

IMPLEMENTATION OF HIGH SPEED 16- POINT FFT PROCESSOR FOR OFDM APPLICATIONS USING HDL (VERILOG)

Miss. Sneha C Gandh¹, Miss. Sneha M Hiremath², Miss. Ashwini A Kotagi³, Miss. Asha S Tamburi⁴, Prof. Sagar S Birade⁵

^{1,2,3,4}VIII Semester Students, ⁵Assistant Professor
E& E Department
Hirasugar Institute Technology, Nidasoshi

Abstract: In this paper, we present a novel fixed 16-point FFT processor developed primarily for the OFDM applications. The 16-point FFT is realized by decomposing it into a two-dimensional structure of 8-point FFTs. This approach reduces the number of required complex multiplications compared to the conventional radix-2 16-point FFT algorithm. The processor completes one parallel-to-parallel (i.e., when all input data are available in parallel and all output data are generated in parallel) 16-point FFT computation in 33 cycles. These features show that it can be used for any application that requires fast operation as well as low power consumption.

Keywords: FFT -Fast Fourier Transform, IFFT- Inverse Fast Fourier Transform, DFT- Discrete Fourier Transform, DIF- Decimation In Frequency, DIT -Decimation In Time, DTFT-Discrete-Time Fourier Transform, OFDM -Orthogonal Frequency Division Multiplexing.

I. INTRODUCTION

Fourth-Generation wireless and mobile systems are currently the focus of research and development. Broadband wireless systems based on orthogonal frequency division multiplexing (OFDM) will allow packet-based high-data-rate communication suitable for video transmission and mobile-Internet applications. Apart from the high speed of operation, the system demands low power consumption since it is primarily aimed at portable and mobile applications. We also showed through simulation that the most computationally intensive parts of such a high-data-rate system are the 16-point Inverse Fast Fourier transform (IFFT) in the transmit direction. Accordingly, an appropriate design methodology for constructing them has to be chosen. For a given functional specification, the main design concerns are: 1) how easily the particular architecture can be made flat for implementation in VLSI; and 2) how small the power consumption can be.

This paper describes a novel 16-point FFT processor, in this paper, a more detailed and complete description of the entire work is provided and the final design of a 16-point FFT processor suggested. The processor performs a forward and inverse 16-point FFT in 90 clock cycles making it suitable for high-speed data communication systems. The rest of the paper describes as follows: Section II describes design approaches for this application. Section III describes the algorithmic development of the proposed FFT processor. In Section IV, the algorithm-to-architecture mapping of the proposed FFT processor is described. In Section V, we present the simulation results of the FFT.

II. ALGORITHMIC FORMULATION

The discrete Fourier transform (DFT) $A(r)$ of a complex datasequence $B(k)$ of length N where $r, k \in \{0, 1, 2, 3, \dots, N-1\}$ can be described as,

$$A(s + Tt) = \sum_{l=0}^{M-1} W_M^{lt} \left[W_{MT}^{sl} \sum_{m=0}^{T-1} B(l + Mm) W_T^{sm} \right].$$

It means that it is possible to realize the FFT of length N by first decomposing it into one M and one T point FFT where $N = MT$, and then combining them. Now considering, $M = T = 8$, one may formulate the 64-point FFT as

$$A(s + 8t) = \sum_{l=0}^7 \left[W_{16}^{sl} \sum_{m=0}^7 B(l + 8m) W_8^{sm} \right] W_8^{lt}.$$

Above equation suggests that it is possible to express the 16-point FFT in terms of a two-dimensional structure of 8-point FFTs plus 16 complex interdimensional constant multiplications. However, since,

