Available online at www.elixirpublishers.com (Elixir International Journal)

Electrical Engineering

Elixir Elec. Engg. 54 (2013) 12460-12462

ISSN: 2229-712X

VLSI implementation of fast Fourier transform used in OFDM

Yesodha.P

Department of ECE, Bharath University, Selaiyur, Chennai – 600 073.

ARTICLE INFO

Article history:

Received: 28 November 2012;

Received in revised form:

27 December 2012;

Accepted: 7 January 2013;

Keywords

FFT- Fast Fourier Transform,
IFFT – Inverse Fast Fourier Transform,
OFDM – Orthogonal Frequency Division Multiplexing,
WiBro – Wireless Broadband,
DAB – Distributed Audio Band,
QPSK – Quadrature Phase Shift Keying.

ABSTRACT

OFDM systems are widely used in both wired and wireless communication systems. Fast Fourier Transform is one of the key processing in the implementation of OFDM systems such as wireless broadband, and ultra wideband systems. In most researches, the implementation of Fast Fourier Transform is focused on reducing the difficulties in multipliers, memory unit and control circuits involved in the FFT process. Modification in the architecture for pipelined FFT processor is proposed, to reduce the register size required for FFT processor. In OFDM, this FFT process is used at the receiver side of the system. The single-path delay-feedback architecture is used to exploit the spatial regularity in signal flow graph of the algorithm.

I. Introduction

IEEE 802.11a, HIPERLAN standard uses Orthogonal frequency division multiplexing as Modulation and Multiplexing technique for many communication systems. OFDM is commonly implemented in many emerging communications protocols because it provides several advantages over the traditional FDM approach in the dispersive environment. More specifically, OFDM systems provides greater spectral efficiency, reduced inter symbol interference (ISI), and it is resilience to multi-path distortion.

FFT/IFFT processor is one of the key components in the implementation of OFDM systems. Many researches have been carried out to design Fast Fourier Transform in terms of area and power. In general, FFT architectures have suitable properties for VLSI implementation of real-time applications. Radix-2^i Single-Path Delay Feedback FFT architecture is one of the efficient FFT structures in terms of memory unit and control circuit. Radix-2^2 FFT algorithms have the same complexity in multipliers as radix-4 algorithms, but still retain the simple radix-2 butterfly structures. Large point-size FFT implementation has been increased due to the highly demand for broadband and mobility in communication systems. If the number of FFT stages increases then the FFT point-size also increases, as the number of required registers increases linearly. The registers have more than 70% of the chip area in large point-size FFT designs, which causes some problems in the chip design due to the area. In OFDM systems, the number of different input signals to FFT in the receiver part is limited. For an example, in 64- QPSK systems, the FFT receives only 64 different input signals. In this, it proposes a register size reduction method which is based on the observation that the number of different FFT input signals are limited in OFDM systems.

In Section II, we review the importance of FFT and IFFT in OFDM systems and the butterfly output signals in FFT are explained in Section-III. The requirement of pipeline method in FFT design is presented in Section IV. In Section V, the pipeline architecture will be explained. The simulation results are presented in Section VI. Finally, brief conclusions are given in Section VII.

II. Role of FFT and IFFT in OFDM

The Fast Fourier Transform (FFT) and its inverse (IFFT) is one of the fundamental operations in the field of digital signal processing. The FFT and IFFT are widely used in the areas such as telecommunication, speech and image processing etc…

FFT and IFFT are used as the one of the key component in OFDM based wideband communication systems. In OFDM serial input data is converted into parallel input data. After conversion, base band modulation technique is applied such as PSK, QPSK, and QAM etc. After this operation, data converted into frequency domain signal. (Complex form) IFFT operation converts the frequency domain signal into the time domain signal. This time domain signal is same as that of the OFDM signal. The sinusoidal signal is used in OFDM signal transmission. Sum of such modulated sinusoidal form the transmit signals. Mathematically, the transmit signal is in the form of the IFFT singnal.

$$x_n = \frac{1}{N}\sum_{n=0}^{N-1} X_k e^{\frac{2\pi i}{N}nk} \quad n = 0,1,\dots,N-1$$

In OFDM systems, subcarriers are generated by IFFT process, on which the information is mapped and is transmitted through the dispersive channel after up-conversion. At the receiver side and the FFT process is used. The Cooley–Tukey algorithm, named after J.W. Cooley and John Tukey, is the most common fast Fourier transform (FFT) algorithm. It re-expresses the discrete Fourier transform (DFT) of an arbitrary

Tele:
E-mail addresses: Yesodha.engineer@gmail.com

composite size $N = N_1N_2$ in terms of smaller DFTs of sizes N_1 and N_2, recursively, in order to reduce the computation time to O(N log N) for highly-composite N (smooth numbers).

