# Comparison and Analysis of GPU Energy Efficiency For CUDA and OpenCL

Joe Jackson

April 17, 2013

## Abstract

The use of GPUs for processing large sets of parallelizable data has increased sharply in recent years. As the concept of GPU computing is still relatively young, parameters other than computation time, such as energy efficiency, are being overlooked. Two parallel computing platforms, CUDA and OpenCL, provide developers with an interface that they can use to work directly with GPUs. CUDA is designed specifically for NVIDA GPUs, while OpenCL can be used with any GPUs, as well as CPUs and FPGAs. In this paper, we analyze the energy efficiency of the two platforms using large matrix multiplication applications as our basis of comparison. We found that CUDA expends less energy over a shorter time than OpenCL when given the same computational workload.

## 1 Introduction

Green Computing is a term used to describe the push towards environmental sustainability in the IT world; it is "The study and practice of designing, manufacturing, using, and disposing of computers, servers, and associated subsystems... efficiently and effectively with minimal or no impact on the environment" [2, 8]. Motivations behind the advancement of green computing include environmental protection, the rising cost of energy, and increasing demand for power, to name a few. Due to these emerging issues, energy efficiency, previously considered one of the least important factors of a computer's overall efficiency, is beginning to rise to the forefront of computing design.

## 2 Related Work

Przybyla and Pegah were some of the first to emphasize the concept of green computing academically, placing an early emphasis on datacenter power consumption due to the large amount of waste, as well as pointing out the lack of non-economic motivation for companies to convert to green computing [10].

Williams and Curtis expand on this concept by pointing out many of the pitfalls businesses face when converting to green computing practices [14]. Gupta examines various computing environments, such as computer labs and datacenters, with respect to green computing, and offers direction for further work [2]. Li and Zhou point out modeling, energy awareness, and networking as specific areas of concern [8], while Kurp explains steps already taken towards energy efficiency by major technology companies such as Microsoft, Google, and Yahoo [7]. Herrick and Ritschard report on their work with datacenters, emphasizing air flow management and long-term renovation planning [3]. Though not directly involved with GPU energy efficiency, these projects not only demonstrate the amount of effort that has been put towards improving the energy efficiency of data centers and business applications, but also the level of interest being generated in green computing.

With a focus on computer labs, Talebi and Way investigate power consumption with a focus on the comparison of desktop sleep, standby, and hibernate modes [12], while Herrick and Ritschard evaluate the pros and cons of virtualization and thin client use [3]. The two research teams demonstrate a changing focus in the study of green computing from large scale concepts to smaller user groups.

In an effort to establish a more focused direction for green computing research, Li and Zhou suggest that a comprehensive model should be created for measuring the energy efficiency of a computer. Wang and Wang emphasize the importance of focusing on personal computers due to rapidly expanding use of personal computers [13]. Current efforts tend to focus on individual aspects of the computational model [8], but as of yet, there is not an adequate amount of research on individual system components to create an overarching model. The narrow focus of our research will help fill this gap.

Focusing singularly on the CPU (central processing unit), Zhong, etal. examine the benefits of concurrency with a focus on energy [15]. In CPU/GPU (graphics processing unit) comparison, Rofouei, etal. establish that greater energy efficiency is achieved when more parallelizable parts of an application utilize the GPU [11]. Kang, etal. show that GPUs are more energy efficient than CPUs in the case of applications that have a significant number of computationally intensive portions, but may not be more energy efficient in situations with a mixed computation intensive/non-intensive workload [6]. Abe, etal. indicate that a higher CPU clock frequency has little effect on the energy consumption of a system that utilizes a GPU, and that Dynamic Voltage and Frequency Scaling (DVFS algorithms) can be used to significantly reduce GPU energy consumption [1]. These research teams thoroughly investigate many facets of CPU/GPU comparison, but lack the single hardware component focus that our research is based on.

