Review Paper on Radix-2 DIT and DIF Fast Fourier Transform

Kausar Ali
PG Scholar
Electronics and
Communication Department
Truba College of
Science and Technology,
Bhopal

Nashrah Fatima
Assistant Professor
Electronics and
Communication Department
Truba College of
Science and Technology,
Bhopal

Paresh Rawat, PhD
Professor
Electronics and
Communication Department
Truba College of
Science and Technology,
Bhopal

ABSTRACT

With the arrival of recent generation in the field of VLSI and communication, there is also an ever growing demand for low power consumption and minimum consumed chip area. It is also a well-known reality that the chip location and maximum combinational path delay (MCPD) unit forms an indispensable a part of processor layout. Due to this regard, high speed and low area architectures grow to be the want of the day. A fast Fourier transform (FFT) is any rapid algorithm for computing the DFT. The decimation-in-time (DIT) fast Fourier remodel (FFT) very often has advantage over the decimation-in-frequency (DIF) FFT for most real-valued programs, like speech/photo/video processing, biomedical sign processing, and time-collection evaluation, and so on., since it does now not require any output reordering.

Keywords

FFT, Decimation in Time, Decimation in Frequency, real Value data

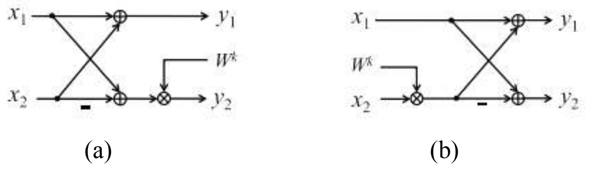
1. INTRODUCTION

Virtual signal processing (DSP) is the mathematical manipulation of an information signal to modify or enhance it in a few manners. It's far characterized via the representation of discrete time, discrete frequency, or other discrete domain alerts with the aid of a sequence of numbers or symbols and the processing of these alerts [1].

The goal of DSP is typically to measure, filter out and/or compress non-stop real-world analog alerts. The first step is typically to convert the signal from an analog to a virtual form, via sampling after which digitizing it using an analog-to-digital converter (ADC), which turns the analog sign right into a circulation of numbers. However, frequently, the specified output signal is another analog output signal, which requires a digital-to-analog converter (DAC). Despite the fact that this procedure is extra complex than analog processing and has a discrete price range, the utility of computational energy to digital sign processing allows for many advantages over analog processing in many programs, including mistakes detection and correction in transmission in addition to information compression. DSP algorithms have long been run on trendy computers, as well as on specialised processors referred to as the virtual signal processor and on purpose-built hardware which include software-specific integrated circuit (ASICs). These days there are additional technology used for digital sign processing together with more powerful widespread reason microprocessors, subject-programmable gate arrays (FPGAs), digital sign controllers (on the whole for commercial apps together with motor manage), and flow processors, amongst others [2-3]. The FFT is one of the most commonly used digital sign processing set of rules. These days, FFT processor has been broadly used in virtual sign processing field applied for OFDM, MIMO-OFDM verbal exchange systems. FFT/IFFT processors are key additives for an orthogonal frequency division multiplexing (OFDM) based Wi-Fi IEEE 802.16 broadband communique system; it's far one of the maximum complex and in depth computation module of diverse wireless requirements physical layer (OFDM-802.11a, MIMO-OFDM 802.11, 802.sixteen, 802.16e) [4].

A few folded pipeline architectures have been proposed for the computation of RFFT [3], [4], in which butterfly operations are multiplexed into a small logic unit. The structures in [3] and [4] may want to provide good enough throughput for a few applications, but the garage complexity of these systems is still very excessive. Some in-area architectures have also been proposed for RFFT the use of specialized packing algorithms [5], [6]. Memory-struggle for examining/write operation is located to be the fundamental challenge in the layout of algorithms and architectures for in-area computation [7]. Lately, an in-place structure and struggle-loose reminiscence addressing scheme were proposed for non-stop processing of RFFT [8].

The FFT algorithms are categorised into vast classes, specifically, the decimation-in-time (DIT) and the decimation-in frequency (DIF) algorithms. The important thing variations between the two are shown in Fig. 1. In case of DIF algorithm (Fig. 1(a)), then enter samples are fed to the computing structure in their natural order, even as the output is generated in bit-reversed order. Then again, in case of DIT set of rules (Fig. 1(b)), the input samples need bit-reversal reordering before being processed, even as the output FFT coefficients are generated in herbal order. In distinctive RFFT programs inclusive of picture and video processing, biomedical sign processing, and time-collection analysis and so on., the whole input series is generally available together at the identical time for the FFT computation. The DIT RFFT has a bonus over the DIF form for those packages, because a DIT RFFT shape need no longer look forward to the advent of input samples however can produce the outputs as quickly as these are computed.



