CO227::Computer Engineering Project – Report

GPU Based Parallel Audio Processing System

Milinda Sandaruwan (E/13/315)

Member’s Name (E/13/073) Shehan Eranga (E/13/377)

****

**http://www.ce.pdn.ac.lk**

Department of Computer Engineering

Faculty of Engineering

University of Peradeniya

Peradeniya 20200

Sri Lanka

**Abstract**

Graphic processing units have been recently used to perform verities of computational tasks including audio processing.

It shows promising results for real time applications due to nature of its parallelism. In musical industry audio processing mainly done by hardware related implementations. Also there can be found software implementations that gives real-time performance for certain extend. Equalization can be found as one of major audio processing technique used in sound industry that needs real time performance. This paper discuss how to implement real time equalization GPU and how to extend this technique to parallel signal equalization. These tasks have been tested on NVidia GeForce graphic card and high performance NVidia tesla GPU. Furthermore this paper discuss how much signal parallelism can be achieved for processing audio signals for different sampling rates in real time.

**CONTENTS**

Abstract i

Contents ii

List of figures iii

List of tables iv

**CHAPTER 1** INTRODUCTION  **1**

**CHAPTER 2** RELATED WORK  **2**

**CHAPTER 3** OBJECTIVES AND SCOPE  **4**

**CHAPTER 4** METHODOLOGY **5** 4.1 OVERVIEW 5

4.2ACHIEVEMENTS 6

4.2.1ANOTHER SUB HEADING 6

4.2.2ANOTHER SUB HEADING 6

**CHAPTER 5** CONCLUSION **7**

References  **8**

Figure 1.2: Second figure name

Table 2.1 First table name

**LIST OF FIGURES**

Figure 1.1 First figure name 1

Figure 1.2 Second figure name 1

**LIST OF TABLES**

Table 2.1 First table name 2

Table 2.2 Second table name 2

**CHAPTER 1**

**INTRODUCTION**

**Introduction to GPU Computing**

Getting use of high-level languages like C/C++ , Graphics Processing Unit (GPU) based applications run the sequential part of the program on the Central Processing Unit (CPU), with single threaded performance while parallel computations done on GPU with multi-threaded performance. This type of computing is called GPU Computing.

The main difference between GPU and CPU is that the CPU consists of a few cores optimized for sequential processing, and the GPU contains smaller but more efficient cores for parallel processing.

Generally GPU was used for games or 3D game rendering. But recently GPU is used for Parallel audio processing, financial modelling, Accelerated Graph Analytics, Circuit Simulation, Chemical Kinetics etc.

**Real time parallel audio processing using GPU**

Today a typical CPUs are capable of supporting real-time audio applications, but there are many limitations. Requirement of high quality real-time parallel audio processing is ricing for large musical events such as orchestra.

Audio processing technique such as equalization (link) is one of major audio processing technique used in musical industry. When the sampling rates of the audio signals are getting higher because of the high quality requirement (I.e. [Direct Stream Digital](https://en.wikipedia.org/wiki/Direct_Stream_Digital) sampling rate: 2,822,400 Hz) the audio processing system need higher performance to calculate everything in real time. Since Equalization can be done parallel, GPU can give a better performance.

In our research we first evaluate the performance between CPU and GPU using optimized algorithms to conclude that, how much real-time Processing power can be achieved using both CPU and GPU. For that we use single audio sample equalization algorithm implemented using MATLAB and CUDA C.

For the second part of our research we talks about Parallel audio signal processing using GPU. There we talk about how much number of audio signal inputs can be equalize parallel using GPU. We are going to test the CUDA C implementation on High performance Tesla GPU and typical laptop NVidia GeForce GPU. In this evaluation we want to conclude that parallel audio signal Processing can be done on a typical laptop GPU with high quality sampling rates like [Direct Stream Digital](https://en.wikipedia.org/wiki/Direct_Stream_Digital) (328000Hz), [Digital extreme Definition](https://en.wikipedia.org/wiki/Digital_eXtreme_Definition) (124455Hz), in real time.

**CHAPTER 2**

**RELATED WORK**

On the research paper entitled Real-time adaptive algorithms using a Graphics Processing Unit written by Jorge Lorente, Miguel Ferrer in 2012 discussed about adaptive channel identifier algorithms implementation in CUDA. They mentioned that the parallelism must be there in the algorithms to implement them using CUDA. In their research adaptive algorithms work sample by sample and they had to implement those algorithms using block algorithms. Furthermore their results conclude that GPU is viable for real -time adaptive applications.

On the paper HIGH-PERFORMANCE REAL-TIME FIR-FILTERING USING FAST CONVOLUTION ON GRAPHICS HARDWARE by Frank Wefers, Jan Berg in 2010 discussed about implementation of filtering on GPU. They mentioned about fast convolution algorithms for audio rendering of complex senses. They talks about the importance of data structures to implementation of fast furrier transformation in GPU. They mentioned about functionalities provided by CUDAFFT libraries. In our research we also used the functions that provided by the CUDAFFT libraries. Such as CUDA R2C, C2R, cudaplanmany (). They also did a performance evaluation on data transfers between CPU and GPU. They mentioned that the "Only for larger data sizes, runtimes vs data size scale nearly linear". Graphs that represent the data clearly indicate that statement. Also they evaluate the BATCHED FFT with CPU and GPU. They mentioned about FFTW library that used to perform furrier transform in CPU. They discussed about the potential for parallelism for higher BATCH sizes. This fact is also related to our research including parallel audio signal equalization. Finally they mentioned that they had achieved 44% computational power using GPU. This fact also proves that GPU computations can achieve higher performance.

