IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197 Volume 1, Issue 6 (Mar. – Apr. 2013), PP 68-75
www.iosrjournals.org

Low Power R4SDC Pipelined FFT Processor Architecture

Anjana R[1], Krunal Gandhi[2], Vaishali lad[3]
Assistant professor [1],Lecturer [2,3], Laxmi Institute of technology,Gujarat

***Abstract**: When the real-time signal processing is required pipelined FFT is the suitable option because of its high throughput and low power demands. A number of FFT architectures are there. Radix-4 single delay commutator (R4SDC) architecture is researched in this paper. R4SDC is the most popular pipeline FFT architectures, because of its efficient use of butterflies and multipliers.*

In this a low power technique for the pipeline FFT architecture is discussed. In this, Conventional R4SDC architecture, complex multiplier, and multiplier-less architecture based on common sub-expression technique are implemented and compared for 16, 64 and 256-point FFT architectures. A new type of multiplier algorithm called Multiplier-less architecture is implemented and compared with the carry save array, Wallace and Conventional complex multiplier (NBW).

I. Conventional R4SDC FFT Architecture

R4SDC was first proposed in [1], a brief introduction is given in Chapter-4. Each stage in R4SDC includes a complex multipliers and a full radix-4 butterfly. The R4SDC architecture can be directly interfaced to a sequential word input without the requirement for input buffers. [1]. A 16-point pipeline FFT processor is shown in Figure 1. Equation 1 defines computation for the first stage [*1*].

$$x_1(q_1,m_1) = W_N^{Q_1 m_1} \sum_{p=0}^{r_1-1} W_{N1}^{pm_1} x_1(N_1p+q_1) \qquad (1)$$

Equation 5.2 defines the computation for the final stage.

$$X(r_1r_2\ldots r_{v-1}m_v + \ldots\ldots + r_1m_2 + m_1) = \sum_{q_{v-1}=0}^{r_{v-1}} W_{rv}^{Qv-1}\ x_{v-1}(q_{v-1}, m_{v-1}) \qquad (2)$$

Equation 5.3 defines the computation for the intermediate stages: [*Ref.[20]*].

$$x_t(q_t,m_t) = W_{Nt-1}^{q_tm_t} \sum_{p=0}^{r_t-1} W_{rt}^{pm_t}\ x_{t-1}(N_tp+q_t m_{t-1}) \qquad (3)$$

Where for both the equation 5.2 and 5.3, , 2 □□*t* □□*v* □1, 0 □ m_t□ r_i*-1* , 0 □□ q_i □*N-1* and 2 □□*i* □□*v*

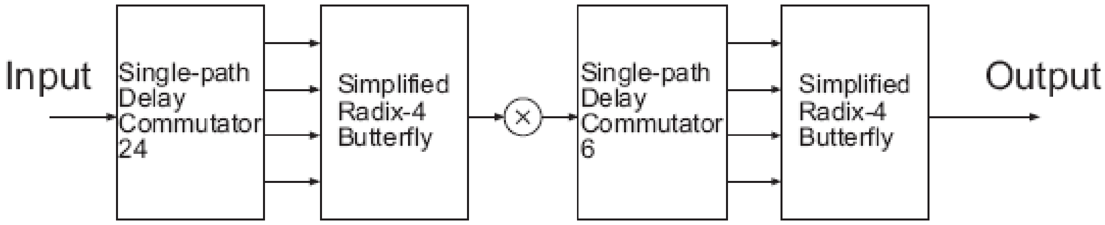

Input
Single-path Delay Commutator 24
Simplified Radix-4 Butterfly
×
Single-path Delay Commutator 6
Simplified Radix-4 Butterfly
Output

Figure 1: 16*-point pipelineR4SDC processor architecture.*

Butterfly:

The butterfly element performs the summation in Equation 1, 2 and 3. The summations can be replaced by six programmable adder/subtractors with the control circuits, as shown in Figure 2. Three complex adder/subtractors (each comprising of a real and an imaginary element) are used instead of eight complex adders. Control signal, stored in ROM unit selects the data fed into add/subs modules, according to the value of *mt.* This butterfly architecture generates *N* outputs consecutively in *N* word cycles, compared to the R4MDC butterfly which generates *N* outputs in *N/4* word cycles, with *N/3* word cycles idle. [1].

