ISSN : 2347-2251

Indo-American Journal of Pharma and Bio Sciences

iajpb.editor@gmail.com
editor@iajpb.com
www.iajpb.com

Indo-American Journal of Pharma and Bio Sciences

Vol. 10, Issuse 1, Jan 2022

An Area Efficient High speed VLSI Architecture for Scalable In-Place Computation of Real valued FFT

SyedSadiqVali

1M.TechStudent,

Abstract: Real value is gained quickly. Physical signals may be simulated using Fourier transforms, which is attracting researchers' attention. In FFT, memory access conflicts and high throughput are among the most difficult problems to solve.' As a result, incorporating imaginary values is superfluous in all real-valued applications. Besides taking up extra space, it also slows down operations. Because the storage unit contains all of the internal processes, this is a good thing. For all the fictitious computations, it takes up a lot of memory space. The best potential method to eliminate memory conflicts and increase current performance is hence RFFT. With more stages and more internal computations, RFFT of Largerbitwidth requires more twiddle factor computations. Using a smaller bitwidth instead of a bigger bitwidth, we were able to improve area and delay while avoiding memory conflicts.

Indexterms: Fast Fourier transform computation in-place..

I.INTRODUCTION

Fast Fourier Transform (FFT) is a common method in digital signal processing. The scholars of today are interested in the computing real-valued FFT since all physical signals are essentially real signals. Real-world signals exhibit conjugate symmetry, which is of particular interest to researchers.

researcher, resulting in the creation of redundant positions. The arithmetic and memory difficulties can be lowered by using this conjugate symmetry characteristic.This includes applications such as spectrum estimate for speech/audio/image/digital communications and radar signal estimation.

processing, and the Fast Fourier Transform is one among the most often used methods (FFT). The data rate of wireless communication is expanding gradually due to the increasing demand for real-time and high data-rate multimedia services.In real-time applications like ECG and EEG, the power spectral density (PSD) of multiple real-time signals must be assessed. The RFFT is periodically measured on numerous overlapping signal windows, where a higher-speed clock can be used by a specific hardware implementation to meet real-time restrictions.

Department.ofECE, JNTU Anantapur, IndiaEmail Id: syedvali51@gmail.com

FFT, on the other hand, considers both real and imaginary values in the computation of memory access conflicts, whereas RFFT considers just real values. Since in RFFT only real values are considered for computation, most of the memory locations are free, allowing it to allocate memory access without conflicts.. Real-valued FFTs must be used in all these situations since all physical signals have real values. FFTs could be replaced with new ones in all of these cases. The problematic issue is avoided with RFFTs since there are no memory accessconflicts.

It's important to note that the higher-point FFTs are a bit more complicated. The number of stages in 8-point FFT and 32-point FFT, for example, is 3 in 8-point FFT and 5 in 32-point FFT. As a result, 32-point FFT computations take longer than 8-point FFT computations. The computation of the twiddle factor is also increased.. As a result, 32-point FFT consumes a significant amount of memory to store twiddle factors and perform computations at each of the five phases.One of the most challenging task inFFTistorealizelargesizeFFTinresourceconstrainedenvironment.

FFTS can be divided into two categories: pipelined FFT and in-place FFT. Classified FFTS typically have a memory unit, a butterfly unit, and a unit that generates random numbers called a twiddle factor generator.

It is unnecessary to undertake imaginary computations because most of the physical signals are genuine. It results in a loss of memory and a conflict in memory access. To overcome this, RFFT is the optimum solution for all physical signals, which not only avoids memory conflicts but also boosts space efficiency.

