Journal of Theoretical and Applied Information Technology
30th April 2014. Vol. 62 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

LOW POWER FEED FORWARD FFT ARCHITECTURES USING SWITCH LOGIC

[1] DHANABAL R, [2] BHARATHI V, [3] SUJANA D.V., [4] SHRUTHI UDAYKUMAR, [5] JOHNY S RAJ, [6] ARAVIND KUMAR V.N

[#1] Assistant Professor (Senior Grade) ,VLSI division,SENSE, VIT University,
*2 Assistant Professor,GGR College of Engineering ,Vellore,
[#3, 4, 5,6] MTECH VLSI DESIGN Students, SENSE Department, VIT University, Vellore- 632014, TN, INDIA.

E-mail: [1]rdhanabal@vit.ac.in, [2] bharathiveerappan@yahoo.co.in, [3]doggalli5sujana@gmail.com, [4]uk.shruthi9020@gmail.com, [5]johny.rkz@gmail.com, [6] vnaravind33@gmail.com

ABSTRACT

The computation of FFT has become necessary in almost all DSP based applications and image processing applications. The radix-2^2 feed forward (MDC) FFT architecture originated for the systems which require high performance. The architecture uses both trivial and non trivial rotators for the computation of FFT. This paper proposes a switch logic in place of trivial rotators thereby reducing computational complexity. The compilation, elaboration and simulation of the design is done using NC launch tool. The design has been synthesized with 130 nm, 90nm, 45nm CMOS technologies using Cadence RTL compiler. The timing, power and area savings are of 26.2, 66, 23.4 percentage respectively for the proposed design. Therefore this design could be used for the systems which demand reduced area, low power and high performance.

Keywords: *Fast Fourier Transform (FFT), Switch logic, Cooley tukey algorithm, Trivial and Non Trivial Rotators, Data shufflers.*

1. INTRODUCTION

Power consumption and delay are two important considerations for VLSI system designer engineers. Our prime motive is to reduce the power and to get less delay that is nothing but the high speed for any design. Adder is one of the fundamental block present in arithmetic logic unit (ALU), floating point unit .In present arena we need fast arithmetic computation cells like adder and multipliers in the very large scale integration (VLSI) designs. The XOR/XNOR is the basic building block in many circuits like Arithmetic circuits.

Compressors, Comparators, Phase detectors, Code converters, Multipliers, Parity Checkers, Error-detecting and Error-correcting codes are some arithmetic circuits. Moreover, adders are very important components in some other applications such as microprocessor and digital signal processing (DSP) architectures. Digital signal processors and Microprocessors mainly rely on highly efficient implementations of generic floating point units and arithmetic logic units(ALU).

Full Adder is one of the core elements in many of the complex arithmetic logic circuits like multiplication, division, addition, exponentiation, etc. To perform an arithmetic operation, mainly even a small circuit can consume very low power at extremely low frequency and also it may even take a long time to complete that operation. There are some standard implementations.

Some different logic styles have been used in the past times for design of the full-adder cells and those techniques are used in this paper. Although they are used for producing similar function and the way of producing transistor count and intermediate nodes are varied. Different logic styles have different advantages such as the size, power dissipation, speed and the wiring complexity of the circuit. Different logic styles have different performance aspects. The propagation delay of a circuit is determined by the number of inversion levels, number of transistors in series, the transistor sizes or channel

Journal of Theoretical and Applied Information Technology
30th April 2014. Vol. 62 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

widths and the intra cell wiring capacitances. The size of the circuit depends on the number of transistors and their sizes and also on wiring complexity of the circuit.

Some may use only one logic style for whole full adder while others can use more than one logic style for their implementation.

I. Background

Power is the main criteria in all the electronic design equipments. So that's why the designers are trying to minimize the power consumption when designing the task. In

CMOS circuits mostly the energy consumed is because of switching activity. Thus the number of nodes in the circuit, energy per every node and also the total number of transaction operations per second, all these factors led to the power consumption. Power dissipation is based on the node capacitances of gate, threshold switching activity and circuit size.

There are four reasons for the power dissipation: dynamic power due to the charging and discharging of capacitance in the circuit because of switching transactions and leakage current is because of reverse bias condition in diode structures, sub threshold leakage, short-circuit current power due to rise and fall times, and static biasing power found in some logic styles (i.e pseudo-NMOS) These three are the major components of power dissipation in (CMOS) circuits:

1. *Dynamic Power:* Power consumed by the circuit node capacitance because of transistor switching.

2. *Short Circuit Power:* Power consumed because of current flowing from VDD to ground during switching of the transistor.

3. *Static Power:* Power because of leakage and static currents.

The increase in demand for low power and low voltage VLSI circuits can be investigated different levels of design, such as the architectural, circuit, layout and the process technology. At the device level, reduction in supply voltage and reduction in the threshold voltage to reduce the power consumption, where as in layout use of short channel transistors, poly and diffusion areas, shorter metal lines for connecting two various devices. It reduces capacitances in circuit and device level.

On an architectural level, CAD algorithms are required for fewer number of gates which reduces the overall power consumption. All these techniques employed most of the times reduce power increase of delay.

