LTE Physical Layer Implementation Using GPU Based High Performance Computing

Sangeeta Bhattacharjee, Satyendra Singh Yadav, Student *Member, IEEE**,
Sarat Kumar Patra, Senior *Member, IEEE***
Department of Electronics and Communication Engineering
National Institute of Technology, Rourkela, Odisha, India-769008
*sangeeta.bhatta@gmail.com, *yadav89satyendra@gmail.com, **skpatra@nitrkl.ac.in*

***Abstract*- 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 paper 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. 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. This paper presents a simulation model 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.**

Keywords: GPU, 3GPP LTE, OFDMA, SC-FDMA, Fast Fourier Transform.

I. INTRODUCTION

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 [1]. 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 [2, 3]. To support such a high data rate application, the system uses Orthogonal Frequency Division Multiplexing (OFDM) in conjunction with Multiple Input Multiple Output (MIMO) OFDM technologies. OFDM technology requires computation of IFFT and FFT at the transmitter and the receiver respectively. Implementing these along with baseband processing is challenging. The IFFT and FFT operations constitute the major computationally intensive and power consuming portion of transceiver design. Thus any performance gain in these blocks can potentially improve the throughput of the whole system significantly. GPU based computing is a recent paradigm in computational research [4]. It has evolved as general purpose GPU (GPGPU) processing technology allowing users to process large blocks of parallel data using an array of low complexity processors. It employs a large number of cores to execute a common set of data operations in parallel.

This paper presents the implementation of LTE physical layer base station with GPU support. The paper considers implementation of transmitter and receiver at the base station for LTE downlink and uplink respectively. Signal processing blocks like pre-equalizer, equalizer, symbol mapper, Peak to Average Power Ratio (PAPR) reduction have not been analyzed. Preliminary results of the simulation of basic framework have been presented.

Following this introduction, the remaining paper is organized as under. Section II discusses the system model with overview of the LTE uplink and downlink physical layer. Section III presents an introductory discussion on GPU architecture. Section IV presents the simulation framework along with a discussion on the result. Section V provides the concluding remarks.

II. LTE PHYSICAL LAYER: OVERVIEW

This section presents a brief description of the physical layer functionalities for LTE uplink and downlink [2, 5-9]. The implementation challenges and a possible solution with GPU support are also presented.

A. *LTE Downlink Physical Layer*

Orthogonal Frequency Division Multiple Access (OFDMA) is the multiple access technique used for LTE downlink transmission [5-7]. The block diagram of OFDM system is shown in Fig. 1. 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. The radio frame structure for LTE in

Frequency Division Duplex (FDD) mode of operation is shown in Fig.2.

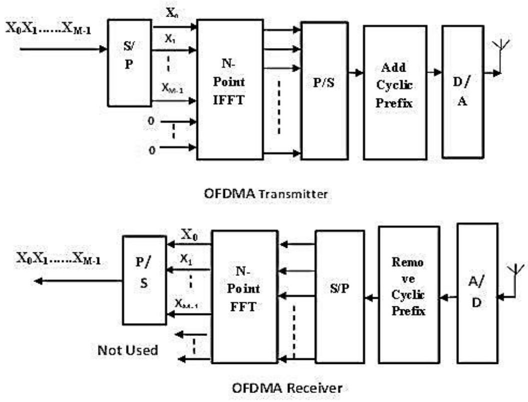

Fig. 1. OFDMA Transmitter and Receiver.

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 [5].

The LTE system parameters for different channel bandwidth are presented in Table I. 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 [2].

B. LTE Uplink Physical Layer

In the uplink LTE uses Single Carrier Frequency Division Multiple Access (SC-FDMA) technique for multiple access [5, 8]. The SC-FDMA block diagram is presented in Fig. 3. 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 [8]. 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. 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 I.

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.

C. 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 II. Here the system uses 140 OFDM symbols with 1200 subcarriers and 2048 point FFT and IFFT based on LTE 200MHz specification.