In FFT the input signals are in the normal order and the output signals are in the reverse order. In IFFT the input signals are in the reverse order and the output signals are in the normal order.

The DFT of the first N/2 points and combine them in a special way with the DFT of the second N/2 points to produce a single N-point DFT. Each of these N/2-point DFTs can be calculated using smaller DFTs in the same way. One (radix-2) FFT begins, therefore, by calculating N/2 2-point DFTs. These are combined to form N/4 4-point DFTs. The next stage produces N/8 8-point DFTs, and so on, until a single N-point DFT is produced [3].

III. FFT processor and butterfly operation

The principle of FFT algorithm is based upon decomposing the computation of DFT of a sequence of length 'n' into successive smaller DFT. Parallel data streams are used as inputs to an IFFT. The input is in the normal order and the output is in the bit-reversed order. The algorithm has an in-place calculation, which leads to reuse of available memory. The basic computational element of the Fast Fourier Transform is the butterfly. Each butterfly requires one complex multiplication and two complex additions.

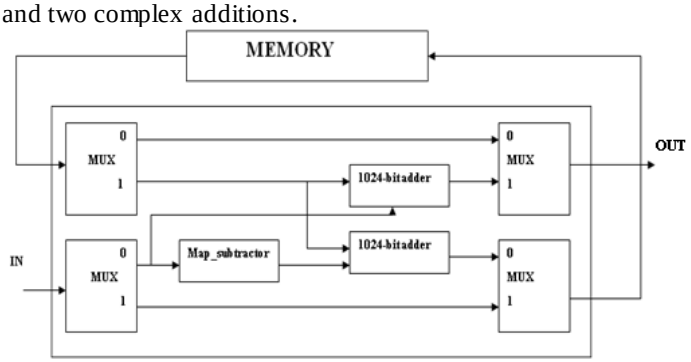

Fig 1. Butterfly-1 Architecture

In the butterfly - 1 architecture, if the mux select value is zero then no operation is performed. If the mux select value is one then the two inputs are added in the N-bit adder unit. At the Map_sub block the negative conversion of the input value is takes place and then is added with another input value in the N-bit adder unit. These output values are stored in the memory for the next operation. The same operation is performed in the butterfly-2 structure, but at the input side –j multiplier operation is performed

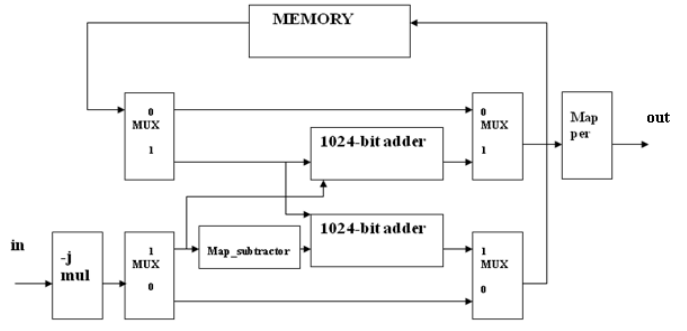

Fig 2. Butterfly-2 Architecture

The pipeline FFT processor can be divided into three main building blocks. i.e. memory, simplified butterfly elements and complex multipliers. Each stage in FFT is computed with a set of processing elements and the result is fed back to the same processing elements for the computation of next stage.

The orthogonality allows for efficient modulator and demodulator implementation using the FFT algorithm on the receiver side, and inverse FFT on the sender side. Although the principles and some of the benefits have been known since the 1960s, OFDM is popular for wideband communications today by way of low-cost digital signal processing components that can efficiently calculate the FFT.

In OFDM serial input data is converted into parallel input data. After conversion, base band modulation technique is applied such as PSK, QPSK, and QAM etc... After this operation, data converted into frequency domain signal (complex form) IFFT operation converts the frequency domain signal into the time domain signal. This time domain signal is same as that of the OFDM signal.