(A) Complementary CMOS Full Adder(C-CMOS):

The complementary CMOS full adder (C_CMOS) is one of the basic full adder circuit shown in below fig .1(a)

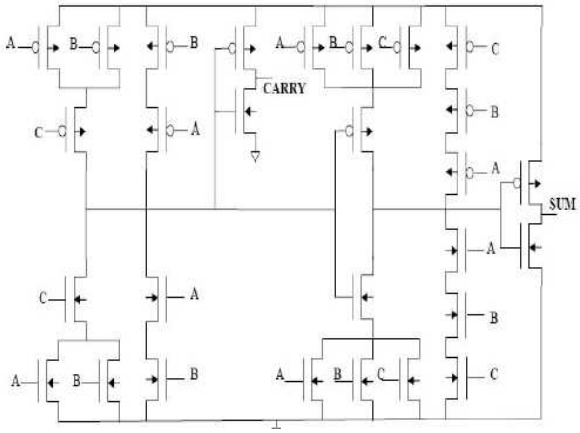


Fig.1(a) C-CMOS Full Adder Cell

The equation for C-CMOS is given as below

Sum =Carry'.(A+B+C) + (A.B.C)

Sum = ABC + AB'C' + A'BC' + A'B'C &

From the equation we have drawn the pull up and pull down networks and from that we have generated the sum and carry outputs. The advantages of the complementary CMOS logic circuit are stability and layout regularity at lower voltages due to smaller number of interconnecting wires and complementary pairs of transistors. The C_CMOS have some robustness against transistor sizing and voltage scaling which can have reliable operation at very low voltages with different transistor sizes.

The second adder is complementary pass transistor logic (CPL) uses 32 transistors with swing restoration. Most CPL gates can have an complexity in interconnection at the layout level with the increase in power and delay. For low power applications Pass Transistor Logic (PTL) is best suitable technique and explanation was given in .The advantage of Pass Transistor Logic (PTL) is that either PMOS or NMOS is enough to

Journal of Theoretical and Applied Information Technology
30th April 2014. Vol. 62 No.3

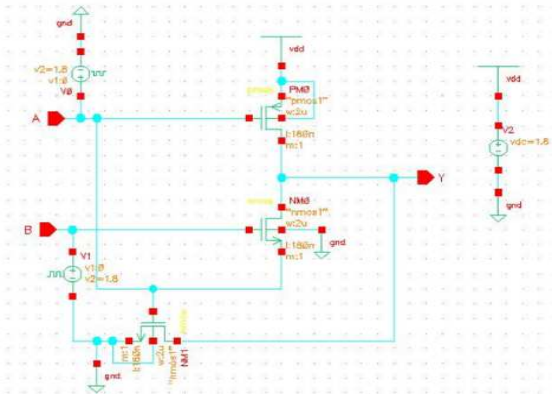
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

implement a complete design and so number of transistors gets decreased and also smaller input loads, especially for NMOS network and also by this PTL we can eliminate short circuit energy dissipation.

(B) 3T XOR:

The modified model of a CMOS inverter and a PMOS pass transistor. Whenever the input B is of logic 1 or logic 0, then the inverter functions like a normal CMOS inverter. And the output Y is the inverter of input A. Whenever the input B is at logic 0, the CMOS inverters output is at high impedance(z). However transistor N3 is in on condition and the output Y gets the same value as input A.

when A=1 and B=0, voltage reduction happens because of threshold drop occurs across the transistor N3 and also the output Y's performance gets deteriorated when compared to the input value. Degradation can be reduced by increasing the W/L ratio of the transistor N3.



Vt is the threshold voltage, γ is bulk threshold coefficient, Fermi potential, thickness of oxide layer are the process dependent materials.

Another problem is current back through transistor N1 occurs when A=1 and B=0. The output of the pass transistor is fed back through transistor N1 which it operates in the active region. This can overcome by reducing the W/L ratio of transistor N1.

(C) 8T ADDER USING XOR GATE:

The 8T adder is implemented using 3T Xor gates The sum output is basically obtained by a implementing exclusive OR of the three inputs in The final sum of the products is obtained using a OR logic of PTL type.

Thus two stages are required to obtain the sum value as output and at most in both the stage delays are to be added. The voltage drop due to the threshold in the transistors M3 and M6 can be reduced by increasing the aspect ratios of the transistors. The input combinations of all the three inputs have been checked by taking the rise and fall time into considerations. The 8-T XOR based Full Adder is shown in fig .2(b)

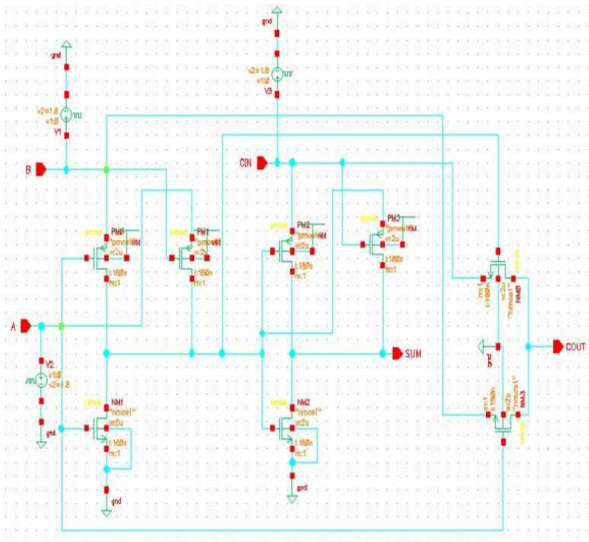


Fig.2(B) XOR Based Full Adder Cell

(D) 3T XNOR:

Similarly like XOR the XNOR also can be performed with the transistors to get better performance and optimized the working functionality is given When A=0 and B=0 transistor then p1 becomes on and N1,N2 becomes off so the output gets charged to VDD. A=0 and B=1 circuit shows logic 0 output, because transistor P1 is off and output node gets discharged by transistor N2. A=1 and B=0 then both transistors P1, N1 are on and output is discharged using N1 and N2 transistors. A=B=1, output gives high logic as N1 is on and thus the logic 1 is sent to the output. The 3-T XNOR based Full Adder is shown in below fig 2(c).

