Thomas Hobiger et al.: GPU Based Software Correlators - Perspectives for VLBI2010, IVS 2010 General Meeting Proceedings, p.40–44
http://ivscc.gsfc.nasa.gov/publications/gm2010/hobiger1.pdf

GPU Based Software Correlators - Perspectives for VLBI2010

Thomas Hobiger [1], *Moritaka Kimura* [1], *Kazuhiro Takefuji* [1], *Tomoaki Oyama* [2], *Yasuhiro Koyama* [1], *Tetsuro Kondo* [1], *Tadahiro Gotoh* [1], *Jun Amagai* [1]

[1)] *Space-Time Standards Group, National Institute of Information and Communications Technology*
[2)] *National Astronomical Observatory of Japan*

Contact author: Thomas Hobiger, *e-mail:* `hobiger@nict.go.jp`

Abstract

Caused by historical separation and driven by the requirements of the PC gaming industry, Graphics Processing Units (GPUs) have evolved to massive parallel processing systems which entered the area of non-graphic related applications. Although a single processing core on the GPU is much slower and provides less functionality than its counterpart on the CPU, the huge number of these small processing entities outperforms the classical processors when the application can be parallelized. Thus, in recent years various radio astronomical projects have started to make use of this technology either to realize the correlator on this platform or to establish the post-processing pipeline with GPUs. Therefore, the feasibility of GPUs as a choice for a VLBI correlator is being investigated, including pros and cons of this technology. Additionally, a GPU based software correlator will be reviewed with respect to energy consumption/GFlop/sec and cost/GFlop/sec.

1. Introduction

Graphics Processing Units (GPUs) have been undergoing tremendous development in recent years, in order to reduce the burden on the central processing unit (CPU) and enable ultra-fast computation of complex scenarios and output the results on the PC's display. As many of the underlying mathematical algorithms can be parallelized, GPUs have been equipped with hundreds of simple computing cores which can be programmed to solve these equations and output the results to the video buffer.

1.1. CPU vs. GPU

Unlike the CPU, which is equipped with a relatively large cache, GPUs work with smaller cache sizes, assigned to each processing core. Although also the arithmetical units are less sophisticated than those on the CPU, the large number of these cores, which are available for parallel processing, yields a significant speed-up of various applications [4].

1.2. General Purpose GPU - GPGPU

Soon after the first powerful graphics cards entered the market, scientists started to get interested in porting their (parallel) computing problems on the GPU, based on OpenGL. NVIDIA was the first company who provided a sophisticated C-like programming environment named "Computing Unified Device Architecture (CUDA)", and other vendors started to work on similar programming environments for their products. Moreover, as for instance CUDA came with a set of libraries (e.g., FFTs and BLAS) and tools (debuggers and profilers), coding became as simple as writing a C-program, and the number of scientific and technical applications which run on a graphics card has grown rapidly and is expected to increase in the future as well. Additionally, the new OpenCL standard, which realizes a multi-platform programming language, is expected to overcome the problem that code is currently only usable for a certain device.

2. GPUs for VLBI

GPUs have been used successfully within the Murchinson Widefield Array post-processing pipeline [3], and tests, which utilize the graphics cards as a software correlator, were made as well [5]. Harris et al. [1] have shown that GPUs can help to speed-up signal convolution, and NICT has developed a GPS software receiver running on a single off-the-shelf graphics card [2].

2.1. Benchmarking the GPU Concerning the Possibility as an FX Correlator

In order to reveal the potential of the GPU as a co-processer on which a software correlator could run, we start with benchmarking an off-the-shelf graphics card (specifications listed in Table 1). As it turned out very soon that an XF-type correlator does not perform well on a parallel architecture, we will focus only on the FX-type implementation of the software correlator. Thereby, we can assume that for each FX engine

$$10 \cdot N \cdot \log_2(N) + 6 \cdot N \qquad (1)$$

floating point operations are being carried out to compute the cross-spectrum of two signals with length N. Thus, measuring the time taken on the GPU for calling such an FX engine in different configurations allows one to draw conclusions on the performance of the GPU. Figure 1 depicts the FX performance in Gflops for different FFT sizes as well as varying configurations of how these engines are called on the graphics card. As expected, the parallel computing power of the graphics card cannot be used when the FX engines are called serially, yielding more than 10 GFlops only for FFT sizes larger than 4096. When the FX engines are called in parallel, one can start taking advantage of the GPU and achieve more than 100 GFlops for FFT sizes larger than 256. Since the amount of shared memory on the GPU is restricted to a few kilobytes per thread, performances saturate as soon as this limit is reached.

Table 1. Specifications of the NVIDIA GTX 280 card used for the benchmark test.