At the receiver side, the original information is added with noise signal. That noise signals removed by simple equalization technique with one tab or two-tab equalizer. Then the FFT operation coverts the time domain signal into the frequency domain signal. After performing these operations, the parallel data is converted into the serial data.

IV. Requirement of pipeline systems

Pipeline FFT processor is a specified class of processors for DFT computation utilizing fast algorithms. It is characterized with real-time, non-stopping processing as the data sequence passing the processor. It is an *AT2* non-optimal approach with *AT2 = O(N3)*, since the area lower bound is *O(N)*. However, as it has been speculated [l] that for real-time processing whether a new metric should be introduced since it is necessarily nonoptimal given the time complexity of *O(N)*. Although asymptotically almost all the feasible architectures have reached the area lower bound [2], the class of pipeline FFT processors has probably the smallest "constant factor" among the approaches that meet the time requirement, due to its least number, O(logN), of Arithmetic Elements (AE).

The difference comes from the fact that an AE, especially the multiplier, takes much larger area than a register in digital VLSI implementation. It is also interesting to note the at least R (1ogN) AEs are necessary to meet the real-time processing requirement due to the computational complexity of R (N1ogN) for FFT algorithm.

V. Architecture of pipeline operation

Another major area/energy consumption of the FFT processor comes from the memory requirement to buffer the input data and the intermediate result for the computation. For large size transform, this turns out to be dominating [3, 4].

Although there is no formal proof, the area lower bound indicates that the "lower bound" for the number of registers is likely to be *Q(N)*. This is obviously true for any architecture implementing FFT based algorithm, since the butterfly at first stage has to take data elements separated *N/r* distance away from the input sequence, where *r* is a small constant integer, or the "radix". Putting above arguments together, a pipeline FFT processor has necessarily R (log, *N)* AEs and R (N) complex word registers. The optimal architecture has to be the one that reduces the "constant factor", or the absolute number of AEs (multipliers and adders) and memory size, to the minimum.

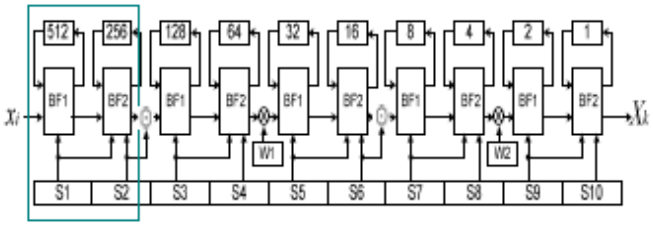

Fig 3. 1024- point FFT Architecture

In this paper a new approach for real-time pipeline FFT processor, the Radix-2 Single-path Delay Feedback, or R2'SDF architecture will be presented. A hardware oriented radix-2' algorithm is then developed by integrating a twiddle factor decomposition technique in divide and conquer approach to form a spatially regular signal flow graph (SFG). The memories dominate in terms of chip area. It is important both from power consumption and chip area point of view to reduce the memory size. This can be done in two ways; first to minimize the internal word length and hence reduce the memory size, second is to select an area efficient memory structure.

VI. Result

Simulation Tools:

- Xilinx ISE Simulator
- Model Sim

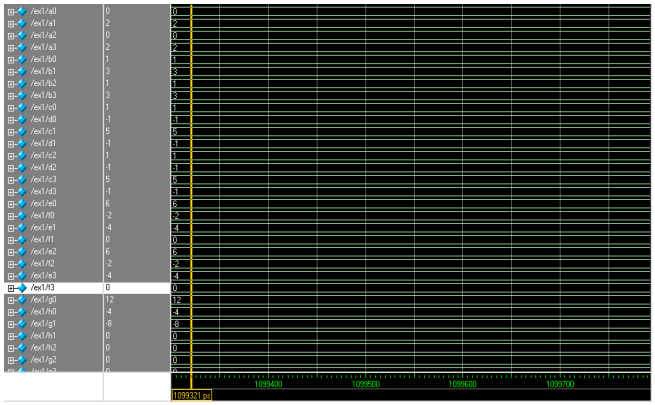

Fig 4. Simulation result for FFT

In the simulation result of Fast Fourier Transform, it gives the output of ordinary fast fourier transform output and it uses more number of input and output pins. So that, the estimated power for the fast fourier transform was increased.

