

Design of Scalable FFT architecture for Advanced Wireless Communication Standard

Rutuja C. Tamhane¹, Shrikant J. Honade²

¹(EXTC , G.H.Raisoni College of Engineering And Management, Sant Gadge Baba Amravati University, Amaravti, India)

²(EXTC Department, G.H.Raisoni College of Engineering And Management, Sant Gadge Baba Amravati University Amaravti, India)

Abstract: Now a day's numerous wireless communication standards have raised additional stringent requirements on each throughput and flexibility for FFT computation. Advanced wireless systems support multiple standards to satisfy the demands of user application necessities. A wireless system whereas supporting multiple standards should also satisfy performance necessities of these supported standards. Meeting performance requirements of multiple standards is a challenge while designing a system. Fast Fourier transformations, a kernel processing task in communication systems, are studied intensively for efficient software and hardware implementations. To design an efficient system, it's necessary to efficiently design its performance critical component. Each system must meet stringent design parameters like high speed, low power, low area, low cost, high flexibility and high scalability, designing FFT processor to support multiple wireless standards whereas meeting the above such performance necessities is a difficult task. This paper proposed a highly efficient scalable architecture, software tools design, and design implementation. The reconstruction of the FFT computation flow is design into a scalable structure. The FFT can be easily expanded for any-point FFT computation. The various parameters satisfied the conditions, gives proper and efficient outputs as compare to other platforms.

Keywords: Code composer studio 3.1, C compiler, Connecting wires, DSK kit, MATLAB, TMS320C6713, Spectrum analyzer.

I. Introduction

In today's era, the demand for high-bit-rate services in wire-less communication system as well as in wired communication system is growing. Fast Fourier Transformation (FFT), the foremost time consuming block in data communication systems, is facing each high flexibility reprogrammability and high data handling capacity requirements in current wireless systems. It should be simply reprogrammed or recon-figured to support numerous standards and operational modes. For instance, the scale of FFT is desired to be change-able under completely different operation environments. The present application-specific integrated circuits (ASICs), cannot provide the specified flexibility and reprogrammability. The fast Fourier transform (FFT) and inverse FFT (IFFT) algorithms are core processing blocks used for conversion from time-to-frequency domain (FFT) and from frequency-to-time domain (IFFT) and represent the foremost computation intensive tasks.

The baseband hardware must be efficient and capable enough to compute FFT among the time constraints necessary to support various wireless standards. Baseband hardware should be scalable so it supports multiple wireless standards likewise as it should meet the performance constraints like high speed, low space and low power consumption. Hence, the system needs a scalable FFT module that meets the demands of end user applications.

WHAT IS FFT?

There are many ways in which to calculate the discrete Fourier transform (DFT), like solving simultaneous linear equations or the correlation technique. The fast Fourier transform (FFT) is another technique for calculating the DFT. The discrete Fourier transform converts a time-domain sequence into identical frequency-domain sequence. The inverse discrete Fourier transform performs the reverse operation and converts a frequency-domain sequence into identical time-domain sequence. The fast Fourier transform (FFT) may be a very efficient algorithmic program technique based on the discrete Fourier transform however with fewer computations needed. The FFT is one among the foremost usually used operations in digital signal process to provide a frequency spectrum analysis. 2 different procedures are introduced to calculate an FFT, the decimation-in-frequency and also the decimation-in-time. Whereas it produces an equivalent result as the other approaches, it's incredibly more efficient, usually reducing the computation time by hundreds.

Frequency analysis of distinct signal is conveniently per-formed on digital signal processors. So as to perform such an analysis one has to transform the signal from time domain to frequency domain representation.

FFT is itself not a transformation however simply a computational algorithm to evaluate discrete Fourier transform (DFT).The N-point discrete Fourier transform (DFT) X (k) of an N-point sequence x(n) is by definition:

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

where the twiddle factor W_N is given by:

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

The overall flow of the FFT is given in below figure.