$l \in \{0, 1, 7\}$, the number of required nontrivial complex multiplications is 128. At first, appropriate data samples undergo an 8-point FFT computation followed by eight multiplications with the inter-dimensional constants or twiddle factors $\{W_{64}^{sl}\}$. However, the number of nontrivial multiplications required for each set of 8-point FFT results is actually seven, since the zeroth term of the first 8-point FFT gets multiplied with 1. Two such computations are needed to generate a full set of 16 intermediate data, which once again undergo a second 8-point FFT operation with the appropriate data ordering (every eighth data forms an input data set for the second 8-point FFT). As in the case of the first 8-point FFT, again two such computations are required. Proper reshuffling of the data coming out from the second 8-point FFT generates the final output of the 16-point FFT. The important point to be noted here is that for realization of an 8-point FFT using the conventional decimation in frequency (DIF) butterfly algorithm; one does not need to use any explicit multiplication operation. This can be shown using the 8-point FFT signal flow graph in Fig. The constants to be multiplied for the first two columns of the 8-point FFT structure are either 1 or and thus, they are mere addition/subtraction operations with the proper data ordering. In the third column, the multiplications of the constants are actually addition/subtraction operation followed by a multiplication. Thus, in principle, an 8-point FFT can be carried out without using any true digital multiplier and thus, provides away to realize a low-power 16-point FFT at a reduced hardware cost. On the other hand, the number of nontrivial complex multiplications for the conventional 16-point radix-2 DIF FFT is 256. Thus, the present approach results in a reduction of complex multiplication compared to that required in the conventional radix-2 16-point FFT. This reduction of arithmetic complexity further enhances the scope for realizing a low-power 16-point FFT processor.

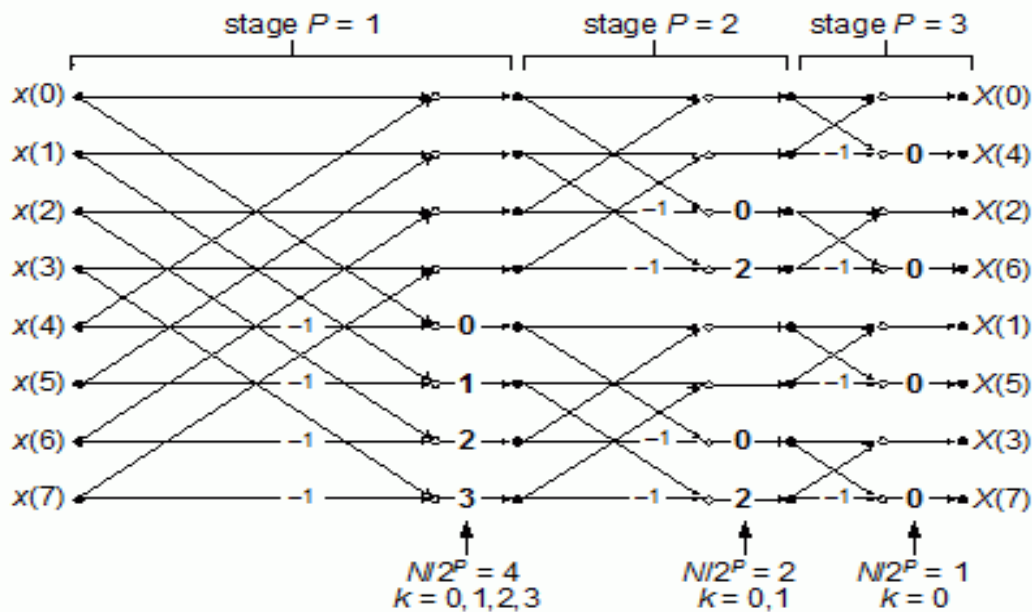


Fig. 1- 8-point FFT flow graph

The IFFT can be performed by first swapping the real and imaginary parts of the incoming data at the primary input, then performing the forward FFT on them and once again swapping the real and imaginary parts of the data at the output. This method allows to perform the FFT and IFFT without changing any of the internal coefficients,

III. ARCHITECTURE OF 64-POINT FFT/IFFT

The block diagram of the 16-point FFT processor derived from above equation is depicted in Fig below. It consists of an input unit (I/P unit), two 8-point FFT units, a multiplier unit, an output unit (O/P unit), and a 5-bit binary counter that acts as the master controller for the entire architecture. However, there are two main performance bottlenecks in such a scheme. First, there are a large number of global wires resulting from multiplexing of the complex data to the 8-point FFTs. Secondly, the construction of the multiplier unit to attain the required speed. To eliminate these two bottlenecks and to make efficient algorithm-to-architecture mapping several special strategies have been adopted in the current architecture.

