LiquidVR AffinityGPUControl SDK Sample

Disclaimer

The information contained herein is for informational purposes only, and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions and typographical errors, and AMD is under no obligation to update or otherwise correct this information.

Advanced Micro Devices, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of noninfringement, merchantability or fitness for particular purposes, with respect to the operation or use of AMD hardware, software or other products described herein. No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of AMD's products are as set forth in a signed agreement between the parties or in AMD's Standard Terms and Conditions of Sale.

AMD, the AMD Arrow logo, ATI Radeon™, CrossFireX™, LiquidVR™ and combinations thereof are trademarks of Advanced Micro Devices, Inc. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

Windows™, Visual Studio and DirectX are trademark of Microsoft Corp.

Copyright Notice

Advanced Micro Devices
One AMD Place
P.O. Box 3453
www.amd.com

Overview

This sample should be used as the starting point for all new AMD DirectX11 samples. It demonstrates the performance gain from using two CrossFire-connected GPUs to render 3D scenes for the left and the right eye in parallel on separate GPUs.

It utilizes the AMD_SDK library to add basic functionality such as:

- Timer
- Magnify Tool
- AMD HUD
- Shader Cache

Pre-requisites

To run the sample, the following hardware and software requirements need to be fulfilled:

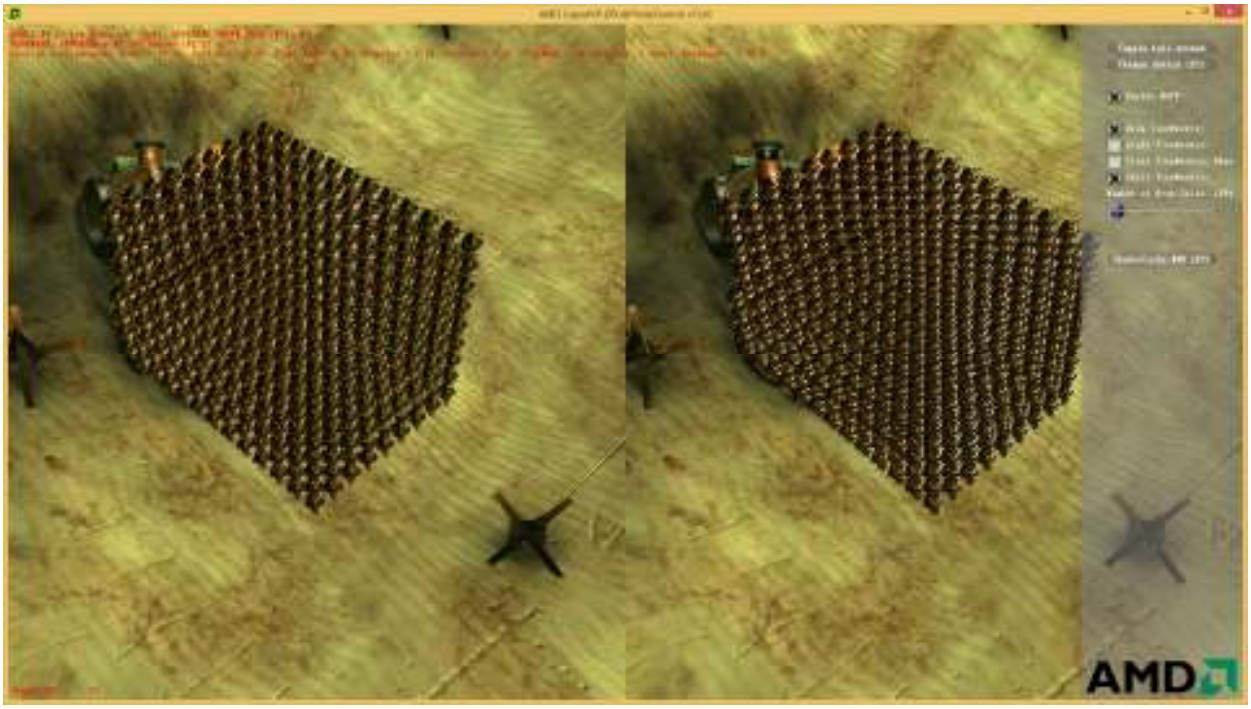
- A PC with a motherboard supporting AMD CrossFireX™ and two CrossFireX™-compatible Radeon graphics cards, with a CrossFireX connector installed.
- A 64-bit Microsoft Windows 7 or Windows 8.1 with the latest graphics drivers (ver 15.10 or newer)
- CrossFire enabled in Catalyst Control Center (under *Gaming/Gaming Performance/AMD CrossFireX™*)
- **Optional:** to compile the sample, Microsoft Visual Studio 2013 is required

Building the AffinityGPUControl Sample

Start Microsoft Visual Studio 2013 and open the *LiquidVR_SDK_1.0\samples\LiquidVR_Samples_VS2013.sln* workspace. Navigate to the *Solution Explorer* pane in Visual Studio, right-click on the *AffinityGPUControl* project and choose the *Set as Startup Project* option from the pop-up menu. Choose *Build Solution* from the *BUILD* menu.

