PERFORMANCE EVALUATION OF OFDM BASED WIRELESS COMMUNICATION SYSTEMS USING GRAPHICS PROCESSING UNIT (GPU) BASED HIGH PERFORMANCE COMPUTING.

A Thesis submitted in partial fulfillment of the Requirements for the degree of

Master of Technology
In
Electronics and Communication Engineering
Specialization: Communication and Networks

By
SANGEETA BHATTACHARJEE

Roll No. : 212EC5170

Department of Electronics and Communication Engineering
National Institute of Technology Rourkela
Rourkela, Odisha, 769 008, India
May 2014

PERFORMANCE EVALUATION OF OFDM BASED WIRELESS COMMUNICATION SYSTEMS USING GRAPHICS PROCESSING UNIT (GPU) BASED HIGH PERFORMANCE COMPUTING.

A Thesis submitted in partial fulfillment of the Requirements for the degree of

Master of Technology
In
Electronics and Communication Engineering
Specialization: Communication and Networks

By
Sangeeta Bhattacharjee
Roll No. : 212EC5170

Under the Guidance of
Prof. Sarat K. Patra

Department of Electronics and Communication Engineering
National Institute of Technology Rourkela
Rourkela, Odisha, 769 008, India
May 2014

This thesis is dedicated to my parents

for their immense love, support

and encouragement.

DEPT. OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA

ROURKELA – 769008, ODISHA, INDIA

Certificate

This is to certify that the work in the thesis entitled **Performance Evaluation Of OFDM Based Wireless Communication Systems Using Graphics Processing Unit (GPU) Based High Performance Computing** by **Sangeeta Bhattacharjee** is a record of an original research work carried out by her during 2013 - 2014 under my supervision and guidance in partial fulfillment of the requirements for the award of the degree of Master of Technology in Electronics and Communication Engineering (Communication and Networks), National Institute of Technology, Rourkela. Neither this thesis nor any part of it, to the best of my knowledge, has been submitted for any degree or diploma elsewhere.

Place: NIT Rourkela

Date: 2nd June 2014

Dr. Sarat Kumar Patra
Professor,
Department of Electronics and Communication, NIT Rourkela

DEPT. OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA

ROURKELA – 769008, ODISHA, INDIA

Declaration

I certify that

a) The work contained in the thesis is original and has been done by myself under the general supervision of my supervisor.

b) The work has not been submitted to any other Institute for any degree or diploma.

c) I have followed the guidelines provided by the Institute in writing the thesis.

d) Whenever I have used materials (data, theoretical analysis, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references.

e) Whenever I have quoted written materials from other sources, I have put them under quotation marks and given due credit to the sources by citing them and giving required details in the references.

Sangeeta Bhattacharjee

2nd June 2014

ACKNOWLEDGEMENTS

I express my deep sense of gratitude and indebtedness to Dr. Sarat Kumar Patra, Professor, Department of Electronics and Communication Engineering, NIT Rourkela for his invaluable guidance, inspiration and encouragement at all stages of the project work. His priceless advice, fantastic stamina and day-to-day monitoring of every minute detail were a constant source of inspiration for me.

I would also like to thank Mr. Brian Fanous, the primary developer of MATLAB GPU computing toolbox, for attending to my query and showing me the right path for programming the GPU using MATLAB.

I would like to express my respect and gratitude to the faculty and staff members of ECE department, NIT Rourkela for their generous help in various ways for the overall completion of this project. I would also like to thank the Ph.D. scholars Mr. Prashanta Kr. Pradhan and Mr.Pallab Maji, department of Electronics and Communication Engineering, NIT Rourkela, for their selfless support and guidance whenever I faced any problem related to the project work.

Last but not the least, I would like to express my love, respect and gratitude to my parents, who have always encouraged me and also provided me with all the logistics to make my stay at NIT Rourkela an unforgettable and rewarding experience.

SANGEETA BHATTACHARJEE

sangeeta.bhatta@gmail.com

ABSTRACT

Wireless communication is one of the fastest developing technologies of current decade. Achieving high data rate under constrained condition demand sophisticated signal processing algorithms which in turn demand complex computational processing. Modern wireless communication techniques using OFDM demand substantial computational resources for implementation. An OFDM system with 2048 subcarriers typically requires a 2048 point IFFT for transmission and 2048 point FFT for reception. When signal processing techniques like PAPR, pre-equalization, equalization, pilot carrier insertion are implemented, the complexity increases considerably. This large complexity demands use of high performance computing systems for efficient implementation. This primary aim of this project was to take up this investigation.

Rapid growth in computing and communications technology has led to the proliferation of powerful parallel and distributed computing paradigm leading to innovation in high performance computing and communications (HPCC). In this project, the performance of advanced wireless communication algorithms on Graphics Processing Unit (GPU) based high performance computing hardware has been evaluated. The computationally expensive multi-carrier wireless communication systems along with associated signal processing techniques have been implemented on GPU with an aim to reduce computation time. This project proposes the use of GPU architecture for efficient implementation of Long Term Evolution (LTE) Physical Layer, Multiple Input Multiple Output (MIMO) OFDM system and Partial Transmit sequence (PTS) technique for Peak-to-Average Power Ratio (PAPR) reduction in OFDM system. The implementation of this new method is expected to provide promising ways to implement complex wireless communication systems using GPU based computing hardware.

CONTENTS

ABBREVIATIONS

1xEV-DV	:	Evolution Data/Voice
3G	:	3rd Generation
3GPP	:	3rd Generation Partnership Project
4G	:	4th Generation
ADC	:	Analog to Digital Converter
ADSL	:	Asymmetric digital subscriber line
AWGN	:	Additive white Gaussian noise
BER	:	Bit Error Rate
BPSK	:	Binary Phase Shift Keying
BTS	:	Base Transmit Station
CCDF	:	Complementary Cumulative Distribution Function
CP	:	Cyclic Prefix
CPU	:	Central Processing Unit
CUDA	:	Compute Unified Device Architecture
DAB	:	Digital Audio Broadcasting
DAC	:	Digital to Analog Converter
dB	:	Decibel
DDR	:	Double Data Rate
DFT	:	Discrete Fourier Transform
DP	:	Double Precision
DV	:	Digital Video
DVB	:	Digital Video Broadcasting

ECC	:	Error Correction Code
E-UTRA	:	Evolved Universal Terrestrial Radio Access
FDD	:	Frequency Division Duplex
FFT	:	Fast Fourier Transform
FLOPS	:	Floating-point Operations Per Second
FPGA	:	Field Programmable Gate Array
GB	:	Gigabytes
GHz	:	Giga Hertz
GPU	:	Graphics Processing Unit
HIPERLAN	:	High Performance Radio Local Area Network
HPC	:	High Performance Computing
HSDPA	:	High-Speed Downlink Packet Access
IDFT	:	Inverse Discrete Fourier Transform
IFFT	:	Inverse Fast Fourier Transform
IQ	:	In-phase and Quadrature
ISI	:	Inter-symbol Interference
LTE	:	Long Term Evolution
MC	:	Multi-carrier
MCCDMA	:	Multicarrier Code Division Multiple Access
MFLOPs	:	Mega Floating-Point Operations Per Second
MHz	:	Mega Hertz
MIMO	:	Multiple Input Multiple Output
MIPs	:	Million Instructions per Second
MRRC	:	Maximal Ratio Receive Combining
OFDM	:	Orthogonal Frequency Division Multiplexing

OFDMA	:	Orthogonal Frequency Division Multiple Access
PA	:	Power Amplifier
PAPR	:	Peak to Average Power Ratio
PRB	:	Physical Resource Blocks
PS3	:	PlayStation 3
PTS	:	Partial Transmit Sequence
QAM	:	Quadrature amplitude modulation
QPSK	:	Quadrature Phase Shift Keying
RAM	:	Random Access Memory
RF	:	Radio Frequency
SC-CDMA	:	Single-Carrier Code Division Multiple Access
SC-FDMA	:	Single Carrier Frequency Division Multiple Access
SDRAM	:	Synchronous dynamic random access memory
SER	:	Symbol Error Rate
SIMT	:	Single Instruction Multiple Thread
SNR	:	Signal to Noise Ratio
SP	:	Single Precision
STBC	:	Space–Time block codes
UMTS	:	Universal Mobile Telecommunications System
V-BLAST	:	Vertical-Bell Laboratories Layered Space-Time
WCDMA	:	Wideband Code Division Multiple Access
Wi-Fi	:	Wireless Fidelity
WiMAX	:	Worldwide Interoperability for Microwave Access
WLAN	:	Wireless Local Area Network

LIST OF FIGURES

LIST OF TABLES

1 ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING IN WIRELESS COMMUNICATION: AN INTRODUCTION

The growing demand for services with high data rates and high spectral efficiency is the key to rapid technological evolution in the field of wireless communication. In the last two decades wireless communication has experienced a massive growth with a mission to provide new services with high data rates. Many new wireless systems have been gradually introduced which include 2^{nd}, 3^{rd} and 4^{th} generation mobile systems as well as Wi-Fi (IEEE 802.11a/b/g/n), WiMAX (IEEE 802.16), LTE, MC-CDMA, SC-CDMA [1]. This revolution in the field of wireless communication is being caused by continuous technological breakthrough to enhance better transmission using signal processing algorithms. The new techniques which are being developed are gradually being incorporated in commercial products and new wireless communications standards are being proposed. Recently, Third generation (3G) and fourth generation (4G) mobile communication systems have been deployed commercially at many places to fulfil the need for packet-based services with high data rate. Moreover lot of advancements has been incorporated in 3G systems to improve the existing data rates. Some of these include- high speed downlink packet access (HSDPA) in wideband code division multiple access (WCDMA) systems, 1x evolution-data, 4G, MIMO-OFDM, MC-CDMA, etc. [2].

But the 3G systems are unable to cope up with the growing demands for wireless multimedia services over the broadband networks. Hence next generation wireless communication systems which include 4G and beyond are being standardized even before the complete deployment of 3G systems in all parts of the world. The next generation wireless systems are expected to support much higher data rates than the existing system.

With the increased demand for higher data rate services such as voice, data, video and multimedia over wired and wireless networks, new baseband processing techniques are required to process the huge amount of data in a less time. These techniques must be able to provide high data rate at permissible bit error rate (BER), and minimum delay. Orthogonal Frequency Division Multiplexing (OFDM) in conjunction with multiple antennas (MIMO-OFDM) is one of such technology expected to provide desired service standards [3, 4]. The first commercial OFDM based system was Digital Audio Broadcasting (DAB) standards developed in 1995. Henceforth, OFDM has been adopted as the technology for some of the most promising standards of wireless industry. Immediately following the development of DAB standards, the European Digital Video Broadcasting (DVB) standards came up which utilized OFDM as the main technology. Following these standards, OFDM was taken up as the technology for wireless LAN (Wi-Fi) with the protocol IEEE 802.11a being established. It was then followed by IEEE 802.11g WLAN which also used OFDM. Currently the most used protocol IEEE 802.11n uses OFDM as the base technology. The IEEE 802.16 standard commonly known as WiMAX uses OFDM coupled with MIMO system. OFDM has been proposed as the principal modulation scheme in 4G communication. [5].

1.1 OFDM Transceiver

OFDM systems transmit multiple parallel low bandwidth channels of data through a wideband channel. This technique achieves high data rate providing transmission using low

bandwidth sub channels within the allocated channel. Here multiple orthogonal subcarriers are used to transmit multi sub channels. The more the number of sub-carriers the better will be the immunity to the frequency selective fading of signals and similarly higher will be the data-rates. However a complex architecture with large number of oscillators and filters are required to implement an OFDM system in hardware [6].

The complexity of an OFDM system was solved by Weinstein and Ebert who proposed implementing OFDM modulation by IDFT and demodulation by DFT [7]. The implementation of OFDM transmission and reception using DFT and IDFT would however have very high complexity as DFT involves N^2 complex multiplications. Thus with the increase in the number of subcarriers, the complexity increases exponentially. This problem has been resolved by the implementation of OFDM transceiver using fast-Fourier-transform-algorithms. Thus OFDM transmitter is implemented by an IFFT block and receiver by a FFT block. This implementation was aided by the improved VLSI implementation of various fast algorithms of FFT/IFFT. The number of sub-carriers could thus be increased without much increase in hardware requirement [8].

1.1.1 OFDM Transmitter

The input serial data is mapped into constellation symbols using a modulation scheme and grouped into a block. The block of data is split up into parallel data streams using serial to parallel converter (S/P), with the number of elements in one parallel block of size, say M, where M is the number of subcarriers. The parallel block of data is then passed through an N-point IFFT block (zero padding is done if M<N) to obtain the OFDM symbol. The obtained OFDM symbol is in digital time domain. Guard time is introduced between two successive OFDM symbols in the form of cyclic prefix to prevent ISI due to channel dispersion. The digital data is then converted to real time waveform using digital-

to-analog converter (DAC). Baseband signal is up-converted to appropriate RF pass-band with IQ mixer or modulator. The transmitter is illustrated in Figure 1-1 [2, 3].

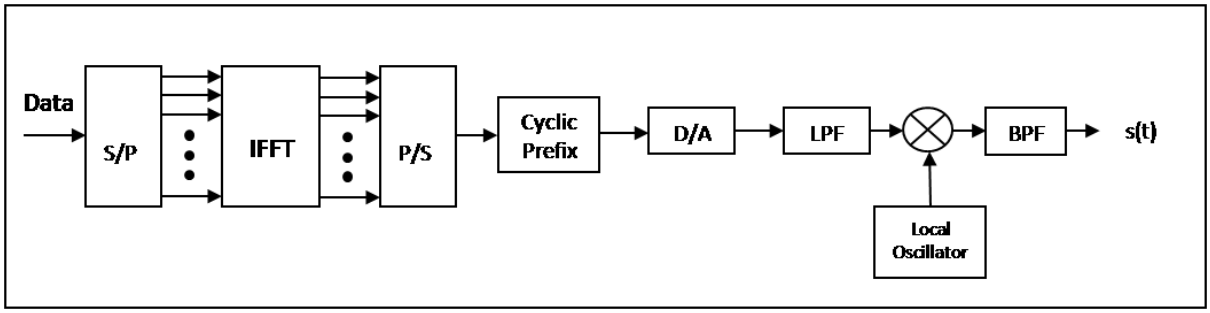

FIGURE 1-1: OFDM TRANSMITTER USING IFFT

1.1.2 OFDM Receiver

OFDM signal is down-converted to baseband with an IQ demodulator. The analog signal is sampled and quantized using an ADC. Demodulation is done by performing DFT. Baseband signal processing is done to recover the data. The receiver is illustrated in Figure 1-2 [3].