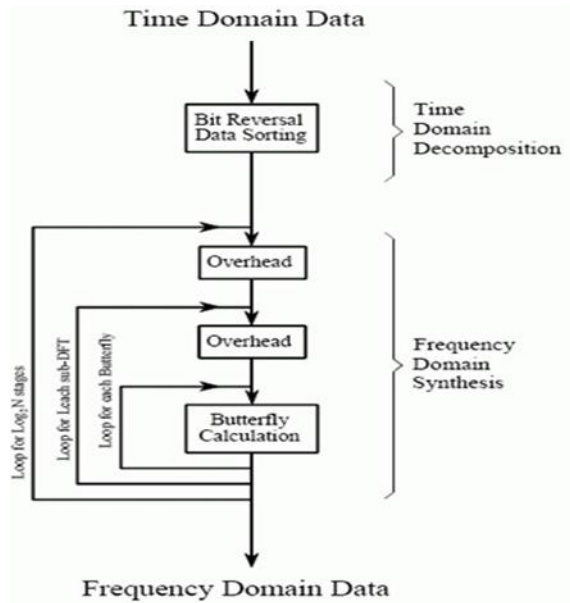


Fig.1-Flow diagram of FFT

II. Objective

Currently in the field of signal processing for communications, there is a rapid development evolving programming formats and in algorithms which act as a key in designing a system. Fourier Transform, Discrete Fourier Transform (DFT), Fast Fourier Transform (FFT) are basis for many signal processing and communication based applications. It is the analysis of the signal in time and frequency domain. From the Fourier Transform of the discrete signals we can develop Discrete Fourier Transform (DFT) and Fast Fourier Transform (FFT).Fast Fourier Transform (FFT) is an efficient way to compute DFT. Although a lot of work on FFT processing in both software and hardware has been done, there is still need of both flexibility and high throughput.

- Design the scalable architecture for FFT Processing.
- Simulation of the proposed design.
- Testing of the architecture for parameter like time.

III. Proposed System

Scalability is the capability of a system, network, or process to handle a growing amount of work. In other words, capability of a system to increase its total output under an increased load when new resources are added. An algorithm, design, networking protocol, program, or other system is said to be scalable if it is suitably efficient and practical when applied to large situations. The FFT operates by decomposing an N point time domain signal into N time domain signals every composed of one point. The second step is to calculate the N frequency spectra equivalent to these N time domain signals. Lastly, the N spectra are synthesized into one frequency spectrum. Here scalable FFT is implemented using code composer studio with DSK kit as shown in the figure 2.

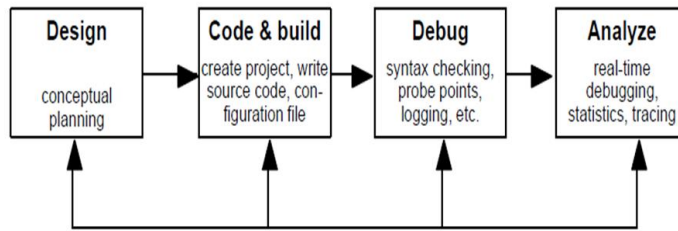


Fig.2-Generalized flow of the system

Experimental Results

We examined the proposed design of Scalable FFT using code composer studio and DSK kit with the existing techniques and obtained the results by using different cases as shown below and the time parameter is obtained which is desirable. Table 1 and Table 2 shows the comparative analysis of different processors. Case 1 and case 2 are explained in detail likewise we can obtain results for different points of FFT.

Case 1-

For N=8point- Figure 3 shows the decomposition of an N-point DFT into 2 (N/2)-point DFTs, for N = 8. As a results of the decomposition method, the X's in Figure 3 are even within the upper half and odd within the lower half. The decomposition method will now be repeated such that every of the (N/2)-point DFTs is further decomposed into 2 (N/4)-point DFTs.