Running the AffinityGPUControl Sample

Run the executable built in the previous step or run a pre-built binary located in *LiquidVR SDK\LiquidVR_SDK_1.0\\bin_x64\Debug\AffinityGPUControl.exe.* Once started, you should see the following image on your primary monitor:



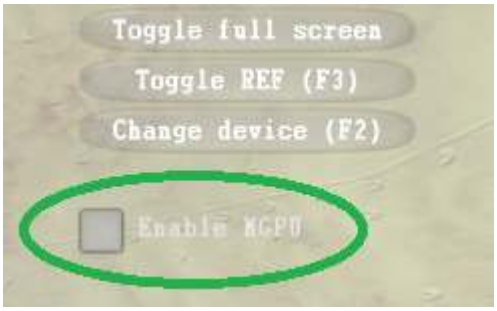
The sample emulates a video game displaying a stereoscopic 3D scene. On a regular non-stereoscopic display the left and the right sides of the image represent what the left and the right eye sees.

A user interface allowing control of various image rendering parameters is displayed on the right in a semi-transparent pane. It can be turned on or off using the F1 function key on the keyboard. It is recommended to turn it off once rendering parameters have been set to achieve maximum rendering performance.

The scene can be navigated by dragging it with a mouse with the left button pressed.

Right-clicking on the scene displayes a magnified image under the mouse cursor when the *Magnify: RMouse* checkbox on the right is checked. The size of the magnified region and the magnification scale can be controlled using the *Magnify Region* and the *Magnify Scale* sliders respectively:

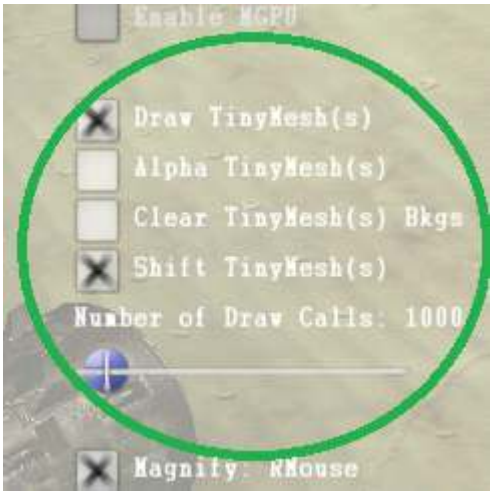
Observe the Enable MGPU checkbox. When AMD CrossFireX is enabled, the checkbox would also be enabled. If it's greyed out, please check AMD CrossFireX settings in the AMD Catalyst Control Center.



The rendering performance statistics is displayed in the upper-left corner of the screen. Observe and note the frame rate when multi-GPU rendering is switched off (*Enable MGPU* checkbox is unchecked, images for both eyes are rendered on a single GPU):

Enable multi-GPU rendering by checking the *Enable MGPU* checkbox. The image for the left eye will be rendered on one GPU and the image for the right eye will be rendered on the other GPU, so that they are rendered in parallel. You will observe a significant (almost 2x) increase in the frame rate displayed. Move the scene around by dragging it with a mouse with the left button pressed. Motion should appear much smoother with multi-GPU rendering enabled.

Experiment with the image of a human figure overlayed over the background of the scene. This figure can be rendered multiple times per each frame, representing dynamic objects that would be drawn over a relatively static background in a real game. To enable rendering of the human figure, check the *Draw TinyMesh(s)* checkbox. You can adjust the number of times the human figure is drawn per frame using the *Number of Draw Calls* slider. You can also emulate other typical game rendering operations, such as alpha blending, pattern replication and background clearing using the *Alpha TinyMesh(s), Shift TinyMesh(s)* and *Clear TinyMesh(s) Bkgs* checkboxes:



Observe the effect of these additional operations on the frame rate with multi-GPU support on and off by toggling the *Enable MGPU* checkbox. You should expect to see significantly higher performance when multi-GPU rendering is enabled.