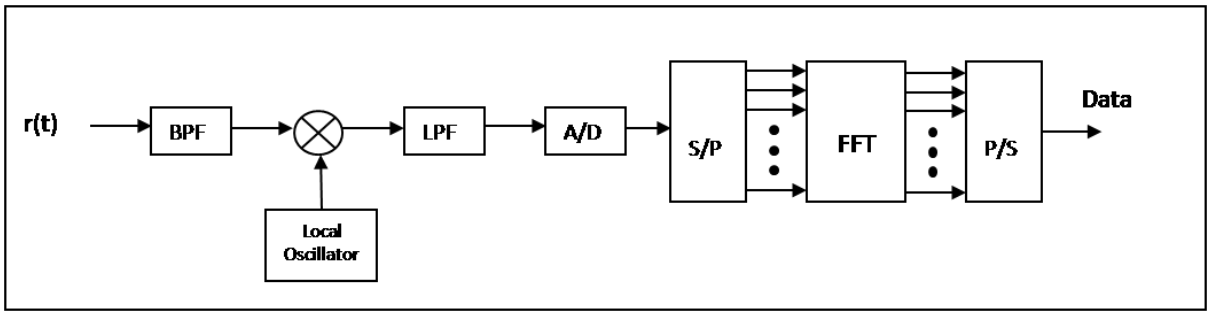

FIGURE 1-2: OFDM RECEIVER USING FFT

1.2 Motivation

Implementation of OFDM requires use of Inverse Fast Fourier Transform (IFFT) and Fast Fourier Transform (FFT) processing at the transmitter and the receiver respectively, this is presented in Figure 1-3. This ensures orthogonality of subcarriers. The IFFT and FFT becomes the most critical module in transmitter and receiver. They also constitute the most computationally intensive part of transceiver design [6].

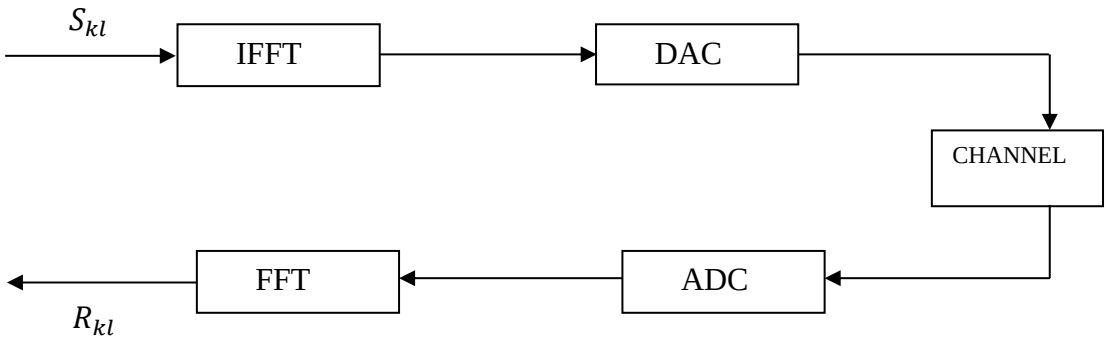

FIGURE 1-3: OFDM IMPLEMENTATION USING FFT

It is possible to achieve higher throughput with minimum penalty in area and power while computing IFFT and FFT. These are achieved using parallel and pipeline processing architecture.

Advanced wireless techniques using OFDM require other forms of signal processing for proper utilization of available bandwidth. These techniques include-

Pre-equalization

Equalization

Peak-to-average power (PAPR) reduction

Pilot carrier insertion

Phase rotation of independent carrier

Implementing signal processing algorithms on all subcarriers of an OFDM system demands computationally complex operation and these pose a limitation on the system performance. Optimization of each of these parameters for different form of wireless communication applications like WLAN, WiMax, MC-CDMA, LTE, 3GPP, etc. makes the task even more complex [1].

Graphics processing unit (GPU) based computing is a recent paradigm in computational research [9]. In recent years GPU has evolved as general processing technology allowing users to compute large array of parallel data process using an array of low complexity processors [10]. The aim of the project is to leverage the power of the GPU processor for

modern wireless communication technology. The objective of this project is to implement and test a few signal processing techniques in modern wireless communication system based on OFDM.

1.3 Objective of the work

The primary objective of this work is to develop computationally efficient algorithms for OFDM based communication systems by providing parallel implementation of FFT/ IFFT and also the complex multiplications using a GPU with an aim to reduce the computation time. The reduced computation time leads to higher system throughput. Moreover computation in GPU helps to reduce power consumption which in turn reduces the overall cost of the system. To realize this objective, following investigations and analysis were undertaken in this project:

- Implementation of fundamental OFDM system and its performance analysis under GPU hardware
- Implementation of MIMO-OFDM system and its performance evaluation under GPU hardware.
- Implementing PAPR reduction techniques under GPU environment and its performance comparison with standard processor architecture.

1.4 Thesis Organization

The thesis has been organized into five chapters. The current chapter gives the introduction to the OFDM technology. Furthermore it describes the computational complexity associated with the OFDM system. The motivation and the objective have been discussed in the penultimate sections while the last section describes the complete thesis organization.

Chapter 2: The second chapter provides an introduction to GPU. The architecture of NVIDIA GPUs and the CUDA programming model is also presented

Chapter 3 : The third chapter deals with the implementation of compute intensive portions of 3rd Generation Partnership Project (3GPP) Long Term Evolution (LTE) physical layer using GPU. This chapter presents a simulation model utilizing the massively parallel architecture of GPU at the base station to reduce computation time of LTE transmission and reception. Simulation results demonstrate that GPU provides a framework for fast data processing in this application.

Chapter 4: The fourth chapter discusses the GPU implementation of a MIMO OFDM system. Maximal Ratio combining (MRRC) and Space-Time Block Coding (STBC) techniques are used for MIMO implementation and from the results a comparison is drawn to show which one of these two techniques suits better for GPU implementation.

Chapter 5: The fifth chapter presents a GPU based implementation of the Partial Transmit sequence (PTS) technique for peak-to-average power (PAPR) reduction in an OFDM system.

Chapter 6: The sixth chapter presents the conclusion to the complete work and discusses the scope of future work to the research work that has been described in the thesis.

2 GRAPHICS PROCESSING UNIT (GPU): AN INTRODUCTION

The demand for high performance computing (HPC) is increasing rapidly since large and complex computational problems have become very common across industries and also in many fields of research. Traditional CPU technology fails to sufficiently address this requirement of faster processor performance. Over the years, size of transistors has become smaller and clock rates have increased which resulted in lesser time for each computation. But gradually it was realized that scaling the processor frequency beyond a certain extent would not be possible because of excessive power consumption. The researchers were left with two choices: either to change the architecture or to adopt multicore processing. Multicore processors were preferred and now-a-days multicore processing finds its applications in many electronic gadgets. In almost every 10 years, the there is a change in computing architectures [11]. The cluster based computing was preferred to vector-based computing and this lead to the supercomputing industry move beyond the petaFLOP performance limit. The next target was to provide computing systems with capability of at least one exaFLOPS, which is a thousand fold increase over the petascale. To achieve this, a new shift in computing architectures was required which led to the emergence of parallel computing [12]. Graphics processing units (GPUs) have rapidly gained significant attention and have evolved as high performance accelerators for parallel computation of data [10]. Modern GPUs contain thousands of smaller processing

units have the ability to achieve up to 1 TFLOPS for single-precision (SP) arithmetic and more than 80 GFLOPS for double-precision (DP) arithmetic [9].

Graphics Processing Unit (GPU) has the ability to process large amount of data in parallel by dividing the problem in a number of smaller units. This capability of GPU allows users to solve the complex computational problems at a faster rate. The use of GPUs for computation is a significant development in the field of HPC. GPUs provide up to 10 to 100 times faster performance to solve problems as compared to traditional sequential processors with x86-based CPUs alone. Moreover GPUs consume much lesser power for solving a particular problem compared to its CPU counterpart [13].

The increasing demand for faster computing performance has resulted in the adoption of a hybrid computing model for HPC, where GPUs and CPUs work in tandem to solve challenging computational tasks. GPUs being parallel processors excel in tacking computation of large amount of similar data because the problem can be divided into hundreds of smaller tasks and calculated concurrently. CPUs being sequential processors are adept in performing sequential tasks such as organizing data and running operating systems. NVIDIA GPUs outperform their contemporaries as they provide the most relevant processor to compute a specific task [14].

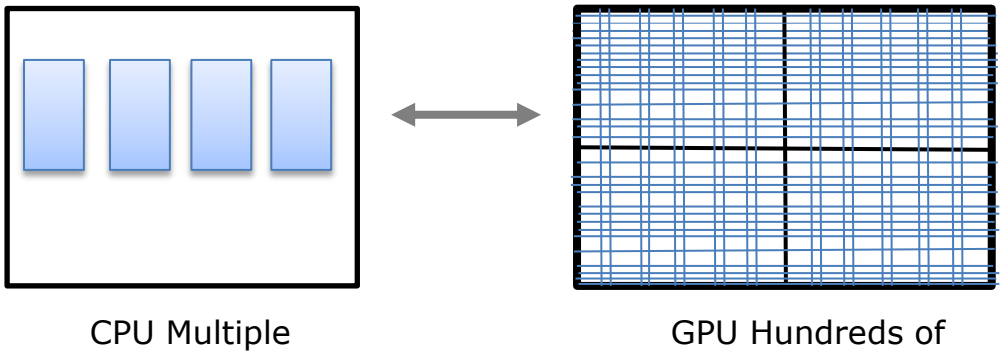

FIGURE 2-1: COMPUTING WITH GPU AND CPU

In GPU-accelerated computing, a graphics processing unit (GPU) works in tandem with a CPU to accelerate scientific, engineering, and industrial applications. GPU-accelerated computing offers significantly faster application performance by executing the

compute-intensive portions of the application to the GPU, while the remainder of the program still runs on the CPU. To harness the parallel computing power of GPUs, the compute intensive portions of a problem are modified to benefit from the hundreds of parallel cores present in the GPU by utilizing all the cores efficiently [9].

2.1 Background

GPUs have been ignored for a long time as a potential source for computation. Initially, the scope of using GPUs was limited to graphics processing only. In 1990s GPUs came to light when there was an increase in 3-D video gaming. In 1999, NVIDIA released its first commercial GPU – "GeForce 256". It contained a 32- bit on board transformation and lighting (T&L) to consumer level 3-D hardware and they performed calculations on triangles on the graphics card instead of the CPU [14]. The GeForce256 GPUs performed advanced video acceleration, motion compensation, etc. In the following years, NVIDIA brought out several improved and faster versions of the GeForce to cope up with the market challenges.

Earlier GPUs were considered only as heterogeneous processors which relied on smaller processors perform tasks at a faster rate. Hence their capabilities were quite limited but gradually researches considered the possibility of GPUs to do more just graphics processing. They realized that "general purpose" GPU (GPGPUs) was the need of the hour as it would make parallel programming relatively easier. By the mid-2000s, NVIDIA started its work to generalize the GPU and their primary goal was to create a scalable processor array which will not only provide better graphics capabilities but also superior graphics processing capabilities. In November 2006, NVIDIA introduce the Tesla architecture in GPUs which provided high performance parallel computing capability to the GPUs.

NVIDIA Tesla GPUs has following special features [11]:

- Superior performance
    - Provides fastest performance for double point floating precision arithmetic
    - Supports large HPC data sets by incorporating larger on board memory
    - Enables faster communication
- Highly Reliable
    - Error checking and correction (ECC) for reliable data transmission
    - Zero error tolerance stress tested

In 2010, NVIDIA released the Fermi architecture which incorporated advanced features to the previous architectures. It allowed a wider range of functionalities on GPU and also enabled greater parallelism with multiple, independent kernels. Table 2-1 presents a comparison between the three major GPU architectures.

TABLE 2-1 COMPARISON OF GPU ARCHITECTURES [11]

Specifications	GeForce 256	Tesla Architecture	Fermi Architecture
Fabrication Technology	220 nm	90 nm	40 nm
Number of cores	4	128	448
Clock Frequency	120 MHz	1.5 GHz	1.15 GHz
Peak GFLOPS	0.48 GF	576 GF	1030 GF
Memory Type	DDR	$GDDR_3$	$GDDR_5$
Memory Size	64	768	3000
Memory Clock	300 MHz	1.08 GHz	3 GHz
Bandwidth	4.8 GB/s	104 GB/s	144 GB/s
Thermal Design Power	~ 20 W	150 W	225 W

2.2 NVIDIA Tesla Architecture: Overview

The principle of multi-carrier transmission is to divide the entire bandwidth into smaller bandwidths each with a different sub-carrier frequency, such that each of these narrow-

band signals is immune to frequency selective fading and the data-rates are improved in comparison to single-carrier system as the total bandwidth can be increased significantly. The block diagram of a GPU (Graphics Processing Unit) based high performance computing system is presented in Figure 2-2 [13, 14] .

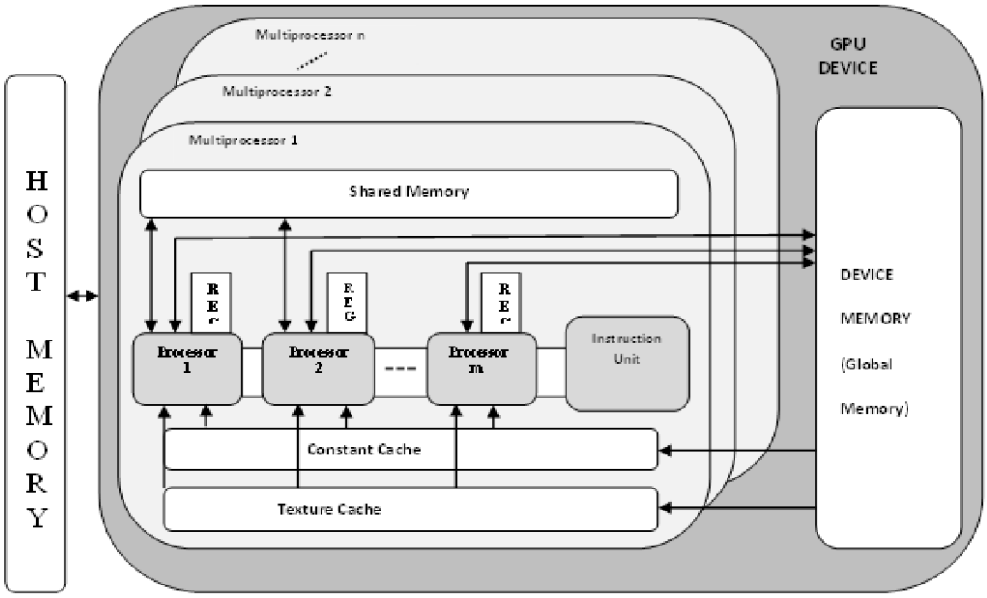

FIGURE 2-2 NVIDIA GPU ARCHITECTURE

A GPU based system uses multiple processors. Along with this it has an array of smaller processors with their shared cache and a shared memory. Currently, a single GPU system can have as many as 512 processing cores [10]. These systems have the capability of high processing throughput through parallelization. CUDA (Compute Unified Device Architecture) is a parallel computing platform and programming model developed by NVIDIA and implemented on the GPUs designed by them. It enables increase in computing performance by utilizing the power of GPU [15].