(a) (b)

Figure 1: (a) DIF FFT butterfly (b) DIT FFT butterfly

2. LITERATURE REVIEW

Pramod Kumar Meher et al. [7], the decimation-in-time (DIT) fast Fourier transforms (FFT) very frequently has gain over the decimation-in-frequency (DIF) FFT for maximum actual-valued packages, like speech/photo/video processing, biomedical sign processing, and time-collection evaluation, etc., because it does not require any output reordering. Except, the DIT FFT butterfly entails less computation time than its DIF counterpart. In this paper, we gift an efficient structure for the radix-2 DIT actual-valued FFT (RFFT). We present right here the essential mathematical components for casting off the redundancies in the radix-2 DIT RFFT, and present a formulation to regularize its float graph to facilitate folded computation with a simple control unit. We propose here a check in-based storage layout which includes significantly much less place on the cost of a touch better latency as compared with the conventional RAM-based garage. They cope with technology for folded in-place DIT RFFT computation with register-based garage is challenging seeing that both examine and write operations are performed within the equal clock cycle at exceptional locations. Therefore, we gift here a simple component of deal with generation for the proposed radix-2 DIT RFFT structure.

The proposed structure includes sixty 1% less vicinity and 40% much less power intake than those of [8], on the common, for FFT sizes 16, 32, 64, and 128. It includes 70% less place-postpone product and fifty seven% much less electricity in keeping with pattern than the ones of the other, on average, for the equal FFT sizes.

Manohar Ayinala et al. [8], this quick affords a unique scalable architecture for in-area speedy Fourier transforms (IFFT) computation for real valued indicators. The proposed computation is based totally on a changed radix-2 algorithm, which eliminates the redundant operations from the flow graph. a brand new processing element (PE) is proposed the usage of two radix-2 butterflies that can procedure four inputs in parallel. A singular war-loose reminiscence-addressing scheme is proposed to ensure the non-stop operation of the FFT processor. Moreover, the addressing scheme is extended to guide multiple parallel PEs. The proposed actual-FFT processor concurrently requires fewer computation cycles and lower hardware fee in comparison to prior work. As an instance, the proposed design with PEs reduces the computation cycles by using a aspect of two for a 256-factor real fast Fourier rework (RFFT) in comparison to a prior paintings while maintaining a lower hardware complexity. The quantity of computation cycles is decreased proportionately with the increase in the range of PEs.

Shashank Mittal et al. [9], fast Fourier transforms (FFT) is one of the most primary and essential operation performed in software program defined Radio (SDR). consequently designing a regular, reconfigurable FFT computation block with low region, put off and strength requirement is very essential. these days it's miles proven that Bruun's FFT is perfectly suited for SDR even if running with higher bit precision to maintain same NSR. on this paper, authors have proposed a brand new architecture for Bruun's FFT using a distributed method for incrementing the variety of bits (precision) with successive ranges of FFT. it's also shown that proposed structure in addition reduces the hardware requirement of Bruun's FFT with negligible changes in its NSR. The proposed layout makes Bruun's FFT, a better alternative for most practical cases in SDR. an in depth contrast of Bruun's traditional and proposed hardware architectures for same NSR is finished and effects of FPGA and ASIC implementations are supplied and discussed.

Byung G. Jo et al. [10], the paper proposes a brand new continuous-float blended-radix (CFMR) fast Fourier transform (FFT) processor that uses the MR (radix-four/2) set of rules and a novel in-area method. the existing in-region strategy supports handiest a set-radix FFT algorithm. In evaluation, the proposed in-place approach can aid the MR set of rules, which lets in CF FFT computations no matter the period of FFT. the unconventional in-place approach is made with the aid of interchanging storage places of butterfly outputs. The CFMR FFT processor offers the MR set of rules, the in-area approach, and the CF FFT computations at the equal time. The CFMR FFT processor requires most effective two -word memories because of the proposed in-place strategy. Further, it makes use of one butterfly unit that can perform either one radix-4 butterfly or radix-2 butterflies. The CFMR FFT processor the usage of the 0.18 m SEC cellular library includes 37,000 gates with the exception of memories, requires simplest 640 clock cycles for a 512-factor FFT and runs at a hundred MHz. Therefore, the CFMR FFT processor can lessen hardware complexity and computation cycles in comparison with existing FFT processors.

