ISSN: 0369-8963

Computation of Real and Complex Valued Signal in Parallel Pipeline FFT Architectures

SURYA P[1], ARUNACHALAPERUMAL C [2], DHILIPKUMAR S[3]

[1]Research Scholar, Anna University, Chennai, India.

[2]Professor in S.A. Engineering College, Chennai, India.

[3]AssistantProfessor in Loyola ICAM College of Engineering and Technology, Chennai, India.

**Corresponding author: Dr.C.Arunachalaperumal*

Abstract

Computated valued Fast Fourier Transform introduces an innovative method to estimate through parallel pipelined architectures. The functional frequency of the recommended architecture can be decremented consecutively decrements the power consumption besides the calculation of Real Fast Fourier Transform (RFFT). In this work with these two above mentioned ideas, Complex Valued Fast Fourier Transform (CFFT) and 128 points Radix-2 Real Valued Fast Fourier Transform (RFFT) architectures are developed. FPGA tool such as XILINX SPARTAN -3E, Virtex-4Xc4vfx12 widely utilized experimental device for the execution of multipliers in this work. The proposed work exercise surplus calculation of FFT samples to deplete the hardware complications. Differentiation among recommended and preceding architectures is carried out. 75 % of power is reduced in 2 parallel CFFT and RFFT architecture.

Keywords: *Fast Fourier Transform, Parallel, Pipeline, radix 2^3, real signal, complex signal and FPGA*

Introduction

FFT is one of the well known technologies in DSP. This algorithm is employed for several applications that include signal processing, image processing, filtering, spectral investigation of digital signals in the field of communication, condition monitoring system. The real time signal processing utilization with respect to hand held devices cost of FFT processors is important.

So far there are several research works have been executed on designing pipelined architecture for computing FFT with complicated valued signals (CFFT). Cooley-Tukey radix 2 FFT is much better in specific [1].

Algorithm that include radix 2^2[2], radix -4[3] and split radix [4] includes the executed drew on fundamental radix 2 FFT technique. Research works from [5-10] have focused these

 https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

techniques. The popular method is the Radix 2 multipath delay commutator (R2MDC) in radix 2 FFT pipelined execution. The Effectual utilization of memory space in R2MDCinitiates the Radix -2 Single Path delay Feedback (R2SDF). This structural design has less memory[11].

Papers [12-14] have addressed FFT parallel architectures. [12]&[13]have proposed specific N-point FFT architecture while in [14] hypercube theory is applied to draw the architecture. Here, the conventional technique has been followed to develop the architecture and it not well incorporated. Besides, the hardware architectures are not completely employed and also more difficulty. High rise throughputs as well as less power models are needed things in high speed communication era to attain the speed and required power. Bearing this in mind, this work focuses to sketch these types of architectures from FFT flow graph.

For real input samples, the spectrum is uniformly arranged and almost 50% of the functions are excess. Several employments such as speech, image processing, communication and biomedical signal processing (CBSP) employ special kit execution [15&16]. Brunnalgorithm [17 & 18] is exerted for pipelined architectures with real value signal. But it is not broadly applicable.

Extracting the CFFT redundancies and calculation of RFFT has been discussed in [19 & 20]. Here Digital signal Processor played the role. In [19], RFFT is computed through CFFT architecture in an effectual way. In this concept, doubling algorithm comes in to compute two RFFT consequently using CFFT. is the later one that structures a complicated series of length N/2 acquiring and even indexed samples of length N and determines N/2 point CFFT of the complicated order. Packing algorithm is the later one that structures a convoluted arrangement of length N/2 gaining the real input odd and even recorded samples of length N and decides N/2 point CFFT of the entangled request. But both of these algorithms need additional functions to achieve the final result.

This Paper discusses a new technique to structure the FFT architecture through folding transformation.In this transformation, several butterflies in the similar column may be mapped to single butterfly unit. For N FFT size, a folding factor of N/2 brings about 2 parallel models. With a folding factor of N/4, 4 parallel architectures can be designed in which 4 samples are progressed in the identical clock cycle. Disparate folding sets brings up a family of FFT

 https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

architecture. Latency reduction and diminishing the number of storage elements are the main intention of designing the folding sets. Here, structure of previous architecture is discussed. Then, new architectures are drawn on for different radices and for several level of similarity and for moreover the Decimation In Time(DIT)or else Decimation In Frequency(DIF) flow graphs.

Here the paper is arranged as follows: Segment II depicts FFT algorithm. Segment III exhibits FFT structure according to the Folding Transformation.

FFT Algorithm:

In general, FFT algorithm is divided into two types: Decimation-in- Time (DIT) as well as Decimation –in –Frequency (DIF) (Figure 1.a & 1.b).