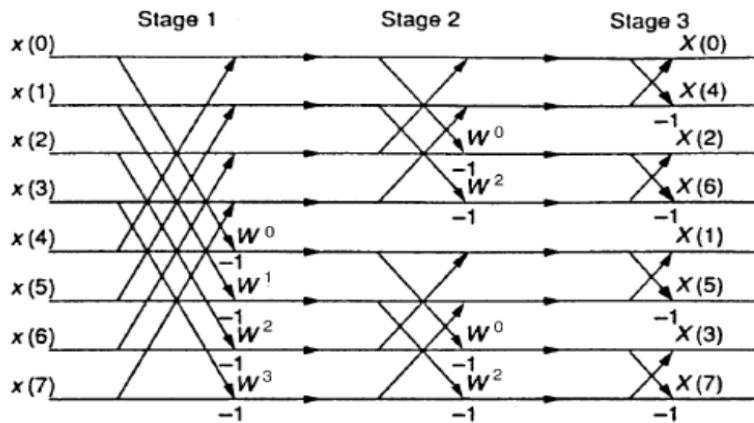


Fig .3- 8 point butterfly architecture

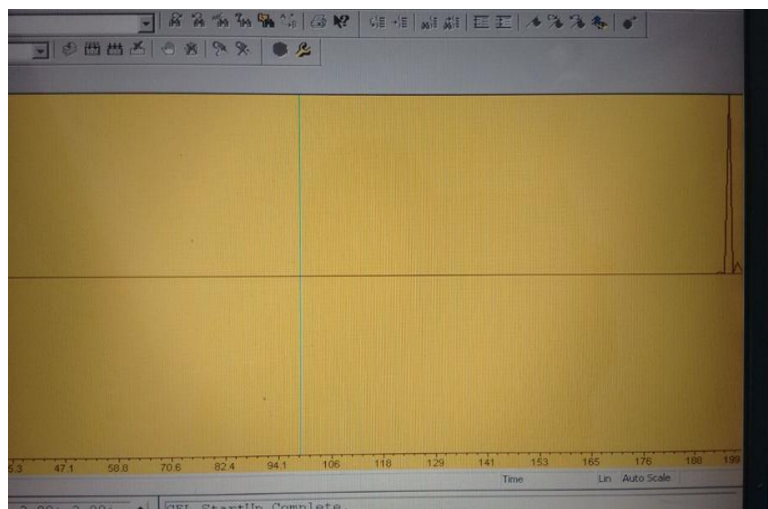


Fig .4-Output on CCS of 8 point FFT

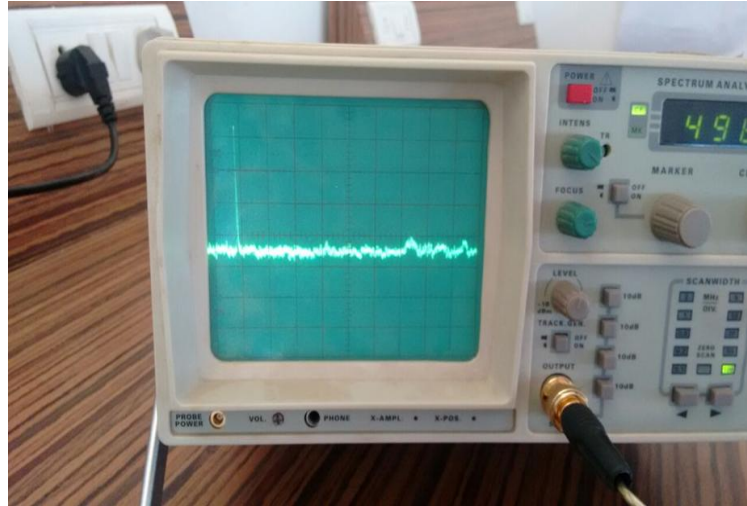


Fig.5-Output on spectrum analyzer

Case 2-
N= 16 point

For N=16 point- Figure 6 shows the decomposition of an N-point DFT into two (N/2)-point DFTs, for N = 16. As a result of the decomposition process, the X's in Figure 6 are even in the upper half and odd in the lower half. The decomposition process can now be repeated such that each of the (N/2)-point DFTs is further decomposed into two (N/4)-point DFTs.