TABLE I. PARAMETERS FOR DOWNLINK TRANSMISSION

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

From Table II, 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.

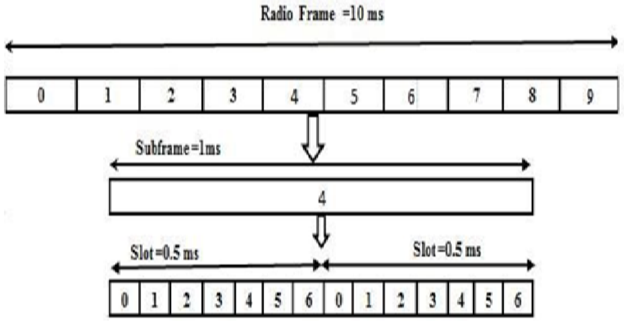

Fig. 2. LTE Radio Frame Structure

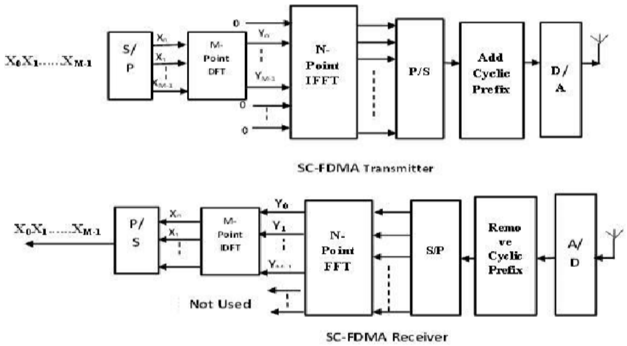

Fig. 3. SC-FDMA Transmitter and Receiver.

TABLE II. 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

D. 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,

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \quad ,0 \leq k \leq N \tag{1}$$

$$W_N = e^{-j2\pi/N} \tag{2}$$

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

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

The 1st term in (3) 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 Fig. 4. The structure is typically called a butterfly structure.

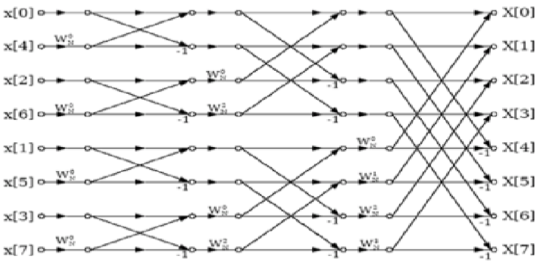

Fig. 4. Flow Graph of Decimation In-Time FFT Algorithm

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 [13]. 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.

III. GPU ARCHITECTURE

The block diagram of a GPU (Graphics Processing Unit) based high performance computing system is presented in Fig. 5. A GPU based system uses multiple processors [10]. 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 [11]. These systems have the capability of high processing throughput through parallelization. CUDA (Compute Unified Device Architecture) is a parallel computing platform and programming model created by NVIDIA and implemented on the GPUs that they produce. It enables increase in computing performance by harnessing the power of GPU.

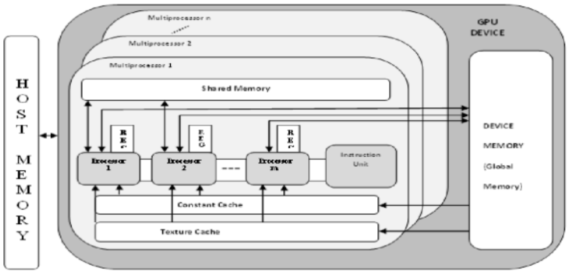

Fig. 5. NVIDIA GPU Architecture.

In CUDA, parallel portion of the application is executed as a kernel [12]. 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 Fig. 6. Threads are virtually mapped to an arbitrary number of streaming multiprocessors (SM) and are executed in 32 parallel thread groups called warps.