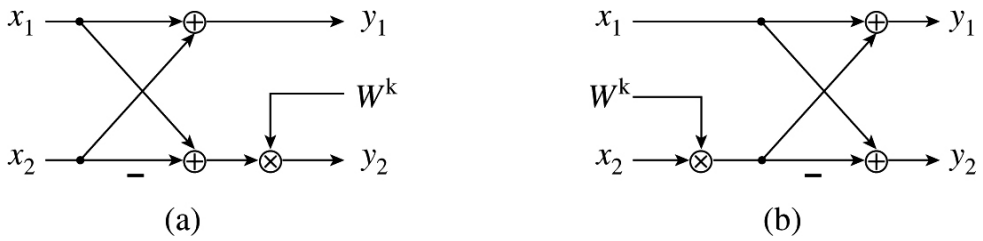

Figure 1. a)DIF-FFT Butterfly b) DIT-FFT Butterfly

Radix -2 DIF Algorithm:

This algorithm contains some set of rules for calculating the DFT of the time domain input series on digital equipment. This designs an input series x(i) within its correspondent frequency domain x(f).The I point DFT of x (i) is stated below eq. 1.

$$X(f) = \sum_{i=0}^{i-1} x(i) W_i^{fi}; \qquad 0 \le k \le I-1 \qquad (1)$$

Where $W_i^{fi}$is mentioned as a complicated twiddle factor that exhibits the rotation of FFT and set as,(eq.2)

 https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

$$W_i^{fi} = e^{\left(\frac{-j2\pi i}{I}\right)} = \cos\left(\frac{2\pi ki}{I}\right) - j\sin\left(\frac{2\pi ki}{I}\right) \qquad (2)$$

Through the usage of periodicity and resemblance of the complex twiddle factors, the DFT of an input data sequence is calculated. Through FFT algorithm, the computation burden in DFT is decreased by changing $O(I^2)$ to $O(I \log_2 I)$. As represented in equation(A), the Radix-2-DIF algorithm decomposes the I point output X(f) into an even numbered samplings X(2f) and odd number samplings X(2f+1).This is specified here(i.e.) eq. (3) and (4).

$$X(2f) = \sum_{i=0}^{\frac{I}{2}-1}\left[x(i) + x(i+\frac{I}{2})\right]\frac{W_I^{ki}}{2} \qquad (3)$$

$$X(2f+1) = \sum_{i=0}^{\frac{I}{2}-1}\left(x(i) - x(i+\frac{I}{2})\right)\frac{W_I^{ki}}{2}W_I^i \qquad (4)$$

Where $0 \le f \le \frac{I}{2} - 1$ and I is an integer power of 2.

Both equations C and D are integrated to articulate the Radixx-2 butterfly process in the DIF algorithm as illustrated in figure 2.

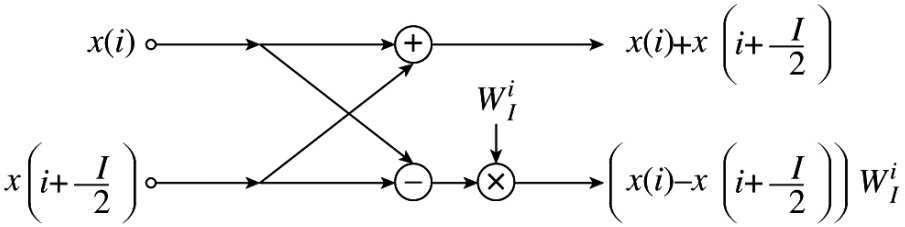

Figure 2. Fundamental Radix-2 butterfly process in DIF.

By repeating this butterfly operation, the FFT is determined. For that, it involves addition, deduction afterward intricate twiddle factor multiplication. In the sequence of i= $\log_2$ I stages, the N-point FFT is enumerated wherein each and every part incorporates sum of $\left(\frac{I}{2}\right) log_2 I$ butterfly units.

The signal flow graph representation of a 8 point Radix 2 FFT DFT is depicted in figure 3 followed by three phases and 16 butterfly units.

 https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

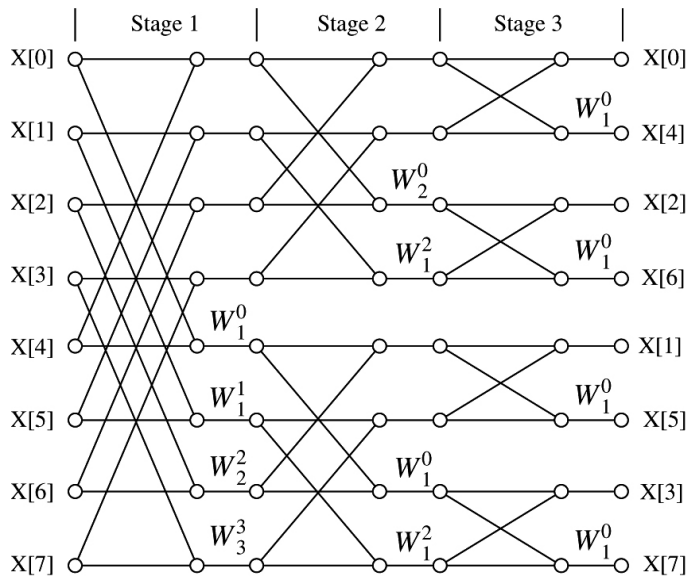

Figure 3. Radix-2 8 point sequence

Suggested Architecture with Real Inputs (RFFT):

For complex inputs, to attain CFFT, it is necessary to do internal calculations and outputs of the flow graph. On the contrary, for real inputs the graph is streamlined by adopting RFFT characteristics as given below:

In RFFT case, the input series is real (i.e)

$\forall n, x[n] \in Real\ Inputs$

It can be easily proved that if $x[i]$ is real, and then output $\mathrm{X}[f]$ is symmetric in nature. Mathematically it is interpreted by equation 5.