II.EXISTINGMETHOD

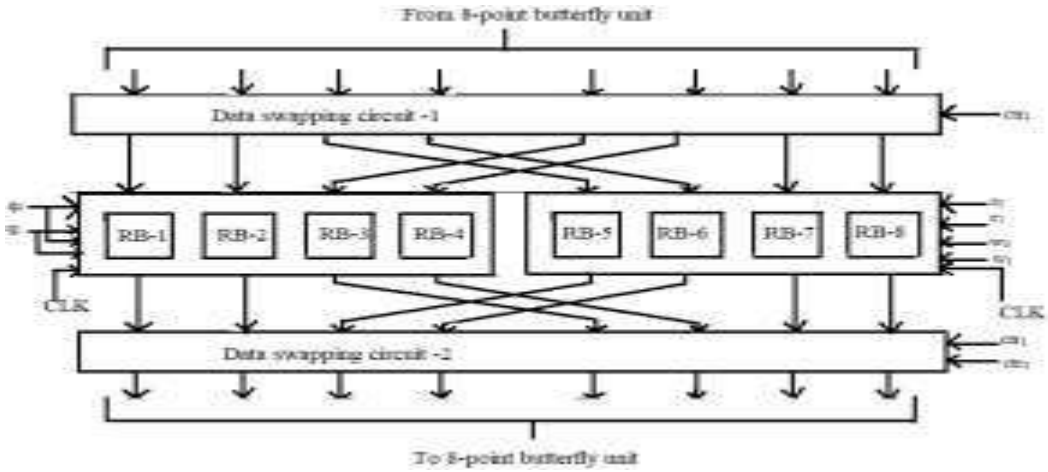


Fig.1. Storage unit design of existingRFFTstructure

Only one butterfly processing unit with an eight-point architecture will be required, along with a data storage unit. The inputs and intermediate outputs produced by different phases of computation are stored in the data storage unit. Each clock cycle, the data storage unit reads eight input samples/intermediate results, which are then written to the same storage locations eight times. Because of the increased amount of butterfly computations, this structure consumes more space and has a longer latency..

Hereintheexistingstructure,8pointbutterfly unit which requires 3stages. Herenotonlytheinternaldatacomputationsare

In 8-bit multipliers, twiddle factors generated are also more numerous than in 4-bit multipliers. There will be significant memory loss as a result of this. If the resources are constrained, memory conflicts may also be the result of this.

III.IMPLEMENTEDMETHOD

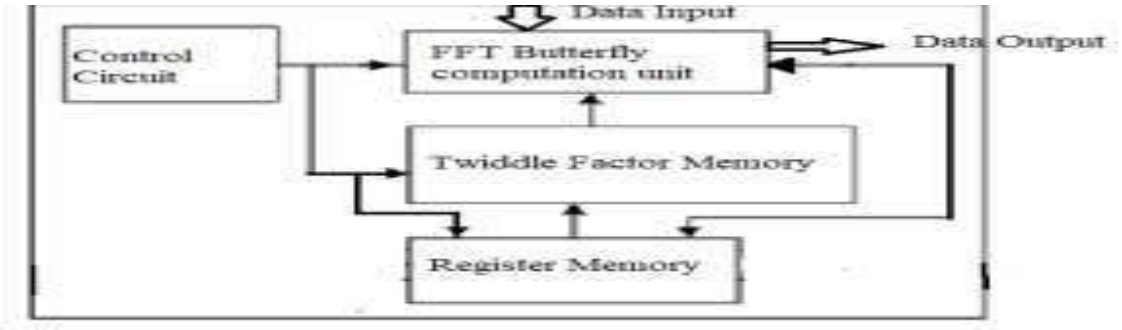


Fig.2.In-placecomputationofRFFTstructure

Above is a diagram illustrating the in-place RFFT structure. This structure includes an FFT butterfly processing unit, register memory, and twiddle factor memory and control circuit. The butterfly computations are carried out using the FFT butterfly unit. The created twiddle factors are stored in a special type of memory called a twiddle factor. The internal data is stored in the register memory. It is necessary to employ a control circuit in order to keep track of everything that is taking place.

