Review on low power high speed 32 point cyclotomic parallel FFT Processor

Ku. Sujata B. Telrandhe1, Associate Prof. M. N. Thakre2

M.Tech Student, Department of Electronics and Telecommunication, B.D.C.O.E, Wardha, Maharashtra, India1
Assistant Professor, Department of Electronics and Telecommunication, B.D.C.O.E, Wardha, Maharashtra, India2

---***---

Abstract - *In this paper, the review of a low power 32 point Cyclotomic parallel Fast Fourier Transform (FFT) Processor has be presented. The Fast Fourier transform is one of the rudimentary operations in field of digital signal and image processing. FFT is a technique which efficiently calculates the DFT by reducing the number of addition and multiplication operations takes place. Cyclotomic FFT is the type of FFT algorithm DFT into several convolutions. This paper shows tradeoff between area and performance. Cyclotomic FFT will help to reduce this trade off over FFT. This paper concentrates on the developments of 32-point Radix-4 FFT by using VHDL as a design entity. The parallel FFT and the combinations of pipeline FFT will be use to improve the efficiency and speed. The synthesis results will show the computation for calculating the FFT by VHDL and their performance.*

Key Words: fast fourier transform (FFT), Radix, VHDL, Cyclotomic, Pipeline FFT, Butterfly, Parallel FFT.

1.INTRODUCTION

The FFT processor is widely used in mobile systems for image and signal processing applications. The requirement for low-power FFT architectures for telecommunication systems in portable form is becoming more and more important. Due to the characteristic of non-stop processing at sample rate, the pipelined FFT is the leading architecture for high throughput or low-power solutions. In pipelined architectures, power consumption is dominated by the commutator and the complex multiplier at each stage.

This proposes the design of 32-points FFT processing block. The work of the project is focused on the design and implementation of FFT. This design computes 32-points FFT and all the numbers follow fixed point formatin to the frequency domain. The pipelined FFT is viewed as the leading architecture for real time applications. However, the use of only one processor element (PE) in each stage limits the throughput of pipelined FFTs. Therefore, an increased throughput requires further parallelization. pipeline architecture is a special class of FFT architecture, which is able to compute the FFT in a sequential manner as they can easily merged with sequential nature of sampling. So the architecture is suitable for real–time applications.

Previous research work is based on either cyclotomic based pipelining or multiplier based parallel processing. Each of these designs have their own advantage. In our research work, we will be combining the advantages of both cyclotomic pipelining and parallel multipler in order to get the best design in terms of power and delay.

Initially, a parallel multiplier based FFT butterfly unit will be created. This unit will consist of an upper and a lower node. Upper node performs addition, lower node performs subtraction. Both of these nods are combined to form a butterfly. For a 32 point FFT, we will need 16 nodes, as 1 node can process 2 inputs. The connection of these nodes will be done on the basis of cyclotomic pipelined design, where we will make sure that when stage 1 processing is done, then stage 2 starts working on the ouput of stage 1, and stage 1 starts working on the new inputs. There by improving the overall throughput and power efficiency of the system.

2. LITERATURE REVIEW

In this paper entitled “Analysis and Design of Low Power Radix-4 FFT Processor using Pipelined Architecture” proposed on low power technique for fast fourier transform. This strategy for Fast Fourier Transform is an elevated form of Discrete Fourier Transform which is much simpler, effective, and faster with lesser number of computations has dominated in various fields. Several low power techniques like sign swap, sub expression elimination alongwith several area reduction techniques like “In Place” addressing, single butterfly element per stage using the pipelined architecture. Pipelined architecture with low power techniques is implemented on both radix-2 and radix-4 FFT processor and compared. Results shows that pipelined Radix-4 FFT consumes11% less power compared to radix-2 FFT for 16 point implementation. Number of clock cycles required for the radix-r algorithm is given by (N/r)/(log N), so for computing 64 point FFT radix-4 algorithm requires 48 clock cycles where to implement the same 64 point FFT radix 2 algorithm requires 192 clock cycles which means radix-2 algorithm is 4 times slower in comparison with the radix-4 algorithm. 16 point radix-4 FFT required of 2 stages and 16 point radix-2 FFT required of 4 stages. Pipelined architecture results in less area consumption since the number of butterfly elements are reduced significantly, By considering implementation of radix-2, 16 point FFT which requires 32 butterfly units where as in the case of pipelined FFT the number of butterfly units required is just one per stage so for implementing radix-2, 16 point requires only 4 butterfly units since it has 4 stages involved, similarly designing 16 point radix-4 requires only 2 butterfly units. This concludes that area occupancy in the case of pipelined architecture is significantly less. Implemented radix-2 and radix-4 FFT processor for 16 input points. The implementation results obtained using Cadence tools are compared in terms of the power, area, computational complexity and speed. Several optimizing techniques are also used. Radix-2 pipelined FFT processor is also presented for better understanding and comparability, but it occupies more area. But the speed is slightly higher for radix-4 processor. For 16 point FFT, we get the outputs 110ns earlier in the case of radix-4 processor .Currently the scope has been limited to 16 point pipelined FFT processor for both radix-2 as well as for radix-4 implementation. Future research work shall include implementation higher point radix FFT processor and evaluating other efficient architecture with the proposed architecture with low power techniques [1].