$$X[I - f] = X^*[f] \qquad (5)$$

Through this characteristic, $\left(\frac{I}{2}\right) - 1$ outputs can be eliminated that are superfluous. Several methodologies in various literatures determine the frequencies through indices. An innovative methodology to find the superfluous sample is followed to find those samples in the graph(Figure 4).

 https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

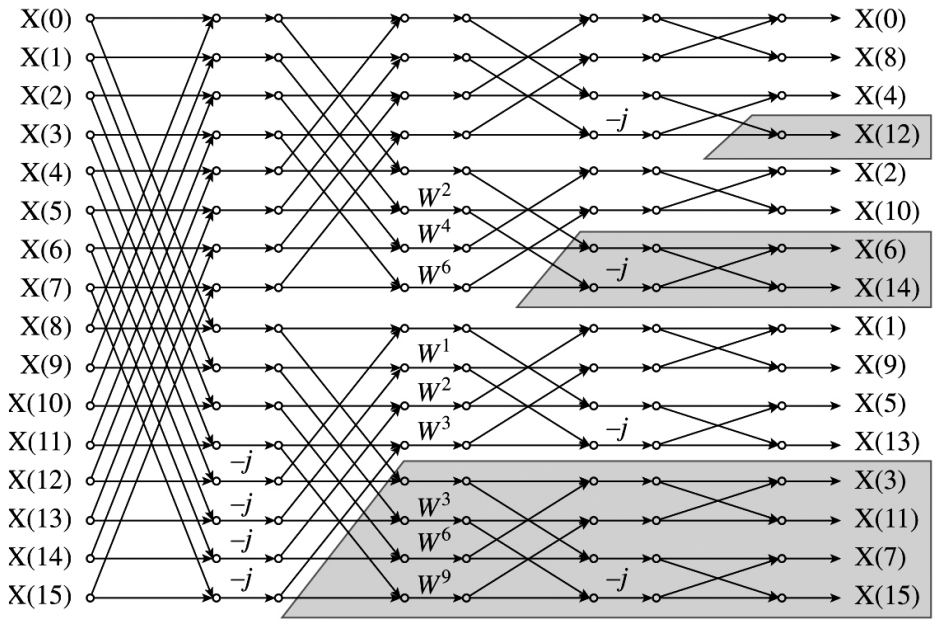

Figure 4. Flow graph of a radix-2 16-point DIF FFT.

Eliminate the shaded portion in fig 4 and FFT output alone is required. Another elucidation is that $\mathrm{Im}(x[I]) = 0$ because of actual inputs. Each part of data is an actual till it is multiplied with the complicated number, hence the summations are actual. In this Paper a new pipelined architecture with respect to these changes can step two samples in parallel.

Two Parallel Radix 2 Architecture:

The pipelined arrangement of RFFT is drawn with succeeding folding sets identical to CFFT arrangement. The nodes from A0,A1,…A7 represent the foremost stage of FFT and B0,B1…B7 embody the next stage of FT and C0,C1….C7 represent the Third stage of FFT and D0,D1…D7 represents the final stage of FFT. The architecture coincides with radix2 DIF CFFT. By continuing this to larger radices, the multiplier burden will be diminished.

Two Parallel Radix - 2^2 Architecture:

Linear mapping of the radix 2 DIF FFT approach to the experimental setup is easy.Radix-2 SDF technique can be followed by altering the first complicated butterfly phase through to reality. However, this cannot exercise the characteristics of RFFT, in which nearly partial samples are redundant.

 https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

Proposed Two Parallel Radix – 2^3 Architecture:

The suggested architecture can measure 128 point input specimens connected in the same direction as to oppose to cascade architectures are shown in figure 5. Two disparate architectures are drawn out through two dissimilar scheduling techniques. That is altering the folding sequence of the butterfly nodes

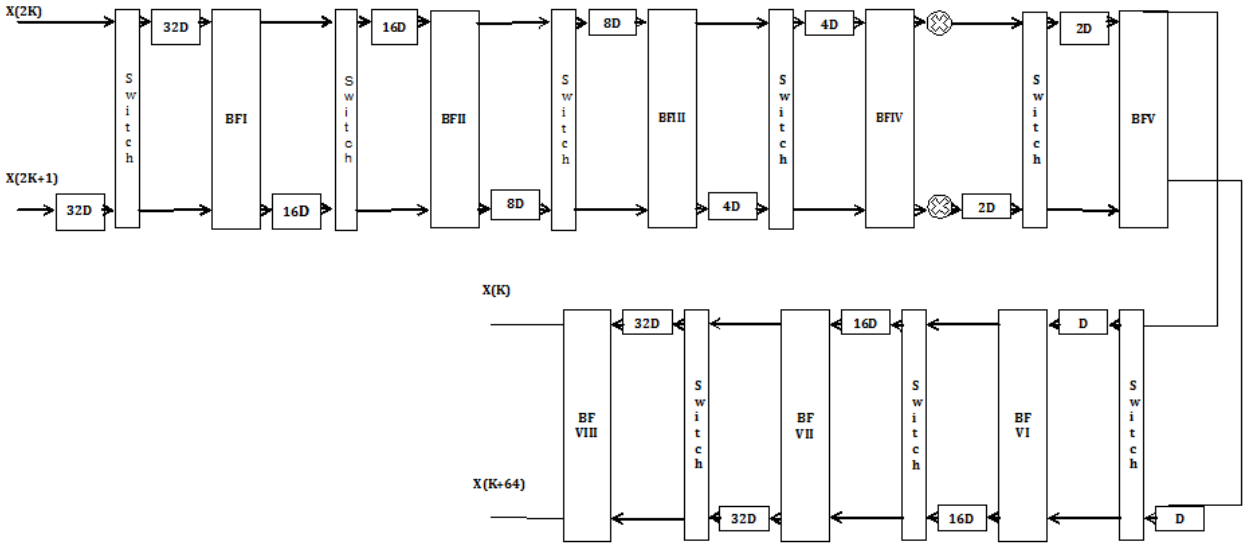

Figure 5. Proposed 2 Parallel 128 Point Radix-2^3 FFT Architecture

Multiplier:

Multiplier plays an imperative role in present day DSP applications. Various literature review reports that about 70% of instructions in DSP applications perform both addition and multiplication operation. This brings up the necessity of high speed multiplier. Fast activating tools with less power utilization are the primary requirement of every customer. By speeding up the tools with diminished power consumption, it leads to enrichment in the performance of the equipment. To diminish the power consumption, it is preferable to minimize the number of functions thereupon decreases the dynamic power. This is a significant part of power utilization. Hence high speed and low power multiplier is mandatory. Disparate types of multipliers are Vedic,

https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

FFT with Vedic Multiplier:

FFT diminishes the complexity in summations and multiplications from N2 to N/2log2N as well as Nlog2N consequently. For N=16, complicated multiplications are 32 and number of summations are 64. In the fundamental butterfly function DIT algorithm, the two inputs are integrated to offer the output. The simple multiplier execution requires 16 rows of fractional product and every row comprises of 16 fractional product sample bits for 16 bits word length.To assemble these 16 partial product rows, huge experimental set up is required to acquire the outcome in sum and carry form. Hence all these multiplications are preferred using Vedic multiplier

Experimental Realization of 128 point Multiplier:

With the generic adder developed, 128 point multiplier is proposed in this work is shown in figure 6.

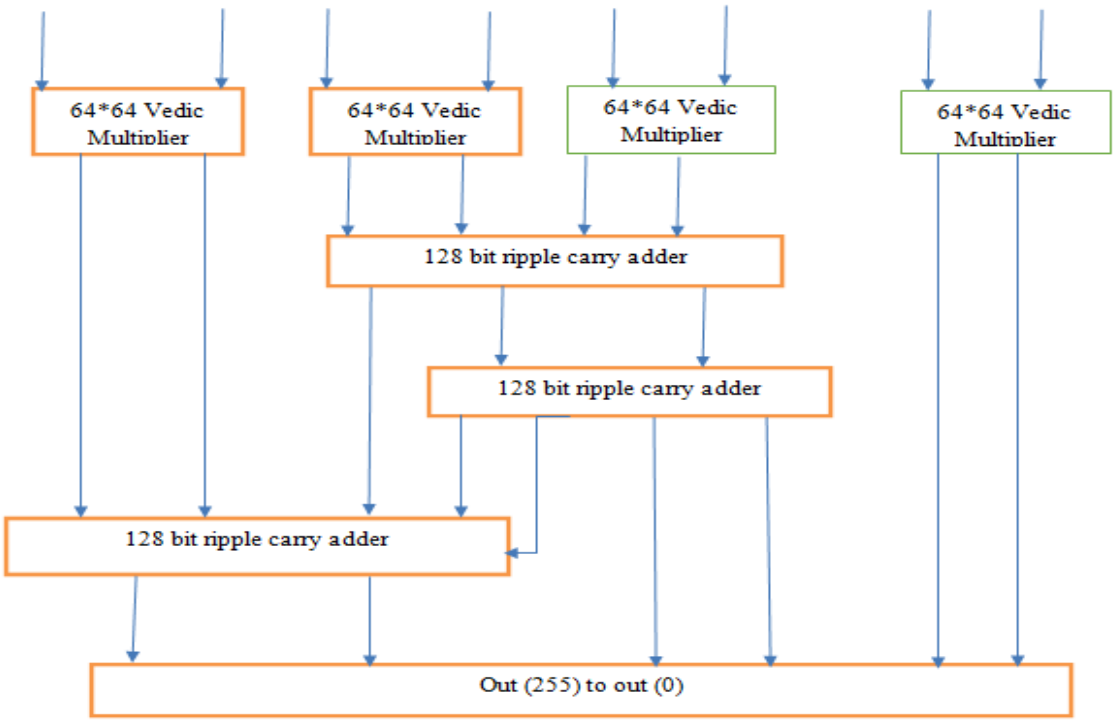

Figure 6. 128 point Vedic Multiplier

https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

Comparison between Array, Wallace and Baugh Wooley Multiplier:

Various parameters that are determined from the simulation results are convenient for deciding the efficiency of the multiplier such as power consumption, area, radix, Size of FFT, Configuration, Application, technology adopted, Frequency and Salient Features. Table represents the comparison of the above-said parameters for the three multipliers.

