninggrave101

07-18-2005 08:53 AM

[QUOTE=FlatLine84]The my question is this, I have a Dell 2001fp, I had always been under the impression that vsync would make no difference on an LCD, but I have texture tearing in Source, when vsync is disabled, no texture tearing when it's enabled. This has baffled me.[/QUOTE]

Tearing is even worse on an LCD then it is on a CRT so you will definitely notice it without vsync enabled in certain games.

Sly

07-18-2005 08:55 AM

Correct me if i'm wrong but since DX doesn't officially have TripleBuffering, i had always thought that [URL=http://img267.imageshack.us/img267/793/nvscr21jm.jpg]this[/URL] was their version of it since it it's basically does the same thing as TripleBuffering. :confused:

burninggrave101

07-18-2005 09:06 AM

[QUOTE=Sly]Correct me if i'm wrong but since DX doesn't officially have TripleBuffering, i had always thought that [URL=http://img267.imageshack.us/img267/793/nvscr21jm.jpg]this[/URL] was their version of it since it it's basically does the same thing as TripleBuffering. :confused:[/QUOTE]

What do you set it to? Your current set refresh rate?

FingerSlut

07-18-2005 09:08 AM

These are the kinds of threads I join forums for. I was not expecting this in the Video card forum ;) Excellent post.

Now someone should explain to me how dual display works with multiple resolutions and refresh rates. :p

Shakezilla

07-18-2005 09:27 AM

Thanks man even I understood that :)
Now I can help when two noobs in CS argue over what VSync does.

I vote for a sticky.

texuspete00

07-18-2005 09:34 AM

So what's the downsides of triple buffering? Performance or otherwise? All these buffers must put strain on the card. Since it helps for being under the refresh rate, performance would seem key. It would be silly for game makers not putting it in if everything is cherry. People complain of mouse lag in fast fps's with this. Is it just the lets say you get 50 fps, on a 60Hz screen, and 10 frames are put up by the monitor twice that make a slight perceived lag?

Found this app.... let's you force triple buffering in DX.
[url]<http://www.beyond3d.com/forum/viewtopic.php?t=19899&postdays=0&postorder=asc&start=0>[/url]

It's just really weird it would all be under lock and key in DX.

Un4given

07-18-2005 09:49 AM

[QUOTE=texuspete00]So what's the downsides of triple buffering? Performance or otherwise? All these buffers must put strain on the card. Since it helps for being under the refresh rate, performance would seem key. It would be silly for game makers not putting it in if everything is cherry. People complain of mouse lag in fast fps's with this. Is it just the lets say you get 50 fps, on a 60Hz screen, and 10 frames are put up by the monitor twice that make a slight perceived lag?

Found this app.... let's you force triple buffering in DX.
[url]<http://www.beyond3d.com/forum/viewtopic.php?t=19899&postdays=0&postorder=asc&start=0>[/url]

It's just really weird it would all be under lock and key in DX.[/QUOTE]

Having the additional buffer consumes some of the video card's memory that could otherwise be used for other purposes, like holding textures.

revenant

07-18-2005 10:12 AM

good read indeed. :D

revenant

07-18-2005 10:27 AM

[QUOTE=burninggrave101]Tearing is even worse on an LCD then it is on a CRT so you will definitely notice it without vsync enabled in certain games.[/QUOTE]

I really noticed this when moving from a CRT to an LCD... I had to enable vertical sync because the tearing was bad... Although it was much more noticable on my 9700Pro then with my SLI setup... Which made me think that the raw speed of the "graphics engine" had something to do with it. Is this also not different depending on the performance of an lcd? Say it would be more noticable on an 25ms "pixel refresh" (combined rise and fall time for one round of display output for a pixel), vs say an 8ms one? I guess this would be the CRT equiv of the speed of the electron gun. Hmm... Oi... I need more coffee before going any further...

Unknown-One

07-18-2005 11:26 AM

[QUOTE=Arkalius]The following is speculative on my part: LCD's don't show tearing for an FPS less than the max refresh rate. The refresh rate on an LCD is simply the maximum rate it can update itself. Unlike CRT's they don't have to constantly update the screen. They only update when new data is available, however they can only update so fast. If the framerate is lower than the refresh rate, you're fine, however if it gets higher, it can produce tearing. Remember this is speculation on my part...[/QUOTE]Hmm, if thats true, than simply using a framerate cap would fix tearing for LCD's.

I'll have to give that a try...

Scali

07-18-2005 12:50 PM

I'd like to add some things.

With D3D, the whole concept of double or triple buffering is generalized.

You can create a chain of buffers, called a swap-chain. This chain can be as long as you want, basically. If you create a chain of two buffers (one frontbuffer, the one actually displayed by the monitor, and one backbuffer, the one being drawn to), you effectively have double-buffering. Add another buffer, and you have triple buffering, etc.

It is up to the programmer, basically.

In OpenGL they sort of 'forgot' to add this feature. ATi allows you to force it. NVIDIA doesn't, for some reason. There are extensions for OpenGL, but not many games make use of them, I think.

Also, the thing about having framerates of 80, 40, 26.67 etc is a bit unclear.

What really happens is that the analog output of the videocard controls the cathode ray of the monitor. There are two special intervals, where nothing is drawn. One is the horizontal blank interval, when the ray moves from the right of the finished scanline to the left of the next scanline. Another is the vertical blank interval, when the ray moves from the bottom right corner back to the top left corner.

