

A novel VLSI Architecture for OFDM Receiver

Parvathy V

PG Scholar

Applied Electronics and Instrumentation
Younus College of Engineering and Technology
Kollam, India

Aswathi B

Assistant Professor

Electronics and Communication Engineering
Younus College of Engineering and Technology
Kollam, India

Abstract—Orthogonal frequency-division multiplexing (OFDM) is the most promising modulation technique. It is a method of encoding digital data on multiple carrier frequencies. The idea is to utilize a number of carriers, spread regularly over a frequency band, in such a way so that the available bandwidth is utilized to maximal efficiency. Fast Fourier transform (FFT) is the critical block in OFDM receiver system. In this study presenting a novel VLSI architecture for OFDM receiver and implemented using Xilinx & ModelSim simulation tool.

Keywords—Coordinate rotation digital computer (CORDIC), Fast Fourier transform (FFT), Orthogonal frequency-division multiplexing (OFDM)

I. INTRODUCTION

OFDM means for orthogonal frequency-division multiplexing. In Frequency Division Multiplexing (FDM) the adjacent bands are nonoverlapping but if overlap is allowed by transmitting signals that are mutually orthogonal (that is, there is a precise mathematical relationship between the frequencies of the transmitted signals) such that one signal has zero effect on another, then the resulting transmission technique is known as Orthogonal Frequency Division Multiplexing (OFDM). It is a method of encoding digital data on multiple carrier frequencies. It used in application such as digital television and audio broadcasting DSL internet access, wireless networks, power line networks, 4G mobile communications.

OFDM is a frequency-division multiplexing scheme. In this scheme a large number of closely spaced orthogonal subcarrier signals are used to carry data on several parallel data streams or channels. The main advantage of OFDM over single-carrier scheme is its ability to cope with severe channel conditions without complex equalization filters.

The Fast Fourier Transform is the main block in the OFDM receiver system. Recently, there is a huge requirement for a power and speed efficient FFT processor of longer length in various applications of OFDM. A typical FFT processor is composed of butterfly calculation units, an address generator and memory units.

The Coordinate Rotation Digital Computer (CORDIC) [1] algorithm is an alternative method to realize the butterfly operation without using any dedicated multiplier hardware. CORDIC algorithm is very suitable for the butterfly operations in FFT [6]. Instead of storing actual

twiddle factors in a ROM, the CORDIC-based FFT processor needs to store only the twiddle factor angles in a ROM for the butterfly operation.

The rest of this brief is structure as follows, Section II deals with a brief overview of the OFDM Receiver. In Section III which deals FFT implementation using CORDIC. Sine and Cosine implementation is presented in Section IV. Last section deals with conclusion of this paper.

II. OFDM RECEIVER

OFDM Orthogonal Frequency division multiplexing is a field that is in vogue and the latest mobile communication technologies have exploited the advantages of the technique. There have been a lot of architectures that have been presented to improve the performance of the OFDM system. The OFDM transceivers employ both Inverse Fast Fourier Transform and Fast Fourier Transform blocks. Here only OFDM receiver section is implemented.

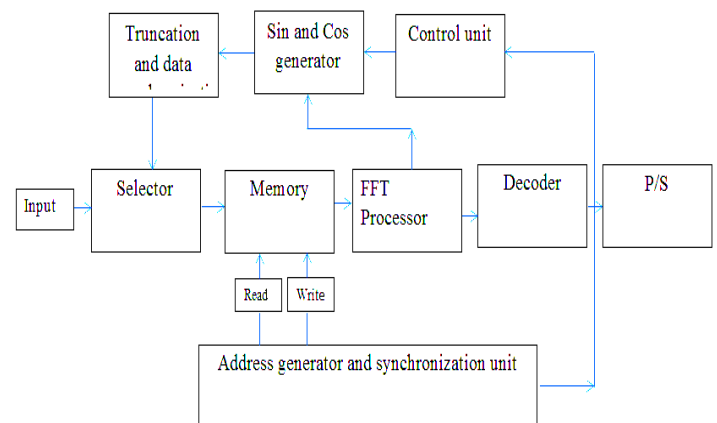


Fig. 1. Block diagram of OFDM receiver.

III. FFT IMPLEMENTATION USING CORDIC

The Fast Fourier Transformation (FFT) was proven to be a faster and more efficient algorithm to compute Fourier transforms [6]. Here implementing radix-4 FFT using butterfly structure based on Cordic algorithm. Basic idea of CORDIC algorithm is given below.