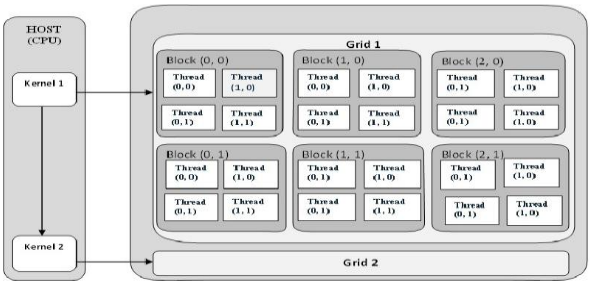

Fig. 6. Parallel Execution of a GPU Kernel in a Grid of Thread Blocks

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.

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 I. 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.

IV. NUMERICAL RESULTS AND DISCUSSION

The performance of LTE transceiver was evaluated through simulation. The simulations were performed using MATLAB software. The system hardware consists of a host PC with Intel® Xeon®, E5-2650 processor operating at 2.0 GHz with Linux operating system. NVIDIA Tesla M2090 graphics card with 512 cores at 1.3 GHz processor clock was used. Parallel computing toolbox in MATLAB was used for simulation in the GPU environment.

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.

A. 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 I. 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 Fig. 7. The time is calculated using the host clock function. From Fig. 7, 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.

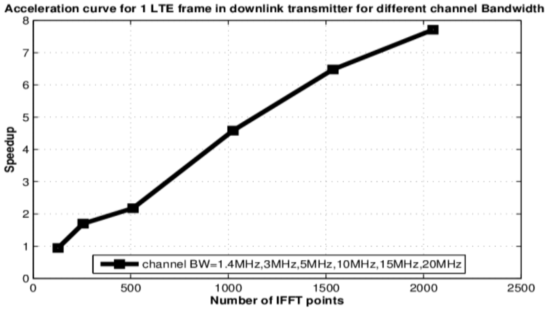

Fig. 7. GPU Performance For transmitting 1 LTE Downlink Frame at Different Channel Bandwidth.

The Symbol error rate curve obtained using Monte Carlo simulation for AWGN channel is shown in Fig. 8. The Monte Carlo simulation with GPU support takes much lesser time to simulate and is completely justified.

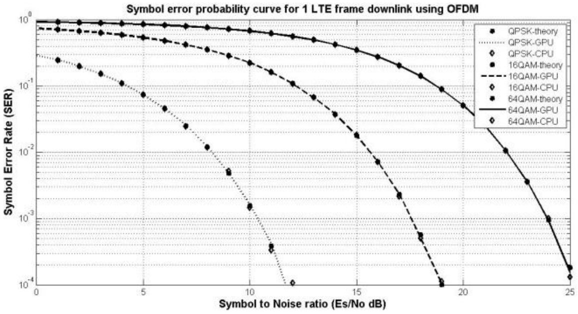

Fig. 8. Symbol Error Rate Curve For LTE Downlink in AWGN Channel.

B. 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 I. 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 speed up obtained for different users with 20MHz bandwidth per user is presented in Fig. 9. Table III, shows the throughput comparison for FFT and IDFT computation in the host CPU and GPU device.

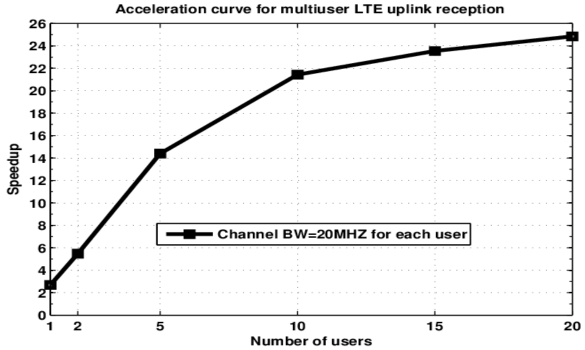

Fig. 9. GPU Performance for Processing Received LTE Uplink Frame for Different Number of Users.