**CHAPTER 3**

**OBJECTIVES AND SCOPE**

**CHAPTER 4**

**METHODOLOGY**

**4.1.1 AUDIO SAMPLING**

A typical signal processing system is shown below.



Analog signals replicate the sound wave as it is but digital audio signal is constructed by geting samples at spesified rate called sampling rate. By Nyquist sampling therorem, ---eqn--

---figure wiki

The quolity of the audio signal depends on the sampling rate,or bit rate. Sampling rate defined as "the number of samples of audio carried per second". Bit rate defined as "the number of bits that are conveyed or processed per unit of time" .

|  |  |
| --- | --- |
| Sampling Frequency | Audio Quality |
| 44,100 Hz | Audio CD |
| 96,000 Hz | DVD Audio |
| 192,000 Hz | HD DVD |
| 352,800 Hz | Super Audio |
| 2,822,400 Hz | Direct Stream Digital (DSD) |
| 5,644,800 Hz | Double-Rate DSD |

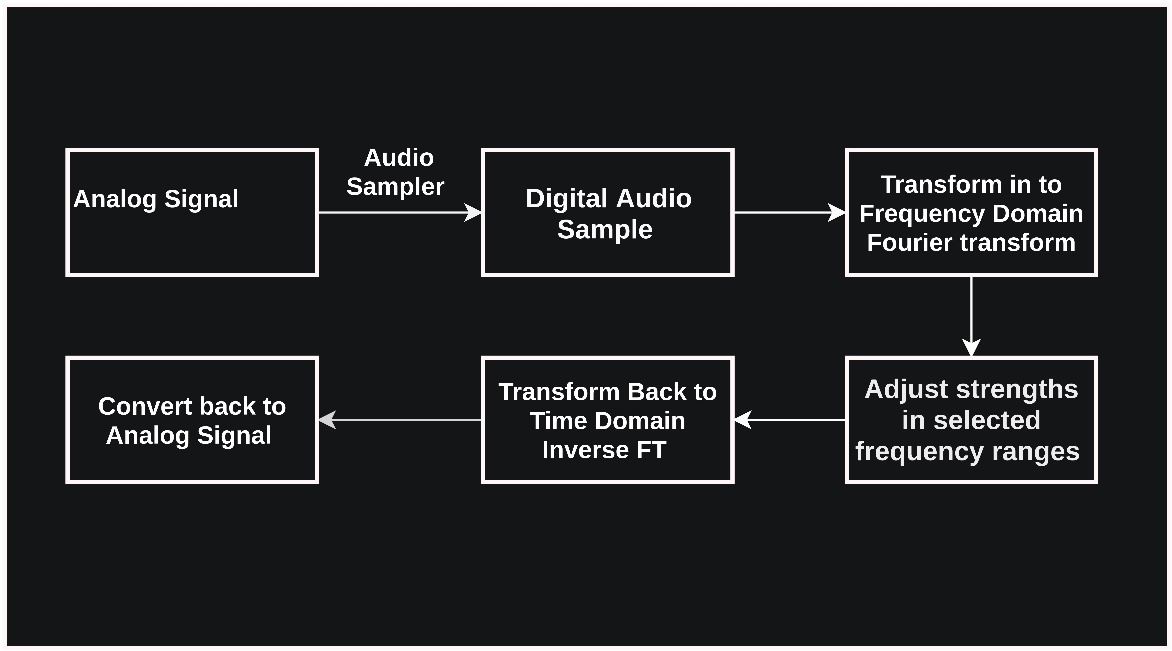
**4.1.2 EQUALIZATION**

General purpose of audio equalization is to make individual instruments and voices sound better.

It involves mixing instrumental and voices sound and boosting bands of frequencies with respect to other bands of frequencies.

In group of instruments play together various notes and harmonics of instruments blend together. This blended frequencies may or may not be in human hearing range (20 -20000).

Every frequency must be treated equally to get a better sound that please the ear. In that scenario equalization comes to picture.



**4.2 ACHIEVEMENTS**

**CHAPTER 5**

**CONCLUSION**

**REFERENCES**

[1] https://en.wikipedia.org/wiki/Sampling\_(signal\_processing)#Sampling\_rate

Accessed on 4th of December 2016 at 10.20 p.m.

[2] Sampling Theory For Digital Audio By Dan Lavry, Lavry Engineering, Inc.

[3] INTRODUCTION TO Signal Processing Sophocles J. Orfanidis Rutgers University

[4] 1999-2002 by Sanjeev R. Kulkarni.Lecture Notes for ELE201 Introduction to Electrical Signals and Systems.

[5] Audio Equalizer Ohio State University Department of Electrical and Computer Engineering March 2009 By Betty Lise Anderson

[6] CUDA FFT Documentation

http://docs.nvidia.com/cuda/cufft/#axzz4RtDvBg2r

Accessed on 6th of December 2016 at 01.18 p.m.

[7] CUDA occupancy

http://docs.nvidia.com/cuda/cuda-c-best-practices-guide/#axzz4RtDvBg2r

Accessed on 4th of December 2016 at 10.20 p.m.

[8] MATLAB FFT ()

https://in.mathworks.com/help/matlab/ref/fft.html

Accessed on 6th of December 2016 at 01.18 p.m.

[9] Programming Massively Parallel Processors

[10] Tutorial on GPU computing With an introduction to CUDA Felipe A. Cruz Tutorial on GPU computing With an introduction to CUDA University of Bristol, Bristol, United Kingdom.