

Low Power and Small Area Implementation for OFDM Applications

K. Umapathy
Research Scholar
JNT University, Anantapur, India

D. Rajaveerappa
Professor, Loyola Institute of Technology,
Chennai, India

ABSTRACT

This paper proposes that several FFT algorithms such as radix-2, radix-4 and split radix were designed using VHDL with the multiplication complexity reduced more than 30% by using the newly proposed CSD constant multipliers instead of the programmable multipliers and the simulations of standard 0.35 μm . The sizes of FFT/IFFT operations are varied in different applications of OFDM systems. The reorganized Mixed Radix 4-2 Butterfly FFT with bit reversal for the output sequence derived by index decomposition execution is our suggested VLSI system architecture to design the module FFT/IFFT processor for OFDM systems. The output shows that the proposed processor architecture can minimize the area cost while keeping a high-speed processing speed, a decrement of more than 70% of the power consumption/area when compared with complex multiplier.

Keywords

FFT/IFFT, OFDM, radix², radix², Multiplier

1. INTRODUCTION

The FFT (Fast Fourier Transform) and its inverse (IFFT) are the key components of the OFDM (Orthogonal Frequency Division Multiplexing) systems. Now days, the demand for a long length, high-speed and low-power FFT processor has increased in wideband OFDM applications. There are three kinds of main design architectures for implementing the FFT processor. One is the single-memory architecture. It has one processing element and one main memory. Hence, it occupies a small area. Another is the dual-memory architecture. It has two memories. The architecture has higher throughput than the single-memory structure because it can store butterfly outputs and read butterfly inputs at the same time. These two memory schemes occupy relatively small area. However, they perform at a lower throughput, and thus require higher clock frequency than the third architecture. The third is the pipeline architecture. It is used for the purpose of high throughput applications. It requires $\log_2(N)$ processing elements; therefore its calculations are $\log_2(N)$ times faster than the processor structured the single memory. However, this scheme has the disadvantage of consuming a large power/area. Hence, a power and area efficient FFT algorithm which can handle this problem should be provided in order to employ the pipeline architecture to the FFT processor of wideband OFDM applications.

2. ORTHOGONAL FREQUENCY-DIVISION MULTIPLEXING (OFDM)

In OFDM, the sub-carrier frequencies are chosen so that the sub-carriers are orthogonal to each other, meaning that crosstalk between the sub-channels is eliminated and inter-carrier guard bands are not required. This greatly simplifies the design of both the transmitter and the receiver; unlike conventional FDM, a separate filter for each sub-channel is not required. The guard bands that were necessary to allow individual

demodulation of subcarriers in an FDM system would no longer be necessary. Orthogonality can also be viewed from the standpoint of stochastic processes. If two random processes are uncorrelated, then they are orthogonal. Given the random nature of signals in a communications system, this probabilistic view of orthogonality provides an intuitive understanding of the implications of orthogonality in OFDM. Later in this paper, we will discuss how OFDM is implemented in practice using the Fast Fourier transform (FFT) as shown figure 1. Recall from signals and systems theory that the sinusoids of the FFT form an orthogonal basis set, and a signal in the vector space of the FFT can be represented as a linear combination of the orthogonal sinusoids. One view of the FFT is that the transform essentially correlates its input signal with each of the sinusoidal basis functions. This transform is used at the OFDM transmitter to map an input signal onto a set of orthogonal subcarriers, i.e., the orthogonal basis functions of the DFT. Similarly, the transform is used again at the OFDM receiver to process the received subcarriers as shown figure 2. The signals from the subcarriers are then combined to form an estimate of the source signal from the transmitter. The orthogonal and uncorrelated nature of the subcarriers is exploited in OFDM with powerful results. Since the basic functions of the FFT are uncorrelated, the correlation performed in the FFT for a given subcarrier only sees energy for that corresponding subcarrier. The energy from other subcarriers does not contribute because it is uncorrelated. This separation of signal energy is the reason that the OFDM subcarriers' spectrums can overlap without causing interference.