TABLE III. PROCESSING THROUGHPUT FOR HOST CPU AND GPU DEVICE

No of user s	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

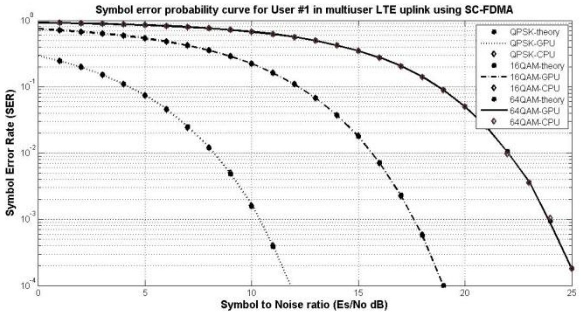

Fig. 10. Symbol Error Rate for User #1 in LTE Uplink.

From Table III, 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. The SER performance for SC-FDMA in LTE uplink in AWGN channel for user #1 transmitting at 20MHz bandwidth is evaluated and presented in Fig. 10.

V. CONCLUSION

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.

On the basis of this result, our future research work would include GPU based implementation of LTE user equipment architecture and performance evaluation of complex signal processing techniques for equalization, PAPR reduction and MIMO for LTE.

VI. REFERENCES

[1] 3GPP, “Technical specification group radio access network; (E-UTRA) and (E-UTRAN); overall description; stage 2," Sep. 2008. http://www.3gpp.org/ftp/Specs/html-info/36300.htm.

[2] Mohammad T. Kawser, LTE Air Interface Protocols, Artech House, 2011.

[3] Ghosh, Amitava, Rapeepat Ratasuk, Bishwarup Mondal, Nitin Mangalvedhe, and Tim Thomas. “LTE-advanced: next-generation wireless broadband technology,” IEEE Wireless Communications, *IEEE* 17, no. 3 (2010), pp. 10-22.

[4] Stephen W. Keckler, William J. Dally, Brucek Khailany, Michael Garland, David Glasco, “GPUs and the future of parallel computing”, IEEE Micro, vol. 31, 2011, pp. 7-17.

[5] 3GPP TS 36.211, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation".

[6] Ciochina, Cristina and Sari, Hikmet, “A review of OFDMA and single-carrier FDMA and some recent results”, Advances in Electronics and Telecommunications, vol.1, 2010,pp. 35-40.

[7] Holma, Harri and Antti Toskala, LTE for UMTS-OFDMA and SC-FDMA based radio access, John Wiley & Sons, 2009.

[8] Myung, Hyung G, Lim, Junsung and Goodman, David J, “Single carrier FDMA for uplink wireless transmission”, IEEE Vehicular Technology Magazine, vol.1, 2006, pp. 30-38.

[9] Mehlfhrer, Christian, Wrulich Martin, Ikuno Josep Colom, Bosanska Dagmar and Rupp Markus, “Simulating the long term evolution physical layer”, Proc. of the 17th European Signal Processing Conference (EUSIPCO 2009), Glasgow, Scotland, vol. 27, 2009, pp. 124.

[10] Erik Lindholm, John Nickolls, Stuart Oberman, John Montrym: NVIDIA, “NVIDIA Tesla: a unified graphics and computing architecture”, IEEE Micro, vol. 28, 2008, pp.39-55.

[11] John Nickolls, William J. Dally: NVIDIA, “The gpu computing era”, IEEE Micro, vol.30, 2010, pp. 56-69.

[12] NVIDIA Corporation, “CUDA compute unified device architecture programming guide,” 2008.

[13] Agarwal, Ramesh C and Gustavson, Fred G and Zubair, Mohammad, “A high performance parallel algorithm for 1-D FFT”, Proceedings of the 1994 ACM/IEEE conference on Supercomputing, 1994, pp. 34-40.