Journal of Theoretical and Applied Information Technology
30th April 2014. Vol. 62 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

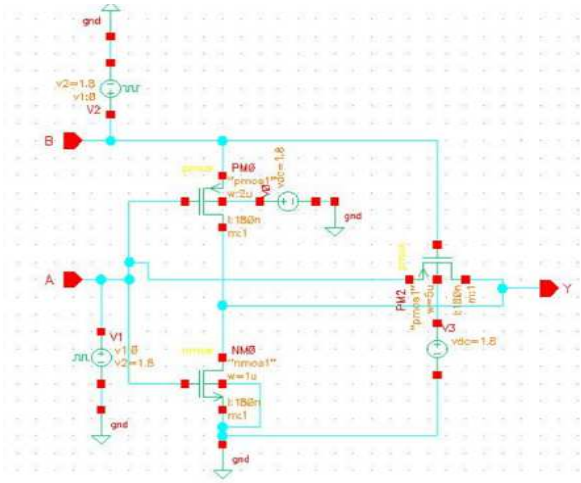
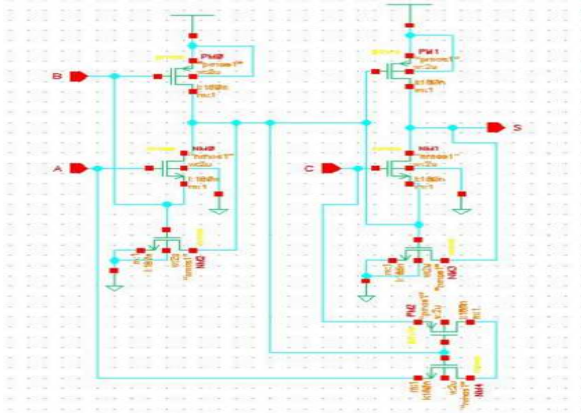


Fig.2(d) 3T XNOR Cell

(E) 8T XNOR:

It is implemented by two XNOR gates with one multiplexer block Sum is generated by using two XNOR gates, Carry out is generated by using two transistors multiplexer block. The XNOR gates with eight transistors has been implemented by using 3T XNOR gate, The simulation has performed form 3.0V to 1.2V to check the levels of output signal circuit in which it shows desired voltage levels. The 8T XNOR based Full Adder Cell was shown in below fig .2(d).



3. PROPOSED ARCHITECTURE

In this proposed adder we will use one XOR and one XNOR and a multiplexer to generate output sum and carry. The expressions for sum and carry are given in below .

$$\text{Sum} = H \text{ XOR } C = H.\ C' + H'.\ C$$

$$\text{Cout} = A.\ H' + C.\ H$$

$$\text{Sum} = (A \oplus B \oplus C) = C'\ (A \oplus B) + C\ (A \oplus B)'$$

Where H is (A XOR B) and H' is compliment of (AXNORB).

The proposed adder cell has 16 transistors and is mainly based up on low power XOR-XNOR pass transistor logic and transmission gates.

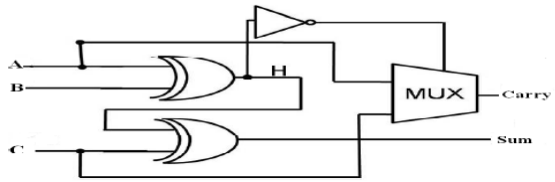


Fig.(a)

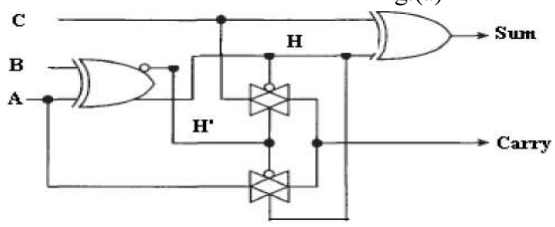


Fig.(b)

Fig.3 General Structure Of Proposed XOR-XNOR Adder

In this proposed adder the two transmission gates are used as multiplexer and the sum can be generated by XOR gates and output carry can be generated by XOR /XNOR gates shown in the above figure and output of XOR gate can be used as the selection line for multiplexer or as the control for the transmission gate which we will get output as the carry. The proposed adder circuit is designed by the combination of the two logic styles in order to get lower power consumption, high speed and good energy efficiency. We know that supply voltage variations will leads to the greater reduction in the power and also in the circuit delay.

IV. Experimental Results

All the circuits presented in this paper are designed by using Cadence VIRTUOSO environment by using CMOS process design kit. The range of supply voltage we have employed was

The proposed adder circuit is designed and simulated for different ranges of supply voltages

Journal of Theoretical and Applied Information Technology
30th April 2014. Vol. 62 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

is 1.3V-1.8V. The results of this proposed adder circuit can be compared with the different conventional adder circuit designs. Totally 16 transistors are needed to design the proposed adder circuit. By this we can clearly decide that the proposed circuit can have lower area overhead than the other conventional adder circuits. The delay values of conventional adder circuits and proposed adder circuit are compared and tabulated in below in table.1and table.2. and from the results it is clear that the proposed adder can have very less delay. The proposed circuit can have the lower power values and also lower PDP values as compared to other conventional adder circuits

Table 1: Simulation Results For Full Adder At 1.3V

Type	Power(µW)
C-CMOS	5.521
CPTL	3.52
XOR	3.265
XNOR	2.985
XOR-XNOR	1.295

Table 2: Simulation Results For Full Adder At 1.3V

Type	Delay(ps)	PDP	# of transistors
C-CMOS	1.65	9.10	28
CPTL	1.257	4.424	32
XOR	0.825	2.64	8
XNOR	0.785	2.345	8
XOR-XNOR	0.653	0.832	16