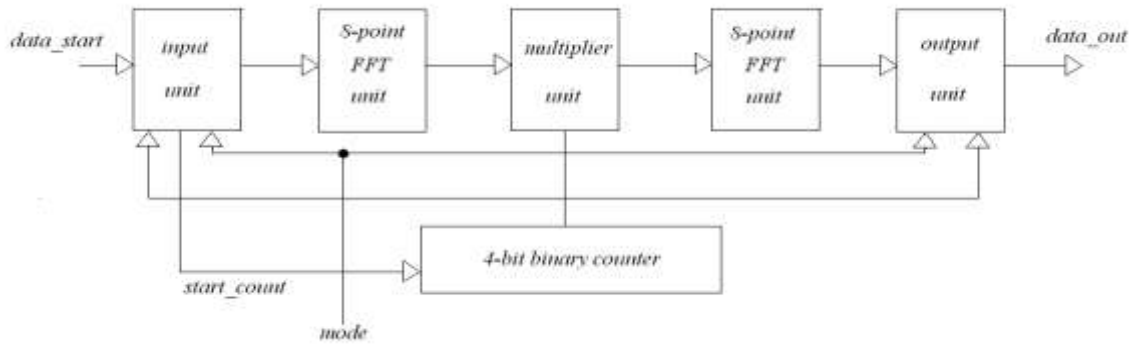


Fig. 2: block diagram of 16-point FFT computation

Input Unit:

The input unit consists of an input register bank (reg (0 to 63)) having 16-bit word length that can store 64 complex data. The data_start signal remains at logic 1 for the next few cycles after its assertion. The logic 1 state of the mode signal indicates the IFFT mode of operation, whereas its logic 0 state indicates the FFT mode.

Eight-Point FFT Units:

To construct the 8-point FFT units, we have chosen the radix-2 DIF 8-point FFT algorithm. The butterfly computations are predominantly addition and subtraction operations. The computation of an 8-point FFT is carried out in 8 clock cycles.

Multiplier Unit:

Nontrivial inter-dimensional constants ($W_{16}^{sl}, s, l \in \{0,1,\dots,7\}$) are to be multiplied to the intermediate results coming out from the first 8-point FFT unit. The use of a single complex multiplier effectively results in a degradation of the speed advantage provided by the first 8-point FFT unit. To achieve the full speed advantage it is necessary to use seven complex multipliers operating in parallel. This approach would result in increased high power consumption.

Output Unit:

For the output unit, we follow the same strategies with the input unit. In essence, the output unit has a complementary structure to the input unit.