Paper entitled "Design of Fast Fourier Transform using Processing Element for Real Valued Signal" proposed on architecture for in-place fast Fourier transform (IFFT) computation for real valued signals. The proposed computation is based on modified radix-2 algorithm, which removes the redundant operations from the flow graph. The modified flow graph contains only real data paths as opposed to complex data paths in a regular flow graph. A new processing element (PE) is proposed which consists of two radix-2 butterflies that can process four inputs signals in parallel. A new conflict-free memory-addressing scheme is proposed to ensure the continuous operation of the FFT processor. The addressing scheme is also used to support multiple parallel PEs. As the proposed PE processes the four parallel inputs that reduce computation cycles and increase the speed compared to prior work. The number of computation cycles is reduced with increase in the number of PEs. As redundant operations are removed, that reduces hardware cost [2].

This paper entitled "Design and Simulation of 32-Point FFT Using Radix-2 Algorithm for FPGA Implementation" proposed on Fast Fourier Transform (FFT) is one of the fundamental operations in field of digital signal and image processing. Some of the very vital applications of the fast fourier transform include Signal analysis, Sound filtering, Data compression, Partial differential equations, Multiplication of large integers, Image filtering etc.Fast Fourier transform (FFT) is an efficient implementation of the discrete Fouriertransform (DFT). This is concentrates on the development of the Fast Fourier Transform (FFT), which is based on Decimation-In- Time (DIT) domain, Radix-2 algorithm, and uses VHDL as a design entity, and their Synthesis by Xilinx. The synthesis results show that the computation for calculating the 32-point Fast Fourier transform is efficient in terms of speed. The proposed FFT block of signal length 32 is been simulated and synthesized. The RTL block thus obtained for the decimation in time domain radix -2 Fast Fourier transform algorithm. The RTL view of the butterfly structure obtained after the simulation of the 32-point FFT block, Decimation in time domain and also the internal architecture of the butterfly, the next is simulation results of the 32-point FFT block. This is the given simulation result is the applied input, its 32 complex numbers. The output in binary format and output is in waveform [3].

Paper contain, "Design and Implementation of Parallel FFT on CUDA" Fast Fourier Transform (FFT) algorithm has an important role in the image processing and scientific computing, and it's a highly parallel Divide-and-conquer algorithm. In this paper, we exploited the Compute Unified Device Architecture CUDA technology and contemporary graphics processing units (GPUs) to achieve higher performance. We focused on two aspects to optimize the ordinary FFT algorithm, multi-threaded parallelism and memory hierarchy. We also proposed parallelism optimization strategies when the data volume occurs and predicted the possible situation when the amount of data increased further. It can be seen from the results that Parallel FFT algorithm is more efficient than the ordinary FFT algorithm [4].

This paper, "High-Performance Low-Power FFT Cores", ETRI Journal, the power consumption of integrated circuits has been attracting increasing attention. Many techniques have been studied to improve the power efficiency of digital signal processing units such as fast Fourier transform (FFT) processors, which are popularly employed in both traditional research fields, such satellite communications, and thriving consumer electronics, such as wireless communications. This paper presents solutions based on parallel architectures for high throughput and power efficient FFT cores. Different combinations of hybrid low-power techniques are exploited to reduce power consumption, such as multiplier less units which replace the complex multipliers in FFTs, low-power commutates based on an advanced interconnection, and parallel-pipelined architectures. A number of FFT cores are implemented and evaluated for their power and area performance[5].