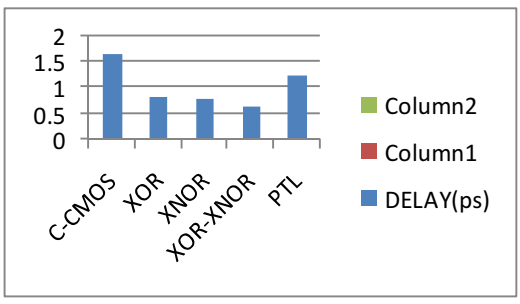


Fig.4 (A): Delay Comparison For Different Adders

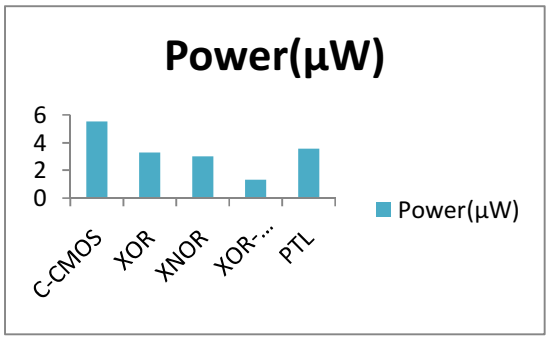


Fig 4(B): Power Comparison Results For Different Adders

V. Conclusion

This paper includes the Implementation of different adder logic styles includes Complimentary CMOS, XOR-XNOR, Complementary Pass Transistor Logic and Simulated by using Cadence Environment .The Comparison of power, delay, PDP are tabulated. From the table it is clear that Complimentary CMOS Adder design having more power consumption and delay so more PDP. This paper includes a new adder design of XOR-XNOR based adder cell with less power, delay, and power delay product (PDP). Cadence simulations results can show that the new adder design have very less PDP when compared with all other designs. The new full adder design can provide good voltage swing at low supply voltages and provides best performance with respect to all other conventional full adder designs in terms of both speed and power. The Fast Fourier transform (FFT) is one of the improvised techniques in digital signal processing to calculate the discrete Fourier transform (DFT) efficiently. The high-speed FFT's are required in orthogonal frequency division multiplexing (OFDM) [6], ultra wide band (UWB) [6] applications. To process multiple samples at the same time, the pipelined parallel feed forward FFT architectures are used .These architectures were proposed for radix-2, radix-4 until radix-2^2 came into existence. The basic element of the architectures is radix-2 butterfly. It is shown in Fig1. The cooley tukey algorithm [4] is used to decompose the N-points into N/2, N/4 and so on until a 2 point sequence is reached. This decomposition is represented in the form of signal flow graph. Then radix-2^k form of FFT originated, which improved the performance. This paper proposes radix-2^2 pipelined parallel feed forward FFT architecture [10][11][12] . The equation for computation of FFT is as follows

X[k] = , k=0,1,……,N-1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Where = and N is a power of 2.

