

---

Subject: : AmigaOS4

Topic: : Warp3DNova shader bugs thread

Re: Warp3DNova shader bugs thread

Author: : Daytona675x

Date: : 2020/6/4 12:43:24

URL:

@Capehill

Quote:

It would be nice to get a warning from compiler about potentially uninitialized variables in cases like this.

This behaviour is not just allowed but also not considered to be worth a warning. Nova is not buggy here, so I have to second Hans that there is no action to be taken. Yes, it would be nice but on the other hand it also will result in unnecessary warnings. And such a analysis is not trivial and not for free. After all there are very valid reasons - unlike in those broken shaders - not to initialize a variable right upon declaration.

@kas1e

Quote:

I not sure what is it mean : that uninitialized variables on win32 are always set to 0.0 but on os4 random values ?

No, it's just that some (maybe even most, or even all, doesn't matter) Windows OpenGL drivers apparently set uninitialized variables to 0 even though the standard explicitly does not require it. Or to put it simple: the respective shaders are buggy, not Nova. Those shaders rely on something they simply must not rely on. However, since real-life shaders often seem to falsely rely on this behaviour, maybe a Nova feature request for an init-vars-to-zero-toggle (default off like now) is a good compromise.

Quote:

Question is : is that the same issue everywhere, or they different ones ?

Novas register handling is still broken and therefore *all sorts* of fun can happen. You or I cannot say for sure which exact issue it is. We cannot even say for sure *if* it has to do with the register allocator, it's likely but we cannot be sure. Maybe it's a different bug with familiar looking symptoms. The thing is: with those Shaderjoy shaders it is natural that whatever is the problem results in pixels being falsely colored, simply because Shadertoy is all about pixel colors and nothing else ;)

Anyway, either create one bug-report named "All sorts of pixel garbage" and then extend it with more and more links to shadertoy-shaders - or create one report for each buggy shader, maybe with a not that you *somewhat guess* that it may be related to bug or report xyz. But don't try to be smart and to categorize further 😊 After all

you are a reporter. It's Hans job to fix that stuff and it's also his job to analyse and eventually merge or split your bug reports.

Quote:

I start to think, that it's all different bugs all the time. I.e. bug the same : wrong content in registers, and their wrong compare/use , and it didn't looks like some single particular issue. Just visually it looks pretty much the same like colors swapped, but it all can be different.

IMHO a rewrite of the register allocator is due. From what I see things get fixed in lib release A only to reappear in a slightly different form or the fixes cause other problems as sideeffects. And then there is also the missing register spilling etc.

For the upcoming ogles2 lib version I threw away and rewrote the whole program introspection system 😁 Yes, certainly no fun but sometimes you have to admit to yourself that what you did before was no good and that a rewrite with a fresh head is the far better choice, instead of putting even more patches on that old rag rug. In my experience this *always* pays off.