Table 1: Comparison of the FFT architecture considering latency and area

Reference	Multiplier	Adder	MUX/ DMUX	Computatio n Time
[10]	12	22	38	N/4 log$_4$N
[9]	9	19	38	N/4 log$_2$N
[8]	4	6	38	N/4 log$_2$N
[7]	4	6	N+12	N/4 log$_2$N

3. FFT ALGORITHM

A fast Fourier transform (FFT) is an algorithm to compute the discrete Fourier transform (DFT) and its inverse. Fourier analysis converts time (or space) to frequency and vice versa; an FFT rapidly computes such transformations by factorizing the DFT matrix into a product of sparse (mostly zero) factors. As a result, fast Fourier transforms are widely used for many applications in engineering, science, and mathematics. Show the butterfly operations for radix-2 DIF FFT in figure 2 and figure 3. The radix-2 algorithms are the simplest FFT butterfly algorithm.

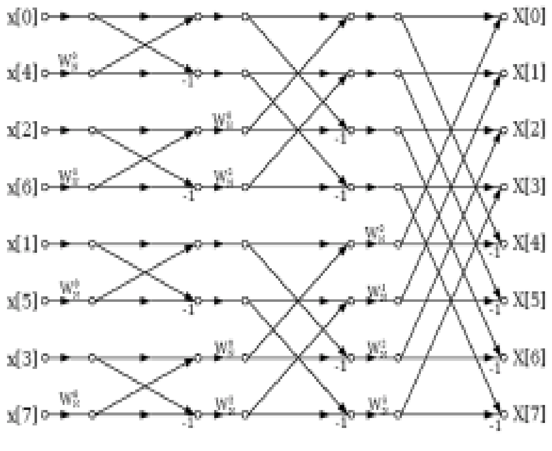


Figure 2: Radix-2 Decimation in Time Domain FFT Algorithm

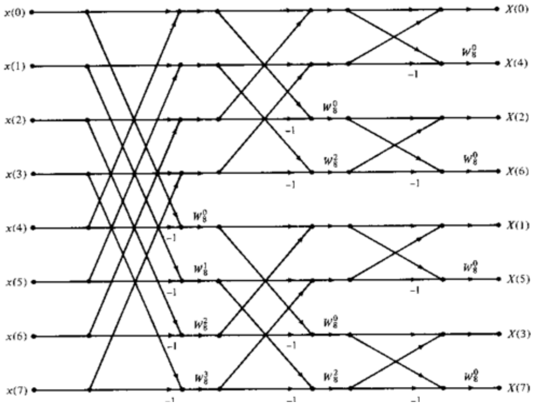


Figure 3: Radix-2 Decimation in Frequency Domain FFT Algorithm

4. PROPOSED METHOD

Since complex multiplication is an expensive operation, wetend to reduce the multiplicative complexity of the twiddlefactor inside the butterfly processor by calculating only threereal multiplications and three add/subtract operations as inThe twiddle factor multiplication:

$$R + jI = (X + jY).(C + jS) \qquad (1)$$

However the complex multiplication can be simplified:

$$R = (C - S).Y + Z \qquad (2)$$

$$I = (C + S).X - Z \qquad (3)$$

$$Z = C.(X - Y) \qquad (4)$$

C and S are pre-computed and stored in a memory table. Consequently it's far essential to save the following 3

Coefficients C, C + S, and C - S. The implemented algorithm of complicated multiplication used on this work uses 3 multiplications, one addition and two subtractions as shown in Fig. 4. This is completed on the cost of an additional 1/3 reminiscence table that's given in (4). Within the hardware description language (VHDL) program, the twiddle issue multiplier was carried out using thing instantiations of three lpm-mult and three lpm-add-sub modules from Altera library.

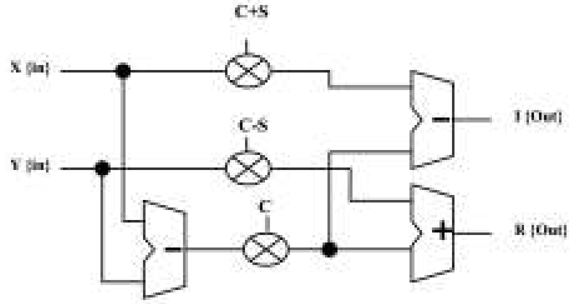


Figure 4: Implementation of complex multiplication

Signed Multiplier:-

Baugh-Wooley set of rules for the signed binary multiplication is based at the concept proven in figure 5. The set of rules specifies that all feasible AND terms are created first, and then despatched thru an array of half-adders and complete-adders with the carry-outs chained to the subsequent maximum enormous bit at every degree of addition. terrible operands can be multiplied the usage of a Baugh-Wooley multiplier.