TRANSMITTER

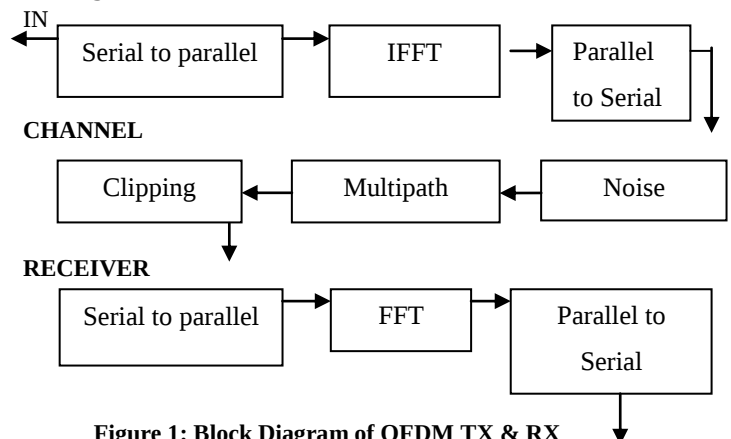


Figure 1: Block Diagram of OFDM TX & RX

3. FAST FOURIER TRANSFORM (FFT)

An FFT computes the DFT and produces exactly the same result as evaluating the DFT definition directly; the only difference is that an FFT is much faster. The Discrete Fourier Transfer (DFT) plays an important role in many applications

of digital signal processing including linear filtering, correlation analysis and spectrum analysis etc.

The DFT is defined as:

$$Z[n] = \sum_{m=0}^{M-1} z[m] W_M^{mn}$$

$$n = 0, 1, \dots, M-1$$

Where $W_M^{mn} = e^{j \frac{2\pi}{M} mn}$ is the DFT coefficient

Evaluating the Equation (1) directly requires M complex multiplications and $(M-1)$ complex additions for each value of the DFT. To compute all M values therefore requires a total of M^2 complex multiplications and $M(M-1)$ complex additions. Since the amount of computation, and thus the computation time, is approximately proportional to M^2 , it will cost a long computation time for large values of M . For this reason, it is very important to reduce the number of multiplications and additions. This algorithm is an efficient algorithm to compute the DFT which is called Fast Fourier Transform (FFT) algorithm or radix-2 FFT algorithm, and it reduce the computational complexity from $O(M^2)$ to $O(M \log_2 M)$.

4. MIXED RADIX 4-2

A mixed radix algorithm is a combination of different radix- r algorithms. That is, different stages in the FFT computation have different radices. For instance, a 128-point long FFT can be computed in two stages using one stage with radix-8 PEs, followed by a stage of radix-2 PEs. This adds a bit of complexity to the algorithm compared to radix- r , but in return it gives more options in choosing the transform length. The Mixed-Radix FFT algorithm is based on sub-transform modules with highly optimized small length FFT which are combined to create large FFT. However, this algorithm does not offer the simple bit reversing for ordering the output sequences.

5. MIXED-RADIX FFT ALGORITHMS

The mixed-radix 4/2 butterfly unit is shown in Figure 2. It uses both the radix-2² and the radix-2 algorithms can perform fast FFT computations and can process FFTs that are not power of four. The mixed-radix 4/2, which calculates four butterfly outputs based on $X(0) \sim X(3)$. The proposed butterfly unit has three complex multipliers and eight complex adders. Four multiplexers represented by the solid box are used to select either the radix-4 calculation or the radix-2 calculation.