In CUDA, parallel portion of the application is executed as a kernel [16]. A kernel is an array of threads executed in parallel and all the threads execute the same code. A kernel is executed as a grid of blocks and threads in Single Instruction Multiple Thread (SIMT) manner as shown in Figure 2-3.

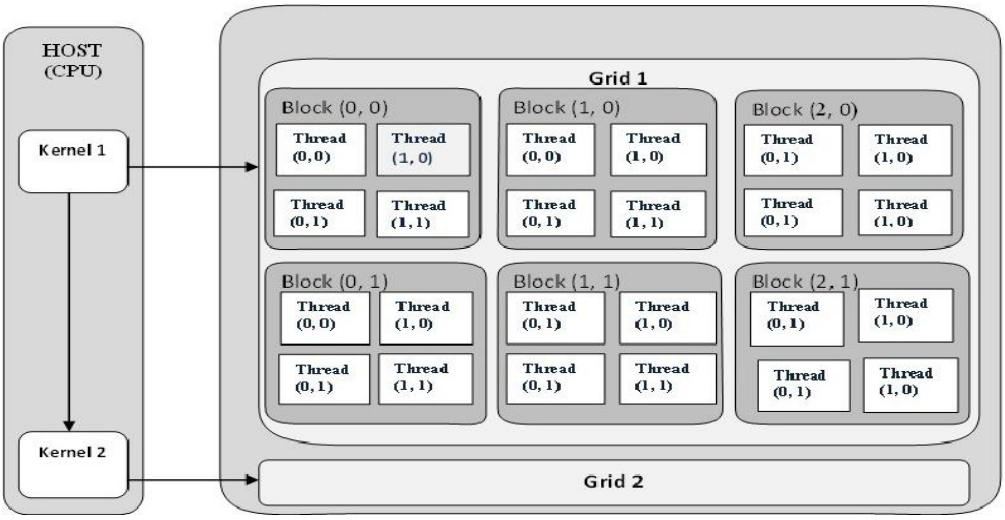

FIGURE 2-3: PARALLEL EXECUTION OF A GPU KERNEL IN A GRID OF THREAD BLOCKS

Threads are virtually mapped to an arbitrary number of streaming multiprocessors (SM) and are executed in 32 parallel thread groups called warps. A GPU has a large amount of off-chip memory called global memory. It is slower than the on-chip memory and register resources. While processing data in GPU, the global memory access must be minimized because it incurs long latency.

2.3 GPU programming using MATLAB

MATLAB Parallel Computing Toolbox for GPU computing allows users to exploit the computational power offered by the NVIDIA Graphics processing unit (GPU) architecture [17].

In this project, Parallel computing toolbox in MATLAB was used for simulation in the GPU environment. The GPU accelerated system hardware consists of a host PC with Intel® Xeon®, E5-2650 processor operating at 2.0 GHz with Linux operating system and a NVIDIA Tesla M2090 graphics card with 512 cores at 1.3 GHz processor clock.The "gpuArray( )" and "gather ( )" functions were used to transfer data from GPU to CPU. All the algorithms were sufficiently vectorized in order to obtain optimum performance in GPU.

3 LTE PHYSICAL LAYER IMPLEMENTATION USING GPU BASED HIGH PERFORMANCE COMPUTING

In recent years Graphics Processing Unit (GPU) has evolved as a high performance data processing technology allowing users to compute large blocks of parallel data using an array of low complexity processors. This chapter proposes the implementation of compute intensive portions of 3rd Generation Partnership Project (3GPP) Long Term Evolution (LTE) physical layer using GPU. LTE employs Orthogonal Frequency Division Multiple Access (OFDMA) in downlink and Single Carrier Frequency Division Multiple Access (SC-FDMA) in uplink [18]. Both these demand computationally complex Inverse Fast Fourier Transform (IFFT) and Fast Fourier Transform (FFT) processing at the transmitter and the receiver. The computational requirements at the base station increases significantly with the increase in number of users. A simulation model of the basic framework is presented utilizing the massively parallel architecture of GPU to reduce computation time of IFFT and FFT operations. Simulation results demonstrate that GPU provides a framework for fast data processing in this application.

3.1 Introduction to Long Term Evolution (LTE)

Long Term Evolution refers to the 3rd Generation Partnership Project (3GPP) Evolved Universal Mobile Telecommunications System (UMTS) Terrestrial Radio Access (E-UTRA) technology and its first version is documented in Release 8 of 3GPP specification.

LTE is considered as one of the most promising technologies to meet the growing demands for high data rate services with high spectral efficiency. This technology is designed to provide a peak data rate of 100 Mbps in downlink and 50 Mbps in uplink when operating in 20MHz bandwidth [19]

3.1.1 LTE PHYSICAL LAYER: OVERVIEW

3.1.1.1 LTE Downlink Physical Layer

Orthogonal Frequency Division Multiple Access (OFDMA) is the multiple access technique used for LTE downlink transmission [18, 2]. The block diagram of OFDM system is shown in Figure 3-1.

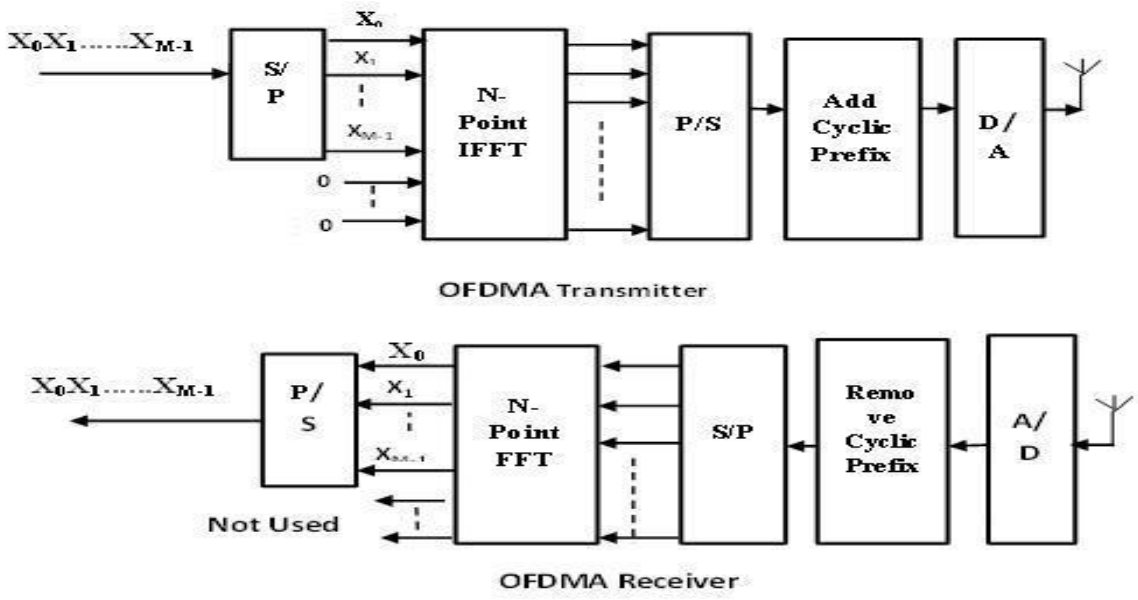

FIGURE 3-1: OFDMA TRANSMITTER AND RECEIVER

The transmit sequence is converted to constellation symbol sequence using any of the three modulation techniques available for LTE. These include QPSK, 16 QAM or 64 QAM. These symbols are converted to M parallel data streams, M being the number of subcarriers. The block of M symbols in each OFDM symbol period passes through an IFFT of size N (N>M). Following this, cyclic prefix is appended to the sequence of OFDM

symbols. The symbols are next transmitted after up conversion. At the receiver the cyclic prefix is removed and N point FFT is applied after serial to parallel conversion.

3.1.1.2 LTE Uplink Physical Layer

In the uplink LTE uses Single Carrier Frequency Division Multiple Access (SC-FDMA) technique for multiple access [18, 20]. The SC-FDMA block diagram is presented in **Error! Reference source not found.**.

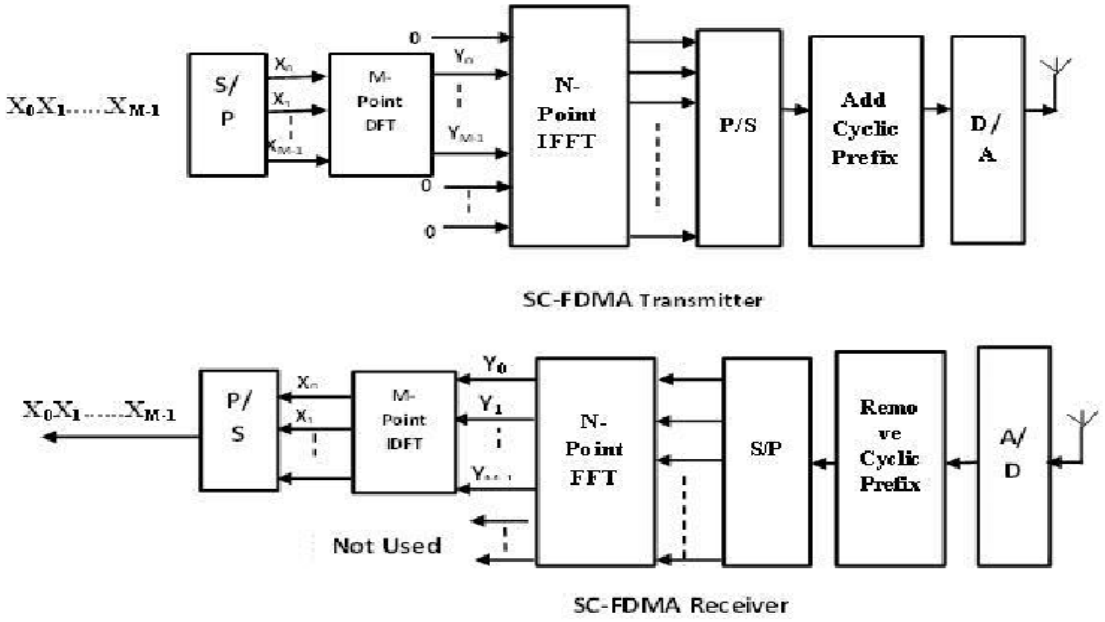

FIGURE 3-2: SC-FDMA TRANSMITTER AND RECEIVER

Blocks of M constellation symbols (QPSK/16-QAM/64 QAM) are converted to frequency using M point FFT. The M FFT outputs are mapped on N subcarriers (N>M). Following this IFFT applied on FFT output padded with zeros. The N–M subcarriers are unused and signal occupies block of M subsequent subcarriers. Hence FFT and IFFT in cascade create a single-carrier signal [21]. In SC-FDMA receiver, N-point FFT operation is done which is followed by M point IFFT processing and the signal is converted back to time domain.

3.1.2 LTE RADIO FRAME STRUCTURE

The radio frame structure for LTE in Frequency Division Duplex (FDD) mode of operation is shown in Figure 3-3 [18, 22]

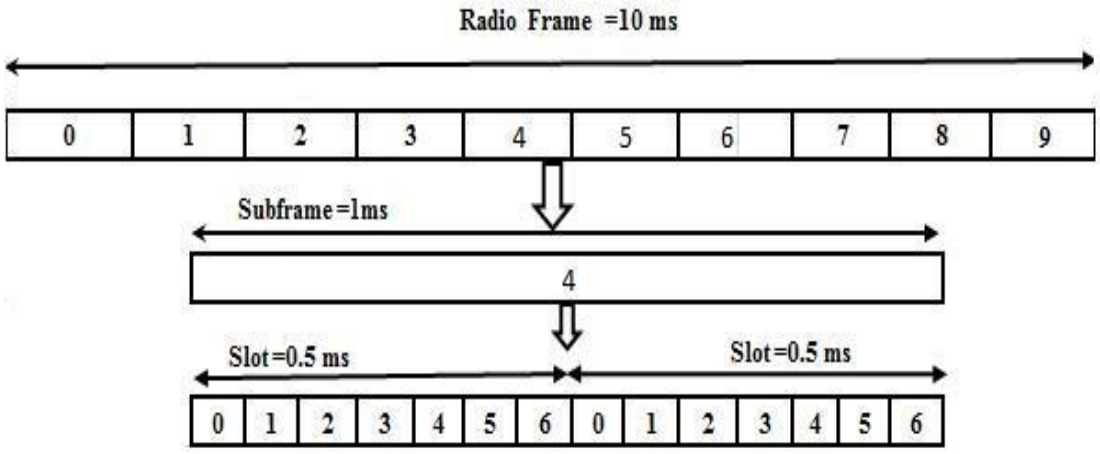

FIGURE 3-3 LTE RADIO FRAME STRUCTURE

The LTE frames are 10 ms in duration which is divided into 10 subframes of 1.0 ms long. Each subframe is further divided into two slots, each of 0.5 ms duration. Slots consist of 6 to 7 ODFM symbols [19].

The LTE system parameters for different channel bandwidth are presented in Table 3-1. Either short or long cyclic prefix is chosen depending on channel delay spread. For short cyclic prefix, the first OFDM symbol in a slot has slightly longer cyclic prefix than the remaining six symbols [23].

TABLE 3-1 PARAMETERS FOR LTE TRANSMISSION [20]

Channel BW (MHz)	1.4	3	5	10	15	20
No. of subcarriers	72	180	300	600	900	1200
FFT Size (N)	128	256	512	1024	1536	2048
Samples per slot	960	1920	3840	7680	11520	15360
Sampling rate (MHz)	1.92	3.84	7.68	15.36	23.04	30.72

The radio frame structure for uplink is same as that used in downlink. The subcarrier spacing is 15 KHz. The parameters for uplink are same as downlink as presented in Table 3-1.

In the frequency domain symbols are grouped in units of 12 subcarriers occupying a total bandwidth of 180 kHz. This block of 12 subcarriers is called a Physical Resource Block (PRB). In time, the length of a PRB is always 1 slot, which is equal to 0.5 ms [22].