Table 1. Comparison between array, Wallace and Baugh Wooley Multiplier

References/ Parameters	**Array Multiplier**	**Wallace Multiplier**	**Baugh Wooley Multiplier**	**Vedic Multiplier**
Power Consumption(mW)	High Power Consumption 60.67	More Power Consumption 63.42	More Power Consumption 61.53	Less Power Consumption-57.41
Area (mm^2)	It contains more area because it utilizes a large number of adders-102	It embodies medium area because Wallace tree used to decrease the operands-69	It comprises less area because adder/ subtracter is nearly as small or fast as adder-72	It comprises less area because adder/ subtracter is nearly as small or fast as adder-29
Number of Slices Used	428	322	496	221
Time Delay	7.495	2.361	2.361	2.272
Operation Speed	Low speed	High speed	Very high speed	Very high speed

Scheduling Methodology - I:

The butterfly flow graph of 16-point radix-2^3 architecture is depicted in Figure 7. The nodes from *A*0,*A1* ...*A*7 correspond to foremost phase of the FFT and *B*0,*B1* ...*B*7 signify the second phase and*C*0,*C1*...*C*7 represent third stage of the FFT *D*0,*D1* ...*D*7 represent Last stage of the FFT. This radix-2^3 feed forward method involves the reduction of computation that produces 2 outputs. In this type of presumption is only valid for the RFFT case owing to the redundant functions.

https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

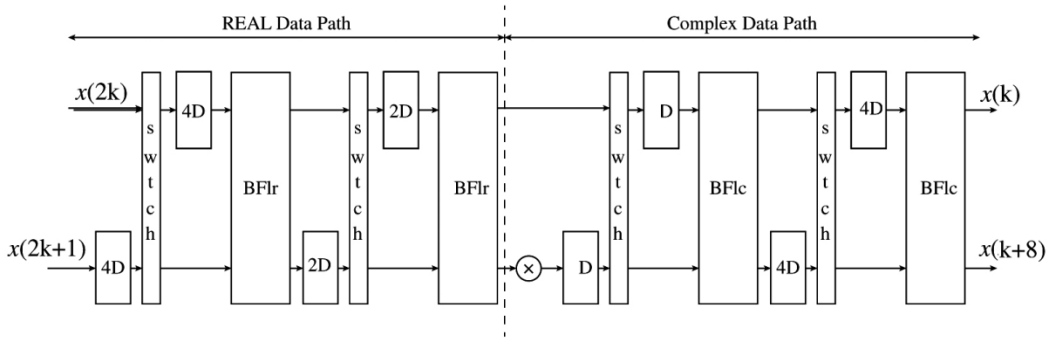

Figure 7. Data Path from real to complex

The architecture design of the flow graph uses four types of butterflies. BF2Ir and BF2IIr are real data path butterflies respectively as shown in Figure 8.

The complex multiplicative factor "-j" used in the flow graph, the real-valued data path are consists of first two stages only. The next stage of the architecture can be designed based on the real and complex values

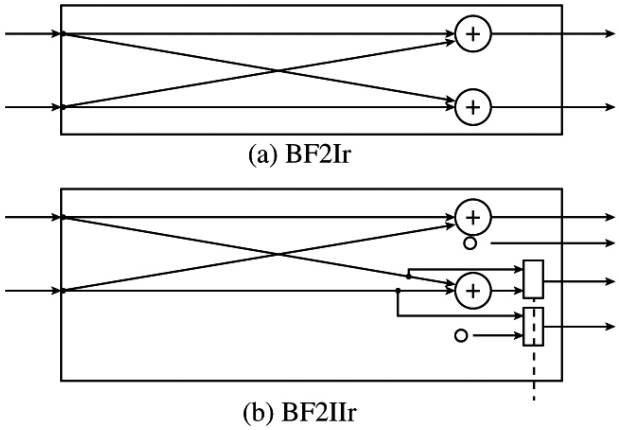

Figure 8. Real data path of BF2Ir and BF2IIr

SchedulingMethodology-II:

Next type of scheduling is proposed 2-parallel pipelined architecture. In this type of scheduling the input signal sequence is processed orderly (Figure 9).

 https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

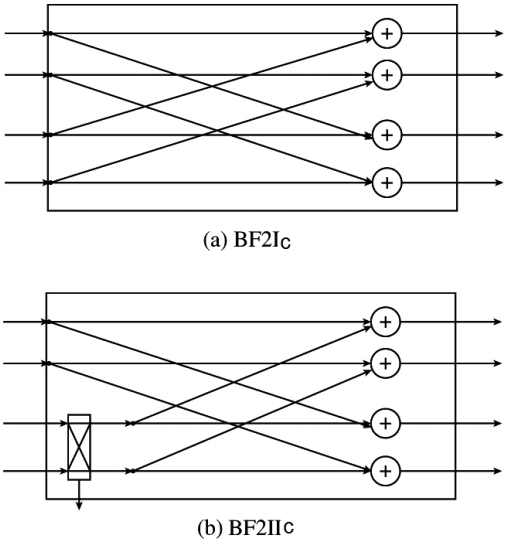

Figure 9. Complex data path of BF2IC and BF2IIc architecture

The final output can be obtained by using the minimum delay elements and also the various types of multipliers. In the final stage, we need to store final signal sequence output samples (Figure 10a,10b,10c,10d).