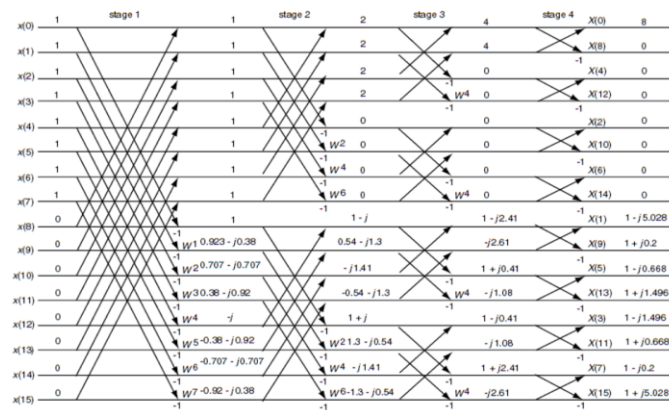


Fig.6- 16point FFT butterfly

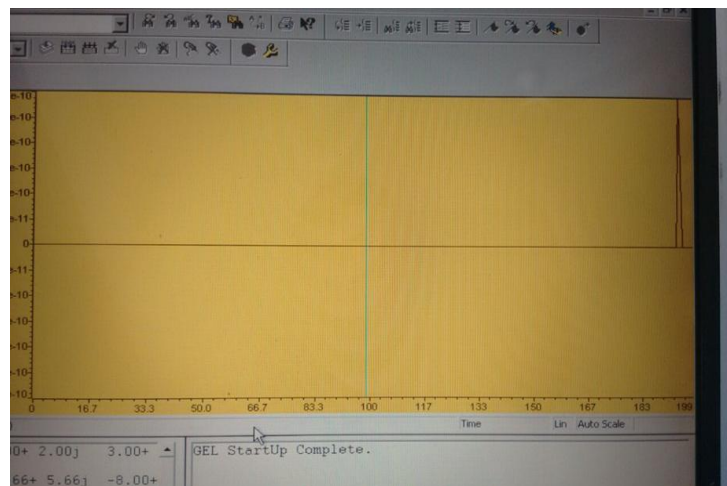


Fig.7-Output on CCS of 16 point FFT

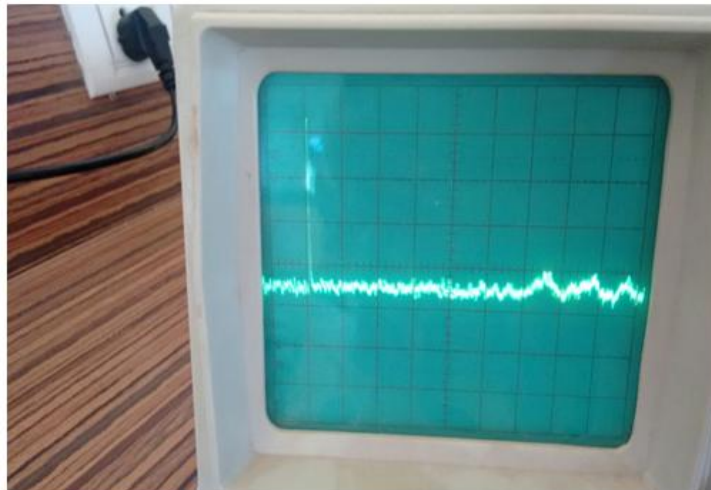


Fig.8-Output on spectrum analyzer

Case 3-
N=32 points decomposition process shows the following output.