A. Unified Cordic Algorithm

- 1) *Rotational -mode Cordic*: Determine coordinate of given vector after rotation through given angle.
- 2) *Vectoring- mode Cordic*: Determine the magnitude and phase of the vector

These are two modes of operation in Cordic algorithm [4]. Using this algorithm decompose the desired rotation angle (θ) Such that the rotation through each of them can be accomplished with simple shift-and-add operation. Basic equations are given below:

$$x(i+1) = x(i) \cos a(i) - y(i) \sin a(i) \quad (1)$$

$$y(i+1) = y(i) \cos a(i) + x(i) \sin a(i) \quad (2)$$

Where $(x(i), y(i))$ and $(x(i+1), y(i+1))$ are the initial and final coordinates of the vector respectively

B. Cordic architecture

CORDIC uses simple shift-add operations for several computing tasks such as the calculation of trigonometric, hyperbolic and logarithmic functions, real and complex multiplications, division, square-root calculation, solution of linear systems, eigenvalue estimation, singular value decomposition, QR factorization and many others.

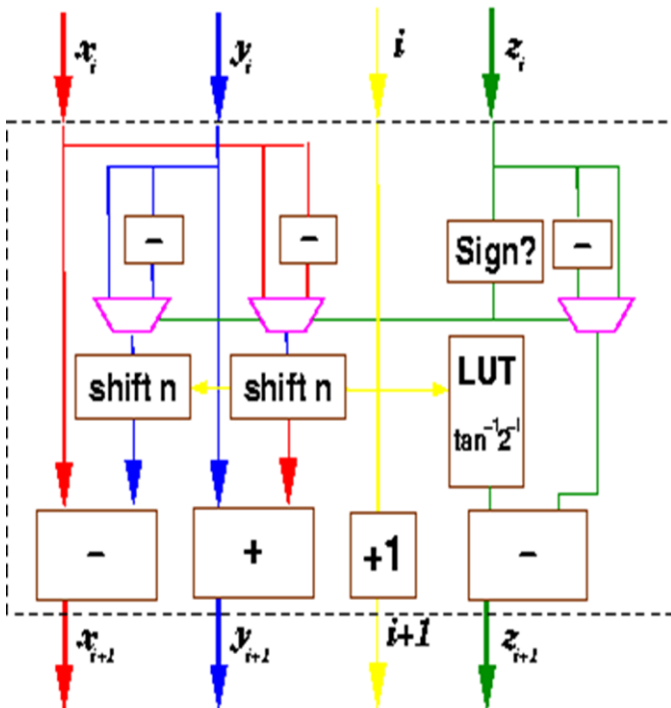


Fig. 2. Basic block of Cordic architecture.

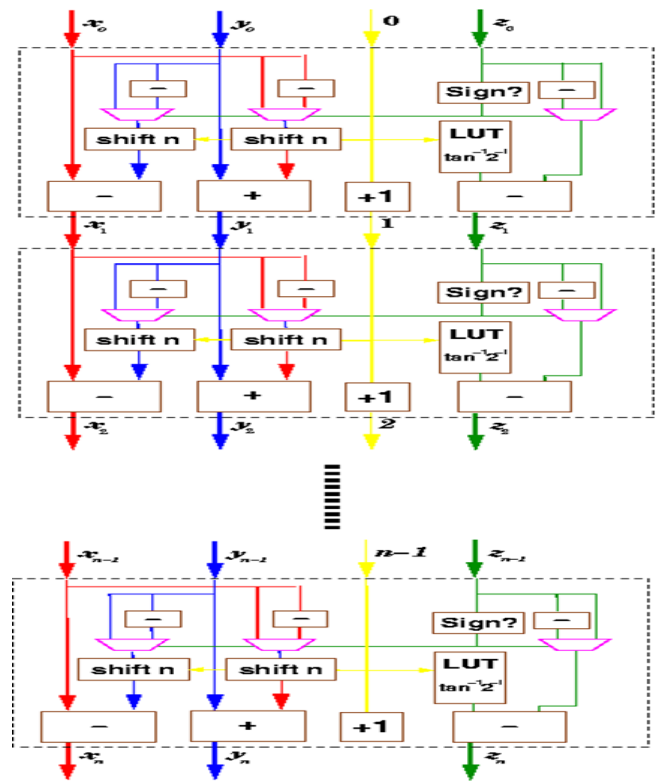


Fig. 3. Combination of n basic blocks.

Basic block diagram of Cordic architecture (shown in Fig. 2) consist of add/subtract/shift units. There is no multiplies needed. Fig. 3 shows combination of n basic blocks each with m-bit add/subtract/shift units.

C. Radix-4 FFT algorithm

In all FFT processors, the basic building blocks are the “Butterfly” which depends on radix of FFT Processor. The radix-4 FFT algorithm is more suitable for digital signal processor which has minimal complex computation than radix-2; radix-8 and structural architecture [2] is also more suitable than other radix FFT algorithms.