3.2 Computational Complexity Issues

From the block diagram of LTE uplink and downlink system, it can be observed that the fundamental system involves A/D and D/A conversion, cyclic prefix addition and removal, P/S and S/P conversion, FFT and IFFT operation. Out of these FFT and IFFT operations consume substantial processing resources. In a typical simulation environment, the execution time of each block using sequential processing is presented in Table 3-2. Here the system uses 140 OFDM symbols with 1200 subcarriers and 2048 point FFT and IFFT based on LTE 200 MHz specification.

TABLE 3-2 EXECUTION TIME OF EACH BLOCK OF AN OFDM BASEBAND SYSTEM IN A SEQUENTIAL PROCESSOR

Transmitter			**Receiver**		
Process	***Time (ms.)***	***%***	***Process***	***Time (ms.)***	***%***
S/P	25.09	11.57	Cyclic Prefix Removal	19.68	9.02
IFFT	**134.45**	**61.98**	S/P	51.30	23.50
P/S	36.59	16.87	FFT	**114.78**	**52.58**
Cyclic Prefix Addition	20.78	9.58	P/S	32.53	14.90

From Table 3-2, it can be observed that IFFT and FFT operations consume majority of the computational resource in a sequential processing environment. Parallel implementation of these is expected to provide performance speed up.

3.3 Parallel Implementation of FFT/IFFT

The computational problem for DFT is to compute the sequence $\{X(k)\}$ of N complex-values for a given sequence of data $\{x(n)\}$ of length N as given in equation (3-1).

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \quad , 0 \le k \le N \qquad (3\text{-}1)$$

where $W_N = e^{-j2\pi/N}$

DFT is calculated more efficiently using the Radix-2 fast Fourier transform or FFT algorithm [24]. Under this

$$X(K) = \sum_{n=0}^{\frac{N}{2}-1} x_e(n) W_{N/2}^{nk} + W_N^K \sum_{N=0}^{\frac{N}{2}-1} x_o(n) W_{N/2}^{nk} \qquad (3\text{-}2)$$

The 1st term in equation (3-2) is the N/2 point DFT of the even indexed sequence $x_e(n)$ and the 2nd term is the N/2 point DFT of the odd indexed sequence $x_o(n)$.

This computation is presented in Figure 3-4. The structure is typically called a butterfly structure.

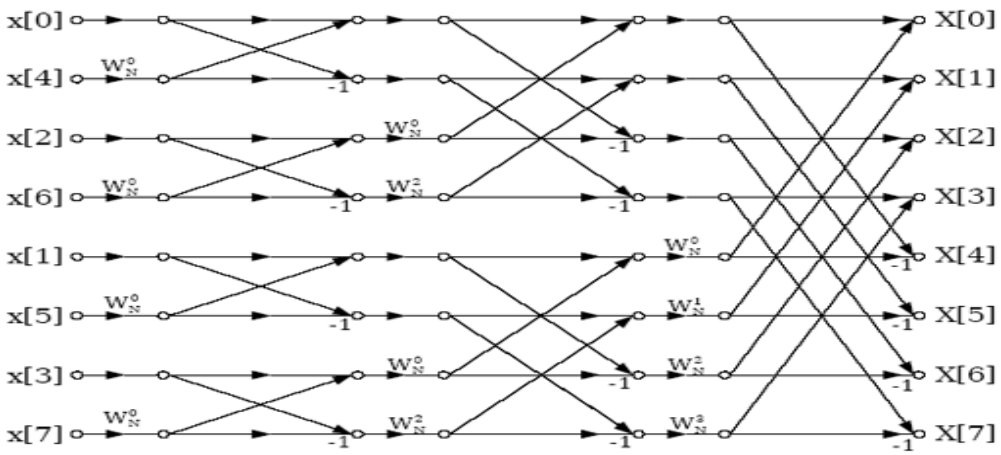

FIGURE 3-4: FLOW GRAPH OF DECIMATION IN-TIME FFT ALGORITHM

This architecture can be extended for larger size FFT computation of the form 2^k where k is an integer. When the FFT size is large the number of parallel computation required constantly increases. The major advantage of using GPU for FFT calculation is to perform the multiplications in parallel using the large array of low complexity processors instead of computing them sequentially. This reduces the computational time and can be used very efficiently for large sequence FFT calculation [25]. Similar analogy can be drawn for IFFT computation in GPU.

Since FFT is a major processing in LTE 3GPP and it can be implemented using parallel architecture, GPU hardware as a tool has been used for efficient implementation of this.

3.4 Implementation Details

Implementation of data processing algorithms in GPU demands transfer of data between host computer (CPU) and the device (GPU). But due to the high latency and low bandwidth in such memory transfers, GPU should be used only for processing large number of data with high complexity of operations so that the cost of transferring and gathering data from the GPU is optimized. In LTE for multiple users the base station has to process huge number of data frames coming from each user. With the increase in channel bandwidth the number of subcarriers increase leading to increased IFFT points as presented in Table 3-1. This increases the computation time. Hence in our simulation model, only the IFFT/FFT computations at the base station are performed using GPU and rest of the blocks are simulated in the CPU.

3.5 Numerical Results and Discussion

The performance of LTE transceiver was evaluated through simulation. The simulations were performed using MATLAB software. The simulations have been carried out using parameters of LTE specifications in FDD mode of operation as specified in Table I for both uplink and downlink. Multiple access has not been addressed in this paper. Hence

simulations were performed considering the entire data frame transmitted by a user using single input single output (SISO) antenna system. In all the simulations, signal power at the transmitter antenna was set to unity. In downlink one frame of data to be transmitted by the base station is processed in GPU for different channel bandwidths. In uplink, the frames transmitted by multiple users, each transmitting at 20 MHz channel bandwidth was processed at the base station in parallel using GPU and the computation throughput was taken as performance index. The symbol error rate (SER) performance for one user was evaluated using Monte Carlo simulation.

3.5.1 LTE Downlink Performance

In downlink the base station transmits a frame using OFDM. The user data is converted to symbols using all three types of modulation used in LTE: QPSK, 16-QAM and 64-QAM. The number of input samples, subcarriers and cyclic prefix are chosen for different channel bandwidth according to the LT specifications given in Table 3-1.

The Symbol error rate curve obtained using Monte Carlo simulation for AWGN channel is shown in Figure 3-5. The Monte Carlo simulation with GPU support takes much lesser time to simulate and is completely justified.

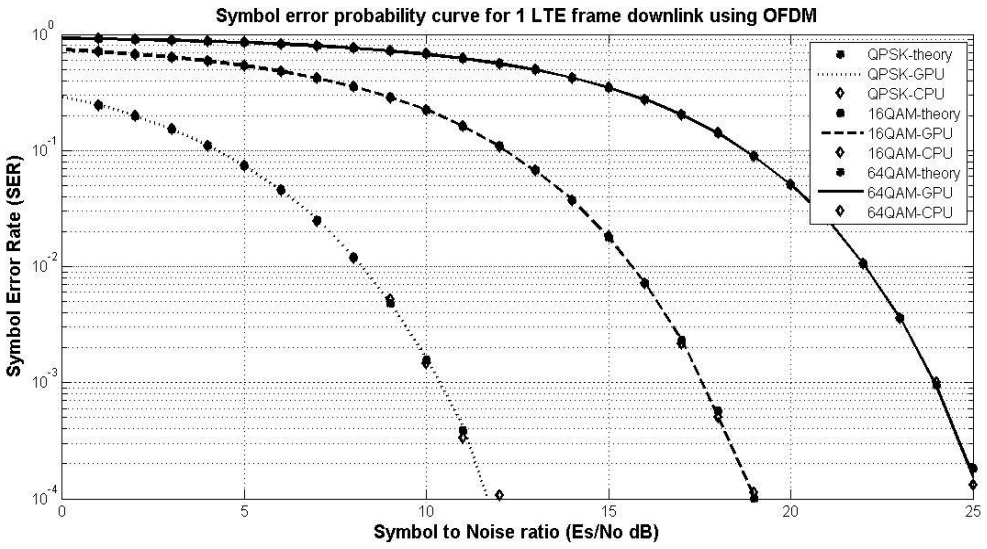

FIGURE 3-5 SYMBOL ERROR RATE CURVE FOR LTE DOWNLINK IN AWGN CHANNEL.

The IFFT computation for the OFDM transmitter is performed in the GPU and the speed up obtained for different channel bandwidths with different number of IFFT points for 1 transmitted LTE frame is shown in Figure 3-6. The time is calculated using the host clock function.

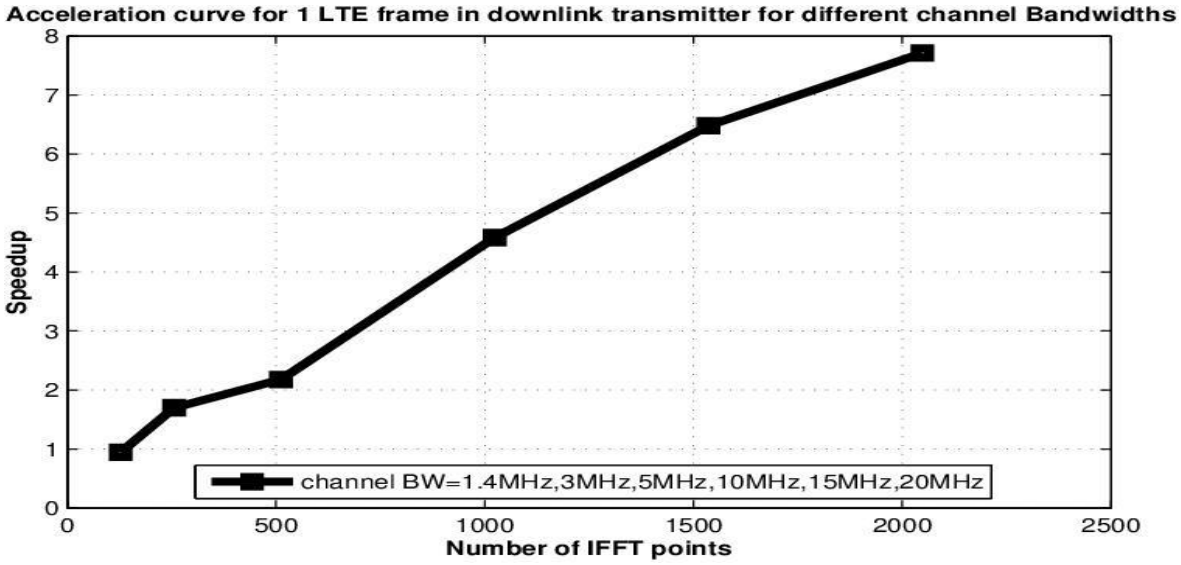

FIGURE 3-6: GPU PERFORMANCE FOR TRANSMITTING 1 LTE DOWNLINK FRAME AT DIFFERENT CHANNEL BANDWIDTH.

From Figure 3-6, it is observed that the speedup is highest for 20 MHz channel bandwidth with 2048 point IFFT while it is lowest for 1.4 MHz channel bandwidth with 128 IFFT points. Thus the performance improves in GPU as the amount of data processed in parallel increases. To fully utilize the multithreading GPU, large amount of data can be processed in parallel at the base station simultaneously and then transmitted frame wise according to LTE specifications. This will optimize the cost of data transfer from the CPU to the GPU.

3.5.2 LTE Uplink Performance

In uplink the base station transmits a frame using SC-FDMA as described in Section II. The number of input samples, subcarriers and cyclic prefix are chose for 20 MHz bandwidth according to the LTE specifications given in Table 3-1. The FFT and IDFT computation for the SC-FDMA receiver is performed in the GPU. Multiple users are

considered to transmit LTE frames and the received frames at the base station are processed simultaneously. The SER performance for SC-FDMA in LTE uplink in AWGN channel for user #1 transmitting at 20MHz bandwidth is evaluated and presented in Figure 3-7.

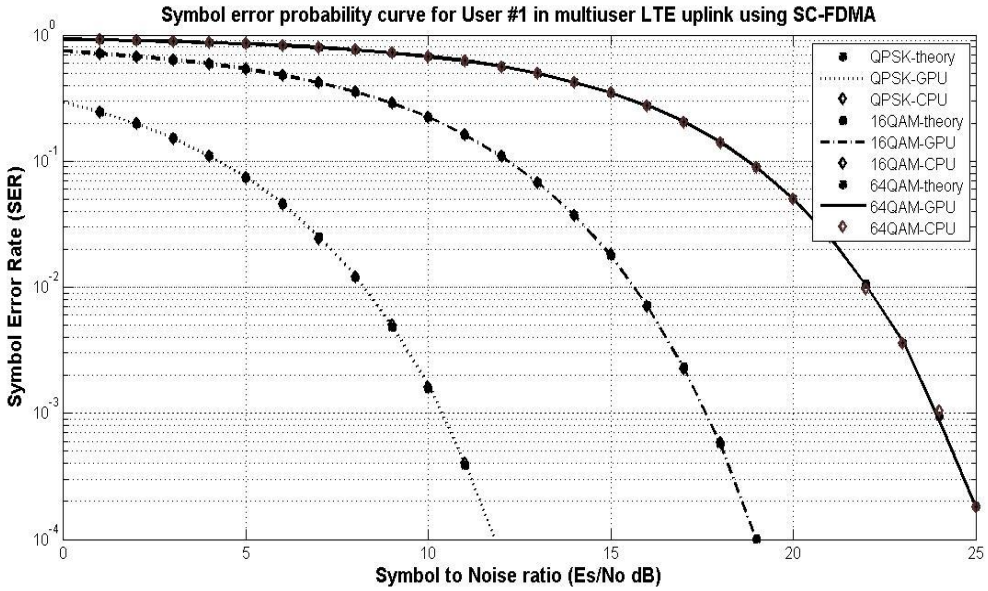

FIGURE 3-7: SYMBOL ERROR RATE FOR USER #1 IN LTE UPLINK.

The speed up obtained for different users with 20MHz bandwidth per user is presented in Figure 3-8.