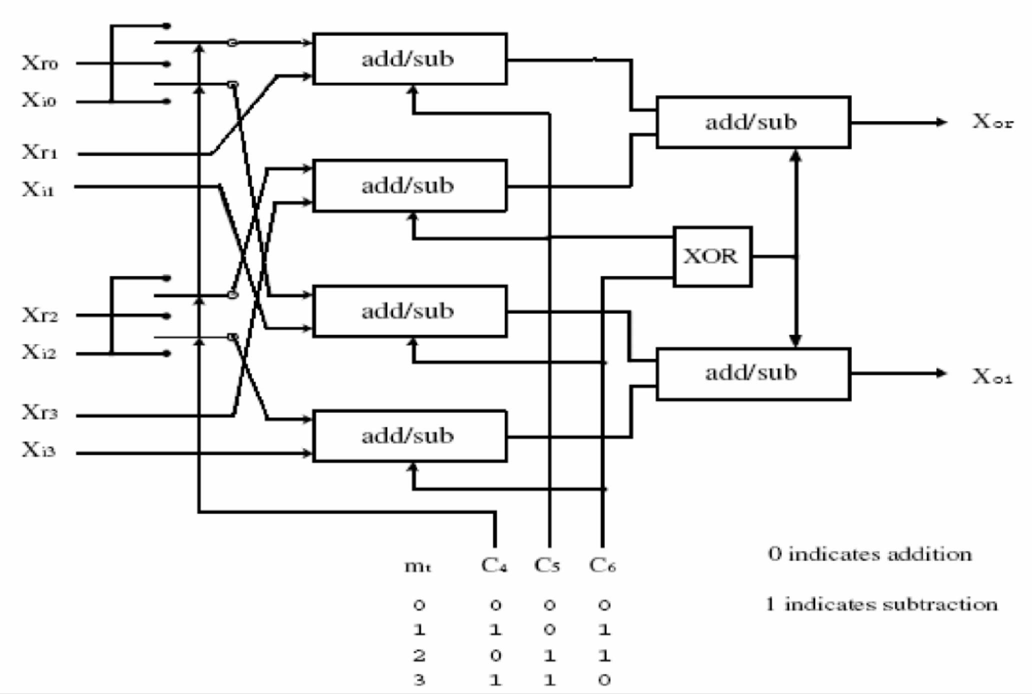


Figure 2: Conventional butterfly architecture for stage t in R4SDC pipelined FFT.
[1]

Conventional Complex Multiplier

In [Ref 8],a conventional complex multiplier accepts two complex inputs namely data (*Xr + jXi*) and coefficient (*Wr + jWi*) and produces a complex output (*XOr + jXOi*). It is constructed by using four real multipliers along an adder and a subtractor. The outputs and inputs are of the complex multiplier are related as:

$$XO_r=(X_rW_r - X_iW_i)$$
$$XO_i=(X_iW_r - X_rW_i)$$

The complex multiplier is shown in Figure 3. The product of the for real multipliers are truncated from 32 bits to 16 bits. The reduced precision achieves significant saving on hardware implementation, with acceptable error.