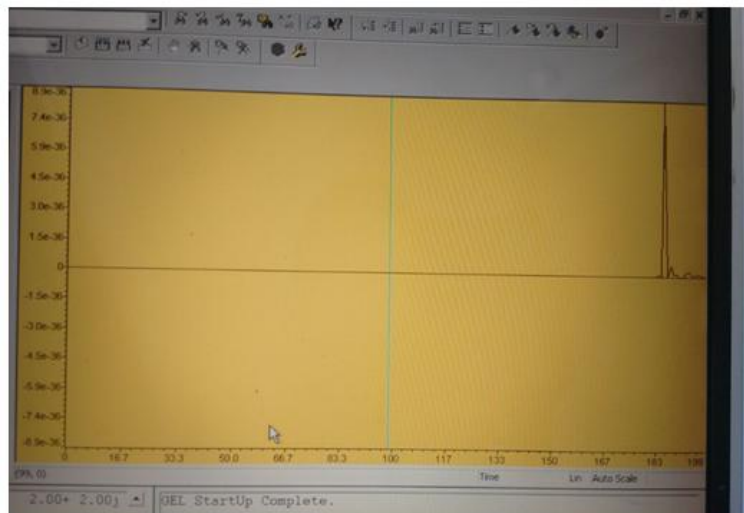


Fig.9-Output on CCS of 32 point FFT

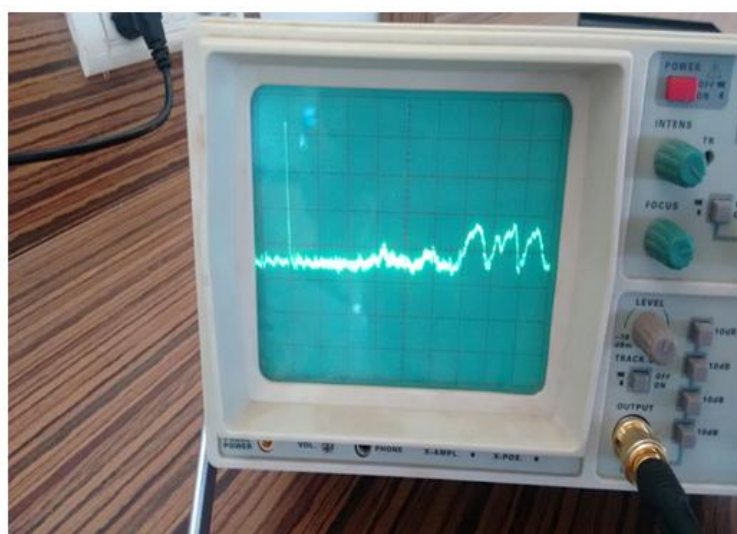


Fig.10-Output on spectrum analyzer

Table No.1-Results with existing and proposed method

Sr.No	Points	MATLAB Dual core i2 (μsec)	TMS3206713 (μs)
1.	8 points	15882400	24
2.	16points	12816247	64
3.	32points	10497500	160
4.	64points	13829500	384
5.	512points	14690000	4608

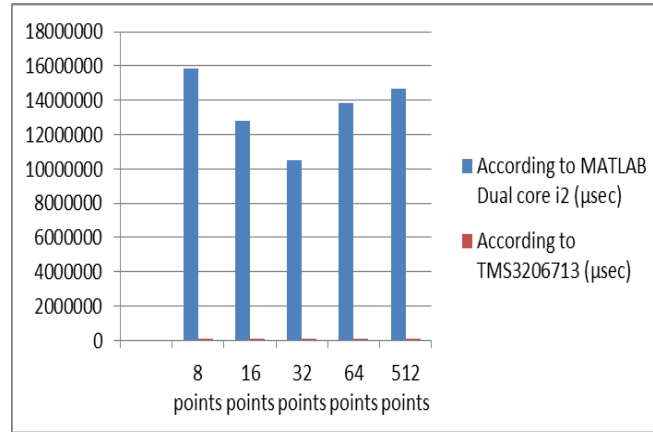


Fig.12 - Graphical representation of existing and proposed method

Table No.2- Comparative analysis of existing and proposed Method

Sr.no	Points	TMS3206713 (μs)	According to Xtensa (μs)
1.	8 points	0.07	31.4
2.	16 points	0.2	31.4
3.	32 points	0.5	31.4
4.	64 points	1.2	31.4
5.	512points	14.4	31.4

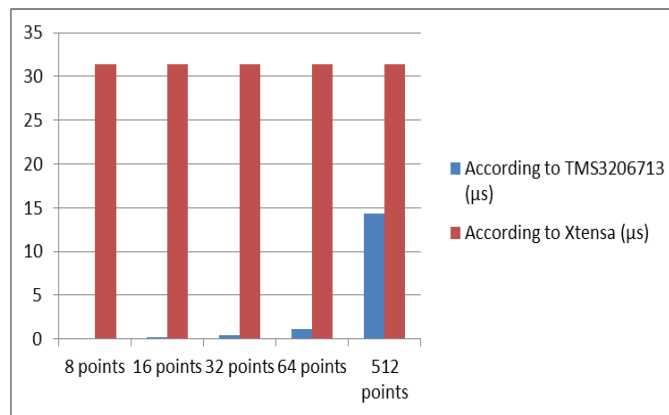


Fig.13 - Graphical representation of existing and proposed method

IV. Applications

FFT and IFFT are the most important blocks in signal processing and image processing, for the conversion of domain as per the applications. Some of the important applications of FFT include-

- For large integer multiplication.
- Efficient matrix-vector multiplication, Filtering algorithms.
- Fast algorithms for discrete trigonometric transforms.
- For JPEG, MP3/MPEG encoding.
- Recovering the non-transformed coefficients.
- Fast Chebyshev approximation.
- Fast Discrete Hartley Transform.
- Solving Difference Equations.

- For Image compression.
- Wireless communication systems like OFDM, UWB, WIMAX, LTE etc.

V. Conclusion

This dissertation proposed a FFT design, which will be flexible and efficient to meet the demands of digital communication standard and wireless communication standard. The structure provide good scalability to any point FFT, and it is easily reprogrammed, flexible for both hardware and software implementation. The cost, and power consumption is acceptable. The results show that the time parameter obtained is better as compared to other platforms.

