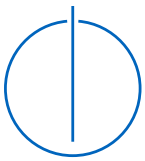


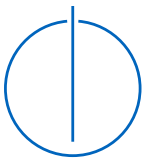
DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics: Game Engineering

Thesis title

Sofia



DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics: Game Engineering

Thesis title

Titel der Abschlussarbeit

Author:	Sofia
Supervisor:	Supervisor
Advisor:	Stepan
Submission Date:	15th March 2022

I confirm that this bachelor's thesis in informatics: game engineering is my own work and I have documented all sources and material used.

Munich, 15th March 2022 Sofia

Acknowledgments

Abstract

Contents

1 Introduction

1.1 Background and Motivation

1.2 Goals and Aims

- develop lightweight tool that has few requirements to gain insight to data movements in a PCIe link

1.3 Delimitation

- Heterogeneous systems have many different interconnects - Focus on PCIE-interface in this thesis - (why)? not sure yet, figure something out

1.4 Structure and Approach

- three approaches / benchmarks - bandwidth: for measuring the raw bandwidth capacity of the system - nvml: for measuring pcie link activity - copy: for measuring pcie link activity

2 A Brief Summary of PCI-Express

2.1 Hardware

2.2 Software

3 Bandwidth Benchmark

3.1 Concept

- Measures raw theoretical maximum bandwidth of pcie link by measuring the duration of memory copies of various chunk sizes - Checks at which chunk sizes the bandwidth of the link is fully saturated

3.2 Implementation

- Compensates for delay - 1 packet with 4B measured as delay - Pageable and pinned memory benchmarks measured - Warmup-feature: first transfer usually has some sort of longer delay, compensates for that (windows-finding, verify on p6000)

3.3 Results

- Transfer durations don't really increase until 8kb - Due to the nature of the PCI-E packet having a max payload of 4kb - First transfer usually has a bit longer delay (warmup?) - On windows: not executable that calls functions, but rather nvcuda64.dll - requires compiling on windows and then using a profiling tool like AMDUprof to look at the program

3.4 Discussion

3.4.1 successes

- gives accurate reading of pcie bandwidths - non-linear scaling of packet transfer durations (2 packets does not equal double the duration of one packet)

3.4.2 shortcomings

- does not really compensate for other bottlenecks, as seen on time-x with gen4 link bandwidth speeds

4 NVML Counters

4.1 Concept

- nvidia has hardware counters, accessible via nvml library - counters measure average bandwidth over the last 20ms, in kb/sec - Transmit and Receive have seperate counters

4.2 Implementation

- method call to read counters takes about 20ms - to increase data granularity and measurement consistency, measuring of TX and RX was done in parallel

4.3 Results

- graphs

4.4 Discussion

4.4.1 successes

- measures bandwidths accurately to some degree - introduces little overhead (probably, still to be measured)

4.4.2 shortcomings

- granularity of method calls prevent more accurate readings -> short memory transfers may be not detected -

5 Link Saturation

5.1 Concept

- if bandwidth is saturated, copy operations should slow down - full duplex, so HtoD and DtoH both need measuring

5.2 Implementation

- Started as a thread that just continuously monitored the counters for set duration of time and printed output into console - Added wrapper and file output in subsequent versions to simplify data-gathering - Added chunk size options for the buffer copy chunks to get as little overhead as possible while getting most consistent data gathering - Delays are measured, no clear correlation between delay and bandwidth - Overhead yet to be properly assessed, however, introduces somewhat significant overhead. TODO: ASSESS OVERHEAD - Measure transmit and receive in the same thread, sequentially

5.3 Results

- graphs - descriptors of graphs

5.4 Discussion

5.4.1 successes

- gives somewhat detailed going-on about PCIe link activities

5.4.2 shortcomings

- introduces some overhead due to occupying PCIe link - Delay compensation sometimes leads to negative values due to delay inconsistencies

6 Summary

Outlook

List of Figures

List of Tables