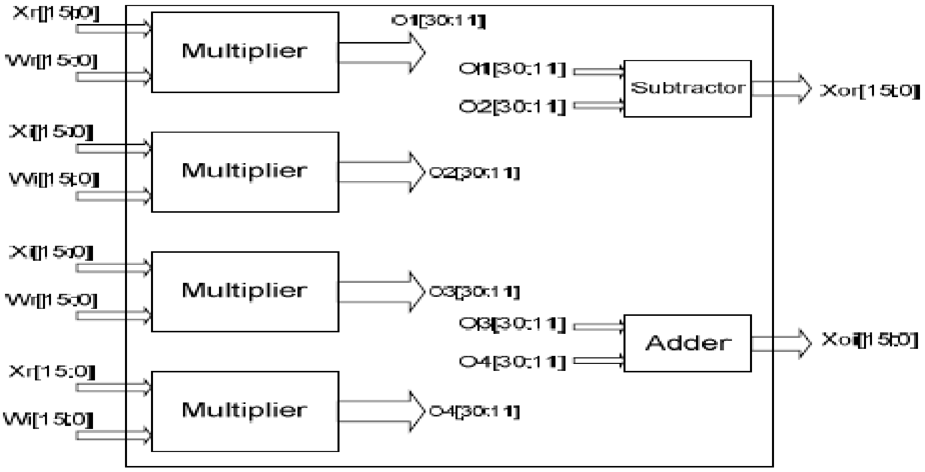


Figure 3: The block diagram of the Conventional Complex Multiplier.[2]

5.1.3 Commutator Architecture

In [9], the commutator architecture is conventional R4SDC FFTs is based on the Shift register architecture (SR) discussed in section 3.2, Chapter-3. Block diagram of the SR architecture is shown in Figure 4.

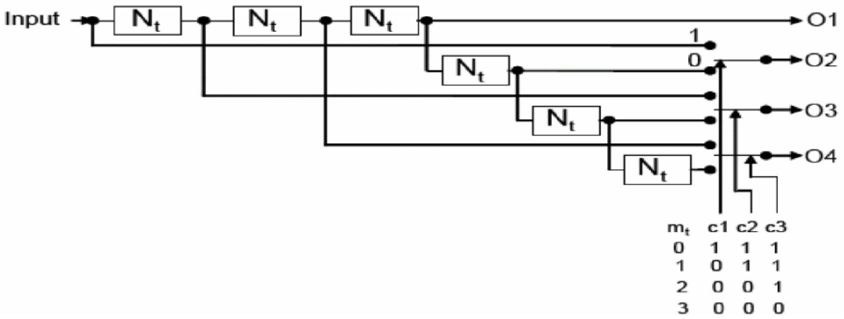


Figure 4: General commutator architecture for the radix-4 pipeline FFT processor.[9]

II. Methodology: Ordered R4SDC FFT Architecture

In this approach coefficient are reordered to save the power consumption by reducing the switching activity between the successive coefficients fed into the complex multiplier. Coefficients are ordered offline. Corresponding to the coefficient ordering, input is also ordered same as to make it Decimation in frequency algorithm and also to reduce the switching activity.

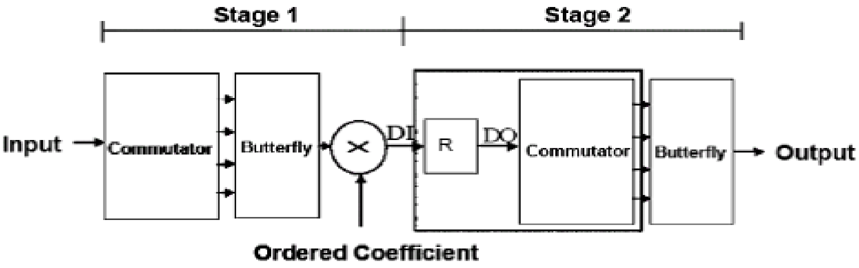


Figure5: 16-point ordered R4SDC pipelined FFT architecture [9]. R – RAM

The coefficients are reordered in order to minimize switching activity between successive coefficients by minimizing the hamming distance for each coefficient transition. The hamming distance is defined as the number of 1's of the XOR operation between two binary coefficients. Both original coefficient sequence and ordered coefficient sequence are encoded with the 16 bit fix point. The switching activity is accumulated by XOR the present coefficient by the previous coefficient sequence. To develop the minimum switching activity,we have developed the transition matrix of the hamming distance beween each coefficient as shown in table.

Our approach involves ordering the coefficient sequence so as to minimize swithing activity between successive coefficients fed to the multiplier for stage 1 of q 16-point FFT as listed in table 1.