There are registers in the videocard that can be checked to see if the ray is in one of these intervals (or on some hardware an interrupt signal is generated). When turning on vsync, the application will draw a frame to the backbuffer, and then wait for the first vertical blank interval to arrive. At this point, it will swap the backbuffer and the frontbuffer, and then start drawing the next frame into the new buffer.

So if the drawing takes slightly longer than the time between two intervals (the refreshrate), then it automatically has to wait for every second interval, because the first one is missed while the frame is still being drawn. The result is that the framerate halves... Or worse depending on how many intervals are missed.

Triple buffering overcomes this problem because after the first backbuffer is finished drawing, the application can start drawing in the extra buffer. There is no time lost waiting for the next interval to free up a buffer for drawing the next frame.

Also, the copperbar effect doesn't actually draw anything. The trick is that the palette was changed on every horizontal interval. You just draw the bars with fixed colours 0, 1, 2, etc. By changing the palette, these values will respond to different colours on screen. With clever palettes, you can seemingly move the bars up and down the screen. Through this trick it is also possible to use more colours on screen than possible in one palette. This trick is no longer possible on modern videocards, since they don't use a palette anymore. The colour values in memory correspond directly to the R, G, B intensities, and are not indices into a palette.

By the way, it is also possible to do single-buffering. This means that you draw directly into the frontbuffer. When you strictly draw from top-left to bottom-right, and you can draw faster than the screen-refresh, you can use this. Else you get a more severe form of tearing, since you are drawing in places that have already been covered by the CRT before they were updated, so the updates are never visible.

With 3d polygons this looks really weird. Polygons generally are in more or less random order, so you will get some interesting fractions of work-in-progress renderings.

Arkalius

07-18-2005 02:30 PM

Scali has it right... Here's another way of describing it.

With double-buffering and VSync enabled, the video card can only swap the two buffers just after the CRT's refresh cycle has occurred. In my 75Hz, 50FPS example, the video card can draw 2/3 of a frame for each refresh cycle, so it draws 2/3 of a frame in the back buffer, the monitor draws frame 1 from the frame buffer, the video card finishes the last third of the frame in the back buffer, but now it has to wait as it can only swap the buffer out right after the monitor refreshes. The monitor grabs frame 1 a second time and then the frames switch. So, even though the video card can draw 50fps, this limitation shows only 37.5FPS (half the refresh rate). With triple buffering, the same process occurs, but after finishing that last third of the back buffer, it can render the first third of a frame in the second back buffer, and after the monitor grabs frame 1 the second time, frame 2 is swapped in, and the video card renders the last 2/3rds of frame 3, frame 2 is grabbed, and frame 3, now finished, can be swapped into the frame buffer. On the next refresh cycle, the monitor grabs frame 3. As you can see, this means there are 2 frames rendered every 3 refresh cycles, giving you the 50FPS you really got. All that happens is every third refresh is the same frame as the previous one.

Essentially double-buffered VSync forces the video card to sit and wait at certain points because it's filled up the buffer but can't swap it out quite yet. Triple buffering solves that by allowing the video card to continue working on the second back buffer while it waits to swap the first one into the frame buffer.

kmeson

07-18-2005 03:34 PM

[QUOTE=Scali]In OpenGL they sort of 'forgot' to add this feature. ATi allows you to force it. NVIDIA doesn't, for some reason. There are extensions for OpenGL, but not many games make use of them, I think.
[/QUOTE]

Most games I know use extensions. This has always been the case, for example most games utilize multitexturing but GL does not provide this functionality out of the box. Over time these extensions

become part of the specification as ratified by the ARB (Architecture Review Board).

Besides the extra ram necessary for triple buffering, it can also introduce what people percieve as input lag. Input lag is the time from when you request an action to be performed to when that action is displayed on screen.

[QUOTE=Scali]Also, the copperbar effect doesn't actually draw anything.[/QUOTE]

Semantics.

[QUOTE=Scali]

This trick is no longer possible on modern videocards, since they don't use a palette anymore.
[/QUOTE]

Are you certain about this? I would be very surprised if this was a hardware issue. Rather I belive it to be an OS issue. It's been a while but I would expect that if I booted into DOS, I could get mode 13h running and have direct access to the VGA register set (3c8h & 3c9h pel, 3dah v & h blank).

Scali

07-18-2005 04:31 PM

[QUOTE=kmeson]Most games I know use extensions. This has always been the case, for example most games utilize multitexturing but GL does not provide this functionality out of the box. Over time these extensions become part of the specification as ratified by the ARB (Architecture Review Board).[/quote]

I meant the specific extensions that enable triple-buffering, obviously :)

[quote]Semantics.[/quote]

Yes, and what are semantic errors? That something works, but doesn't do what you want it to do. Same here. Semantics are very important in programming.

[quote]Are you certain about this? I would be very surprised if this was a hardware issue. Rather I belive it to be an OS issue. It's been a while but I would expect that if I booted into DOS, I could get mode 13h running and have direct access to the VGA register set (3c8h & 3c9h pel, 3dah v & h blank).[/QUOTE]

These modes are often emulated. Back in the early Windows days I benched my Matrox card, and it turned out that 32-bit mode was faster than 8-bit mode, even though the 32-bit mode had to do the palette lookup manually. Also, nobody would still use a palette mode today for any kind of graphics (for one, there is no 3d acceleration in palette mode), so the copperbar effect cannot be applied in any practical situations. It never was very easy to do on PC anyway. As stated, it was an effect on C64, since the particular hardware on that machine made it relatively easy to implement. PC hardware has always been relatively poor compared to home computers.