Processor Cores	240
Processor Clock	1296 MHz
Memory	1 GB GDDR3
Price	∼ 300 $
Power consumption	200 W

2.2. Data-transfer CPU ↔ GPU

In order to make sure that neither the data-transfer from the CPU memory to the GPU nor the way back turns out to be a severe bottleneck, tests were carried out to determine the transfer

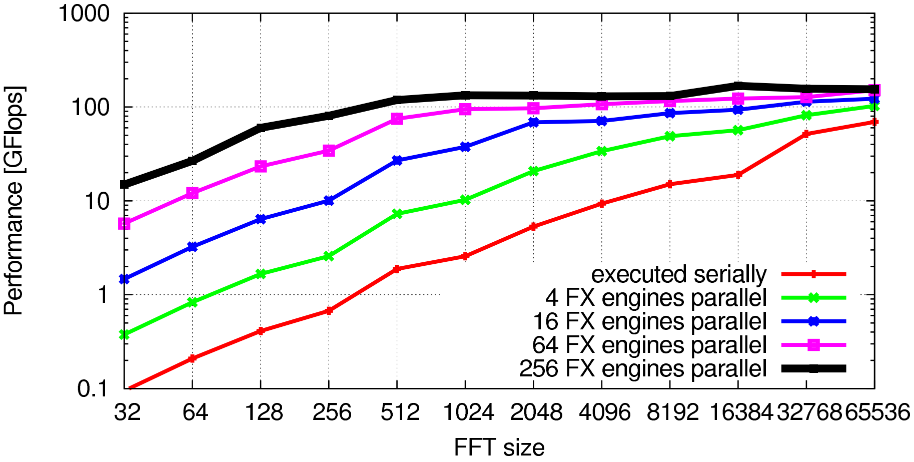

Figure 1. FX performance on the GPU.

rates. Figure 2 summarizes the results obtained for different data sizes. For small blocks of data (i.e., from 1 KB to 100 KB) moderate transfer rates are observed, which have the potential to slow down any application. But as the FX engines of a VLBI correlator are expected to be fed with rather large amounts of data, one can be sure that data sizes larger than 1 MB will be used. Thus, for such data sizes, more than 5 GB/sec can be achieved in both directions, which appears to be fast enough to not be a severe bottleneck of the correlator. Moreover, modern graphics cards recently even support asynchronous data-transfer, which allows running the code while data is transferred to/from the device.

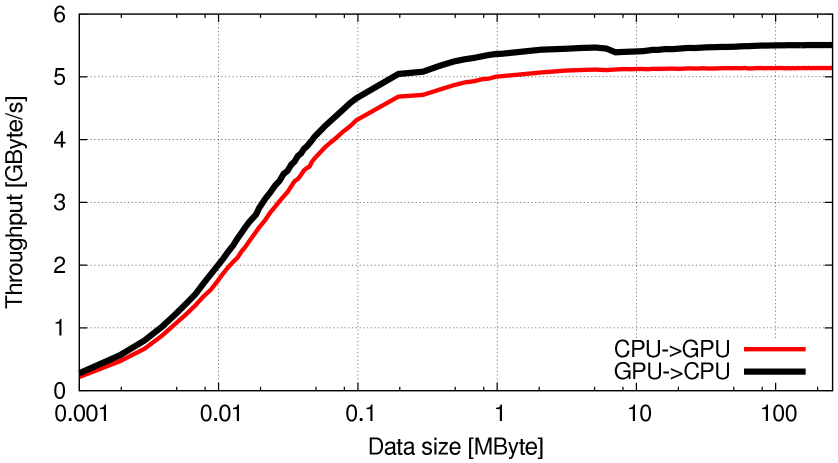

Figure 2. Data transfer rates to/from the GPU.

3. Implementing the Correlator

Based on two off-the-shelf graphics cards (NVIDIA GTX295) for a price of less then 800 US$ a complete software correlator was implemented. The correlator carries out the following tasks:

1. data-transfer from the CPU to the GPU,
2. unpacking of the data,
3. fringe stopping,
4. FFT,
5. delay tracking,
6. correlation and integration.

All modules are implemented on the GPU, which is expected to be a little bit slower as compared to the tests with the FX engine only (prior section). Test runs with recorded data (1024 Msps, 2 bits) were made in single- (i.e., auto-correlation) and multi-station mode. The results, showing the performance in Msps, are displayed in Figure 3. Thereby, it turns out that four stations (six baselines) can be correlated in real-time, when a sampling rate of 1 Gsps is chosen. Moreover, eight stations with a sampling rate of 500 Msps can be processed as well in real-time, when slightly larger FFT sizes are selected.