References

- [1] Xuan Guan, Yunsi Fei and Hai Lin, "Hierarchical Design of an Application-Specific Instruction Set Processor for High-Throughput and Scalable FFT Processing," *March 2012 IEEE. VOL. 20, NO. 3*, pp.551-563
- [2] Deepak Revanna, Omer Anjum, Manuele Cucchi, Roberto Airoidi, Jari Nurmi, "A Scalable FFT Processor Architecture for OFDM Based Communication System," *2013 IEEE* Pp.19-27
- [3] Yuan Chen, Yu-Wei Lin, and Chen-Yi Lee National Chiao Tung University, Hsinchu, Taiwan MediaTek Inc., Hsinchu, "A Block Scaling FFT/IFFT Processor for WiMAX Applications," *2006 IEEE* Pp.203-206.
- [4] Wei Hun, A.Erdogan, Arslun, und M. Hasan, "The Development of High Performance FFT IP Cores through Hybrid Low Power Algorithmic Methodology," *2005 IEEE*. Pp.549-552.
- [5] Yu-Wei Lin, Hsuan, Yu Liu, and Chen-Yi Lee, "A 1-GS/s FFT/IFFT Processor for UWB Applications," *2005 IEEE*.Pp.1726-1735
- [6] Ramesh Chidambaram, Rene van Leuken Marc Quax, Ingolf Held, Jos Huisken, "A Multistandard FFT Processor for Wireless System-on-Chip Implementations," *2006 IEEE*.pp.1099-1102
- [7] Kiran George and Chien-In Henry Chen, "Configurable and Expandable FFT Processor for Wideband Communication," *2007 IMTC* Pp.1-6.
- [8] Adnan Suleiman, Hani Saleh, Adel Hussein, and David Akopian, "A Family of Scalable FFT Architectures and an Implementation of 1024-Point Radix-2 FFT for Real-Time Communications," *2008 IEEE* PP-321-327
- [9] Xuan Guan, YunsiFei, and HaiLin, "A Hierarchical Design of an Application-specific Instruction Set Processor for High-throughput FFT," *2009 IEEE*.pp.2513-2516.
- [10] Youn Ok Park Jong-Won Park, "Design of FFT Processor for IEEE802.16m MIMO-OFDM Systems," *2010 IEEE* Pp.191-194
- [11] J.Manikandan, B.Venkataramani, K.Girish, H.Karthic and V.Siddharth, "Hardware Implementation of Real-Time Speech Recognition System using TMS320C6713 DSP," *2011 IEEE* Pp.250-255.
- [12] Mingxi Sun, LiyuTian and Dongmin Dai, "Radix-8 FFT Processor Design Based on FPGA," *2012 IEEE* Pp.1453-1457
- [13] Yazan Samir Algnabi, Furat A. Aldaamee, Rozita Teymourzadeh, Masuri Othman and Md Shabiul Islam, "Novel Architecture of Pipeline Radix 22 SDF FFT Based on Digit-Slicing Technique" *2012 IEEE* PP-470-474
- [14] Shang-Ho Tsai, Kai-Jiun, "MDC FFT/IFFT Processor With Variable Length for MIMO-OFDM Systems" *APRIL 2013 IEEE, VOL. 21, NO. 4*,
- [15] Navneet Basutkar, Peng Xue, Kyeong yeon Kim, and Young-Hwan Park, "Software-Defined DVB-T2 Demodulator using Scalable DSP Processors" *May 2013 Vol. 59, No. 2,IEEE*.
- [16] Manohar Ayinala, Yingjie Lao, IEEE, and Keshab K. Parhi, "An In-Place FFT Architecture for Real-Valued Signals" *OCTOBER 2013 VOL. 60, NO. 10*,
- [17] Ting Chen1a, Xiaowei Pan, Hengzhu Liu, Tiebin Wu, "Rapid Prototype and Implementation of a High-Throughput and Flexible FFT ASIP Based on LISA 2.0," *2014 IEEE*.pp.681-687
- [18] J.Greg Nash, "High-Throughput Programmable Systolic Array FFT Architecture and FPGA Implementations," *Presented at the 2014 International Conference on Computing, Networking and Communications (ICNC), CNC Workshop (ICNC'14 - Workshops - CNC), Honolulu, HI, Feb 2014*.
- [19] Rulph Chassaing, "DSP Applications using C and TMS320C6X DSK", *A Wiley-Interscience Publication JOHN WILEY & SONS, INC,2002*
- [20] Texas Instruments, "TMS320C6713B, Floating-Point Digital Signal Processors, Data Sheet", Dallas, TX, June 2006
- [21] Steven W. Smith, "The Scientist and Engineer's Guide to Digital Signal Processing, Second Edition", *California Technical Publishing, 1999*.