Table 1
The transition matrix of switching activity
between each two coefficient with 16 word length

	W0	W1	W2	W3	W4	W6	W9
W0	0	15	17	19	3	21	13
W1	15	0	14	16	12	20	24
W2	17	14	0	14	14	16	14
W3	19	16	14	0	16	12	18
W4	3	12	14	16	0	20	16
W6	21	20	16	12	20	0	16
W9	13	24	14	18	16	16	0

From this transition matrix, we can arrange the twiddle factor in order to minimize the switching activity easily. The Coefficients are ordered so as to minimize switching activity between successive coefficients by minimizing the hamming distance between them.The ordered coefficient set is obtained by first arranging only imaginary part of the coefficient set on the basis of Hamming distance. It is followed by picking up the corresponding real part of the coefficient or its two's complement depending upon the hamming distance with respect to the previously arranged real part.

The design complexity of ordered FFT and the size of the additional RAM increases as the size of the additional RAM increase as the size of the FFTs increases. Hence the reordering technique is suitable for stage-1 of a 16-point radix-4 FFT processor due to the need of restoring data ordering for the following stage.

Complex Multiplication

First, we discuss the implementations of complex multiplications with real multiplication.The product of complex numbers,X=A+jB and Y=C+jD is (A+jB)(C+jD)=(AC-BD)+j(AD+BC). The direct computations of complex multiplications requires four real multiplications and two two additions and thus requires large chip area and power consumption.

Another method to compute a complex multiplications is to modify the original computation is to modify the original computation as follows.

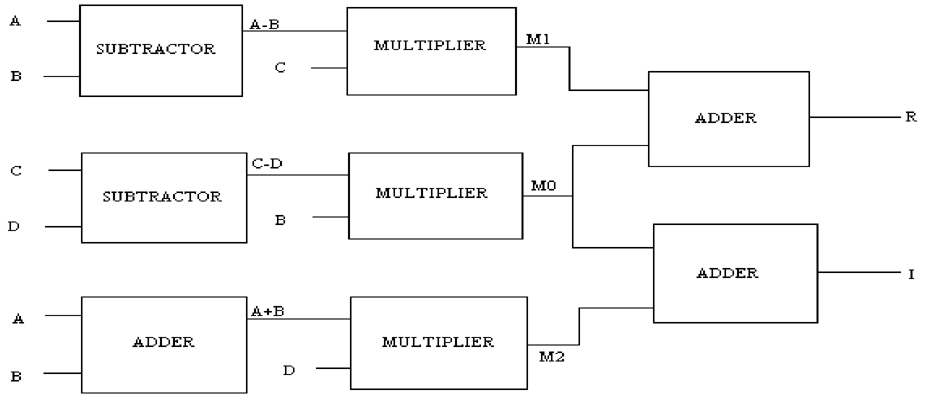


Figure 6 Multiplierless architecture for complex multiplier [9].

Butterfly Architecture: The most important element in FFT processor is a butterfly structure. It takes two signed fixed-point data from memory register and computes the FFT algorithm. The output results are written back in same memory location as the previous input stored. This method is called in-placement memory storage whereby it can reduce the hardware utilization. The butterfly architecture is shown in Fig. 6. The adder sums the input before being multiplied by the twiddle factor. The multiplier forms the partial product of the complex multiplication and produce two times bigger then input bit. Shift register would shift the bits to avoid overflow issue. Output of this butterfly would be kept in the register for the subsequent stage.