In this paper," Design of Parallel FFT Architecture Using Cooley Tukey Algorithm" This paper represents, a parallel FFT architecture is proposed to give an efficient throughput and less energy consumption with the help of Cooley Tukey algorithm for radix 8.In this algorithm the DFT of N size is divided into smaller sizes ofN/2 and repeated until final DFT scalars are found. It divides the DFT in even index and odd index term. The computation time which is calculated by the pre-defined formula (Nlog2(N)) is reduced by the use of parallel architecture. Energy is defined aspower used per unit time. Parallel architecture helps to performnumber of operations simultaneously. As less time is required, the energy is efficiency is increased. The aim of this paper is to check throughput and efficiency using Cooley Tukey algorithm for higher radix. The recent trends of this algorithm is development of FPGA that is Field Programmable Gate Array as it can perform signal processing tasks in parallel, execute pipeline structure as well as speed up the computation of tedious algorithms. The main advantage of Cooley Tukey algorithm is that it reduces arithmetic computations as well as fast processing. As this algorithm divides the DFT into smaller DFTs, it can be combined with any other algorithm simultaneously [6].

3.EXPERIMENTAL STUDY

The proposed methodology through which we are going to mention along with the block diagram.

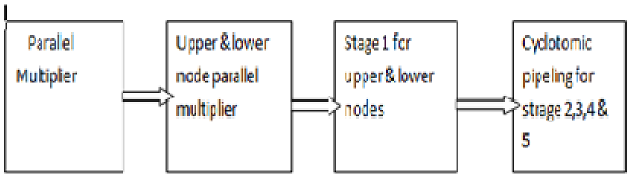
Paralel Multiplier
Upper & lower node parallel multiplier
Stage 1 for upper & lower nodes
Cyclotomic pipeling for strage 2,3,4 & 5

Fig -1: Block diagram of proposed methodology

3.1 model 1-

The first model of our project is parallel multiplier which is used for the multiplication of complex number. The multiplication is one of the most important arithmetic function especially when implement in programmable logic. Following figure shows the complex multiplier in galios field and output waveform.