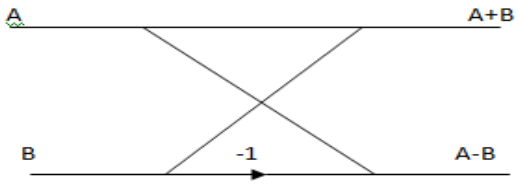


Fig 1: radix 2 butterfly

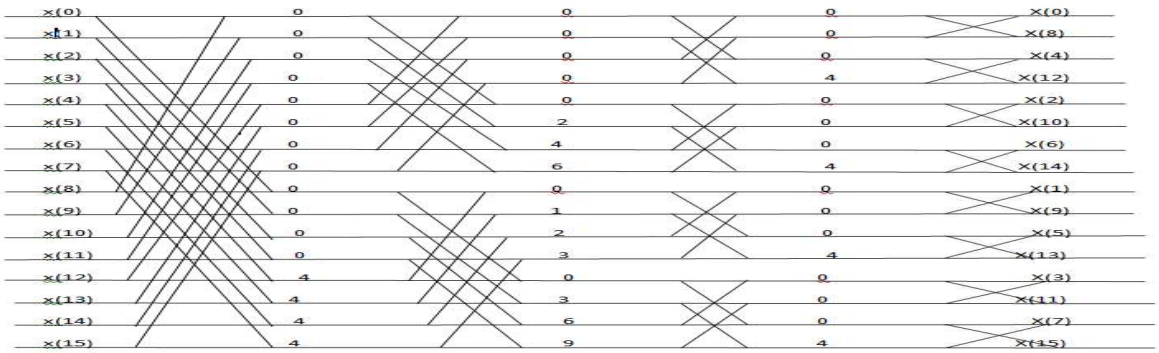


Fig 2: Butterfly Diagram For Radix-2^2

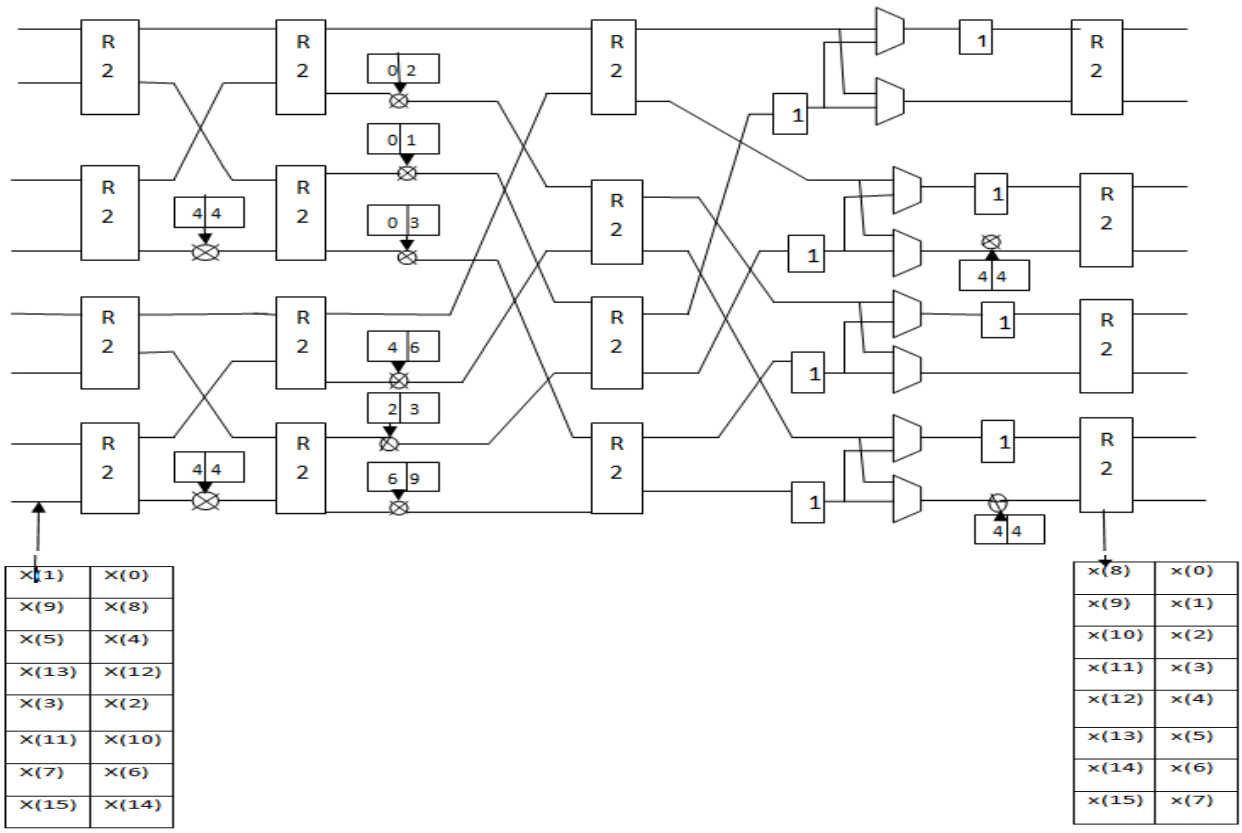


Fig 3: 8 Parallel -16 Point, Radix-2^2 Feed Forward FFT Architecture

Journal of Theoretical and Applied Information Technology
30th April 2014. Vol. 62 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

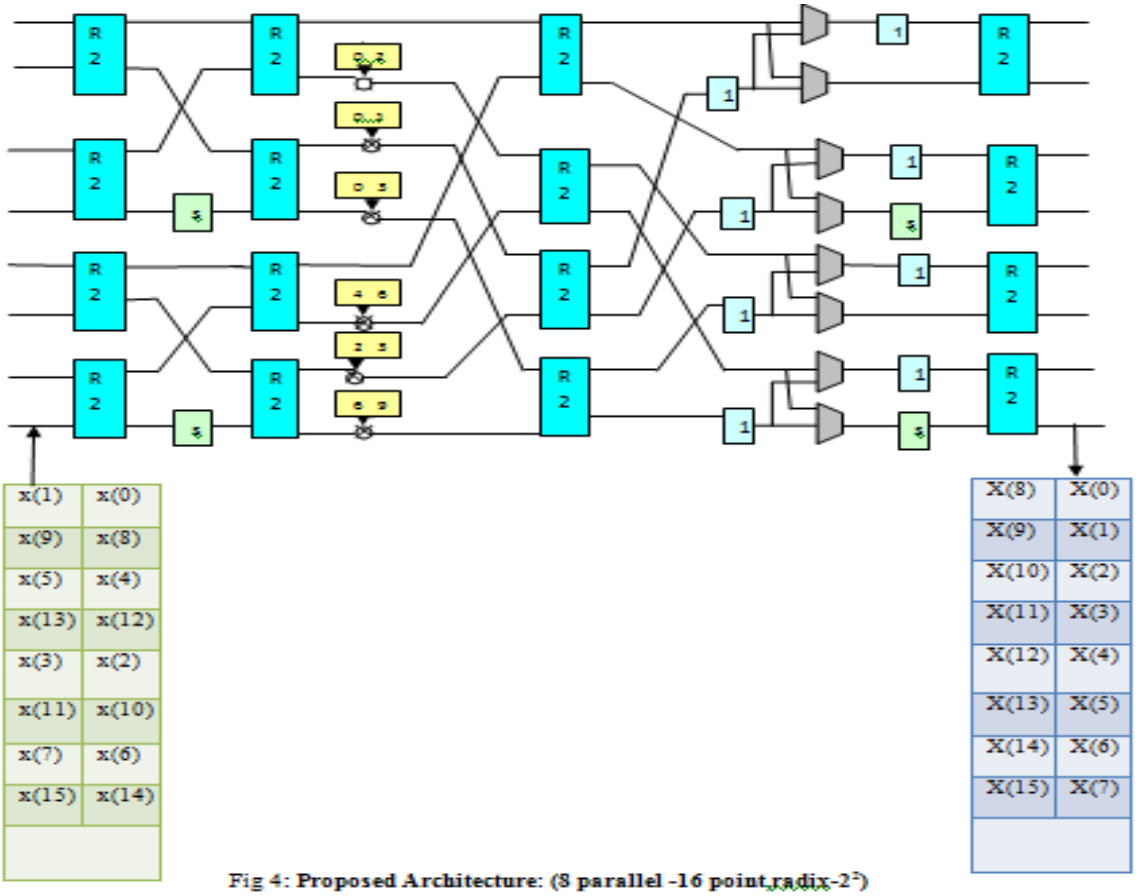


Fig 4: Proposed Architecture: (8 parallel -16 point radix-2^2)

4. PREVIOUS WORK

In [2] the computation of FFT was performed for real-valued signals .A folding transformation technique was used. Two different scheduling approaches have been proposed here which reduced the complexity and the delay elements. In [3], MDC, single delay feedback (SDF), single delay commutator (SDC) architectures of FFT were compared for radix-2, radix-4 and radix-2^2 FFT. The results have shown that radix-2^2 SDF architecture proved to have less number of multipliers and storage compared to others. In [1], multi delay commutator (MDC) architecture was proposed for the 2-parallel, 4-parallel and 8-parallel samples. The architectures were designed for radix-2^2, radix-2^3, radix-2^4 [5][9] using parallel processing and pipelining[7][8]. The comparison of these architectures was done and the results show that the radix-2^k reduces latency and improves throughput compared to its lower levels. The designs are efficient both in area and performance, and can obtain throughputs of the order Giga samples/sec .The design has low latency as well. Thus for much more efficient design, an architecture is proposed in the paper.