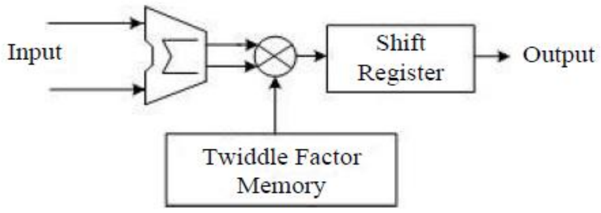


Figure 7 Butterfly architecture

III. Results:

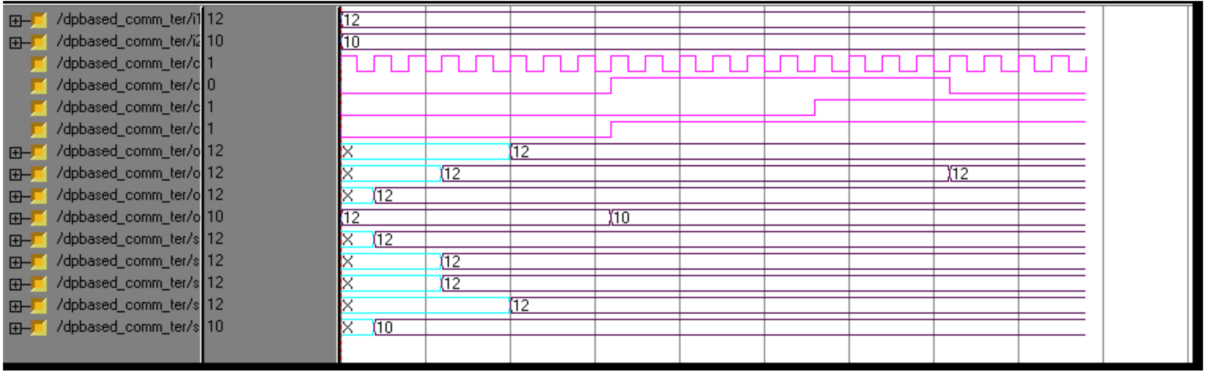
The results are compared with the different FFT architecture implementations In this, as per the project requirements, Conventional 16-point and Scheme I 16-point FFT architectural implementation are discussed with the area and power calculations. All the other proposed architectural implementations and results are discussed briefly.

The 16-point R4SDC is synthesised at 16ns clock cycle, using the Cadence RTL Compiler targeted at 0.18 □ □ CMOS technology library. Power evaluations were carried out, usingCadence RTL compiler, at 16ns clock cycle for 16-point FFT. Table 5 and 6 provide information about the main modules for each implementation.

IV. Simulation Results

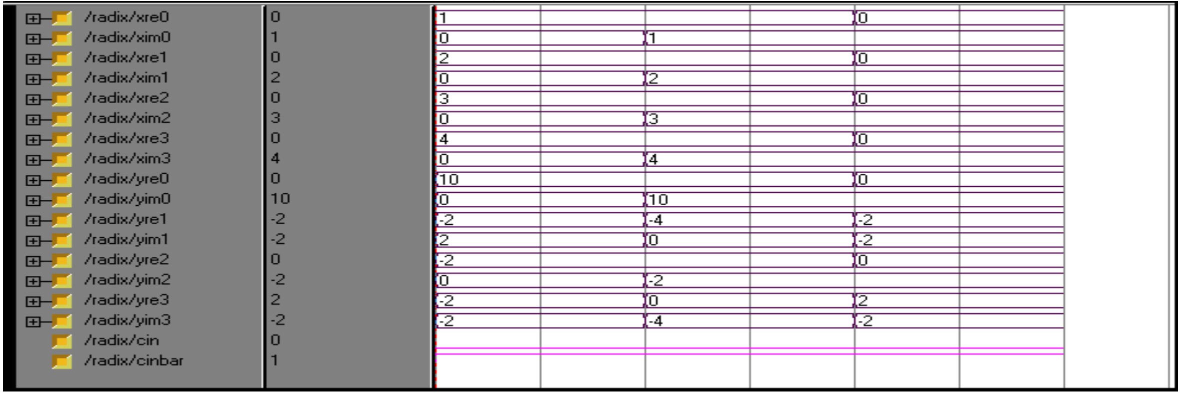
Analysis :
Commutator converts serial input to parallel output so that butterfly can receive these outputs at different clock with N_t delays.