The N-point FFT can be decomposed to repeated micro Operations called butterfly operations. [6] When the size of the Butterfly is r, the FFT operation is called a radix-r FFT. For FFT hardware realization, if only one butterfly structure is implemented in the chip, this butterfly unit will execute all the Calculations recursively.

If parallel and pipeline processing techniques are used, an N point radix-r FFT can be executed by $N/(r \log N)$ clock cycles. This indicates that a radix-4 FFT can be four times faster than a radix-2 FFT Fig. 4 shows the signal flow graph of 64-point radix-4 FFT, For radix-r FFT, r banks of memory are needed to store data, and each memory bank could be two-port memory. With "in-place" strategy, the r outputs of the butterfly can be written back to the same memory locations of the r inputs, and replace the old data. In this case, to realize parallel and pipelined FFT processing, an efficient addressing scheme is needed to avoid the data conflict. A popular addressing scheme for radix-r ($r > 2$) was presented by Johnson, however due to the modulo-r addition, this method is slow and the speed depends on the length of FFT.

D. Architecture of radix-4 FFT butterfly

For N-point sequence, the radix-4 FFT algorithm consist of taking number of 4 data points at a time from memory, performing the butterfly computation and returning the result to memory. This procedure repeated many times, i.e., ((Nlog₄N)/4) times in the computation of N-point data DFT. Therefore, memory requirement is essential factor for FFT processor design. The requirement of memory size is 2N for the input sequence which is complex number and 2N for the output sequence. So the capacity of memory for N-point FFT processor is 4N. Bevan M. Baas [5] described about different type memory architecture like single memory architecture, Dual memory architecture, Array architecture and cached memory architecture. These type memory architectures are not suitable for FFT processor.

E. Generation of sine and cosine using Cordic

The sine and cosine of the input angle can be computed simultaneously. Setting the y component of the input vector to zero reduces the rotation mode result to:

$$X_n = A_n X_0 \cos Z_0 \tag{4}$$

$$Y_n = A_n X_0 \sin Z_0 \tag{5}$$

By setting X₀ equal to 1/A_n, the rotation produces the unscaled sine and cosine of the angle argument, Z₀. Very often, the sine and cosine values modulate a magnitude value [7].

IV. IMPLEMENTATION OF SINE AND COSINE

FPGA implementation for calculating the sine and cosine values of given angle using CORDIC algorithm is given below. The module was implemented by using Xilinx ISE Suite and VHDL. The ModelSim simulator was used to verify the functionalities of the module and this module is

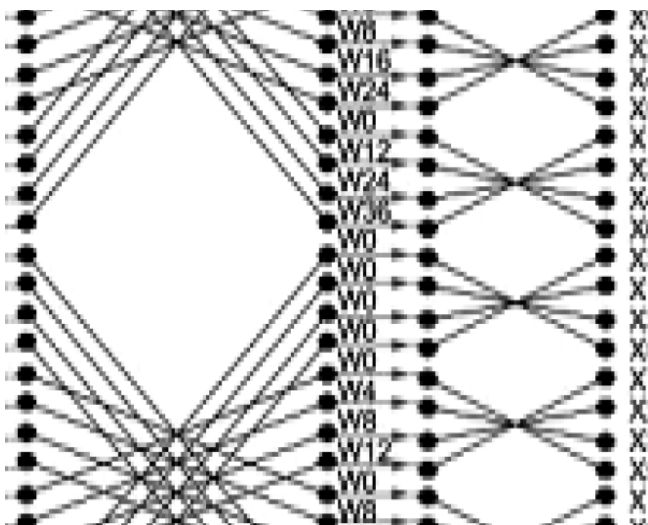


Fig. 4. Signal flow graph of 64-point radix-4 butterfly FFT

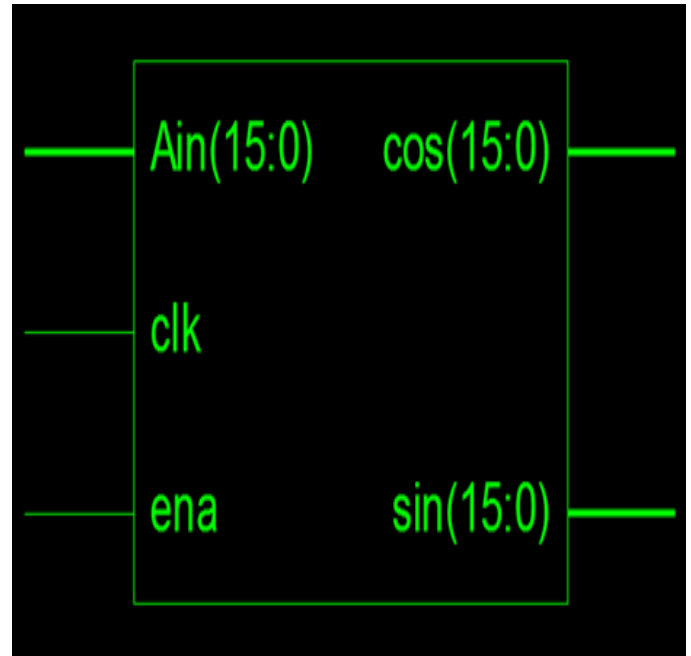


Fig. 5. Top-level RTL schematic of the Sine/Cosine generators.