Fig 5. BF1 OUTPUT

INPUT: **OUTPUT:**

a1=01,b1=10,mux1=0,mux2=1 **C1=011,C2=110**

In the butterfly -1 output simulation result, the first stage of fast fourier transform operation is performed. In this result, if the mux1 as 0 then the input a1 as 01 and b1 as 10 will be added and the ouput produced at the output pin c1 as 110. In this result, if the mux2 as 1 then the input a1 as 01 and b1 as 10 will be subtracted and the output produced at the output pin c1 as 011. Due to this operation, the number of input and output pins are reduced and it estimates power uses for this operation is also be reduced.

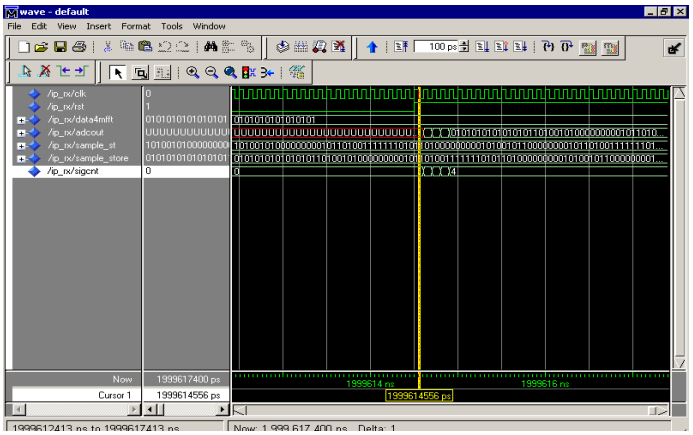

Fig.6 Output of 64-bit QPSK

The 64-bit QPSK act as the output of the modulator and that is given as the input to the fast fourier transform operation.

VII. Conclusion

Performance	Ordinary FFT	Butter fly-1	Butter fly-2
No. of bonded IOBs	128 out of 173 73%	12 out of 97 12%	22 out of 97 13%
Total Estimated Power	86mW	56mW	60mW

Thus from the simulation, the above observation is made. It compares the conventional FFT process and the pipelined process as given by the proposed architecture. Simulation result clearly indicates that pipelined architecture uses less IOBs(Input and Output Bonds)The reduction is around 1/6th of that required for the conventional method. Similiarly the power dissipation by piplined architecture in average is around 37% less than the conventional one.

Thus this paper tried to achieve a reduction in area required for FFT process. Also it gives reduction in power dissipated.

References:

[1] TTAS.KO-06.0064R1, Specifications for 2.3GHz band Portable Internet Service - Physical Layer, Telecommunications Technology Association, Dec. 2004.

[2] ETSI, "Digital Video Broadcasting (DVB): Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television," *ETSI EN 300 744 v1.4.1*, 2001.

[3] E. H. Wold and A. M. Despain, "Pipeline and parallel pipeline FFT processors for VLSI implementation," *IEEE Trans. Comput*., C-33(5), pp.414-426, May 1984.

[4] S. -S. He and M. Torkelson, "A new approach to pipeline FFT processor," *Proc. 10th Int'l Parallel Processing Symp*. (*IPPS* '96), pp. 766-770, 1996.

[5] Y. Chang and K. K. Parhi, "An efficient pipelined FFT architecture," *IEEE Transactions on Circuits and Systems II*, vol. 50, no. 6, pp. 322- 325, Jun. 2003.

[6] El-Khashab, A. M., Swartzlander, E. E., Jr. "A modular pipelined implementation of large fast Fourier transforms," *IEEE Transactions on Signals, Systems and Computers*, vol. 2, pp. 995-999, 2002.

[7] S. He and M. Torkelson, "Designing pipeline FFT processor for OFDM (de)Modulatioin," in *Proc. IEEE URSI Int. Symp. Signals, Syst., Electron*., pp. 257-262, 1998.

[8] J. Y. Oh and M. S. Lim, "New radix-2 to the 4th power pipeline FFT processor," *IEICE* Trans. Electron., vol. E88-C, no. 8, pp.1740-1746, Aug. 2005. 881.