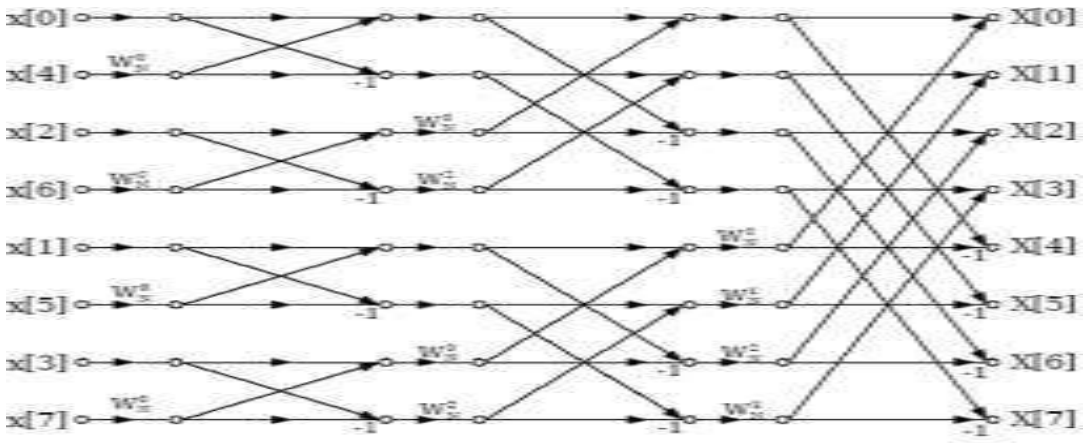


Fig.3.EightpointFFTbutterflystructure.

It takes three steps to perform an 8-point FFT. It is necessary to reduce an eight-point sequence to a two-point sequence in this case. The two-point DFT is computed for each two-point sequence. The 4-point DFT can be calculated from the 2-point DFT result. The 8-point DFT can be calculated from the 4-point DFT result.

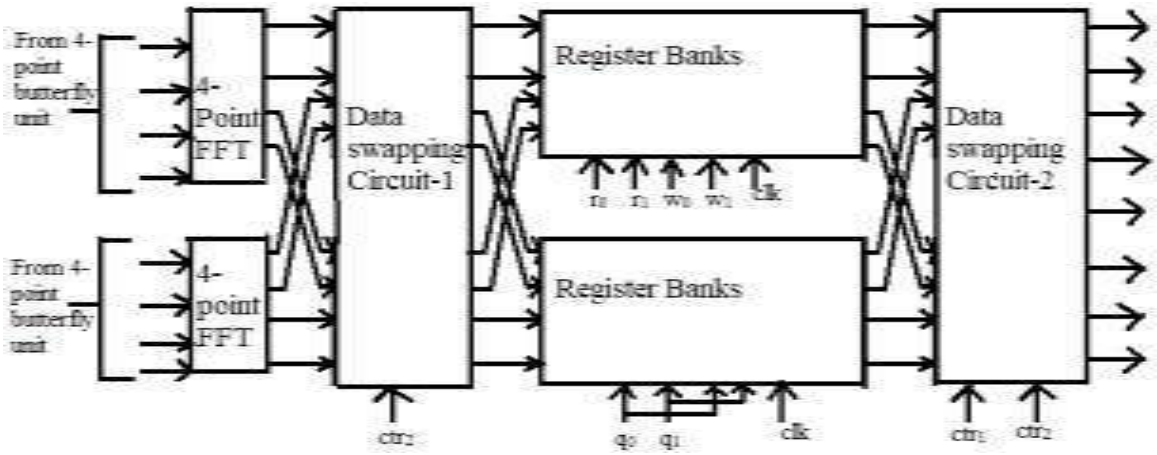


Fig 4. Implemented FFT structure for butterA flying unit and a storage unit

8 point FFT is used to build the circuit when complexity is increased in the present design.

There are two 4 point FFT multiple delay commutators and a novel butterfly unit proposed in the proposed approach. Area overhead, latency, and memory requirements will all be decreased as a result of this strategy. There are no changes to the data selectors or other storage components. With eight register banks and two data exchanging devices, it is referred to as a "multiplexer." Multiple memory banks are implemented in one unit by the register banks, which makes use of a single address decoding mechanism for all of the banks.