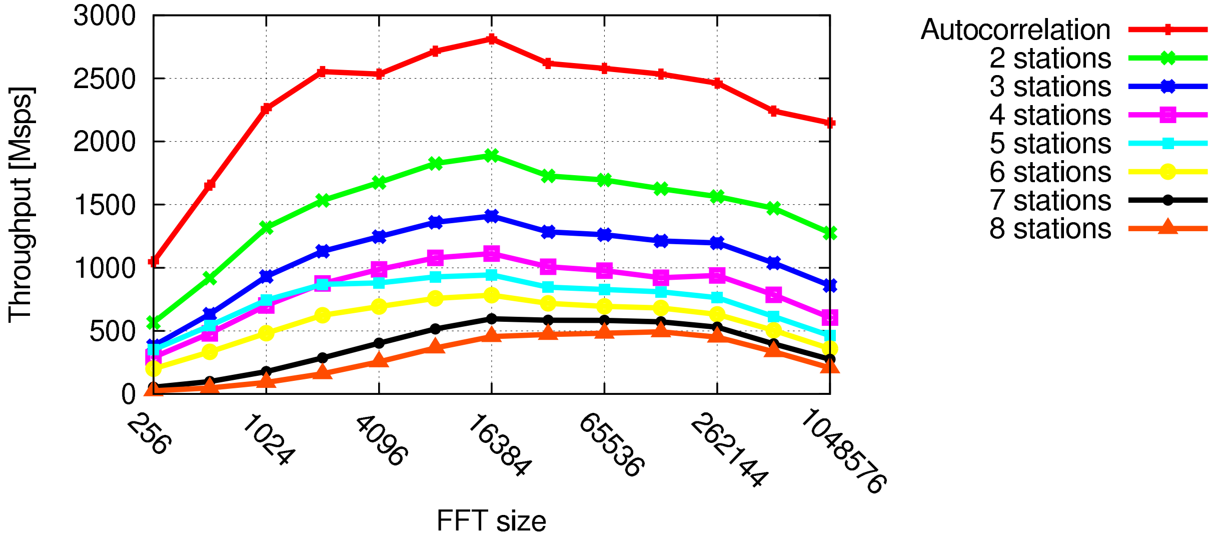

Figure 3. Data transfer rates to/from the GPU.

4. Summary

Graphics processing units seem to be an alternative to CPU-based software correlators. Considering that vendors provide simple programming environments, there appears to be no real hurdle in using these devices. Table 2 compares the graphics cards described in Section 2.1 w.r.t. to a modern CPU. Additionally, data-transfer, as discussed in Section 2.2, should not be a severe restriction for the usage of graphic cards as VLBI correlators.

Table 2. Comparison of the GPU w.r.t. a CPU.

	GPU	CPU (3.6 GHz Pentium 4)
Perf.	∼ 200 Gflops (unoptimized)	12 Gflops (based on best FFTW score)
Cost	∼ 0.5 Gflops/$	∼ 0.1 Gflops/$
Energy	∼ 0.8 Gflops/Watt	∼ 0.2 Gflops/Watt

5. Outlook

The next generation of GPUs, which will have twice as many computing cores as current devices, will enter the market in April 2010. Moreover, the new cards will have larger shared memory blocks as well as L2 cache, which will help to improve the speed of the FFTs and other highly parallel operations. Thus, given that the developmental progress of GPUs continues to be faster than those of the CPU, the graphics card appears to be a really strong competitor for the realization of a software correlator, running on off-the-shelf components.

Acknowledgements

Parts of this work were supported by a Grant-in-Aid for scientific research (KAKENHI, No. 24241043) from the Japan Society for the Promotion of Science (JSPS).

References

[1] Harris, C., K. Haines, L. Staveley-Smith, GPU accelerated radio astronomy signal convolution. Exp Astron, 22, 1-2,129–141, 2008.

[2] Hobiger, T., T. Gotoh, J. Amagai, Y. Koyama, T. Kondo, A GPU based real-time GPS software receiver, GPS Solutions, 14, 2, 207–216, 2010.

[3] Ord, S., L. Greenhill, R. Wayth, D. Mitchell, K. Dale, H. Pfister, R. Edgar, GPUs for data processing in the MWA, ADASS XVIII, ASP Conference Series, 411, 127-130, 2009.

[4] Nguyen, H., GPU Gems 3: programming techniques for high-performance graphics and general-purpose computation, Addison-Wesley Professional, 2007.

[5] Wayth, R.B., L. Greenhill, F. Briggs. A GPU-based Real-time Software Correlation System for the Murchison Widefield Array Prototype, Publications of the Astronomical Society of the Pacific, 121, 857-865, 2009.