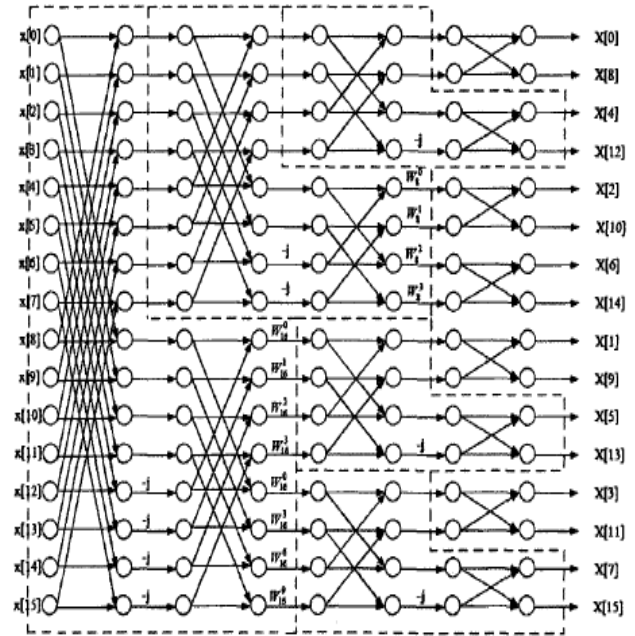


Figure 2: The Basic Butterfly for Mixed-Radix 4/2 DIF FFT Algorithm

In order to verify the proposed scheme, 64-points FFT based on the proposed Mixed-Radix 4-2 butterfly with simple bit reversing for ordering the output sequences is exemplified. As shown in the Figure 2, the block diagram for 64-points FFT is composed of total six-teen Mixed-Radix 4-2 Butterflies. In the first stage, the 64 point input sequences are divided by the 8 groups which correspond to $n_3=0, n_3=1, n_3=2, n_3=3, n_3=4, n_3=5, n_3=6, n_3=7$ respectively. Each group is input sequence for each Mixed-Radix 4-2 Butterfly. After the input sequences pass the first Mixed-Radix 4-2 Butterfly stage, the order of output value is expressed with small number below each butterfly output line in the figure 3. The proposed Mixed-Radix 4-2 is composed of two radix-4 butterflies and four radix-2 butterflies. In the first stage, the input data of two radix-4 butterflies which are expressed with the equation $B_4(o, n_3, k_j)$ $B_4(i, n_3, k_l)$, are grouped with the $x(n_3)$, $x(N/4 \pm n_3)$, $x(N/2 \pm n_3)$, $x(3N/4 \pm n_3)$ and $x(N/8 \pm n_3)$, $x(3N/8 \pm n_3)$, $x(5N/8 \pm n_3)$, $x(7N/8 \pm n_3)$ respectively. After the each input group data passes the first radix-4 butterflies, the outputted data is multiplied by the special twiddle factors. Then, these outputted sequences are inputted into the second stage which is composed of the radix-2 butterflies. After passing the second radix-2 butterflies, the outputted data are multiplied by the twiddle factors. These twiddle factors $W_Q(1+k)$ is the unique multiplier unit in the proposed Mixed-Radix 4-2 Butterfly with simple bit reversing the output sequences. Finally, we can also show order of the output sequences Fig above. The order of the output sequence is 0,4,2,6,1,5,3 and 7 which are exactly same at the simple binary bit reversing of the pure radix butterfly structure. Consequently, proposed mixed radix 4-2 butterfly with simple bit reversing output sequence include two radix 4 butterflies, four radix 2 butterflies, one multiplier unit and additional shift unit for special twiddle factors.

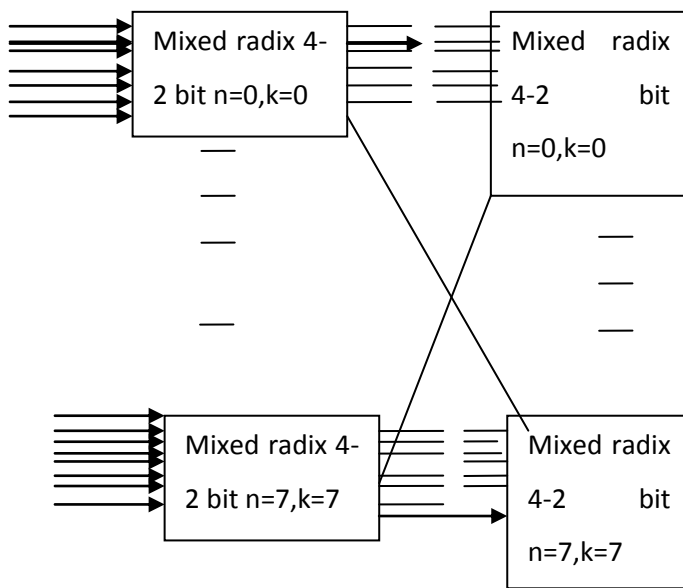


Figure 3: Proposed Mixed-Radix 4-2 Butterfly for 128 point FFT

6. RESULTS & FUTURE WORK

Employing the parametric nature of this core, the OFDM block is synthesized on one of Xilinx's Virtex-II Pro FPGAs with different configurations. The results of logic synthesis for 64 point FFT based orthogonal frequency division multiplexing (OFDM) using Radix-2, Radix-4, split Radix and mixed radix 4-2 are presented in Table 1. We analysis the 64-point FFT based ofdm is chosen to compare the number of CLB slices and Flip Flop for different FFT architectures.