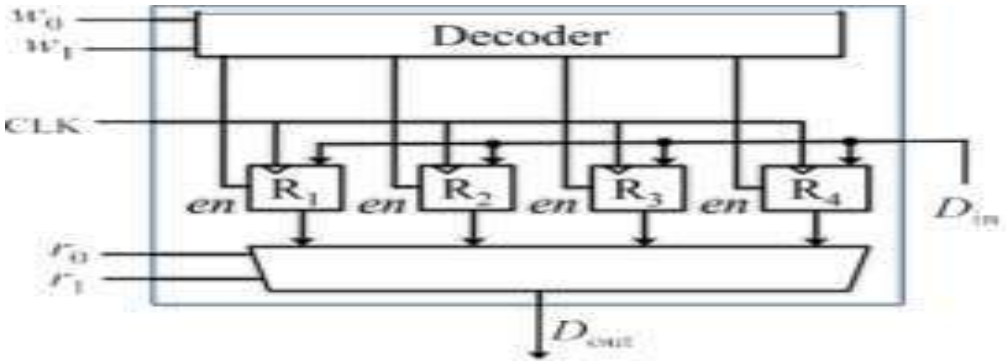


Fig.5.RegisterbankforN=32

These register banks are used to store thedata,intheformofregisters.ThedataTwodat a-swapping circuits, DS-1 and DS-2, that are utilized to exchange data. As a result of the control circuit's signal,operations. The ctr2 signal governs the DS-1 and DS-2 data swapping operations. Using a common address decoding circuitry, the multi-channel register bank supports many memory banks in a single device. It has four rows and four columns of 16 registers each. One bank channel is formed by each column's four registers.

A 2:4 To write one block of input data, a decoder is utilized to decode the two-bit write-address (w0,w1) and activate the clock signal CLK for a particular row of registers belonging to four separate banks (w1,w2).

In the suggested technique, we replace the 8-point FFT with two 4-point FFTs in order to minimize memory conflicts and produce an area and delay efficient RFFT. In 8point FFT, the number of twiddle factors generated is greater than in 4point FFT, if inspected attentively. Additionally, the internal data computations are greater in an 8-point FFT because of the additional twiddle factor calculations. When memory is freed, it can be used for other computations, avoiding conflicts with existing computations.

FFTcomputationsandtwiddlefactorsaremore.And the structureresults in the morearea and delaybecause the twiddle factorsarerequiredmorefor8pointFFTandrequires3stagestocomputethe8-pointFFT. This can be accomplished by splitting an 8-point FFT into two 4-point FFT multiple delay commutators in order to reduce the amount of twiddle factors and arithmetic operations. Area overhead, latency, and memory requirements will all be decreased as a result of this strategy.

the number of stages, and a bitwidth of 8 for the 32-point

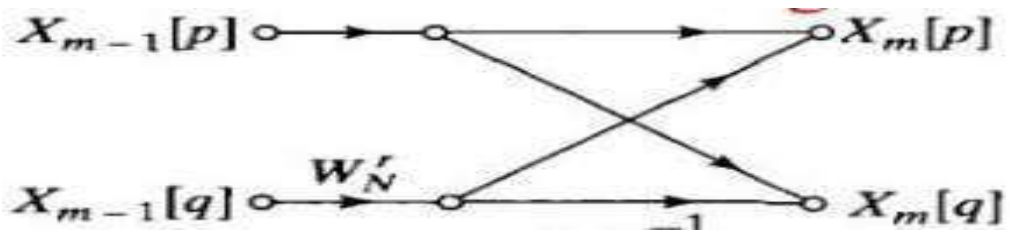
Stage1		Stage2		Stage3	
Inputs	Outputs	Inputs	Outputs	Inputs	Outputs
x(0),x(4)	x10r,x11r	x10r,x12r	x20r,x22r	x20r,x24r	y0r,y4r
x(2),x(6)	x12r,x13r	x11r,x13r	x21r,x23r	x21r,x25r	y1r,y5r
x(1),x(3)	x14r,x15r	x14r,x16r	x24r,x26r	x22r,x26r	y2r,y6r
x(5),x(7)	x16r,x17r	x15r,x17r	x25r,x27r	x23r,x27r	y3r,y7r

Table1:-3stagesinputsandoutputsfor8pointFFT

Stage1		Stage2	
Inputs	Outputs	Inputs	Outputs
x(0),x(2)	x10r,x11r	x10r,x12r	x20r,x21r
x(1),x(3)	x12r,x13r	x11r,x13r	x22r,x23r

Table 2:-2 stages inputs and outputs for 4 pointFFT.