Figure 8:Commutator



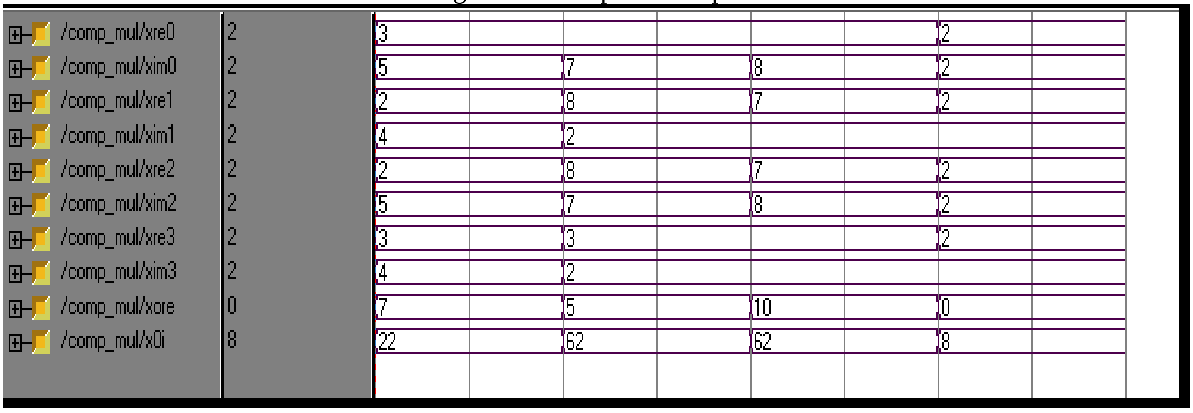
Analysis
The butterfly element is used to perform addition and subtraction. It accepts four input and produces four output.Here xre0,xre1,xre2,xre3,xim0,xim1,xim2,xim3 are the inputs and yre0,yre1,yre2,yre3,yim0,yim1,yim2,yim3 are the outputs.

Figure 9: Radix-4 FFT



Analysis
Complex multiplier multiplier accepts two complex inputs namely data (*Xr* + *jXi*) and coefficient (*Wr* + *jWi*) and produces a complex output (*XOr* + *jXOi*). It is constructed by using four real multipliers along an adder and a subtractor.

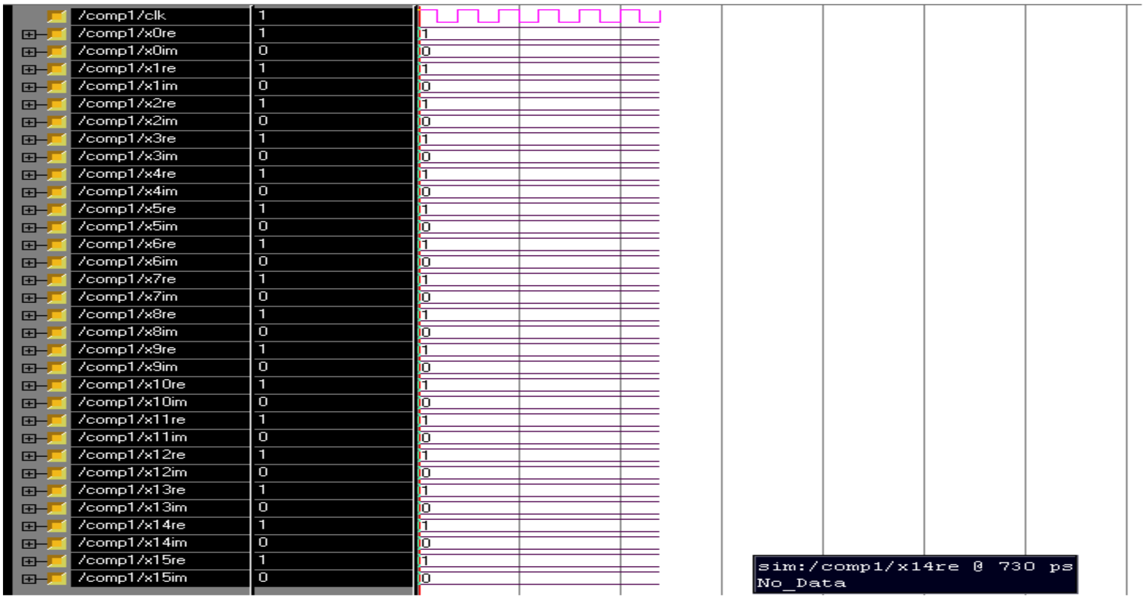
Figure 10: Complex Multiplier



Analysis

The pipelined FFT Processor accepts serial input and produces the output depending upon the applied clock. The input is 32 bit complex data and output is 32 bit complex data. For easy understanding all the inputs and the outputs are shown.

