Android PerfHUD ES quick start guide

Version 1.0001

Contents

Introduction

This quick start guide is designed to assist a new user in setting up and starting PerfHUD ES profiling. PerfHUD ES enables experimentation and in-depth analysis of OpenGL ES applications. Through data provided by the OpenGL ES driver and the NVIDIA GPU, developers are given access to GPU performance and bottleneck information in order to target optimizations where they are most needed. The debugging capabilities give full access to the state of the OpenGL ES pipeline, related textures and shaders, and all rendering states to help find the causes for improper setup and rendering anomalies.

For detailed information on the use of PerfHUD ES please refer to the PerfHUD ES User Guide that is installed with the host application.

Setup

Requirements

Minimum Host PC Hardware Requirements

- Windows based PC with 1 USB port free; ideally with a 2GHz CPU and 2GB RAM.

> **Note: Windows XP is currently the officially-supported host OS for PerfHUD ES. While Windows Vista and Windows 7 have been shown to generally work, it has not been heavily tested.**

- For direct Ethernet connections, you'll need a **crossover** network cable, long enough to reach from the Tegra devkit's built-in Ethernet jack to the host PC. Since the Ethernet connections are configured to automatically obtain an IP address some form of DHCP server must be configured and running on the host PC.
- For wireless Ethernet connections, you'll need a wireless access point that your host PC is configured to communicate with either directly or on an accessible network.

A PerfHUD ES enabled Android OS image which may be found at the Tegra developers site; http://tegradeveloper.nvidia.com/tegra/downloads

A PerfHUD ES host installer which may be found at the Tegra developers site; http://tegradeveloper.nvidia.com/tegra/downloads

Setting up the Tegra 250

Please consult the Tegra 250 devkit setup guide for information on how to assemble the devkit and connect the USB cable between the devkit and the host PC. The guide may be found at the Tegra developers site;
http://developer.download.nvidia.com/tegra/docs/tegra_250_hw_setup.pdf

Flashing the Tegra 250 with a PerfHUD ES enabled Android OS

Please consult the Tegra 250 Development Kit Android Setup Experience, specifically STEP 5: Flash the Devkit with the Android OS for information on how to flash the Android OS to your devkit. The guide may be found at the Tegra developers site;
http://developer.download.nvidia.com/tegra/docs/android_setup_experience_v492.pdf

Installing the Windows PerfHUD ES Host.

1) Run the PerfHUD ES host installer
2) Click “Next”
3) Decide to accept the EULA and then click “Next”
4) Decide on your installation location and then click “Next”
5) Click “Install” and wait for installation to complete
6) Click “Finish”

Connecting to the PerfHUD ES client

Common startup procedure

1) Run the application to be profiled on the devkit.
2) Run the PerfHUD ES host application on your Windows PC, you will now be prompted with the target IP setting dialog.

Connecting to the PerfHUD ES client using TCP/IP

If you are connecting to the PerfHUD ES client over wired or wireless Ethernet using TCP/IP

1) Enter the IP address for your target into the dialog.
2) Click “Advanced”
3) Click “Refresh graphics process list”
4) Select your application from the process and click “Connect”

Connecting to the PerfHUD ES client using ADB

1) On your windows PC open a command prompt and enter “adb logcat” noting the port assigned to the application. This will be presented in the log with a message like the following;

```
Communication layer init succeed, listen on port 7876.
```

2) On your windows PC enable port forwarding for the applications port by entering “adb forward tcp:<port> tcp:<port>” with <port> replaced by the previously noted port number.
3) Enter the “localhost” as the IP address into the dialog.
4) Click “Advanced”
5) Click “Refresh graphics process list”
6) Select your application from the process and click “Connect”

Common problems

The windows ding.wav plays over and over again.

The sound effect is played each time a buffer object is uploaded, to disable open the view dropdown menu and select audio notifications.

The driver time never seems to indicate anything on the dashboard.

Please select a non-zero value for driver time sampling as the default is zero.

The speed bar is not visible and the frame profiler is not selectable.

Please ensure that your application is using the "EGL_NV_Perfmon" extension to acquire all application side timing information as detailed in the "NVIDIA PerfHUD ES User Guide".

Known issues

Application hangs with adb logcat showing "Output FIFO does not refill, context read is stuck".

There is no current workaround.

My application fails to run with GL_OUT_OF_MEMORY and W/GraphicBufferMapper out of memory messages.

Please try restarting the Android device, if the issue persists there is no current workaround.

The timing graph varies wildly for a static scene.

There is no current workaround.

The frame debugger takes a long time to complete or fails to complete.

Please try closing any other applications or increasing the memory available on your host PC.

The texture viewer does not appear.

Please try restarting the PerfHUD ES Host to resolve this issue.

The shader viewer shows "Microcode disassembler is not available" for vertex and fragment microcode modes.

Vertex and fragment microcode disassembly is not supported by PerfHUD ES, please ignore these options as they will be removed in the future and select fragment or vertex source mode.

The driver time sampling interval is ignored after a connection lost.

Please try setting the sampling interval to another value and then back again to your preferred value.

Framebuffer objects are displayed incorrectly when attached to a texture sampler.

There is no current workaround.

The geometry viewer stutters on very complex geometry.

Please try closing any other applications and ensure you have the latest drivers for your graphics card.

The IP Address dialog forgets my selected port.

Please re-enter your selected port or select the application from the advanced button.

Vertical scroll bars appeared clipped along the vertical axis.

There is no current workaround.

Portions of the Performance Dashboard overlay the Frame Debugger.

Please try restarting the PerfHUD ES Host to resolve this issue.

Notice

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, “MATERIALS”) ARE BEING PROVIDED “AS IS.” NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE.

Information furnished is believed to be accurate and reliable. However, NVIDIA Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. NVIDIA Corporation products are not authorized for use as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

Trademarks

NVIDIA, the NVIDIA logo, Tegra, GeForce, NVIDIA Quadro, and NVIDIA CUDA are trademarks or registered trademarks of NVIDIA Corporation in the United States and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

NVIDIA Corporation
2701 San Tomas Expressway
Santa Clara, CA 95050
www.nvidia.com