In-PlaceComputationofFFT



The memory locations are represented by the two input nodes Xm-1 [p] and Xm-1 [q]. A new set of values is stored in place of input values following calculation of Xm[p] and Xm[q].

An algorithm that uses the,same location tostore both the input and output sequences iscalled„in-place"algorithm

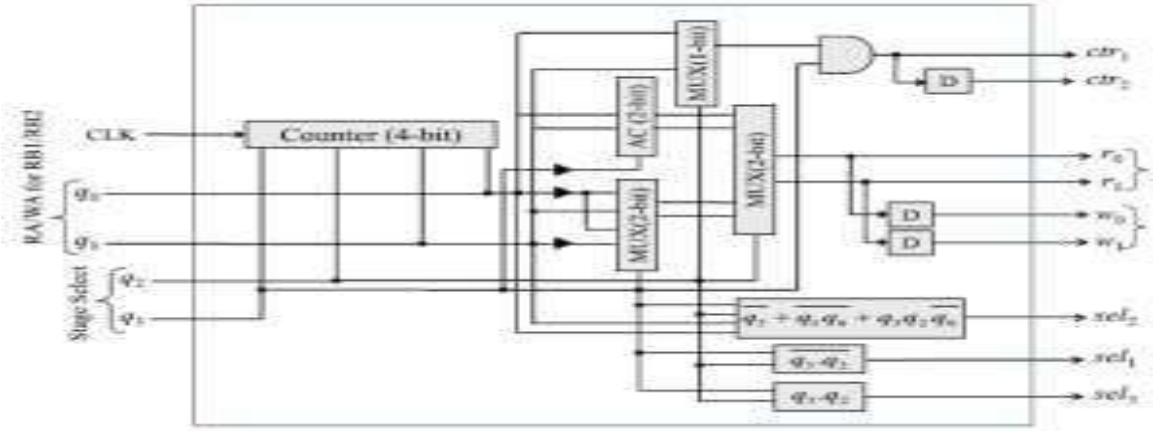
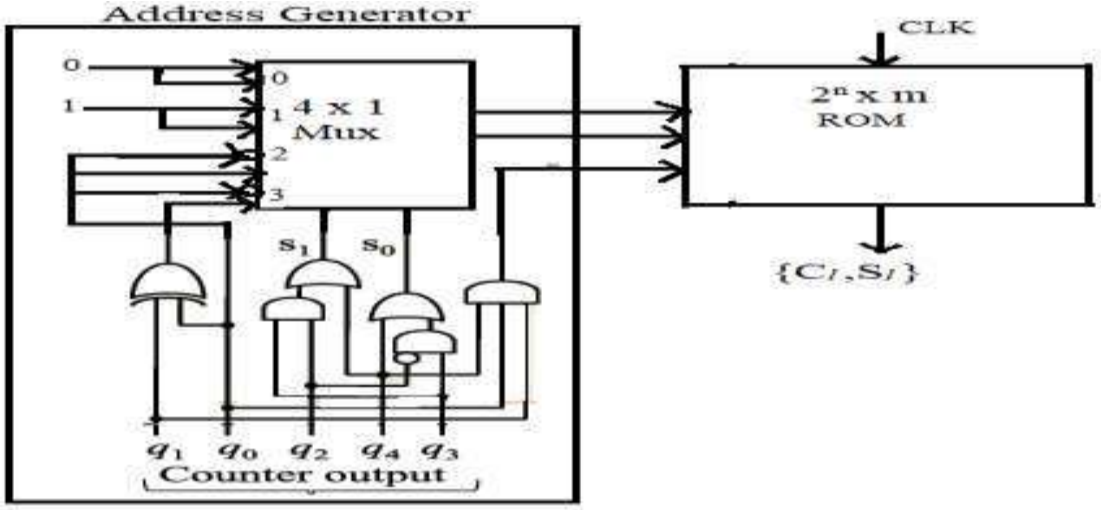