Figure 11: Inputs for pipelined FFT



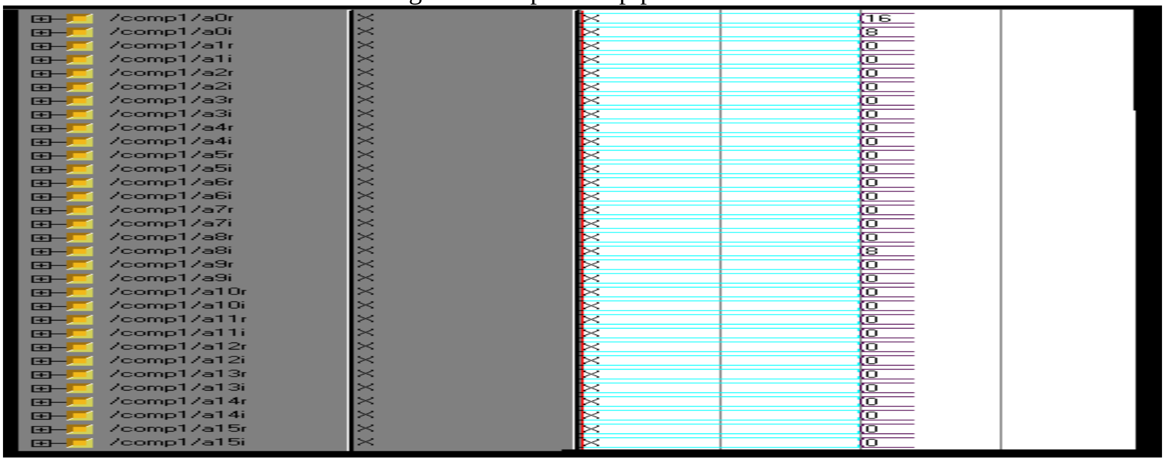
```
/comp1/clk     1
/comp1/x0re    1
/comp1/x0im    0
/comp1/x1re    1
/comp1/x1im    0
/comp1/x2re    1
/comp1/x2im    0
/comp1/x3re    1
/comp1/x3im    0
/comp1/x4re    1
/comp1/x4im    0
/comp1/x5re    1
/comp1/x5im    0
/comp1/x6re    1
/comp1/x6im    0
/comp1/x7re    1
/comp1/x7im    0
/comp1/x8re    1
/comp1/x8im    0
/comp1/x9re    1
/comp1/x9im    0
/comp1/x10re   1
/comp1/x10im   0
/comp1/x11re   1
/comp1/x11im   0
/comp1/x12re   1
/comp1/x12im   0
/comp1/x13re   1
/comp1/x13im   0
/comp1/x14re   1
/comp1/x14im   0
/comp1/x15re   1
/comp1/x15im   0
```

```
sim:/comp1/x14re @ 730 ps
No_Data
```

Analysis

The output is 32 bit complex data. Commutator accepts serial input and produces the parallel output with Nt delays. The size of commutator is 3N/2.so the output is delayed by 3N/2 bits.

Figure 12 outputs for pipelined FFT



```
/comp1/a0r    X   16
/comp1/a0i    X   8
/comp1/a1r    X   0
/comp1/a1i    X   0
/comp1/a2r    X   0
/comp1/a2i    X   0
/comp1/a3r    X   0
/comp1/a3i    X   0
/comp1/a4r    X   0
/comp1/a4i    X   0
/comp1/a5r    X   0
/comp1/a5i    X   0
/comp1/a6r    X   0
/comp1/a6i    X   0
/comp1/a7r    X   0
/comp1/a7i    X   0
/comp1/a8r    X   0
/comp1/a8i    X   8
/comp1/a9r    X   0
/comp1/a9i    X   0
/comp1/a10r   X   0
/comp1/a10i   X   0
/comp1/a11r   X   0
/comp1/a11i   X   0
/comp1/a12r   X   0
/comp1/a12i   X   0
/comp1/a13r   X   0
/comp1/a13i   X   0
/comp1/a14r   X   0
/comp1/a14i   X   0
/comp1/a15r   X   0
/comp1/a15i   X   0
```

The graphical power and area comparison between the all the 5 architectures is shown in Figure 10 and 11.