Focusing on the GPU, Huang, etal. present the highly synchronized relationship between GPU performance, energy consumption, and energy efficiency [4]. Jiao, etal. examine GPU memory and core frequencies using OpenCL, finding that when working on computation intensive applications, high core frequencies resulted in greater energy consumption than high memory frequencies

when working on memory intensive applications [5]. GPU-centric research has yet to explore the effects of the application framework on energy efficiency – the goal of our project.

# 3 Theory

Our research compares two different parallel computing platforms, NVIDIA's CUDA and Apple's OpenCL. Parallel computing is the process of carrying out many computation simultaneously, while a platform is a combination of hardware architecture and a software framework. CUDA, the older of the two, was designed to work exclusively on NVIDIA GPUs. Meanwhile, the design focus of OpenCL was the creation of a framework that could interface with CPUs, GPUs, and FPGAs (field-programmable gate arrays) regardless of the hardware developer. We believe that due to OpenCL's focus on portability and CUDA's more specialized field of use, CUDA achieves greater energy efficiency than OpenCL. The basis for our hypothesis is that the developers of CUDA were able to design the platform for their own GPUs, whereas OpenCL's developers may have sacrificed efficiency in some areas in order for OpenCL to be adaptable to various kinds of hardware.

# 4 Methods

In our experiments, we compared the two platforms via large matrix multiplication of two 6144x6144 matrices, the dimensions of which are the largest values we found useable while still being able to fit the matrices into the GPU's global memory and still achieve optimal performance. The CUDA version of the application uses the CUDA 4.3 SDK, while the OpenCL version uses the OpenCL 1.1 API. Both versions run on an Ubuntu Linux 9.10 operating system with NVIDIA driver 310.32. An NVIDIA GeForce 9800 GT graphics card was used as the GPU device for both applications.

Current readings were obtained every .005 seconds using Vernier LabPro current sensors in conjunction with LoggerPro software, while voltage readings were obtained via a Keithley 197A Autoranging Microvolt DMM multimeter. The results of these readings were used to determine power consumption over the course of application execution with the following equation:

$$P = I \cdot V$$

In this equation, P = Power (measured in Watts), I = Current (measured in Amps), and V = Voltage (measured in Volts).

We modified a PCI Express (PCIe) 16x flexible riser cable and a PCIe 6-Pin extension cable to facilitate power measurements from the motherboard to the graphics card and from the computer's power supply to the graphics card, respectively. PCIe is a computer expainsion bus which relays power and information from the motherboard of the computer to peripheral hardware. The

PCIe architecture is two sided, with each side consisting of 82 pins. The PCIe riser cable is an extension of the motherboard's PCIe slots, connecting pins on the graphics card to slots via two sides of 82 wires. For the two sides A and B, Side A wires 2 and 3 run 12V current and wires 9 and 10 run 3V current, while Side B wires 1 - 3 run 12V current and wires 8 and 10 run 3V current.

Prior to our experiment, wires were tested for current and internal connectivity. All 3V wires were found to have no current, indicating that they are unused. This is understandable as PCIe is used for other systems. Side A wires 2 and 3, as well as Side B wire 1 were found to be internally connected, as were Side B wires 2 and 3. This meant that soldering these wires together to gather current and voltage measurements would not disrupt the graphics card's functionality. An example of the riser cable can be found in Figure 1. The PCIe 6-Pin extension cable, which consists of three 12V wires and three grounding wires, was also tested. Wires 1 and 3 were found to be internally connected, while wire 2 was unused.