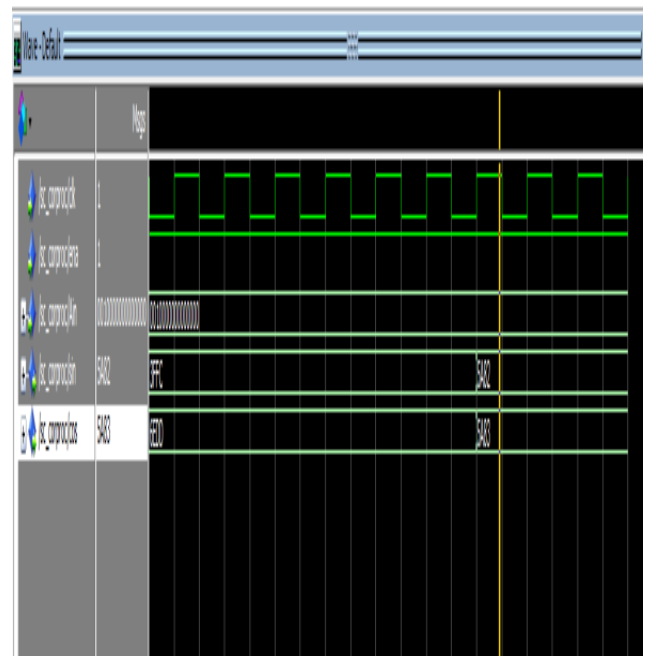


Fig. 6. ModelSim results

described in VHDL and synthesized using the Xilinx ISE Design Suite. Fig. 5 shows the top-level RTL schematic of the Sine/Cosine generators.

ModelSim result for binary input angle Ain = 45deg and binary outputs X_n (cos (Z₀)), Y_n (sin (Z₀)) in the form of waveform, and their corresponding magnitude is shown in Fig. 6.

V. CONCLUSION

Orthogonal Frequency Division Multiplexing (OFDM) receiver has been design using Xilinx tool and Simulation have been carried out using ModelSim simulation tool. VHDL is used to program all the components of the OFDM Receiver and verification of functionality of all components has done by giving different input and output is verified. Main component of OFDM is FFT processor, which is implemented by Cordic architecture.

ACKNOWLEDGEMENT

I thankfully acknowledges Principal Dr. M Abdul Majeed, in providing me with necessary requirements to help us finish the work in time. I would also like to extend my gratitude to the Head of the Department of Electronics and Communication, Mr. Rajeev S K, who were always ready to help me with ideas and suggestions for rectifying the mistakes that crept up time to time during the completion of this venture. I would also like to thank my friends and last but not the least the staff of ECE department for their whole hearted support and encouragement. Above all, I am thankful to the God almighty.

REFERENCES

- [1] R. Andraka, "A survey of CORDIC algorithms for FPGA based computers," *Proceedings of the 1998 ACM/SIGDA sixth international symposium on Field programmable gate arrays*, pp. 191 – 200
- [2] W. Hussain, F. Garzia, J. Nurmi, "Evaluation of Radix-2 and Radix-4 FFT Processing on a Reconfigurable Platform", 13th IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems, 2010
- [3] Neha V. Mahajan, Dr. J. S. Chitode "Simple Computation of DIT FFT" IJARCSSE, Vol 4, Issue 5, May 2014
- [4] Meggitt J. E., "Pseudo division and pseudo multiplication processes" IBM Journal, vol. 6, no. 2, pp. 210–226, 1962.
- [5] Baas, B. M., "A low-power, high-performance 1024-point FFT processor", IEEE Journal of Solid State Circuits. pp. 380-387, 1999.
- [6] Shi Jiangi; Tian Yinghui; Wang Mingxing; Yang Zhe; "A Novel design of 1024-point pipelined FFT processor based on CORDIC algorithm"; Intelligent System Design And engineering Application (ISDEA) 2012 second International Conference on Digital Object Identifier.
- [7] K. Maharatna, S. Banerjee, E. Grass, M. Krstic, and A. Troya, "Modified virtually scaling free adaptive CORDIC rotator algorithm and architecture," IEEE Trans. Circuits Syst. for Video Technol., vol. 15, no. 11, pp. 1463–1474, Nov. 2005