Table 1: Different of FFT Algorithm based on CLB Slices, DFF, Function Generators

OFDM with 128 pt FFT	CLB Slices	UTILIZATION	DFF	Function Generator
Radix-2	2793	7.21%	5786	3285
Radix-4	2536	6.01%	3482	5672
Split Radix	2477	6.10%	5764	4178
Mixed Radix 4-2	2172	4.87%	4734	5242

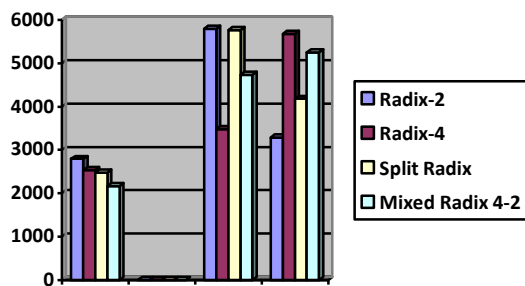


Fig 5 :Effect of slices

Table 2: Synthesis Simulation Results

Type(12*12)	Area (cell)	Power (mW)	Latency(nS)
Fixed with MB multiplier	2771(100%)	364.6(100%)	8.4(100%)
Proposed Multiplier	1249(45%)	141.9(39%)	9.91(118%)

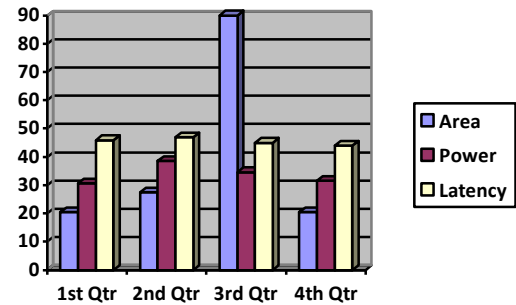


Fig 6: Effect of synthesis

7. CONCLUSION

The newly proposed modified radix-2/4 algorithm and its pipeline architecture R24SDF, which is regular and extensible for any $2n$ -point FFT. The R24SDF architecture allows one half of the programmable multipliers used in the R22SDF architecture being replaced by the newly proposed CSD constant multipliers. In this paper, we design an OFDM for different algorithms implemented in OFDM modem are identified. It was found during the algorithm design that many blocks need complex multipliers and adders and therefore special attention needs to be given to optimize these circuits and maximize reusability. In particular, the models have been applied to analyze the performance of mixed-radix FFT architectures used in OFDM. Actual hardware resource requirements were also presented and simulation results were given for the synthesized design. We proposed complex multipliers were able to provide a reduction of more than 30% in both power consumption and area in terms of multiplication complexity. The 128-point Mixed Radix FFT based OFDM architecture was found to have a good balance between its performance and its hardware requirements and is therefore suitable for use in OFDM systems.

8. REFERENCES

- [1] Bruno Fernandes , Helena Sarmento "Implementation of an 128 FFT for a MB-OFDM Receiver" 978-972-789-304-1 REC'2010 page no 45-48
- [2] Tai-cheng Lee,Yen-chuan Huang" The design and analysis of a Miller-divider-based clock generator for MBOA-UWB application" solid state circuits.IEEE journal v41,Isu 6 page no.1253-1261.
- [3] "A High-Speed Low-Complexity Modified FFT Processor for High Rate WPAN Applications "Very Large Scale Integration (VLSI) Systems, IEEE Transactions on volu :pp,Issue :99 page no 1-5
- [4] K.J. Cho, K.C. Lee, J.G. Chung, and K.K. Parhi, "Design of low-error fixed-width modified booth multiplier," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol.12, no.5, pp.522–531, 2004.
- [5] Shousheng. He and Mats Torkelson, "Design and Implementation of a 1024-point Pipeline FFT Processor",