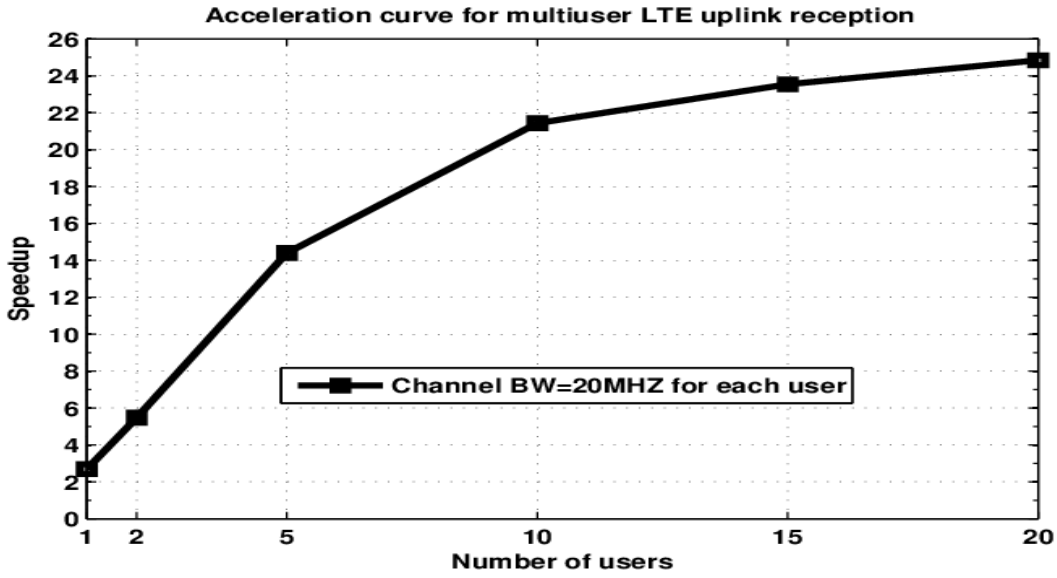

FIGURE 3-8 GPU PERFORMANCE FOR PROCESSING RECEIVED LTE UPLINK FRAME FOR DIFFERENT NUMBER OF USERS.

TABLE 3-3 PROCESSING THROUGHPUT FOR HOST CPU AND GPU DEVICE

No of users	CPU (ms)	GPU (ms)			Throughput (QPSK modulation)	
		Kernel	*Data Transfer*	*Total*	*CPU (Mbps)*	*GPU (Mbps)*
1	15.96	5.96	3.81	9.77	56.92	93.08
2	28.69	5.24	5.94	11.18	63.39	162.69
5	92.89	6.45	12.65	19.1	48.95	238.07
10	190.88	8.91	23.6	32.51	47.64	279.74
15	296.36	12.59	36.46	49.05	46.03	278.12
20	399.33	16.07	50.93	67	45.54	271.47

Table 3-3 shows the throughput comparison for FFT and IDFT computation in the host CPU and GPU device. From Table 3-3, it can be observed that as the numbers of users increase the processing throughput increases significantly. Though the time to transfer the data from CPU to GPU is quite high, still the large amount of data processed in parallel compensates for this latency and gives a computation throughput which is much higher than that in CPU.

A basic framework simulation assuming a simple LTE model has been efficiently implemented on GPU and presented in this paper. The simulation model utilized the massively parallel architecture of GPU to reduce the computation time at the base station for LTE uplink and downlink. The computation throughput of the GPU implementation is shown to outperform the conventional sequential implementation. The implementation of this new method is expected to provide promising ways to implement complex wireless communication systems using GPU based computing hardware.

4 MIMO OFDM PERFORMANCE UNDER GPU ENVIRONMENT

The multiple-input multiple-output (MIMO) wireless technology in conjunction with OFDM is perceived as a very promising technique to support high data rate and high performance [26]. Specifically, coding over the space, time, and frequency domain in MIMO-OFDM provides a much more reliable and robust transmission over the harsh wireless environment [27]. In OFDM the total available bandwidth is divided into a set of orthogonal subchannels. At the receiver, the received signal at each antenna for each subcarrier comprises of a signal which is a combination of data streams from multiple transmit antennas. Hence a higher complexity detector is required to reconstruct the transmitted signal vector as compared to single antenna systems.

In this chapter, performance comparison of 2×1 space-time (ST) coded MIMO-OFDM detection with the classical 1×2 Maximal Ratio combining scheme (MRC) under GPU environment is presented.

4.1 MIMO-OFDM SYSTEM MODEL

The goal of future broadband wireless systems is to provide high data rate and high performance over wireless channels that may be time selective and frequency-selective. OFDM combined with MIMO is considered to have the potential of meeting this stringent requirement. MIMO can boost the capacity and the diversity of the system and OFDM can mitigate the detrimental effects caused due to multipath fading. A general MIMO-OFDM

system is shown in Figure 4-1, where M_t transmit antennas, M_r receive antennas, and N subcarriers are used [27].

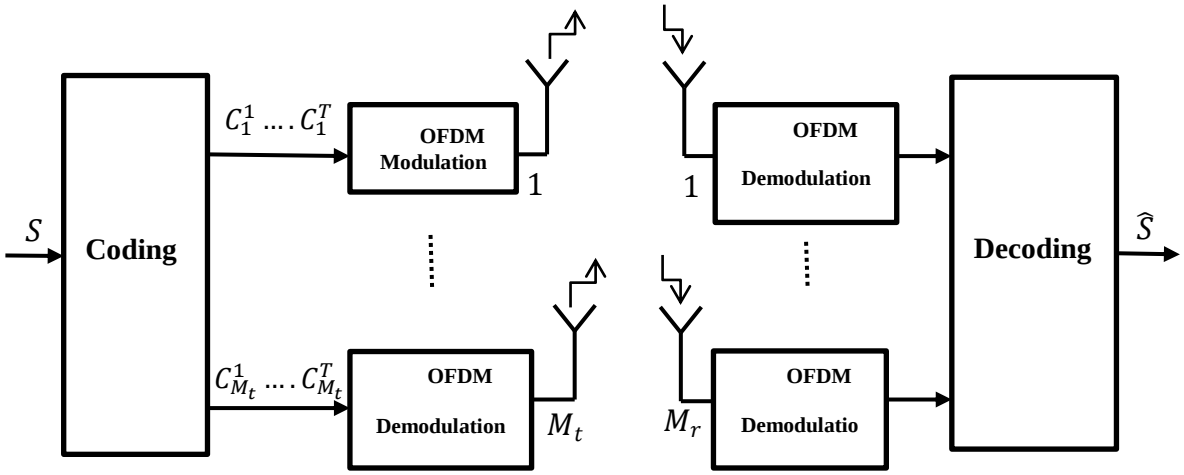

FIGURE 4-1: SIMPLIFIED BLOCK DIAGRAM OF MIMO OFDM SYSTEM

The input information bits are into data symbols using a specific modulation scheme. Following this blocks of N_s data symbols $S = [s_1, s_2, \ldots . , s_{Ns}]$are encoded into a code word matrix C of size $NT \times M_t$, which is then transmitted through $M_t$antennas in T OFDM blocks. Each block of the matrix consists of N sub channels. Thus vectors $c_j^1, c_j^2 \ldots c_j^T$ are transmitted from the transmit antenna j in OFDM blocks $1, 2, \cdots, T$ respectively, where C_j^ndenotes a vector of length N, for all $j = 1, 2, \cdots, M_t$and $n = 1, 2, \cdots, T$.

The codeword matrix C is presented in Equation (4-1)

$$C = \begin{bmatrix} c_1^1 & \cdots & c_{M_t}^1 \\ \vdots & \ddots & \vdots \\ c_1^T & \cdots & c_1^1 \end{bmatrix} \quad (4\text{-}1)$$

After cyclic prefix is addition on each OFDM block, c_j^n is transmitted from the jth transmit antenna in the nth OFDM block. The signals after passing through the MIMO channels reach the receiver. In the receiver OFDM demodulation is done which is followed by MIMO decoding.

4.2 SPACE-TIME CODED OFDM

Space-Time (ST) coding for MIMO is a powerful scheme. It combines coding along with transmit diversity to achieve higher diversity performance in wireless communication systems. Orthogonal ST block code (OSTBC) design was first proposed by Alamouti in 1998, now referred to as the Alamouti code [28].

4.2.1 STBC OFDM Encoding Scheme

In STBC design, the information symbols from two transmit antennas are transmitted after encoding in a specific order at consecutive time intervals as shown as in Table 4-1

TABLE 4-1 ALAMOUTI ENCODING SCHEME.

Time	Antenna 1	Antenna 2
t	s_1	s_2
t+T	$-s_2^*$	s_1^*

In ST coding with 2 antennas, the information symbols s_1 and $-s_2^*$are sent through sub channel k of antenna 1 in OFDM blocks n and $n+1$ respectively. Signals s_2 and s_1^* are sent through sub channel k of antenna 2 in OFDM blocks n and $n+1$ respectively [28, 27].

The channel at time t can be modeled by a complex multiplicative distortion $h_1(t)$ and $h_2(t)$ for transmit antenna 1 and for transmit antenna 2. Assuming that fading is constant across two consecutive symbols and T being the symbol duration, it can be written

$$h_1(t) = h_1(t+T) = h_1$$
$$h_2(t) = h_2(t+T) = h_2 \quad (4\text{-}2)$$

The received signals can be expressed as shown in Equation (4-3)

$$r_1 = r(t) = h_1 s_1 + h_2 s_2 + n_1$$

$$r_2 = r(t+T) = -h_1 s_2^* + h_2 s_1^* + n_2 \quad (4\text{-}3)$$

where r_1 and r_2 are the received signals at time t and t+T and n_1 and n_2 are complex random variables representing receiver noise and interference.

4.2.2 STBC combining scheme

The combiner reconstructs the two combined signals as shown in Equation (4-4) that are sent to the detector.

$$\tilde{s}_1 = h_1^* r_1 + h_2 r_2^*$$
$$\tilde{s}_2 = h_2^* r_1 - h_1 r_2^* \quad (4\text{-}4)$$

Since the two signals are transmitted in two symbol periods, the coding rate of STBC is one. As STBC codes are orthogonal to each other, the signals can be recovered at the receiver using simple linear combining scheme.

4.3 Maximal Ratio Receive Combining (MRRC) Scheme

MRRC is a form of receiver space diversity where there are multiple antennas at the receiver. Figure 4-2 presents a two-branch (1 × 2) MRRC scheme [28, 26].

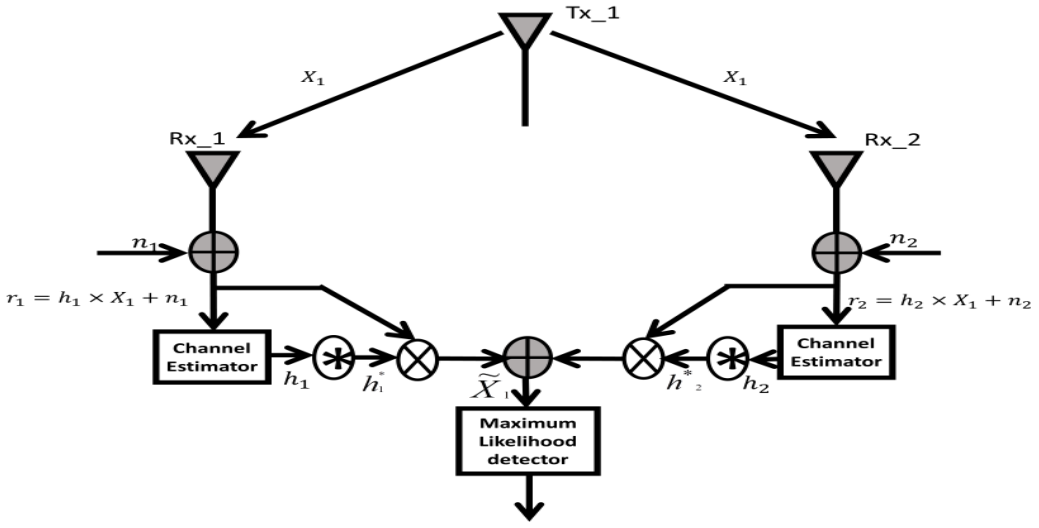

FIGURE 4-2: TWO BRANCH MRRC SCHEME

4.3.1 MRRC transmission scheme

In 1×2 MRRC scheme the signal X_1 transmitted from the antenna passes through two different with coefficients h_1 and h_2 between the transmit antenna and receive antenna 1 and receive antenna 2 respectively. The received base band signal r_1 and r_2 at receive antennas 1 and 2 with noises n_1 *and* n_2 can be represented as shown in equation (4-5).

$$r_1 = h_1 x_1 + n_1$$
$$r_2 = h_2 x_1 + n_2 \quad \text{(4-5)}$$

4.3.2 MRRC Combining scheme

The receiver combining scheme for two-branch MRRC is presented in Equation (4-6) where, $\tilde{X}_1$ is the expected received signal for the transmitted signal X_1.

$$\tilde{x}_0 = h_1^* r_1 + h_2^* r_2 \quad \text{(4-6)}$$

4.4 GPU Implementation of STBC and MRRC schemes

It is observed from equation (4-4) and equation (4-6) that both STBC and MRRC MIMO OFDM schemes involve complex multiplications at the receiver for combing. Hence MRRC and STBC schemes are implemented with GPU hardware support with an aim to reduce the computation time [33, 34].

4.5 Simulation results and discussion

The 2×1 STBC and 1×2 MRRC schemes are implemented on the host CPU with GPU support at the MIMO combining block at the receiver. BPSK modulation is done to map the symbols into constellation. Following this, OFDM modulation is done. The OFDM signal transmitted from the antennas (2 for STBC and 1 for MRRC) pass through Rayleigh flat fading channels. At the receiving end after OFDM demodulation, combining is performed (assuming known channel coefficients). The parameters chosen for OFDM

modulation are listed in Table 4-2 and the computation time is evaluated with GPU support at the MIMO receiver. A comparison is shown between sequential implementation using CPU and GPU accelerated implementation.

TABLE 4-2 PARAMETERS CHOSEN FOR MIMO OFDM SIMULATIONS

No. of subcarriers	256	512	1024	2048
FFT Size (N)	256	512	1024	2048
Samples per slot	1920	3840	7680	11520

4.5.1 BER performance comparison for MRRC and STBC schemes

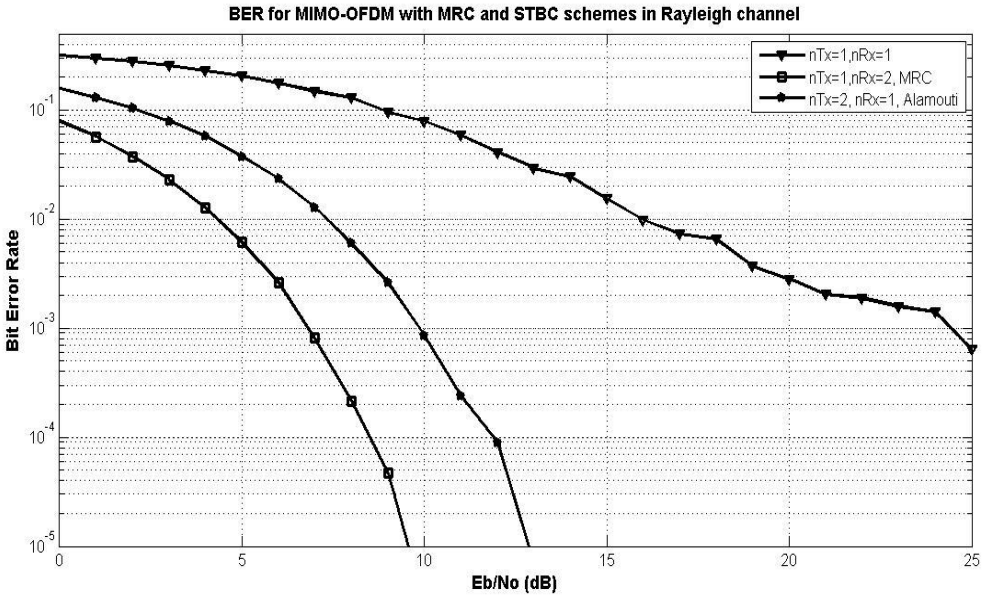

FIGURE 4-3: BER COMPARISON OF STBC AND MRRC SCHEMES WITH SISO OFDM

The BER performance of 2×1 STBC and 1×2 MRRC schemes are evaluated and compared with single antenna OFDM transmission using Monte-Carlo simulation in Rayleigh flat fading (single tap) channel. From Figure 4-3 it can be observed BER significantly improves for MIMO antenna systems. Also it can be observed that MRRC scheme has a better BER performance of approximately 3 dB at BER of 10^{-3} in comparison to STBC scheme.