Fig6.Controlcircuit (CU)

The control circuitconsists of counter andmultiplexers andtheseoutputsare

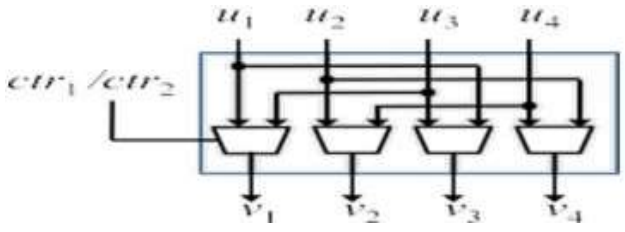


senttothedifferentblockslikebutterflyunitand

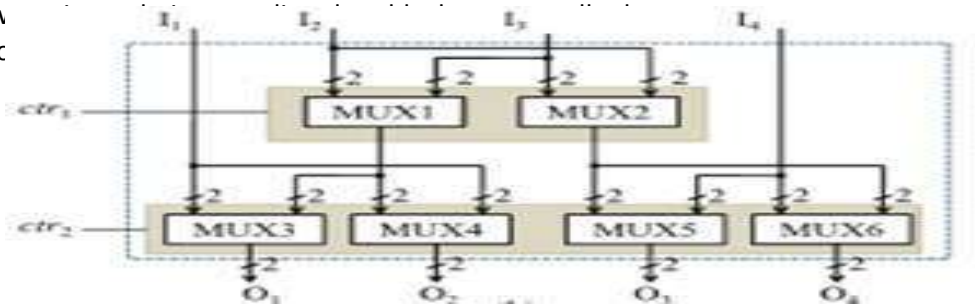
twiddlefactorunittocontroltheoperation.

Fig.7.Twiddle factor storage unit for FFTsizeN =32 .

The output of control circuit is given to



thetwiddlefactorstorageunitforaddressgenerationofparticularvaluetobemultiplied.Fig.8.(a)Dataswapping circuit-1(DS-1).(b) Dataswapping circuit-2(DS-2).Twodata-swappingcircuits(DS1andDS2)areusedbythestorageunittoconductthenecessarydata-sw BCU tc