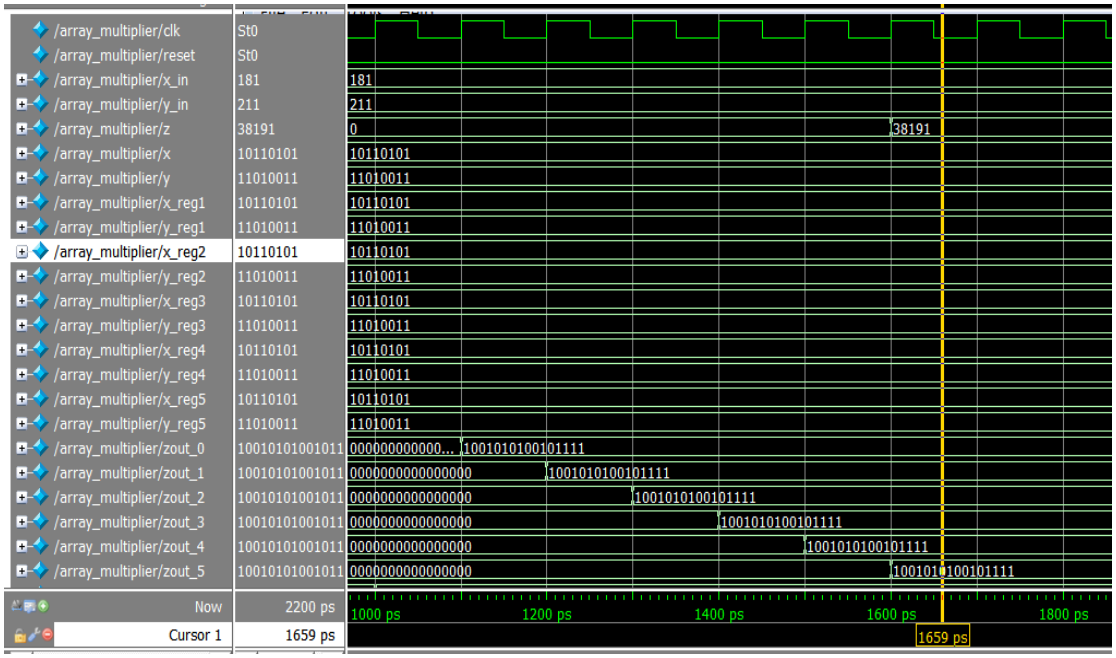

Figure 10a. 128-point Array multiplier radix-2^3 FFT scheduling Methodology output

https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

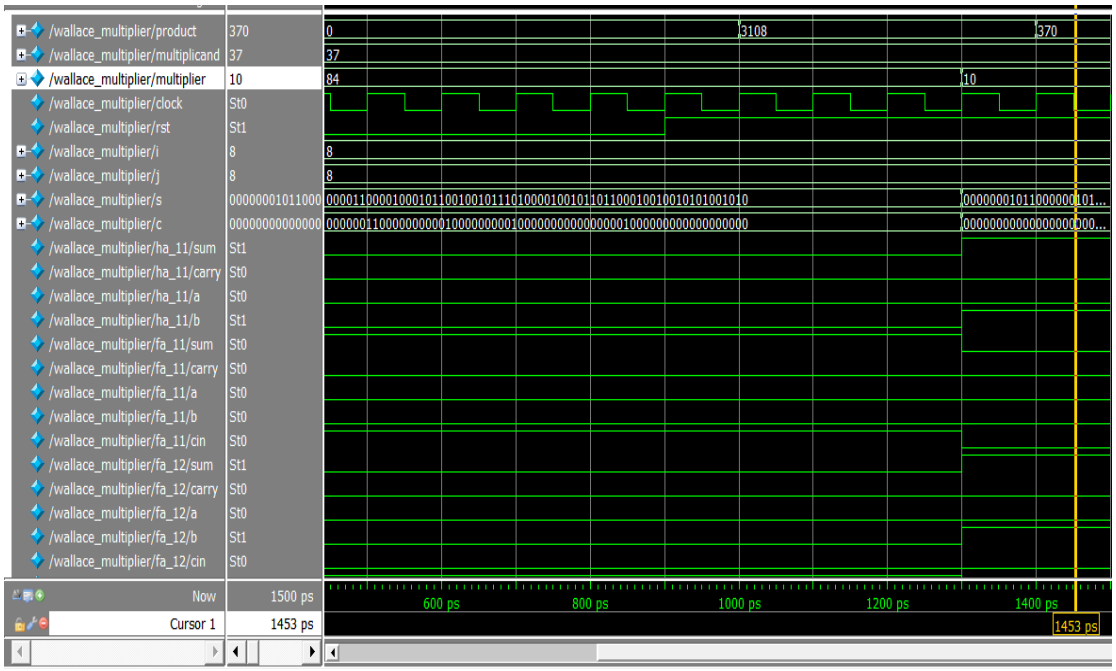

Figure 10b. 128-point Wallace multiplier radix-2^3 FFT scheduling Methodology output