4.5.2 Computation Time comparison for MRRC and STBC combining schemes in GPU and CPU

The receiver combining of STBC and MRRC schemes require complex multiplications. Hence receiver combining is done in GPU while rest of the code runs in the CPU. The computation time required with fully CPU implementation is compared with that with GPU support at the MIMO combiner for different number of subcarriers as listed in is evaluated.

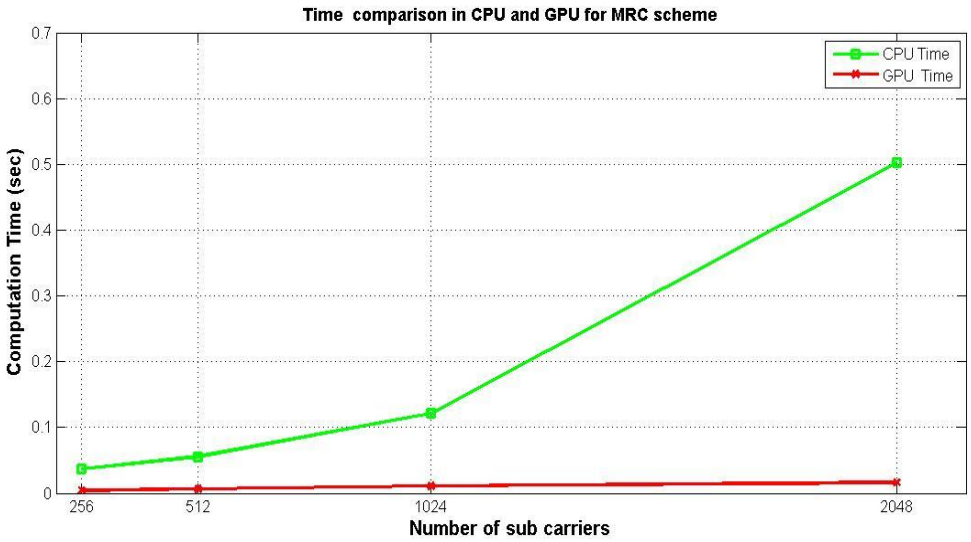

FIGURE 4-4: COMPUTATION TIME FOR MRRC COMBINING IN GPU AND CPU ENVIRONMENT

In MRRC scheme data from both the receivers is transferred CPU to GPU after OFDM demodulation. The MIMO detection of OFDM signals from both the receivers are computed in parallel using GPU. From Figure 4-4, it can be observed that combining in GPU takes very less time as compared to CPU implementation. Though a large amount of data has to be transferred from GPU to CPU, yet the performance gain in GPU gives a much higher overall processing throughput as compared to CPU implementation.

Figure 4-5 shows the computation time comparison for combining between CPU and GPU environment for 2×1 STBC scheme.

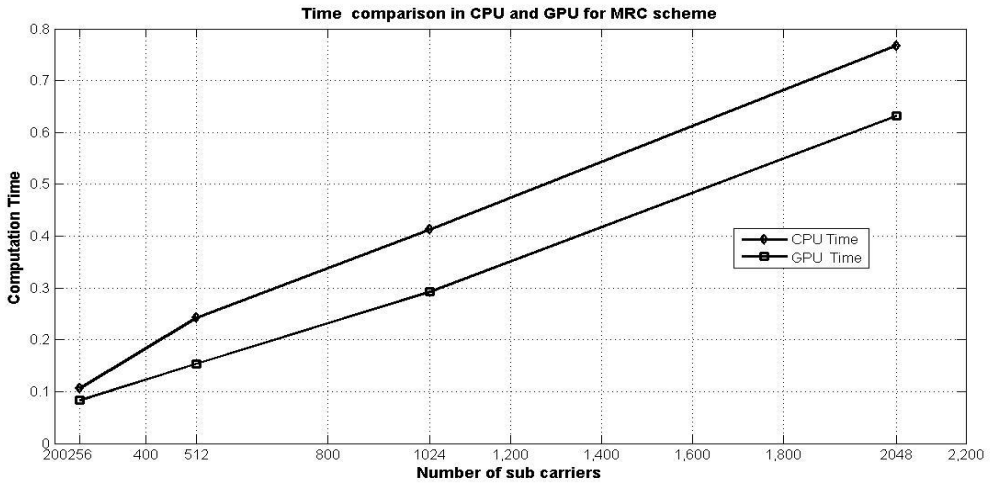

FIGURE 4-5: COMPUTATION TIME FOR STBC COMBINING IN GPU AND CPU ENVIRONMENT

It can be observed that there is not much performance gain obtained in GPU. The reason is STBC transmit data from both the antennas and in receiver this huge amount of data has to be transferred to the GPU which consumes large amount of time due to high latency of GPU.

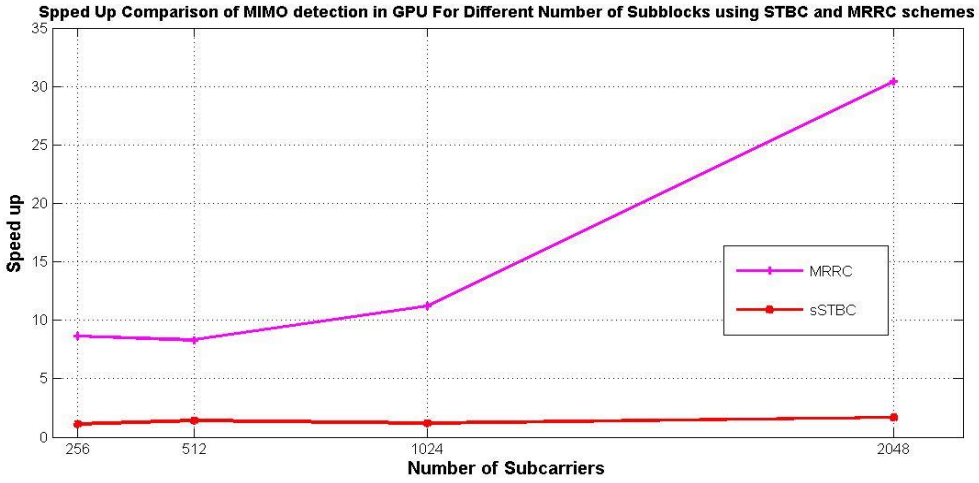

FIGURE 4-6: SPEED UP COMPARISON FOR MRRC AND STBC COMBINING SCHEMES

Figure 4-6: Speed Up Comparison For MRRC and STBC Combining Schemes. Thus it can be concluded that GPU implementation is very suitable for MRRC scheme. The MRRC scheme provided better BER performance and with GPU support computation time also reduces significantly and the overall computation throughput of the system will be high.

5 PAPR REDUCTION IN OFDM SYSTEM USING GPU BASED HPC

5.1 The Peak Power problem in OFDM system

An OFDM signal consists of a number of subcarriers which are modulated independently. This gives rise to high peak values of transmit signals in time domain when the subcarriers are added up coherently. When N signals are added with same phase, the peak power produced in N times the average power of the signals. High Peak-to-Average Power Ratio (PAPR) is a major drawback of OFDM system since it degrades the efficiency of the RF power amplifier in the transmitter and also increases the complexity of analog-to-digital (A/D) and digital-to-analog (D/A) converters [3].

Due to saturation characteristics of power amplifiers (PA), the linear amplifiers also cause a nonlinear distortion on their outputs which is caused by an input much larger than its nominal value. Non-linear operation of PA results in generation of unwanted frequencies by non-linear modulation of the incoming signal. This results in following detrimental effects:

- Inter-modulation of carriers
- Out-of-band radiation due to spectral leakage

5.1.1 Introduction to PAPR

PAPR of a passband signal $\tilde{s}(t)$ is the ratio between the maximum power and the average power of the complex passband signal s (t) is given by [33]

$$PAPR\{\tilde{s}(t)\} = \frac{\max|Re(\tilde{s}(t)e^{j2\pi f_c t})|^2}{\mathrm{E}\{|Re(\tilde{s}(t)e^{j2\pi f_c t})|\}^2} = \frac{\max|s(t)|}{E\{|s(t)|^2\}} \quad \textbf{(5-1)}$$

Let us consider an OFDM system with a data block denoted as $\boldsymbol{X} = [\boldsymbol{X}_0, \boldsymbol{X}_1, \dots, \boldsymbol{X}_{N-1}]$, N being the number of sub-carriers of the OFDM system. The complex baseband representation of this OFDM signal with subcarrier spacing Δf and time period for pulse-shaping symbol T is:

$$x(t) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X_n e^{j2\pi n \Delta f t} \qquad ,0 \le t < NT \quad \textbf{(5-2)}$$

The PAPR of the above transmitted signal can be defined by:

$$PAPR = \frac{\max_{0 \le t < NT} |x(t)|^2}{\frac{1}{NT}\int_0^{NT} |x(t)|^2 \, dt} \quad \textbf{(5-3)}$$

However equation (5-3) gives PAPR for the analog OFDM signal. In practice, the PAPR for the continuous-time baseband signal can be estimated from the discrete-time signal x[n]. To obtain PAPR for a digital OFDM signal, only NL equidistant samples of x(t) are considered where L is the oversampling factor (L>1). The signal samples can be represented as $\boldsymbol{x} = [x_0, x_1, \dots, x_{N-1}]$. It is shown that x[n] can show almost same PAPR as x(t). Thus the OFDM signal can be represented in digital domain as:

$$x_k = x\left(\frac{k.T}{L}\right) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X_n e^{\frac{j2\pi k \Delta f n T}{L}}, k = 0,1, \dots, N-1 \quad \textbf{(5-4)}$$

The PAPR of the OFDM signal described by the equation (5-4) is given below:

$$PAPR = \frac{\max_{0 \le k < NL-1} |x_k|^2}{E[|x_k|^2]} \quad \textbf{(5-5)}$$

5.1.2 CCDF for PAPR

The most common and frequently used measurement index for evaluating the performance of PAPR reduction techniques is Complementary Cumulative Distribution Function (CCDF) of PAPR [30, 33]. The amplitude of the OFDM signal has a Rayleigh distribution, while the power distribution becomes a central chi-square distribution with two degrees of freedom and zero mean with a cumulative distribution function (CDF) given by

$$F(z) = 1 - e^{-z} \tag{5-6}$$

CCDF represents the probability that the PAPR of a data block exceeds a certain threshold value. The CCDF of PAPR of a OFDM signal with N subcarriers sampled at Nyquist rate is given as

$$P(PAPR > z) = 1 - P(PAPR \leq z) = 1 - F(z)^N = 1 - (1 - e^{-z})^N \tag{5-7}$$

The assumption made in equation (5-7) is that the samples of the OFDM signal are mutually uncorrelated and the equation does not hold true when oversampling is applied. It is quite difficult to find an exact solution for the peak power distribution of an oversampled OFDM signal and hence an approximation has been proposed where the effect of oversampling is tackled by adding a certain number of extra independent samples. The distribution of PAPR for oversampled OFDM signal, L being the oversampling factor is given by:

$$P(PAPR > z) = 1 - (1 - e^{-z})^{L-N} \tag{5-8}$$

If the CCDF graph is plotted against the threshold values, the more vertical the graph is, the better is the PAPR reduction performance.

5.1.3 Eliminating distortion due to high PAPR

The effect of high PAPR on OFDM signals can be handled in a multiple number of methods which are described along with their pros and cons as below [34]:

- The peak transmit power may be limited by either regulatory or application constraints to reduce the average power allowed under multi-carrier transmission. However this technique reduces the range of multi-carrier transmission and obstructs one of the major advantages of OFDM to be used.
- The dynamic range of the power amplifier can be increased to accommodate the maximum power to be transmitted. But this technique is expensive as PAs with higher dynamic range comes at a greater cost. Again this requires hard-coding of the maximum allowed power which in turn limits the range of OFDM.
- The most feasible method yet improvised, is to use number of techniques that reduce the PAPR of the generated OFDM signal to an acceptable limit before being transmitted. These techniques require an extra set of computation but allow the OFDM range to expand as required and also do not add on to the overall cost of the transmitter as the processing can be done in baseband.

5.2 PAPR Reduction Techniques

Over the years, a number of techniques have been proposed for PAPR reduction of OFDM signals. The main objective of all these techniques is to reduce the PAPR of the OFDM signal to an acceptable value before the signal is sent for transmission. The PAPR reduction techniques can be broadly classified into following categories [36]:

- Amplitude Clipping and Filtering
- Coding
- Partial Transmit Sequence Technique
- Selected Mapping Technique

- Interleaving Technique
- Tone Reservation Technique
- Tone Injection Technique
- Active constellation extension technique
- Clustered OFDM
- Two-dimensional pilot symbol assisted modulation [used in coherent OFDM for channel estimation]

5.2.1 Criteria for selection of PAPR Reduction Techniques

There are a number of parameters or factors which are considered about the PAPR reduction techniques. Not all the criteria can be fully satisfied by any of the existing PAPR reduction techniques. A tradeoff is required between these factors to select the most appropriate technique depending on the system under consideration [34, 35].

The factors are as listed below:

1. ***PAPR Reduction capability***: The PAPR reduction capability is described by the reduction of PAPR value (in dB) after the technique is applied to OFDM transmission system. It is measured by CCDF graph.
2. ***Power Increase in transmit signal***: The technique must not increment the total power level that is being transmitted. If it does happen, the increment in power has to be within a permissible limit.
3. ***BER increase at the receiver***: The technique must not introduce unwanted errors into the transmitted bit stream, such that the overall BER at the receiver is increased. In other words the technique must not distort the signal.
4. ***Loss in data rate***: The technique may use some extra bits and this may result in a loss of data rate. The loss is acceptable up to certain value dependent on the system under consideration

5. ***Computational complexity***: The technique may satisfy all the other criteria but at the cost of a very high computational complexity. If this complexity is exceedingly high, the technique might not be suitable for hardware implementation as it will incur higher cost, power and time which are not desirable in speedy networks based on OFDM.