Proposed architecture

[1] Properties

The proposed architecture is based on analyzing the FFT flow graph (butterfly diagram) in Fig 2 and extracting the properties of the algorithm. In the flow graph, the input samples are decimated in frequency (DIF) to get the output samples. The number of stages depends on the number of input samples it processes at a time. The properties of radix 2^2 FFT algorithm [3] are shown in the Table: 1. The basic radix-2 butterflies in the flow graph whose indices differ in bit b_{n-s}, where subscript 'n' indicates number of stages of FFT operate in pair. The other two properties in the table are for the rotations at odd stage and even

Journal of Theoretical and Applied Information Technology
30th April 2014. Vol. 62 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

stage. At the odd stages, trivial rotators are used and at the even stages non-trivial rotators are used. The twiddle factor of FFT = where ϕ is the rotation which can be trivial or non trivial rotator. If ϕ belongs to ϕ mod N/4, it is said to be trivial rotation otherwise it is non-trivial. Fig 3 shows 16 point 8 parallel feed forward radix 2^2 FFT flow graph. In the first and third stages (odd) twiddle factors are calculated with rotation angle phi mod N/4. In the second stage, twiddle factors are computed according to radix-4 FFT [13][14] .

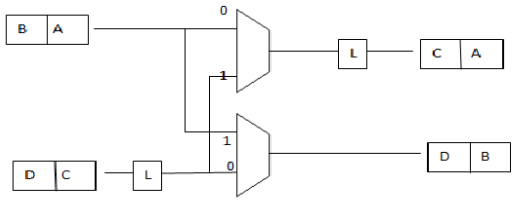
In the proposed architecture shown in Fig 4, 8-input samples are taken at a time and processed for 16 point FFT computation. The input samples flow from left to right. The indexed samples (0,8,4,12,2,10,6,14) arrive in parallel at the inputs of the circuit at the first clock cycle and at the next consecutive cycle (1,9,5,13,3,11,7,15) samples arrive in parallel. The architecture is made up of radix-2 butterflies, non trivial rotators, trivial rotators and data shuffling structures which are made up of multiplexers and buffers. Here basic radix-2 butterflies are used at different stages. Instead of trivial rotators, switch logic shown in Fig 5 is used. For the reordering of samples at the output, data shufflers are used at the essential stages. For this 8-parallel, 16 point DIF-FFT architecture, the data shufflers are used at the third stage. The data shuffler used here is made up of two muxes and buffers of required length. It is shown in Fig 6.

The properties of radix 2^2 FFT algorithm [3] are shown in the table:

Properties	DIF
Butterflies	b_{n-s}
Trivial rotations(odd stage)	$b_{n-s} . b_{n-s-1} = 1$
Non trivial rotations(even stage)	$b_{n-s+1} + b_{n-s} = 1$

-1 A-B

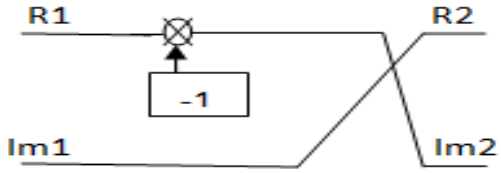
At the next stage, the complex multipliers are used where floating point multiplication is incorporated for decimal numbers for both trivial and non trivial multiplication. At the last stage, multiplexers and buffers to carry out data shuffling. Here buffer length used is 1,so one sample should be ready for one clock cycle. The basic data shuffler architecture is shown below:



Switch Logic:

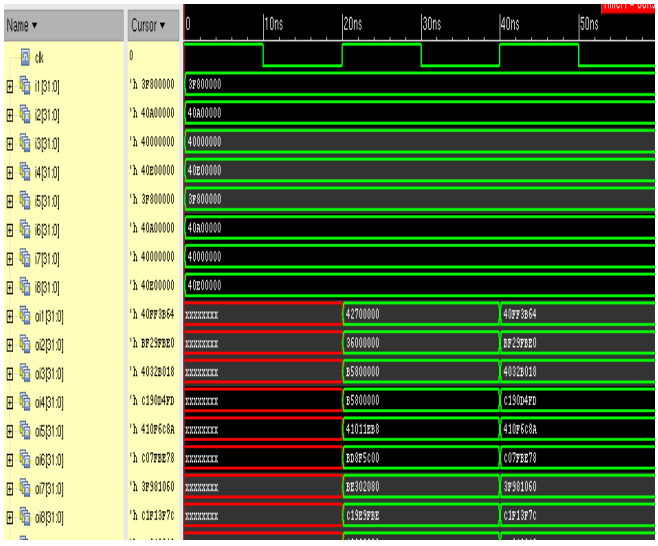
The input samples used are single precision floating point complex numbers. To perform any multiplication of these numbers, it requires much hardware and computational effort. As shown in fig 1 the trivial multiplication in each stage involves complex multiplication, which in turn involves two real multiplication and addition. So if any of these multiplications or additions are reduced, it can save much hardware and can speed up the system as well. For this reduction, a switch logic is proposed in this paper which uses only one real multiplier in it, instead of many complex multipliers and adders. So the trivial rotators used at different stages of the design are now replaced by the switch logic.