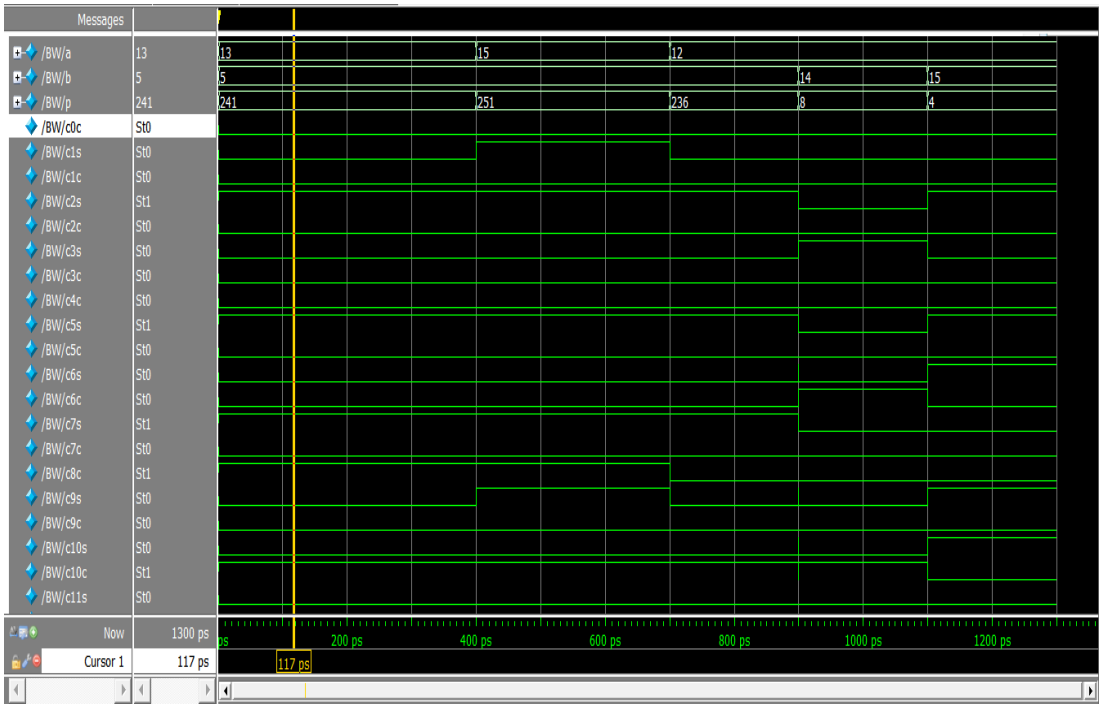

Figure 10c.128-point Baugh Wooley multiplier radix-2^3 FFT scheduling Methodology output.

https://doi.org/10.37896/pd91.4/91466

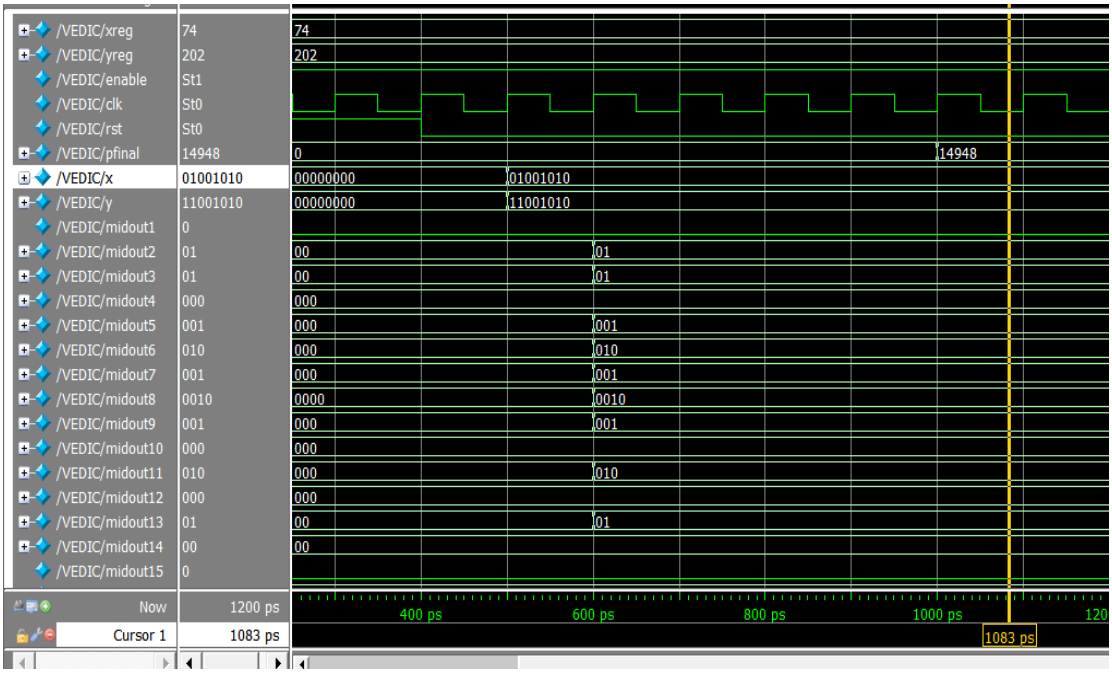

Figure 10d. 128-point Wallace multiplier radix-2^3 FFT scheduling Methodology output

Conclusion

The architecture of 2-parallel Radix2^3 DIF-RFFT & CFFT for two different methodology of scheduling approaches small complexity in control logic. By using various high speed multipliers in the DIF-FFT parallel pipelined architecture, the vedic multiplier computation produces less power consumption and also small delay. It's very high speed multiplier compare to other multipliers.70% of hardware utilization can be achieved. It's very useful method for signal computation process. Coming times work will be indented through to design of RFFT & CFFT architectures can be designed with booth multiplier, Wallace code multiplier instead of canonic signed digit multiplier (CSD).