Table 5-1 presents the comparison of all these techniques based on the criteria presented above.

TABLE 5-1: COMPARISON OF PAPR REDUCTION TECHNIQUES [36]

Technique name	Power increase	Distortion-less	Loss in data rate	Computational Complexity
Amplitude clipping & filtering	No	No	No	Low
Coding	No	Yes	Yes	Medium
Partial Transmit Sequence	No	Yes	Yes	Very High
Selected Mapping	No	Yes	Yes	High
Interleaving	No	Yes	Yes	Medium
Tone Reservation	Yes	Yes	Yes	Medium
Tone Injection	Yes	Yes	No	Medium
Active constellation extension	Yes	Yes	No	Medium

The Partial Transmit Sequence technique has a very high computational complexity but delivers a remarkably good performance in terms of PAPR reduction. The higher the complexity of the technique and the more extensive the technique is, better is the PAPR reduction performance [39]. Hence this technique has been worked upon by a number of researchers with an objective to reduce the computational complexity so as to avail the PAPR reduction performance in an efficient manner.

5.3 Objective of the Work

The PTS technique is involves large number of complex computation which in turn demands more time, power and hardware resources thus increasing the cost as well. The

main objective of this work is to reduce the overall computation time of the existing PTS technique by implementing certain portions of the PTS algorithm in GPU. To realize the objective, the following analysis and investigations were required to be undertaken:

- To study and analyze the portions of the PTS algorithm that can be implemented in GPU so as to achieve higher computation throughput.
- To reduce the overall computation time of the PTS algorithm even when the number of sub blocks are increased to provide a better PAPR reduction performance.

5.4 PTS Technique: Description

The PTS technique was proposed by Müller and Huber in 1997 as a modification to the existing Selective Mapping technique [39]. The principle of the technique can be quoted as:

"***The coordination of appropriately phase rotated signal parts to minimize the peak power of the multiplex signal***"

The block diagram for PTS technique is shown in Figure 5-1.

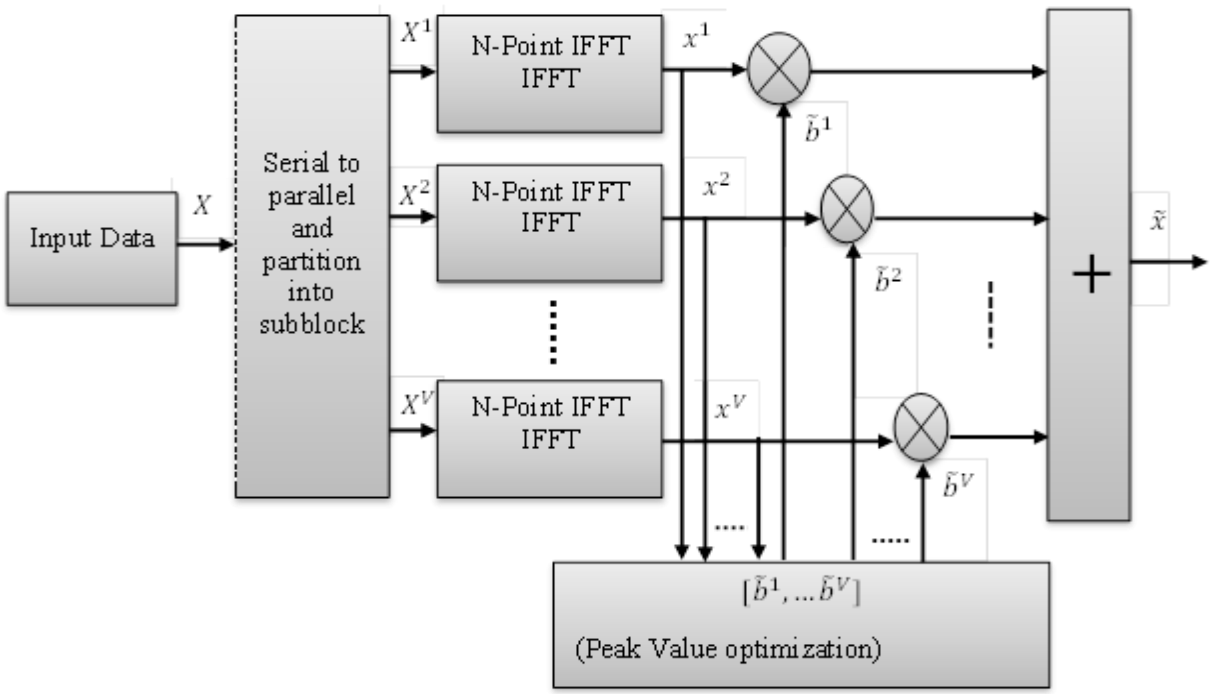

FIGURE 5-1 BLOCK DIAGRAM OF PARTIAL TRANSMIT SEQUENCE (PTS) TECHNIQUE

5.4.1 Mathematical Analysis of PTS Technique

The Partial sequence (PTS) technique partitions an input data of N symbols into V disjoint sub blocks as given below [32] :

$$X = [X^0, X^1 X^2, \ldots, X^{M-1}] \quad (5\text{-}9)$$

where X^i are the sub blocks that are consecutively located and are of equal size. Each of the sub-blocks is then passed through an N-point IFFT block.

After IFFT computation in each sub block scrambling (rotating its phase independently) is applied to each sub block. Each partitioned sub block is multiplied by a corresponding phase complex phase factors. Generally the first sub-block is left as it is, and rest of the sub blocks is rotated. The phase vector b is chosen such that the PAPR is minimized.

$$b^v = e^{j\varphi v}, \mathrm{v} = 1,2, \ldots \mathrm{V} \quad (5\text{-}10)$$

In general, the selection of phase factors is limited to a set of elements to reduce the search complexity. Let the number of allowed phase factors are W. So the phase factors can be expressed as:

$$b_w = e^{j\theta_w}, w = 0,1,..,W-1 \quad (5\text{-}11)$$

Generally the values used for phase factors are either [1,-1] or [1, +j, -j, -1] [40].

The outputs from each sub block after phase factor multiplication is added to obtain the corresponding PTS signal which given by:

$$x = IFFT\{\sum_{v=1}^{V} b^V X^V\} = \sum_{v=1}^{V} b^V . IFFT\{X^V\} = \sum_{v=1}^{V} b^V x^V \quad (5\text{-}12)$$

Where $\{x^V\}$ is referred to as the partial transmit sequence (PTS) and corresponding time domain signal with the lowest PAPR vector is denoted by $\tilde{x}$.

The PAPR performance of PTS technique is affected not only by the number of sub blocks V and the number of allowed phase factors, but also the sub block partitioning. Three different types of sub-block partitioning schemes have been proposed in literature [39]. These are adjacent, interleaved and pseudorandom. All of these sub-block division schemes assign equal number of non-zero sub-carriers to each sub-block. It has been proved mathematically by Müller and Huber that if the random distribution of sub-carriers is more, lesser is the correlation that exists among the sub-blocks. Hence pseudo-random sub block partitioning is known to provide the best PAPR reduction performance [41].

5.5 PTS Technique Implementation in GPU: Design Analysis

The partial transmit sequence involves large number of IFFT computation for each of the sub-blocks. These IFFT blocks itself have a complexity of $N*\log_2 N$ where N is the number of sub-carriers. When multiple numbers of such IFFT blocks are used, the complexity is increased manifolds thereby increasing the computation time. The process of formation of candidate signals after phase factor optimization also contributes significantly to the computation complexity. The entire process needed to generate one candidate signal is repeated W^{V-1} number of times and hence the overall complexity is increased.

The design approach has thus been concentrated on two major factors:

1. Reducing the PAPR by increasing the number of sub blocks.
2. Reducing the computation time of IFFT blocks by processing all the sub blocks in parallel using GPU.

In the PTS technique it is observed that as the number of sub block increases the PAPR performance improves [35]. But the search complexity increases exponentially with the increase in the number of sub blocks. Let N be the number of subcarriers in an OFDM system. Let us define the computation times associated with the PTS technique as presented in Table 5-2.

TABLE 5-2 COMPUTATIONAL TIME PARAMETERS IN PTS TECHNIQUE

t_{ifft}	Time taken for one N-point IFFT computation
t_{add}	Time taken for one N-point vector addition
t_{phase}	Time taken for one Phase rotation
t_{PAPR}	Time taken for PAPR computation of one N-point symbol
t_{cmpr}	Time taken for one PAPR comparison

In the PTS technique, one OFDM symbol will be generated in time t_{sym} given by:

$$t_{sym} = t_{ifft} + t_{phase} + t_{add} + t_{PAPR} \quad (5\text{-}13)$$

Let W be the number of allowed phase factors and V be the number of sub blocks. So the process will be completed to generate the OFDM symbol with minimum PAPR in t_{pts} given by:

$$t_{pts} = t_{sym} * W^{V-1} + t_{cmpr} * (W^{V-1} - 1) \quad (5\text{-}14)$$

In a typical simulation environment, the major computation time parameters of PTS technique are evaluated using sequential processing as listed in Table 5-3. Here the system uses 128 sub carriers and 64 sub blocks. From Table 5-3, it is observed that IFFT computation in sub blocks and searching optimum phase factors by computing PAPR of each N-point symbol consume the major computation resources in the PTS technique.

TABLE 5-3 EXECUTION TIME OF MAJOR BLOCKS OF PTS ALGORITHM IN A SEQUENTIAL PROCESSOR

Time Parameters	**Computation Time in CPU (ms.) For 1 sub block**	**Computation Time in CPU (s) for 16 sub blocks**
t_{ifft}	309	**12.58**
t_{add}	151	2.284
t_{phase}	116	3.303
t_{PAPR}	165	**13.670**
t_{cmpr}	242	**15.604**

In the proposed technique, GPU support is used at the sub blocks for computing the IFFT of the data blocks in parallel. The algorithm is implemented such that the IFFT processing at the sub blocks are executed in parallel using the GPU and rest of the processes run in the CPU. As a result, the computation complexity due to multiple IFFT blocks and repetition of the same process to generate large number of candidate signals has been reduced immensely because the IFFT points at each sub block are computed in parallel using hundreds of smaller cores of the GPU. Thus the computation throughput at the sub blocks become very high and it is observed that due to high performance gain in the sub blocks, the overall computation time for the PTS technique is reduced even when the number of sub blocks are increased to obtain a better PAPR performance.

5.6 PTS Technique using GPU: Implementation Details

The original PTS technique as presented in Figure 5-1 is simulated with GPU support at the sub blocks. The input data block is partitioned using adjacent sub block division method as shown Figure 5-2. [42].

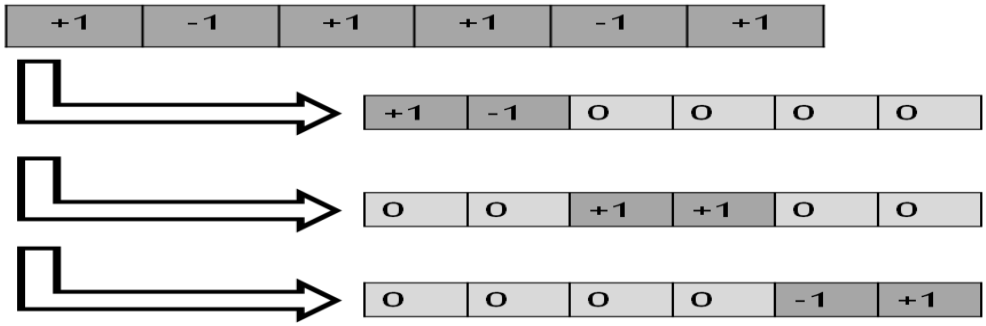

FIGURE 5-2: ADJACENT SUBBLOCK DIVISION METHOD

The implementation of the original PTS technique is modified to take advantage of the hundreds of parallel cores in the GPU thereby making efficient use of all cores in the system. To harness the parallel computing power of GPUs, the IFFT operations in the sub blocks are computed simultaneously in parallel. The remaining program runs in the GPU.

In this method of partitioning, if the number of sub-carriers N is a multiple of the number of sub-blocks V, then the first N/V sub-carriers are assigned to the first sub-block.

Similarly the second sub-block will have non-zero values for the next set of N/V sub-carriers. In this case, the correlation among the sub-blocks is very high.

5.7 Simulation results and discussion

PTS technique is implemented using the parameters shown in

TABLE 5-4 PARAMETERS FOR PTS IMPLEMENTATION

Number of sub blocks	2	4	8	16
Number of subcarriers	256			
Allowed phase factors	[1,-1]			
Symbol Mapping	BPSK Modulation			
Oversampling factor	4			

5.7.1 PAPR performance

The CCDF of PAPR using PTS technique is shown in Figure 5-3.It can be observed that PAPR performance improves as the number of sub blocks is increased. But with the increase in sub blocks, the computational complexity also increases exponentially.

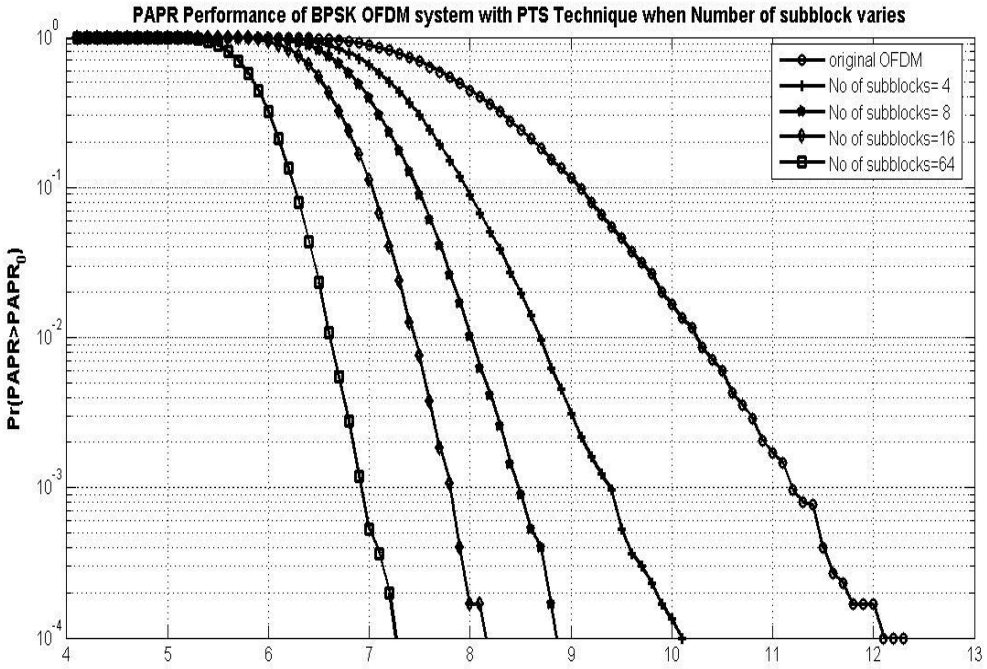

FIGURE 5-3: CCDF OF PAPR OF AN OFDM SIGNAL FOR DIFFERENT NUMBER OF SUBBLOCKS

5.7.2 Performance evaluation of PTS technique in GPU

In order to reduce computation time in PTS technique the sub blocks have been implemented in parallel. The data is transferred from CPU to the GPU. Following this sub block partitioning is done and the N point IFFT computations are done in parallel at all the sub blocks using GPU.