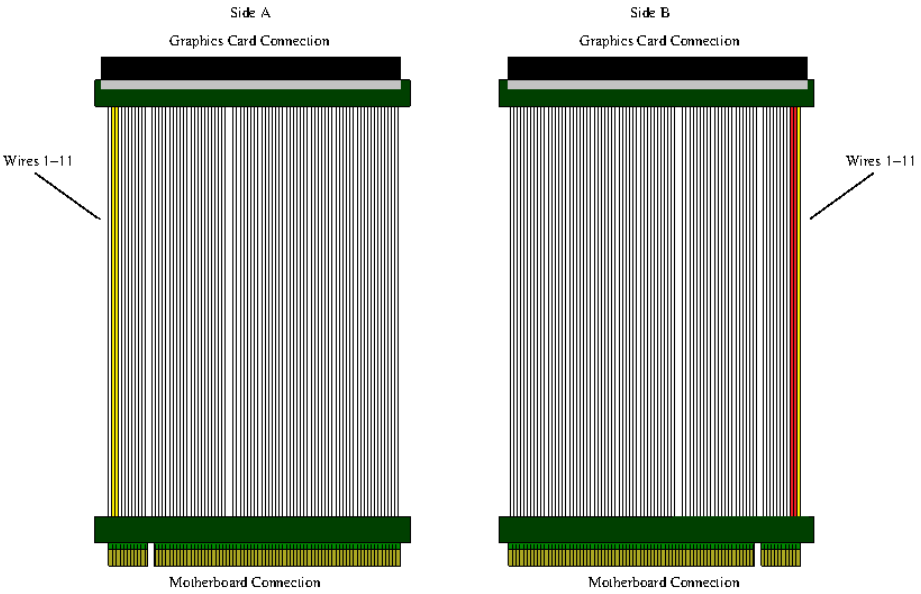


Figure 1: PCIe Riser Cable

# 5 Results

During application execution, voltage measurements stabilized at 11.6V for the external power source and 11.65V for the motherboard power source. Both applications completed computation in between 7.5 and 8 seconds. Over the course of ten trials, CUDA consumed 421.4 W on average, while OpenCL consumed