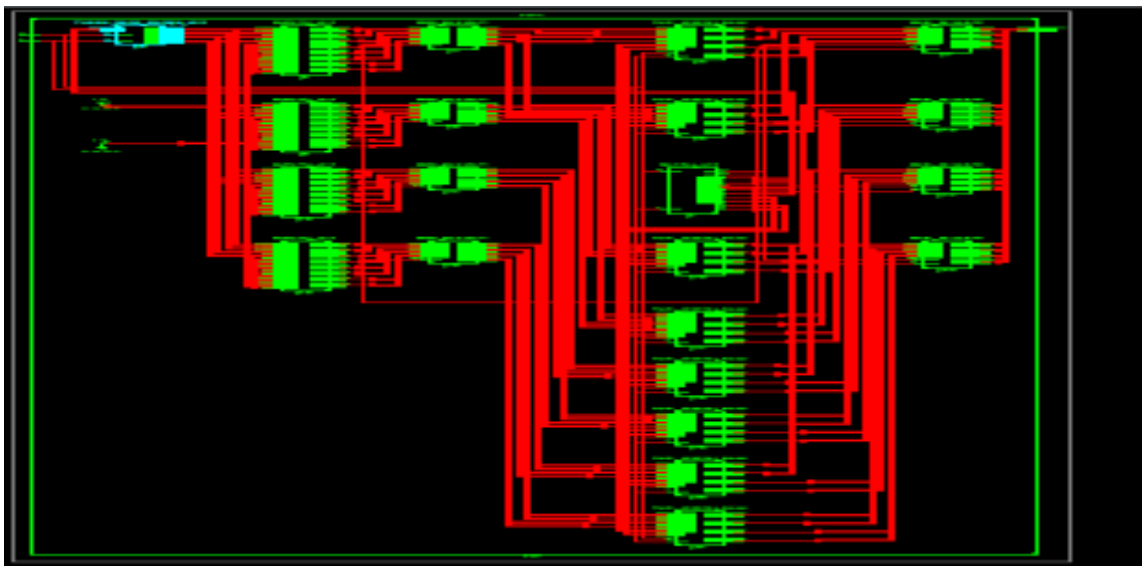
IV. RESULTSANDDISCUSSION

Due to the limited resources, an area and delay-efficient RFFT is proposed in this study, which separates an 8-point FFT into two 4-point FFTS in order to minimize memory conflicts. The schematic and simulation results are displayed below. Xilinx ISE 14.7 is used to simulate all designs.

Evaluation tableforArea,Delay:

	FFT size	(Lut,,s)	Delay(ns)
Implemented	32	4489	19.61

RTLSchematic:



SimulationResults:

V. CONCLUSION

It is proposed to use an area and delay efficient design strategy to construct an efficient architecture for in-place RFFT that can be scaled up to handle higher throughput and larger FFTs. For data storage and switching, an 8-point butterfly unit is proposed and used as an input to a storage device that has more complexity. Two 4 point FFTs are used to build a new 8-point FFT architecture in the suggested design. When compared to the current design, the proposed solution consumes less space and delays less. Xilinx ISE 14.7 was used to verify the synthesis and simulation results.

REFERENCES

[1]For VLSI applications, E. H. Wold et al. proposed pipeline and parallel-pipeline FFT processors for VLSI implementations in IEEE Transactions on Computers, vol. C-33, no. 5, pp. 414–426, May 1984.

Y.-N.Circuits Systems II: Analog Digital Signal Processing, vol. 50, number 6, 2003, pp 322–325, Chang and K. K. Parhi "An efficient pipelined FFT architecture."

[2]Real-valued rapid Fourier transform algorithms" by H. Sorensen, D. Jones, M. Heideman, and C. Burrus Vol. ASSP 35, no. 6, June 1987, pp 849–863. IEEE Trans Acoust Speech Signal Processing.

[3]The authors' paper, "Design of Low-Cost and High-Throughput Linear Arrays for DFT Computations: Algorithm, Architecture, and Implementation," appeared in IEEE Transactions on Circuits Systems II: Analog Digitized Signal Processing, vol 47, no 11, November 2000, pages 1188–1203.

[4]An FFT design for real-valued signals that is pipelined has been proposed by Garrido, Grajal, and Parhi in IEEE Transactions on Circuits and Systems I, Regulated Papers, which is published in December 2009.

[5]Ayinala, Brown, and Parhi, "Pipelined parallel FFT architectures via folding transformation," IEEE Trans. VLSI Syst., vol. 20, no. 6, pp. 1068–1081, June 2012. [6] Ayinala

[6]Energy-aware architectures for realvalued FFT implementation, A. Wang and A. P. Chandrakasan, Proc..Low Power Electron.Design, Aug. 2003, pp.360–365.

[7] H.-F.ChiandZ.-H.Lai,“Acost-effectivememory-basedreal-valuedFFTandHermitiansymmetricIFFTprocessorforDMT-basedwire-

linetransmissionsystems," in Proc. IEEE Int. Symp. CircuitsSyst., vol. 6, May2005,pp. 6006–6009.

[8] L. G. Johnson, "Conflict free memoryaddressingfordedicatedFFThardware, "IEEE Trans. Circuits Syst. II, Analog Digit.Signal Process., vol. 39, no. 5, pp. 312–316,May1992.

[9] M. Ayinala, Y. Lao, and K. K. Parhi,"Anin-placeFFTarchitectureforreal-valued signals," IEEE Trans. Circuits Syst.II, Exp. Briefs, vol. 60, no. 10, pp. 652–656,Oct.2013.