References

[1] A. V. Oppenheim, R.W. Schafer, and J.R.Buck, Discrete-Time Signal Processing, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1998.

[2] P. Duhamel, "Implementation of split-radix FFT algorithms for complex, real, and real-symmetric data," IEEE Trans. Acoust., Speech, Signal Process., vol. 34, no. 2, pp. 285–295, Apr. 1986.

 https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

[3] S. He and M. Torkelson, “A new approach to pipeline FFT processor,” In Proc. of IPPS, 1996, pp. 766–770.

[4] L. R. Rabiner and B. Gold, “Theory and Application of Digital Signal Processing”. Englewood Cliffs, NJ: Prentice-Hall, 1975.

[5] E. H. Wold and A. M. Despain, “Pipeline and parallel-pipeline FFT processors for VLSI implementation,” IEEE Trans. Comput., vol.C-33, no. 5, pp. 414–426, May 1984.

[6] A. M. Despain, “Fourier transfom using CORDIC iterations,” IEEETrans. Comput., vol. C-233, no. 10, pp. 993–1001, Oct. 1974.

[7] E. E. Swartzlander, W. K. W. Young, and S. J. Joseph, “A radix-4delay commutator for fast Fourier transform processor implementation,”IEEE J. Solid-State Circuits, vol. SC-19, no. 5, pp. 702–709,Oct. 1984.

[8] E. E. Swartzlander, V. K. Jain, and H. Hikawa, “A radix-8 wafer scaleFFT processor,” J. VLSI Signal Process., vol. 4, no. 2/3, pp. 165–176,May 1992.

[9] G. Bi and E. V. Jones, “A pipelined FFT processor for word-sequentialdata,” IEEE Trans. Acoust., Speech, Signal Process., vol. 37, no. 12, pp.1982–1985, Dec. 1989.

[10]E. H. Wold and A. M. Despain, “Pipeline and parallel-pipeline FFTprocessors for VLSI implementation,” IEEE Trans. Comput., vol.C-33, no. 5, pp. 414–426, May 1984.

[11]Y. W. Lin, H. Y. Liu, and C. Y. Lee, “A 1-GS/s FFT/IFFT processorfor UWB applications,” IEEE J. Solid-State Circuits, vol. 40, no. 8, pp.1726–1735, Aug. 2005.

[12] J. Lee, H. Lee, S. I. Cho, and S. S. Choi, “A High-Speed two parallel radix- FFT/IFFT processor for MB-OFDM UWB systems,” in Proc.IEEE Int. Symp. Circuits Syst., 2006, pp. 4719–4722.

[13] J. Palmer and B. Nelson, “A parallel FFT architecture for FPGAs, “Lecture Notes Comput. Sci”, vol. 3203, pp. 948–953, 2004.

[14]D. Sinha and A. H. Tewfik, “TDCS, OFDM, and MC-CDMA: A brieftutorial,” IEEE Commun.Mag., vol. 43, no. 9, pp. S11–S16, Sep. 2005.

 https://doi.org/10.37896/pd91.4/91466

ISSN: 0369-8963

[15] M. H. Cheng, L. C. Chen, Y. C. Hung, and C. M. Yang, “A real-timemaximum likelihood heart-rate estimator for wearable textile sensors,” In Proc. IEEE 30th Annu. Int. Conf. EMBS, 2008, pp. 254–257.

[16] T. Netoff, Y. Park, and K. K. Parhi, “Seizure prediction using costsensitivesupport vector machine,” in Annu. Int. Conf. EMBS, 2009,pp. 3322–3325.

[17] Y. Wu, “New FFT structures based on the Bruun algorithm,” IEEETrans. Acoust., Speech Signal Process., vol. 38, no. 1, pp. 188–191,Jan. 1990.

[18]B. R. Sekhar and K. M. M. Prabhu, “Radix-2 decimation in frequencyalgorithm for the computation of the real-valued FFT”,IEEE Trans.Signal Process., vol. 47, no. 4, pp. 1181–1184, Apr. 1999.

[19] W.W. Smith and J. M. Smith, “Handbook of Real-Time Fast Fourier Transforms”. Piscataway, NJ: Wiley-IEEE Press, 1995.

[20] Ashish Raman, Anvesh Kumar, R.K.Sarin, “High Speed Reconfigurable FFT Design by Vedic Mathematics”, journal of Computer Science and Engineering, vol.1, pp 59-63 May 2010.

[21] Anvesh Kumar, Ashish Raman, “Small Area Reconfigurable FFT Design by Vedic Mathematics”, vol 5, IEEE pp 836-838, 2010.

[22] Laxman P. Thakre, Suresh Balpande,UmaehAkare, SudhairLande, “Performance evaluation and Synthesis of Multiplier Used in FFT Operation Using conventional and Vedic Algorithm”, International Conference on emerging trends in Engineering and Technology, pp 614-619, 2010.

[23] M.E.Paramasivam, R.S.Sabeenian, “An Efficient Bit Reduction Binary Multiplication Algorithm using Vedic Methods”, IEEE pp 25-28, 2010.

[24] Anvesh Kumar, Ashish Raman, “Low Power ALU Design by Ancient Mathematics”, vol 5, IEEE pp 862-865, 2010.

 https://doi.org/10.37896/pd91.4/91466