434.8 W. An example of the current and total power consumption results can be seen in Figures 2 and 3, respectively.

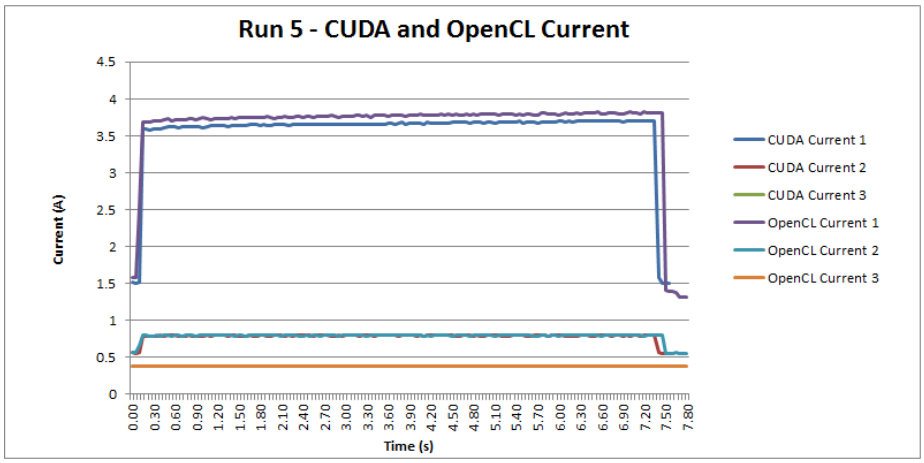


Figure 2: Run 5 Current Measurements for CUDA and OpenCL

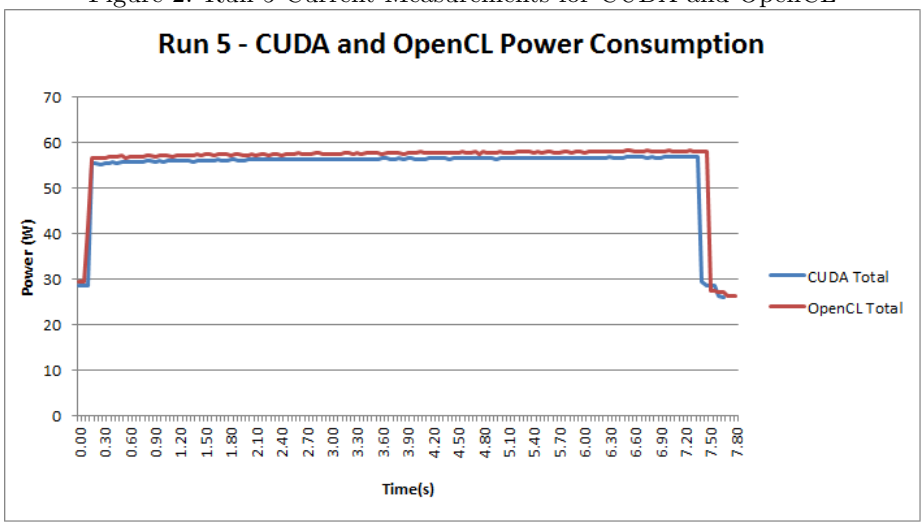


Figure 3: Run 5 Power Consumption for CUDA and OpenCL

A single iteration of the matrix multiplication operation was deemed sufficient, as the variance for CUDA and OpenCL measurements was 3.47 W and 2.27 W, meaning that there was no overlap between the two platforms. While both platforms measured the same voltage, CUDA averaged .11A/s less current than OpenCL. CUDA also consumed 13.43 W less on average than OpenCL. Figure 4 displays the results of the ten trials for both platforms.

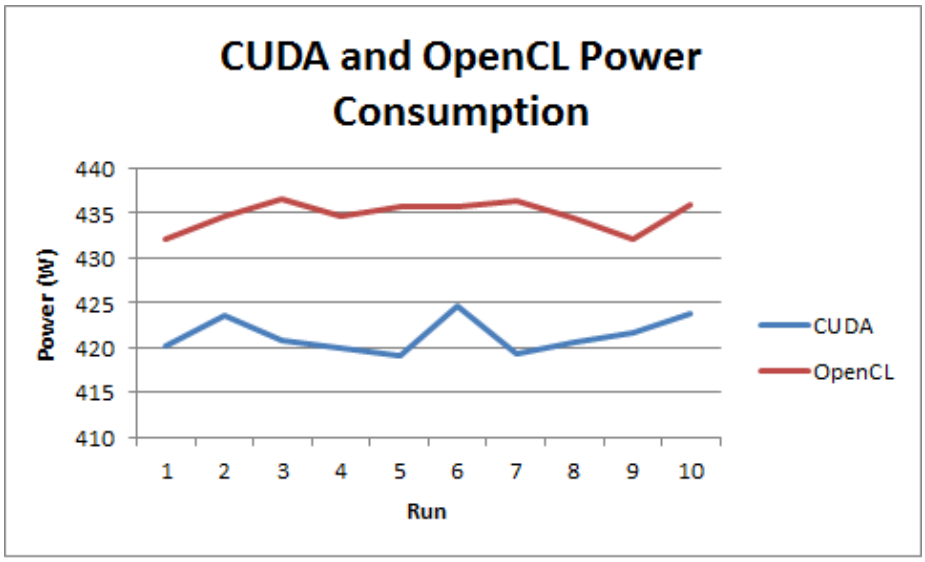


Figure 4: Comparison of CUDA and OpenCL Power Consumption Over Ten Runs

# 6 Analysis

Through our experiments, we found that CUDA consumes less power over less time than OpenCL. However, CUDA's measurements revealed more erratic results than OpenCL, with a power consumption range of 6.9V compared to OpenCL's 4.5V. The power consumption of the two platforms converted to kWh (the U.S. standard for power measurement) is 196.637 kWh for CUDA and 201.337 kWh for OpenCL. To put this difference into perspective, given the U.S. national average price of energy of $0.129 per kWh [9], the use of CUDA rather than OpenCL results in a savings of $0.606 per hour.

The primary source of the differerence in power consumption between the two platforms was the power drawn from the computer's power supply. Power drawn from the motherboard remained fairly identical for both platforms. As not all graphics cards receive extra power from the computer's power supply, these results most likely will not be the same for those cards.