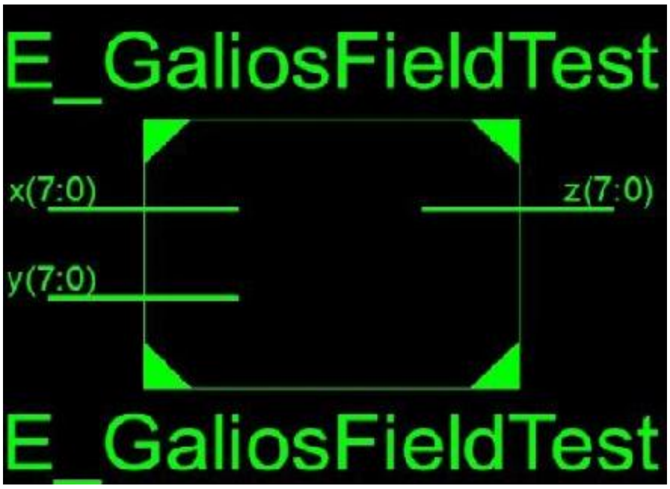

Fig -2: Galios Field

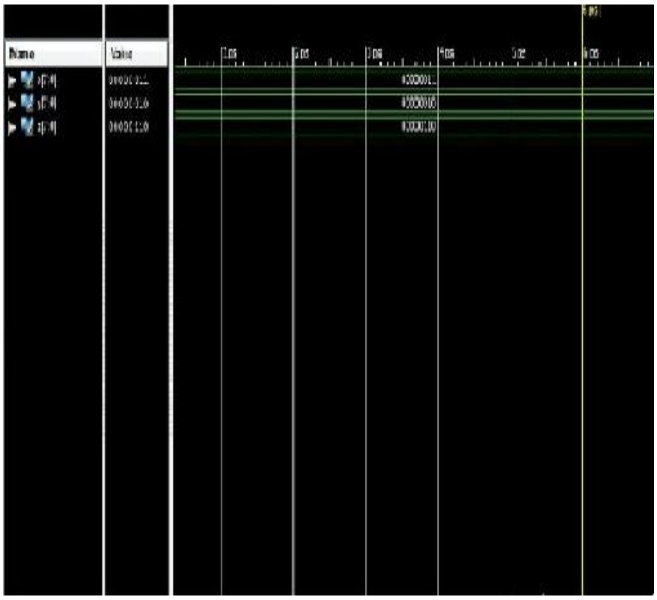

Fig -3: The output of Complex Multiplier

4. CONCLUSIONS

In multiplication the partial product increases for that method of multiplication can be change by using algorithms. Operation using FFT required large amount of delay at it haslarge number of multiplication and addition operations, so this will create large delay and power obtain is large have to reduced delay and power. We can use parallel processing with five stages, so that more than one operation can be perform in the single clock cycle for improving the speed, reducing delay and power.Future scope is to extend the concept of FFT in order to generate IFFT using FFT and transposition units also to apply FFT to sound processing using FPGA.

ACKNOWLEDGEMENT

We would like to thanks Mr. M.N.Thakre, Associate Professor, Electronics and Telecommunication department, B.D.C.O.E for their valuable suggestions. We would thanks to our college for providing valuable facilities which helps us in our research work. We also express thanks to our parents, friends and colleagues.

REFERENCES

[1] P.AugustaSophy, R.Srinivasan, J.Raja, M.Avinash, "Analysis and Design of Low Power Radix-4 FFT Processor using Pipelined Architecture",2015 International Conference on Computing and Communications Technologies (ICCCT'15).

[2] Ankush R. Kuralkar, "Design of Fast Fourier Transform using Processing Element for Real Valued Signal" This full-text paper was peer-reviewed and accepted to be presented at the IEEE ICCSP 2015 conference.

[3] .Afreen Fatima , "Designing and Simulation of 32 Point Fft Using Radix-2 Algorithm for Fpga", Department : Electronics Affiliated To : JNTU (HYD).

[4] Xueqin Zhang ,Kai Shen, ChengguangXu ,Kaifang Wang ,"Design and Implementation of Parallel FFT on CUDA", 2013 IEEE 11th International Conference on Dependable, Autonomic and Secure Computing

[5] Wei Han, Ahmet T. Erdogan, Tughrul Arslan, and Mohd. Hasan, "High-Performance Low-Power FFT Cores",ETRI Journal, Volume 30, Number 3, June 2008.

[6] RuchiraShirbhate, TejaswiniPanse and ChetanRalekar," Design of Parallel FFT Architecture Using Cooley TukeyAlgorithm",This full-text paper was peer-reviewed and accepted to be presented at the IEEE ICCSP 2015 conference.

[7] ShymnaNizar, N.S,Abhila, R Krishna, "An Efficient Folded Pipelined Architecture For Fast Fourier Transform Using Cordie Algorithm", 2014 IEEE International Conference on Advanced Communication Control and Computing Technologies (lCACCCT).

[8] SharadSingh,MrsJyotiKedia, "Pipelined FFT Architectures: A Review",-978-1-4799-7678-2/15/$3100.Elecrical, Electronics, Signals, Communication and optimization(EESCO),2015 International conference on©2015 IEEE.

[9] Mohammed A. El-Motaz, Ahmed M. El-Shafiey, Mohamed E. Farag,"Speeding-up Fast Fourier Transform",2015 IEEE International conference on Electronics, Circuits and systems(ICECS9)78-1-5090-0246-7/15/$31.00 ©2015 IEEE 510.

[10] WeihuaZheng, Kenli Li, Member, KeqinLi,Hing Cheung So, "A Modified Multiple Alignment Fast FourierTransform with Higher Efficiency". 10.1109/TCBB.2016.2530064, IEEE/ACM Transactions on Computational Biology and Bioinformatics.

[11] Wei Han, Ahmet T. Erdogan, TughrulArslan, and Mohd. Hasan, "High performance Low power FFT Cores". ETRI Journal, Volume 30, Number 3, June 2008.

[12] IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE) e-ISSN: 2278-1676,p-ISSN: 2320-3331, Volume 9, Issue 1 Ver. III (Jan. 2014), PP 42-50 www.iosrjournals.org www.iosrjournals.org 42, "Designing and Simulation of 32 Point Fft Using Radix-2 Algorithm for Fpga",Afreen Fatima.