IV. Simulation Results

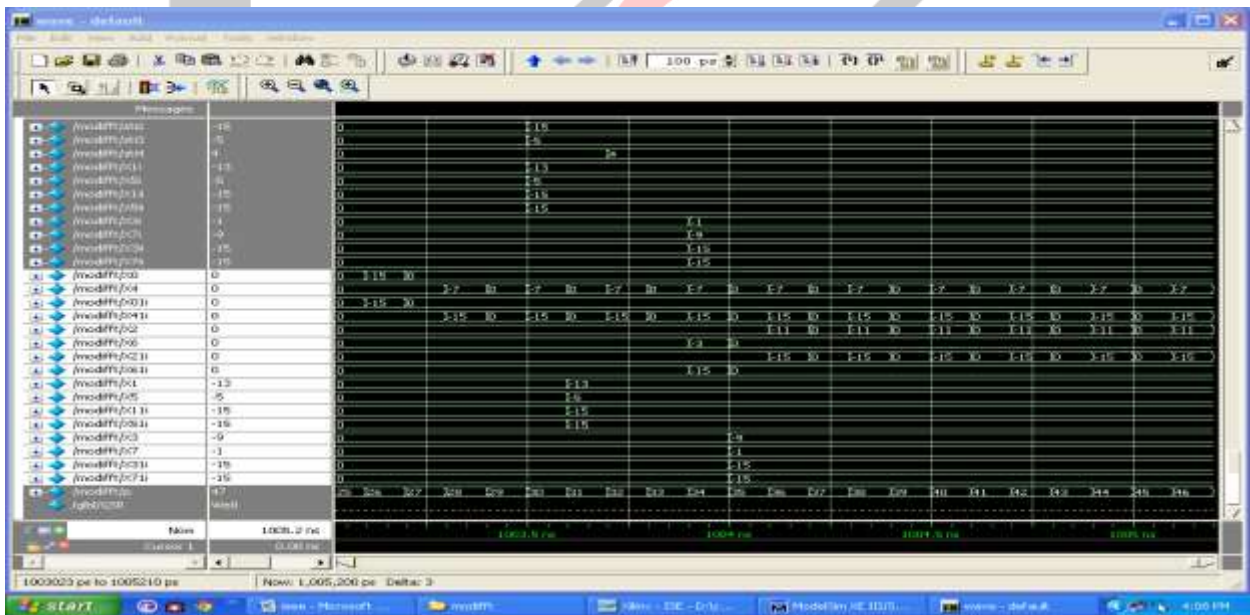


Fig. 3: FFT simulation result

V. CONCLUSION

16-point FFT architecture for high-speed wireless systems based on OFDM transmission has been presented. This architecture is based on a decomposition of the 16-point FFT in to two 8-point. It exhibits numerous attractive features from a VLSI point of view, which include regularity, modularity, simple wiring, and high throughput. A new low-power high-performance 16-point FFT chip for wireless applications has been successfully designed based on the architecture described. The design computes a 16-point parallel to parallel FFT in 90 clock cycles. The new design is deemed to result in a considerable reduction in cost, size, and power dissipation for the existing wireless systems.

Future Work

As it is discussed that the 2, 8-point FFT's combined to build the 16-point FFT in the same way we can design 32-point FFT by combining 2, 16-point FFT's. This process can be used to build any number of FFT by using 2 blocks; for ex: 64-point FFT can be built by 2, 32-point FFT's, 128-point FFT by 2, 64-point FFT's and so on.

Some recommendations are suggested to overcome the problem encountered during development of this project. First is to use higher fixed point representation for point value representation. Floating point format also can be considered as the solution to reduce error number representation especially for twiddle factor value which is 0.7071. Although floating point consume processing time and output latency, but it is an excellence method to overcome accuracy problem.

In the same way we can use Distributed Arithmetic Algorithm to implement the Complex Multiplier so that further power will be reduced.

References:

- [1] Draft supplement of standard /for/ Information Technology- Telecommunication and Information Exchange Between Systems- Local and Metropolitan Area Networks -Special Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High Speed Physical Layer in the 5 GHz Band, IEEE P802.11a/D7.0
- [2] E. Grass, K. Tittelbach. U. Jagdhold, A. Troya, G. Lippert, O. Krueger, J. Lehmann, K. Maharatna. N. Fiebig, K. Dombrowski, R. Kraemer and P. Machoenen. "on the single chip implementation of a hiperlan/2 and IEEE 802.11a capable modem." IEEE Pers. Commun., 8, pp.48-57. Dec. 2001
- [3] J. O'Brien, J. Mather, and B. Holland, "A 200 MIPS single-chip 1 k FFT processor," in Proc. IEEE Int. Solid-State Circuits Conf., 1989, vol. 36, pp. 166-167., 327.
- [4] B. Gold and T. Bially, "parallelism in fast Fourier transform hardware." IEEE Trans. Audio Electroacous., vol. 21, pp. 5-16. Feb 1973.
- [5] K. Maharatna, E. Grass and U. Jagdhold, "A lower-power 64-point FFT/IFFT architecture for wireless broadband communication." Inproc. 5th int. OFDM workshop. Hamburg, Germany. Sept.2000.p.36.
- [6] Xilinx Product Specification. High Performance 64-point complex FFT/IFFT V1.0.5 [online]. Available: <http://www.xilinx.com/ipcenter>
- [7] Product Design Data Sheet, FFT-1024 complex 1024-points FFT/IFFT processor. Icomm Technologies Inc. [online]. Available: http://www.spinnaker.co.jp/jp/datasheet/tb_fft_1_V002.pdf
- [8] Fast Fourier Transforms, Connexions online book edited by C. Sidney Burrus, with chapters by C. Sidney Burrus, Ivan Selesnick, Markus Pueschel, Matteo Frigo, and Steven G. Johnson (2008).