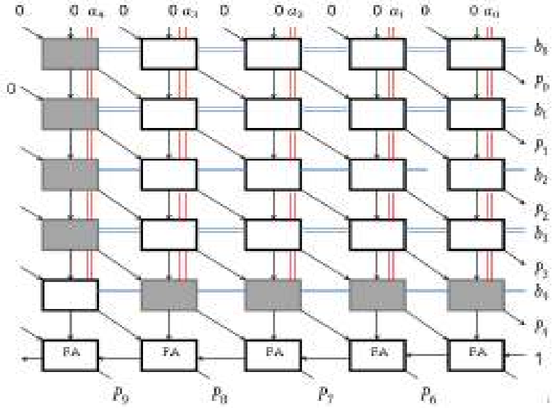


Figure 5: Signed Multiplier

5. DELAY AND AREA METHODOLOGY

The delay and vicinity assessment technique considers all gates to be made from AND, OR, and Inverter (AOI), every having postpone identical to 1 unit and location identical to one unit.

We then upload up the range of gates inside the longest route of a common sense block that contributes to the maximum postpone. The region evaluation is performed by means of counting the overall quantity of AOI gates required for each common sense block.

6. METHOD OF DESIGN

1. Design fast fourier transform (FFT) using radix-2, radix-3 and radix-4 in different device family i.e. Spartan-3, Virtex-4 and Virtex-7.
2. Design FFT using 8-point,16-point, 32-point, 64-point and 128-point in different radix algorithm.
3. Design FFT using Signed input and complex input.
4. Design FFT using decimation in time (DIT) and decimation in frequency (DIF) and reduced the time and area.
5. Hand calculation of area and delay in FFT and compared existing algorithm.

7. CONCLUSION

The top objective is to construct a FFT in an effort to have low electricity consumption and lesser region. The parameters (i) energy intake (ii) area occupancy had been given due consideration for evaluating the proposed circuit with different FFTs. The circuits had been simulated the usage of version-Sim 6.3c and synthesized with Xilinx ISE 14.1.The performance of various sixty four factor FFT including Radix-2, Radix-four, break up Radix, mixed -radix four-2, R2MDC and the proposed changed R2MDC were performed and their performance were analyzed with appreciate to the quantity of CLB slices, utilization component and power intake.

8. REFERENCES

[1] Charles. Roth Jr., "Digital Systems Design using VHDL", Thomson Brooks/Cole, 7th reprint, 2005.

[2] S. S. Kerur, Prakash Narchi, Jayashree C N, Harish M Kittur and Girish V A, "Implementation of Vedic multiplier for Digital Signal Processing", International Conference on VLSI, Communication & Instrumentation (ICVCI) 2011, Proceedings published by International Joural of Computer Applications® (IJCA), pp.1-6.

[3] Himanshu Thapaliyal and M.B Srinivas, "VLSI Implementation of RSA Encryption System Using Ancient Indian Vedic Mathematics", Center for VLSI and Embedded System Technologies, International Institute of Information Technology Hyderabad, India.

[4] Jagadguru Swami Sri Bharati Krishna Tirthaji Maharaja, "Vedic Mathematics: Sixteen simple Mathematical Formulae from the Veda", Delhi (2011).

[5] Harpreet Singh Dhillon and Abhijit Mitra, "A Reduced-bit Multiplication Algorithm for Digital Arithmetic", International Journal of Computational and Mathematical Sciences, Febrauary 2008, pp.64-69.

[6] Sumit Vaidya and Depak Dandekar. "Delay-power performance comparison of multipliers in VLSI circuit design". International Journal of Computer Networks & Communications (IJCNC), Vol.2, No.4, July 2010.

[7] Pramod Kumar Mehe*,* Basant Kumar Mohanty*,* Sujit Kumar Patel, Soumya Ganguly, and Thambipillai Srikanthan*, "*Efficient VLSI Architecture for Decimation-in-Time Fast Fourier Transform of Real-Valued Data*",* IEEE Transactions on Circuits And Systems—I: Regular Papers, Vol. 62, No. 12, December 2015.

[8] M. Ayinala, Y. Lao, and K. K. Parhi, "An in-place FFT architecture for real-valued signals," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 60, no. 10, pp. 652–656, Oct. 2013.

[9] Shashank Mittal, Md. Zafar Ali Khan and M.B. Srinivas, "Area Efficient High Speed Architecture of Bruun's FFT for Software Defined Radio", 1930-529X/07/$25.00 © 2007 IEEE.

[10] B. G. Jo and M. H. Sunwoo, "New continuous-flow mixed-radix (CFMR) FFT processor using novel in-place strategy," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 52, no. 5, pp. 911–919, May 2005.