# 7 Conclusion and Future Work

This paper presents our comparison of the CUDA and OpenCL platforms for GPU computation. Using matrix multiplication as our basis for comparison, we were able to determine that CUDA has lower power consumption and runtime than OpenCL when given the same computational workload.

For future work, the extension of other parallelizable functions would be

useful in comparison to our single example. The utilization of newer hardware, corresponding SDKs/APIs, operating systems, and drivers should also be achieved, so as to determine if changes made in the last few years have altered results. An investigation on the effects of system heat may also prove useful, as we did not factor post-execution cool down time into our results.

# References


[1] Yuki Abe, Hiroshi Sasaki, Martin Peres, Koji Inoue, Kazuaki Murakami, and Shinpei Kato. Power and performance analysis of gpu-accelerated systems. In *Proceedings of the 2012 USENIX conference on Power-Aware Computing and Systems*, HotPower '12, 2012.

[2] S. Gupta. Computing with green responsibility. In *Proceedings of the International Conference and Workshop on Emerging Trends in Technology*, ICWET '10, pages 234–236, 2010.

[3] Dan R. Herrick and Mark R. Ritschard. Greening your computing technology, the near and far perspectives. In *Proceedings of the 37th annual ACM SIGUCCS fall conference*, SIGUCCS '09, pages 297–303, 2009.

[4] S. Huang, S. Xiao, and W. Feng. On the energy efficiency of graphics processing units for scientific computing. In *Proceedings of the 2009 IEEE International Symposium on Parallel&Distributed Processing*, IPDPS '09, 2009.

[5] Y. Jiao, H. Lin, P. Balaji, and W. Feng. Power and performance characterization of computational kernels on the gpu. In *Proceedings of the 2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing*, GREENCOM-CPSCOM '10, 2010.

[6] SeungGu Kang, Hong Jun Choi, Cheol Hong Kim, Sung Woo Chung, DongSeop Kwon, and Joong Chae Na. Exploration of cpu/gpu co-execution: from the perspective of performance, energy, and temperature. In *Proceedings of the 2011 ACM Symposium on Research in Applied Computation*, RACS '11, pages 38–43, 2011.

[7] Patrick Kurp. Green computing. *Communications of the ACM*, 51(10):11–13, Oct 2008.

[8] Qilin Li and Mingtian Zhou. The survey and future evolution of green computing. In *Proceedings of the 2011 IEEE/ACM International Conference on Green Computing and Communications*, GREENCOM '11, pages 230–233, 2011.

[9] U.S. Bureau of Labor Statistics. Average energy prices in the los angeles area, March 2013.

[10] David Przybyla and Mahmoud Pegah. Dealing with the veiled devil: eco-responsible computing strategy. In *Proceedings of the 35th annual ACM SIGUCCS fall conference*, SIGUCCS '07, pages 296–301, 2007.

[11] Mahsan Rofouei, Thanos Stathopoulos, Sebi Ryffel, William Kaiser, and Majid Sarrafzadeh. Energy-aware high performance computing with graphic processing units. In *Proceedings of the 2008 conference on Power aware computing and systems*, HotPower'08, 2008.

[12] Mujtaba Talebi and Thomas Way. Methods, metrics and motivation for a green computer science program. In *Proceedings of the 40th ACM technical symposium on Computer science education*, SIGCSE '09, pages 362–366, 2009.

[13] Luyang Wang and Tao Wang. Green computing wanted: Electricity consumptions in the it industry and by household computers in five major chinese cities. In *Proceedings of the 2011 IEEE/ACM International Conference on Green Computing and Communications*, GREENCOM '11, pages 226–229, 2011.

[14] Joseph Williams and Lewis Curtis. Green: The new computing coat of arms? *IT Professional*, 10(1):12–16, Jan 2008.

[15] Benjamin Zhong, Ming Feng, and Chung-Horng Lung. A green computing based architecture comparison and analysis. In *Proceedings of the 2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing*, GREENCOM-CPSCOM '10, pages 386–391, 2010.