Fig 5: **Switch logic**



Simulation result:

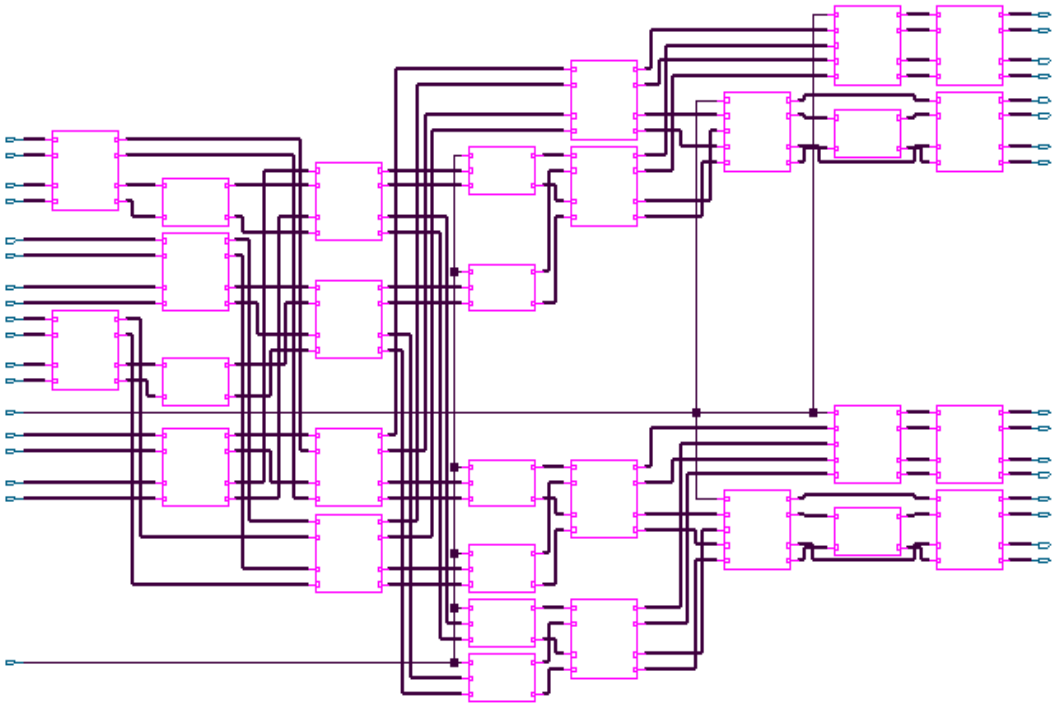
Proposed architecture:



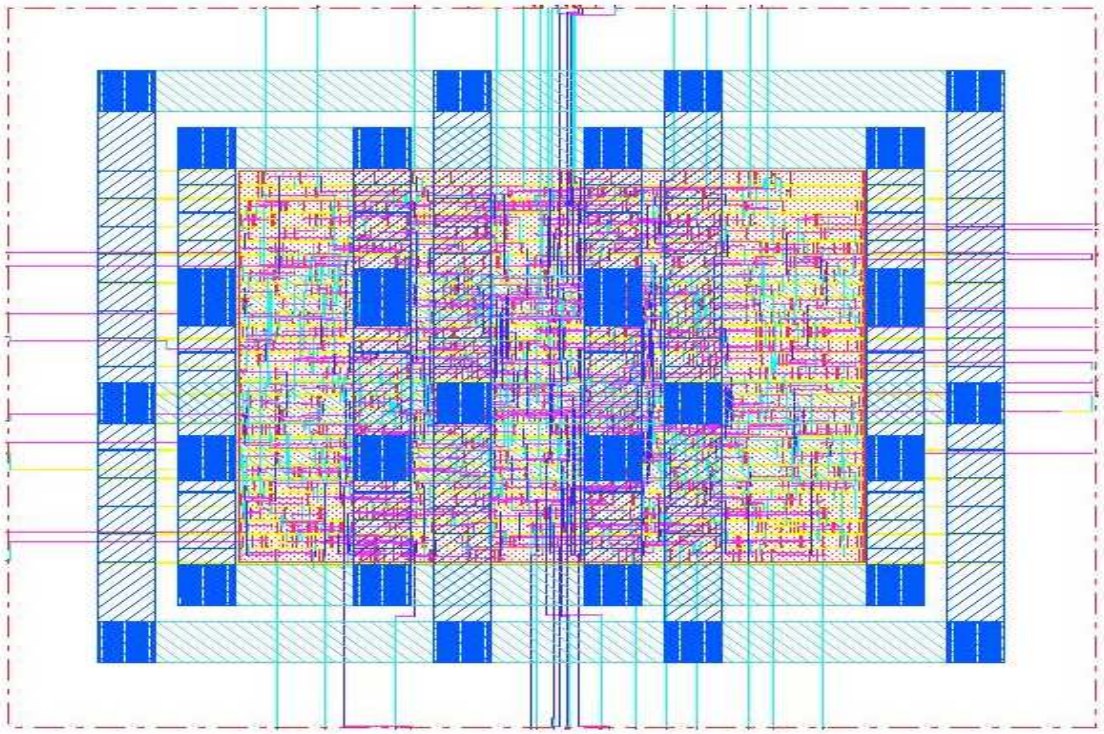
Journal of Theoretical and Applied Information Technology
30th April 2014. Vol. 62 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

RTL view proposed architecture:



Back end design view (45nm technology):



ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Comparison table:

The existing radix 2^2 architecture is compared with the proposed swap logic architecture in different technologies 45nm, 90nm, and 130nm. The following observation is made as shown below.