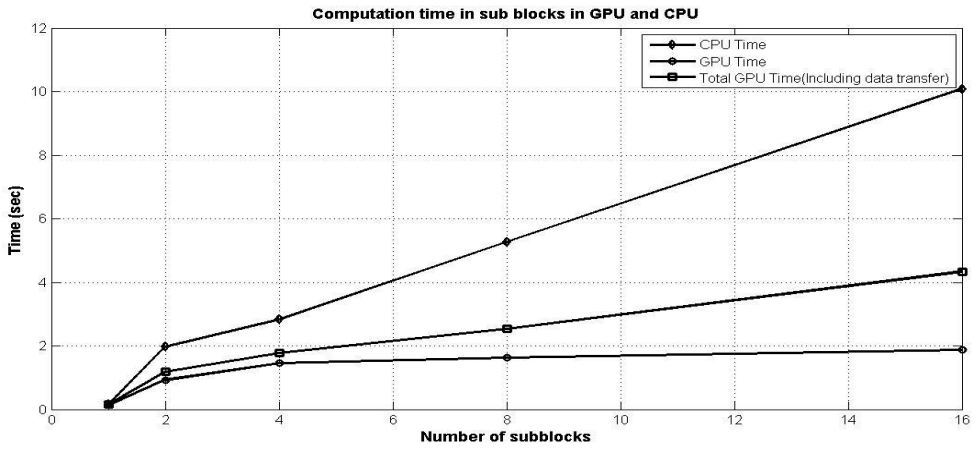

FIGURE 5-4: COMPARISON OF COMPUTATION TIME IN SUB BLOCKS IN GPU AND CPU

From Figure 5-4, it can be observed that the parallel implementation of sub blocks for computation of IFFT takes much lesser time as compared to sequential implementation. Though the data transfer from GPU to CPU is time consuming, yet overall computation performance gain is obtained in GPU for IFFT calculations in the sub blocks.

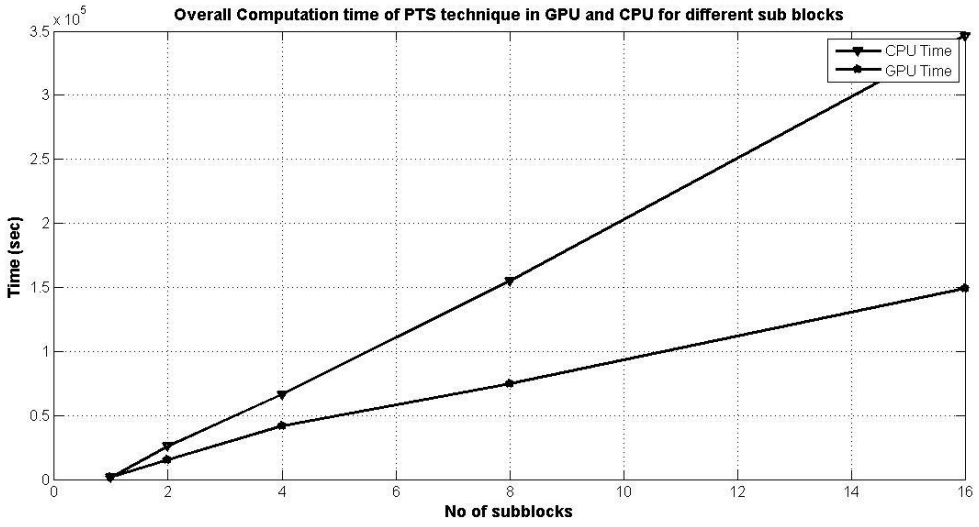

FIGURE 5-5: COMPUTATION TIME FOR PTS TECHNIQUE IN CPU AND GPU ENVIRONMENT

Figure 5-5 shows the comparison of overall computation time required to implement the PTS technique in a CPU and GPU environment. It can be observed that when sub blocks are increased, though the computation complexity increases due to increase in complexity for searching the optimum phase vector, yet an overall performance gain is obtained when sub block computations are implemented in parallel using GPU. For 16 sub blocks, the computation of PTS technique is nearly 2.5times faster than that in CPU.

Thus it can be concluded that the implementation of PTS technique with GPU support at the sub blocks can be very advantageous. The computation time becomes less and a better PAPR performance is obtained with the increase in the number of sub blocks.

6 CONCLUSION

High Performance Computing using parallel processing with GPU is very essential to meet computational challenges in terms of complexity and scalability of signal processing in wireless communication systems. In this project, baseband processing of various wireless communication systems based on OFDM and the signal processing techniques used in OFDM based systems such as MIMO-OFDM and PAPR reduction which require complex computations, were implemented using GPU. The CUDA architecture, using MATLAB/C programming languages, was utilized for programming the GPU. The major importance of GPU in wireless communication applications that uses OFDM is to make computations faster by parallel processing, thereby increasing the throughput of baseband signal processing which is the demand of next generation high data rate wireless communication systems.

A basic framework simulation assuming a simple LTE model, MIMO-OFDM models and PTS technique for OFDM has been efficiently implemented on GPU. The simulation model utilized the massively parallel architecture of GPU to reduce the computation time. The computation throughput of the GPU implementation is shown to outperform the conventional sequential implementation. The implementation of this new method is expected to provide promising ways to implement complex wireless communication systems using GPU based computing hardware.

6.1 Future Work

On the basis of this result, our future research work would include

- Implementation of OFDM based communication system on GPU based CUDA platform
- Implementation of high complexity communication system like SC-CDMA, MC-CDMA, WCDMA, V-Blast etc. on GPU and performance comparison with standard architecture
- Development of computationally efficient algorithms for OFDM based communication systems by providing parallel implementation of FFT and IFFT.
- Performance analysis of multicarrier OFDM systems for WLAN, WiMAX, MC-CDMA, MIMO-OFDM, LTE, 3GPP on GPU based computing platform.
- Implementation of signal processing algorithms for equalization, pre-equalization, PAPR reduction, optimization of PAPR reduction techniques and pilot carrier insertion for OFDM based system under GPU environment.
- Performance analysis of signal processing algorithms for wireless communication under constrained computing resources.

DISSEMINATION:

Bhattacharjee, S; Yadav, S.S.; Patra, S.K.; "**LTE PHYSICAL LAYER IMPLEMENTATION USING GPU BASED HIGH PERFORMANCE COMPUTING**", *IEEE International Conference on Advanced Communication, Control and Computing Technologies, 2014,* Syed Ammal College of Engineering, Ramanathapuram.

BIBLIOGRAPHY:

[1] T. Cooklev, Wireless Communication Standards: A Study of IEEE 802.11, 802.15, 802.16, Standards Information Network, 2004.

[2] K. Fazel and S. Kaiser, Multi-carrier and spread spectrum systems: from OFDM and MC-CDMA to LTE and WiMAX, Wiley, 2008.

[3] R. van Nee and R. Prasad, OFDM for wireless multimedia communications, Artech House, Inc., 2000.

[4] N. J. LaSorte, W. J. Barnes and H. H. Refai, "The History of Orthogonal Frequency Division Multiplexing," in *GLOBECOM*, 2008.

[5] A. G. Manushree Bhardwaj and D. Soni, A Review on OFDM: Concept, Scope & its Applications.

[6] H. Liu and G. Li, OFDM-based broadband wireless networks: design and optimization, John Wiley \& Sons, 2005.

[7] S. Weinstein and P. Ebert, "Data transmission by frequency-division multiplexing using the discrete Fourier transform," *Communication Technology, IEEE Transactions on*, vol. 19, no. 5, pp. 628-634, 1971.

[8] T. N. Bajwa, A. Khan and S. Baig, "Evolution of Orthogonal Frequency Division Multiplexing Modulation to Discrete Wavelet Multitone," in *Frontiers of Information Technology (FIT), 2011*, 2011.

[9] J. Nickolls and W. J. Dally, "The GPU computing era," *Micro, IEEE*, vol. 30, no. 2, pp. 56-69, 2010.

[10] S. W. Keckler, W. J. Dally, B. Khailany, M. Garland and D. Glasco, "Gpus and the future of parallel computing," *Micro, IEEE*, vol. 31, no. 5, pp. 7-17, 2011.

[11] [Online]. Available: www.nvidia.com.

[12] [Online]. Available: www.elmtec.co.uk.

[13] [Online]. Available: http://www.sgi.com/.

[14] E. Lindholm, J. Nickolls, S. Oberman and J. Montrym, "NVIDIA Tesla: A unified graphics and computing architecture," *Micro, IEEE,* vol. 28, no. 2, pp. 39-55, 2008.

[15] E. Wynters, "Parallel processing on NVIDIA graphics processing units using CUDA," *Journal of Computing Sciences in Colleges,* vol. 26, no. 3, pp. 58-66, 2011.

[16] *CUDA compute unified device architecture programming guide,* 2008.

[17] [Online]. Available: http://www.mathworks.in/company/newsletters/articles/gpu-programming-in-matlab.html.

[18] H. Holma and A. Toskala, LTE for UMTS OFDMA and SC-FDMA BBase Radio Access, John Wiley & Sons, 2009.

[19] *Technical Specification Group Radio Access Network; Physical Channels and Modulation (Release 8).*

[20] D. J. G. H. G. Myung, Single Carrier FDMA, John Wiley & Sons, 2008.

[21] D. Sinanovic, G. Sisul and B. Modlic, "Comparison of BER characteristics of OFDM and SC-FDMA in frequency selective channels," in *Systems, Signals and Image Processing (IWSSIP), 2011 18th International Conference on*, 2011.

[22] *Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) radio transmission and reception.*

[23] M. T. Kawser, LTE Air Interface Protocols, Artech House, 2011.

[24] E. a. R. E. M. Brigham, "The fast Fourier transform," *Spectrum, IEEE*, vol. 4.12, pp. 63-70, 1967.

[25] R. C. Agarwal, F. G. Gustavson and M. Zubair, "A high performance parallel algorithm for 1-D FFT," 1994.

[26] L. Hanzo, J. Akhtman, L. Wang and M. Jiang, "MIMO-OFDM for LTE, Wi-Fi and WiMAX," *A John Wiley and Sons, Ltd*, 2011.

[27] W. Zhang, X.-G. Xia and K. Ben Letaief, "Space-time/frequency coding for MIMO-OFDM in next generation broadband wireless systems," *Wireless Communications, IEEE*, vol. 14, no. 3, pp. 32-43, 2007.

[28] S. Alamouti, "A simple transmit diversity technique for wireless communications," *Selected Areas in Communications, IEEE Journal on*, vol. 16, no. 8, pp. 1451-1458, 1998.

[29] D. A. Gore and A. J. Paulraj, "MIMO antenna subset selection with space-time coding," *Signal Processing, IEEE Transactions on*, vol. 50, no. 10, pp. 2580-2588, 2002.

[30] Y. S. Cho, J. Kim, W. Y. Yang and C. G. Kang, MIMO-OFDM wireless communications with MATLAB, John Wiley & Sons, 2010.

[31] M. Wu, Y. Sun, S. Gupta and J. R. Cavallaro, "Implementation of a high throughput soft MIMO detector on GPU," *Journal of Signal Processing Systems*, vol. 64, no. 1, pp. 123-136, 2011.

[32] D. Sui, Y. Li, J. Wang, P. Wang and B. Zhou, "High throughput MIMO-OFDM detection with graphics processing units," in *Computer Science and Automation Engineering (CSAE), 2012 IEEE International Conference on*, 2012.

[33] S. H. Muller and J. B. Huber, "A novel peak power reduction scheme for OFDM," in *Personal, Indoor and Mobile Radio Communications, 1997. Waves of the Year 2000. PIMRC'97., The 8th IEEE International Symposium on*, 1997.

[34] S. H. Han and J. H. Lee, "An overview of peak-to-average power ratio reduction

techniques for multicarrier transmission," *Wireless Communications, IEEE,* vol. 12, no. 2, pp. 56-65, 2005.

[35] R. Prasad, OFDM for Wireless Communications Systems, Artech House, Inc., 2005.

[36] J. Hou, J. Ge and J. Li, "Peak-to-average power ratio reduction of OFDM signals using PTS scheme with low computational complexity," *Broadcasting, IEEE Transactions on,* vol. 57, no. 1, pp. 143-148, 2011.

[37] S. H. Muller and J. B. Huber, "OFDM with reduced peak-to-average power ratio by optimum combination of partial transmit sequences," *Electronics letters,* vol. 33, no. 5, pp. 368-369, 1997.

[38] S.-J. Ku, C.-L. Wang and C.-H. Chen, "A reduced-complexity PTS-based PAPR reduction scheme for OFDM systems," *Wireless Communications, IEEE Transactions on,* vol. 9, no. 8, pp. 2455-2460, 2010.

[39] Y. Xiao, X. Lei, Q. Wen and S. Li, "A class of low complexity PTS techniques for PAPR reduction in OFDM systems," *Signal Processing Letters, IEEE,* vol. 14, no. 10, pp. 680-683, 2007.

[40] P. Varahram, W. F. Al-Azzo and B. M. Ali, "A low complexity partial transmit sequence scheme by use of dummy signals for PAPR reduction in OFDM systems," *Consumer Electronics, IEEE Transactions on,* vol. 56, no. 4, pp. 2416-2420, 2010.

[41] J. J. Sanchez, D. Morales-Jimnez, G. Gmez and J. Enbrambasaguas, "Physical layer performance of long term evolution cellular technology," in *Mobile and Wireless Communications Summit, 2007. 16th IST*, 2007.

[42] C. Mehlfhrer, M. Wrulich, J. C. Ikuno, D. Bosanska and M. Rupp, "Simulating the long term evolution physical layer," in *Proc. of the 17th European Signal Processing Conference (EUSIPCO 2009), Glasgow, Scotland*, 2009.