- IEEE Custom Integrated Circuits Conference, May. 1998, pp. 131-134.
- [6] Shousheng He and Mats Torkelson, "Designing Pipeline FFT Processor for OFDM (de)Modulation", IEEE Signals, Systems, and Electronics, Sep. 1998, pp. 257-262.
- [7] Shousheng He and Mats Torkelson, "A New Approach to Pipeline FFT Processor", IEEE Parallel Processing Symposium, April. 1996, pp.776-780.
- [8] C. Sidney Burrus, "Index Mapping for Multidimensional Formulation of the DFT and Convolution", IEEE Trans. Acoust., Speech, and Signal Processing, Vol. ASSP-25, June. 1977, pp. 239-242.
- [9] Lihong Jia, Yonghong GAO, Jouni Isoaho, and Hannu Tenhunen, "A New VLSI-Oriented FFT Algorithm and Implementation", IEEE ASIC Conf., Sep. 1998, pp. 337-341.
- [10] Martin Vetterli and Pierre Duhamel, "Split-Radix Algorithms for Length-ptmDFT's", IEEE Trans. Acoust, Speech, and Signal Processing, Vol. 37, No. 1, Jan. 1989, pp. 57-64.
- [11] Daisuke Takahashi, "An Extended Split-Radix FFT Algorithm", IEEE Signal Processing Letters, Vol. 8, No. 5, May. 2001, pp. 145-147.
- [12] Y.T. Lin, P.Y. Tsai, and T.D. Chiueh, "Low-power variable-length fast Fourier transform processor", IEE Proc. Comput. Digit. Tech, Vol. 152, No. 4, July. 2005, pp. 499-506.
- [13] Byung G. Jo and Myung H. Sunwoo, "New Continuous-Flow Mixed-Radix (CFMR) FFT Processor Using Novel In-Place Strategy", IEEE Transactions on Circuits and Systems, Vol.52, No. 5, May. 2005.
- [14] Gordon L. Demuth, "Algorithms for Defining Mixed Radix FFT Flow Graphs", IEEE Trans Acoust, Speech, and Signal Processing, Vol. 37, No. 9, Sept. 1989, pp. 1349-1358.
- [15] Kyung L. Heo, Jae H. Baek, Myung H. Sunwoo, Byung G. Jo, and Byung S. Son, "New In-Place Strategy for a Mixed-Radix FFT processor", IEEE SOC Conference, Sep. 2003, pp. 8 1-84.
- [16] S. He and M. Torkelson, "Designing pipeline FFT processor for OFDM (de) modulation," Proc. IEEE URSI Int. Symp. Signals, Syst., Electron., pp.257–262, 1998.
- [17] J. Melander, Design of SIC FFT Architectures, Linköping Studies in Science and Technology, Thesis no.618, Linköping University, Sweden, 1997.
- [18] J.Y. Oh, J.S. Cha, S.K. Kim, and M.S. Lim, "Implementation of orthogonal frequency division multiplexing using radix-N pipeline fast Fourier transform (FFT) processor," Jpn. J. Appl. Phys. 1, vol.42, Regul. Pap. Short Notes, no.4B, pp.1–6, 2003.
- [19] K.K. Parhi, VLSI digital signal processing systems, John Wiley & Sons, USA, 1999. [5] S.M. Kim, J.G. Chung, and K.K. Parhi, "Design of low error CSD fixed-width multiplier," IEEE Inter. Symp. Cir., Syst., vol.1, pp.1-69–1-72, 2002.2002.

9. AUTHORS PROFILE

K. Umapathy received his B.E degree in Electronics and Communication Engineering from Madurai-Kamraj University in the year 1992 and M.S degree in Systems and Information Engineering from Birla Institute of Technology, Pilani in the year 1997. Presently he is pursuing PhD in JNT University, Anantapur. He is working as Professor of the Department of IT in Arunai College of Engineering, Tiruvannamalai, Tamilnadu. He has more than 18 years of experience in the field of Engineering Education and Administration. He is a Life member of ISTE.

Dr. D. Rajaveerappa received his BE degree in Electronics & Communication Engineering from National Institute of Technology, Tiruchirapalli in 1985 and MTech degree in Communication Systems from Indian Institute of Technology, Madras in 1988. He obtained his PhD degree in Multimedia Wireless Networks from Indian Institute of Science, Bangalore in 2005. Presently, he is working as Professor in the Department of ECE at Loyola Institute of Technology, Chennai. He has more than 23 years of experience in the field of Engineering Education and Research.