

---

Subject: : AmigaOS4

Topic: : GL4ES: another OpenGL over OpenGL ES2 emulation - some tech. info and porting progress

Re: GL4ES: another OpenGL over OpenGL ES2 emulation - some tech. info and porting progress

Author: : kas1e

Date: : 2019/9/17 17:05:26

URL:

@Capehill, Theiller

Hans explain those things a bit for me, and that what i understand :

That "DMA support in graphics.library" in Sam440, Sam460 & x1000 is not real DMA , its just a hack, which used their CPU's DMA to speed up those ram->vram transfers , but only for internal use inside of graphics.library , and only for bitmaps. I.e. its "DMA", but CPU's DMA and not real DMA or GART of anything of that sort. Just some little "speed up".

In other words, is of no use for drivers (like for minigl, for warp3dnova, radeon drivers, etc), and more of it , its only for use for those apps which use graphics.library and rely on those parts which is "hacked" inside of graphics.library to speed things ups (named those "graphics.library's bitmaps").

Probabaly, it may help somewhere and not only in benchmarks (at least it was added not just because), but it didn't help at all with drivers, so that explain why we didn't have a single difference when test irrlicht engine examples between x5000 (without that "cpu dma hack") or on x1000 (with that "cpu dma hack"). Those examples done for gl4s, which works over ogles2.library, which works on top of warp3dnova, which in turn, didnt have any kind of DMA acceleration for RAM->VRAM transfers. There is only proper implementation of GART can help.

Quote:

Certainly let Nova do the reordering was not a good idea as datas are then accessed several times (vs a cpu that will write to real GPU vram directly the reordered datas)

As i aware now, warp3dnova's BufferUnlock() do not only writing from RAM to VRAM, but also do endian conversion from big-endian to little-endian (as gfx card is little endian). So we have 2 stop factors there :

1. no real DMA (GART) is used , that mean transferring from RAM->VRAM are slow. We should be happy we even have something usable without it. We at least have 100fps in quake3 without GART, that for sure not bad.

2. Endian conversion inside of BufferUnlock(), may slow all things down, expectually if it didn't compiled with -O3 optimisation enabled by any of reassons. As it also mean buffers, working with them, etc, so pretty possible

that add another bottleneck.

All of this probably explain well why some things works on minigl and on gl4es almost the same by speed, like that quake3, lugaru, supertuxkart : all those games do a lot of drawing per frame, which is limited by speed because of no GART, so both minigl.library which working over warp3dnova, and gl4es , limited by the same.

But that code which writen "right", i.e. not thousands of draw calls per frame with lots of data, those ones speeduped well by usage of VBO and co.

Imho, but i think pretty close to truth.