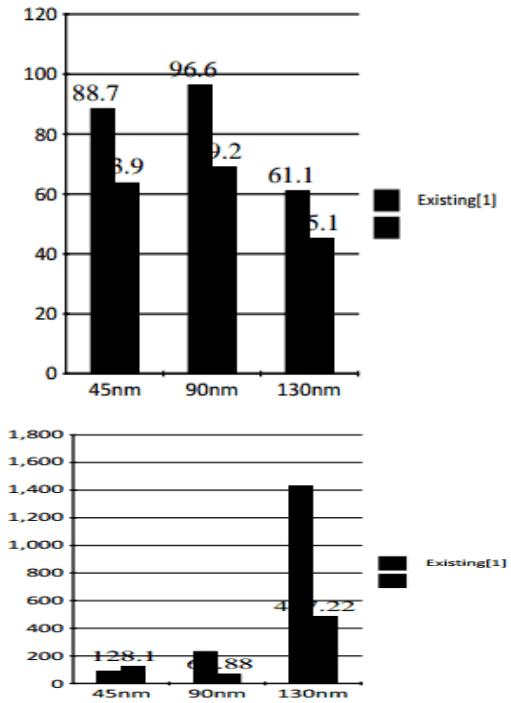
Technology	Area/power/timing	library	[1]	Proposed
45nm	Net area(micro square meter)	Slow	1923767.18	1475384
	Net power(nw)		89963277.75	128120990.2
	Timing(worst path delay in ps)		88670.1	63996
90nm	Net area(micro square meter)	slow	5498365	4216828.78
	Net power(nw)		234920683	69882686.95
	Timing(worst path delay in ps)		96559.201	69178.7
	Net area(micro square meter)	fast	5498365	4216878.78
	Net power(nw)		353176903.610	110357950.44
	Timing(worst path delay in ps)		26075.70	18645.10
	Net area(micro square meter)	typical	5498365	4216828.78
	Net power(nw)		263378554.963	63378554.963
	Timing(worst path delay in ps)		41767.60	31876.50
130nm	Net area(micro square meter)	fast	16799701	12867857.94
	Net power(nw)		1432990853	487276640.7
	Timing(worst path delay in ps)		61140	45013.6
	Net area(micro square meter)	typical	1654285.46	12671101.82
	Net power(nw)		478017856.97	690463824.27
	Timing(worst path delay in ps)		96143.80	70825.50

Journal of Theoretical and Applied Information Technology
30th April 2014. Vol. 62 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

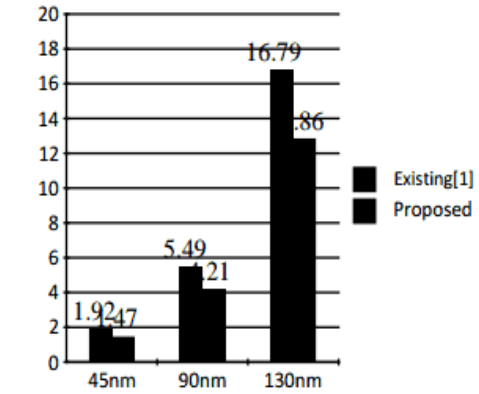
Bar chart

Timing in ns for different technology



Power in mw for different technology

Area in mm^2 for different technology



5. CONCLUSION:

In this paper we developed an architecture using switch logic for 8 parallel- 16 point radix- 2^2 FFT feed forward systems. The proposed architecture has achieved betterment in area, power and performance when compared to the previous works [9]. The timing, power and area savings are of 26.2, 66, 23.4 percentage respectively for 130nm (slow.lib) for the proposed design. Because of the above merits it can be concluded that the proposed architecture could be used for DSP/ image processing applications which require optimal hardware, good speed and low power.

REFRENCES:

[1] Garrison, M. and Raja, J. and Sanchez, M.A. and Gustafson, O., Pipelined Radix-2^k Feed forward FFT Architectures, Very Large Scale Integration (VLSI) Systems, IEEE Transactions on ,Vol.21,pp. 23-32(2013)

[2] Ayinala, M. and Parhi, K.K., Parallel - pipelined radix-2^2 FFT architecture for real valued signals, pp .1274-1278, 2010.

[3] Shousheng He and Torkelson, M., Parallel Processing Symposium, 1996., Proceedings of IPPS '96, The 10th International, A new approach to pipeline FFT processor, pp. 766-770,1996

[4] Parhi, K.K., VLSI digital signal processing education Signals, Systems and Computers, vol.2, pp. 1303-1308,1994.

[5] H. Liu and H. Lee,, A high performance four-parallel 128/64-point radix-2^4 FFT/IFFT processor for MIMO-OFDM systems, in Proc. IEEE Asia Pacific Conf. Circuits Syst., pp. 834–837, 2008.

[6] L. Liu, J. Ren, X. Wang, and F. Ye, Design of low-power, 1 GS/s throughput FFT processor for MIMO-OFDM,UWB communication system, in Proc. IEEE Int. Symp. Circuits Syst., pp 2594–2597, 2007.

[7] S. He and M. Torkelson,, "Design and implementation of a 1024-point pipeline FFT processor," in Proc. IEEE Custom Integer. Circuits Conf., 1998, pp. 131–134.

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

[9] J. Lee, H. Lee, S. I. Cho, and S.-S. Choi, "A high-speed, low-complexity radix-2^4 FFT processor for MB-OFDM UWB systems," in Proc. IEEE Int. Symp. Circuits Syst., 2006, pp. 210–213.

[10] L. Yang, K. Zhang, H. Liu, J. Huang, and S. Huang, "An efficient locally pipelined FFT processor," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 53, no. 7, pp. 585–589, Jul. 2006.

[11] J. A. Johnston, "Parallel pipeline fast Fourier transformer," IEE Proc. F Commun. Radar

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Signal Process., vol. 130, no. 6, pp. 564–572, Oct. 1983.

[12] N. Li and N. P. van der Meijs, "A radix based parallel pipeline FFT processor for MB-OFDM UWB system," in Proc. IEEE Int. SOC Conf., 2009, pp. 383–386.

[13] W. Xudong and L Yu, "Special-purpose computer for 64-point FFT based on FPGA" in Proc. Int. Conf. Wirel. Commun. Signal Process., 2009, pp. 1–3

[14] C. Cheng and K.K. Parhi, "High-throughput VLSI architecture for FFT computation," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 54, no. 10, pp. 863–867, Oct. 2007.