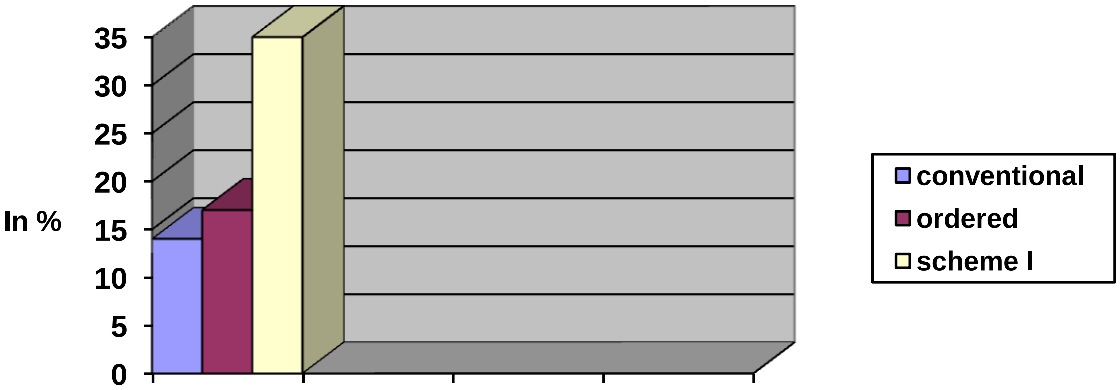


Figure 13: *Power reduction of Ordered and Scheme I-III relative to ordinary FFT*

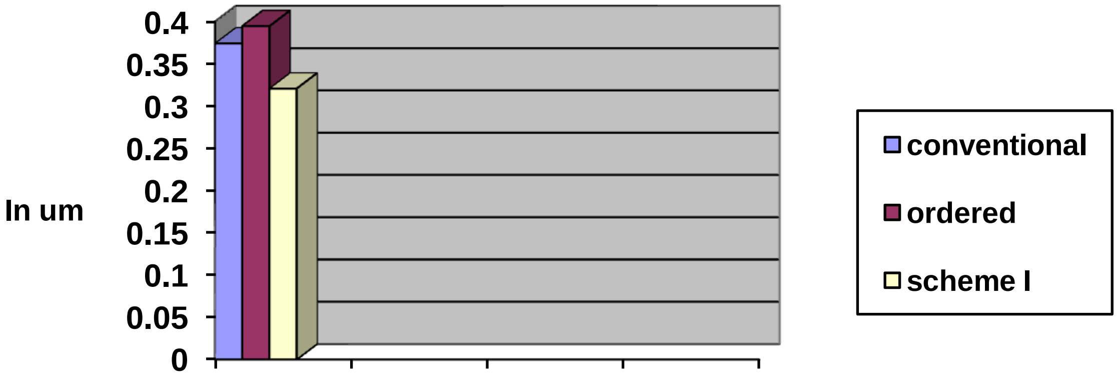


Figure 14: Area of Conventional FFT, Ordered FFT and Scheme I-III

This comparison result gives us very brief and concise information that which architectural combination is best for the design? As can bee seen by the figures above the scheme III outperforms all the other architectures both in power and area. So in respect with the above comparisons results we will compare the area and power for our designed architectures.

The comparative power and area results are shown in Figure 5.13 and 5.14 respectively. Clearly, for the scheme II-III for the 16-point FFT, the best possible power savings results are achieved.

Table 2

16 point FFT	Slack Time (ns)
Conventional	7.89
Scheme II	7.6
Scheme III	0.049

Timing Analysis for 16-point R4SDC FFT.

V. Summary.

In this work, we have discussed low power design techniques for multiplier and butterfly units. Based on the combination of the above two low power techniques with the ordered commutator architecture proposed in Chapter-4, low power 16-point R4SDC FFT architecture is implemented. Power and area parameters are calculated and discussed in the end of the chapter.

The multiplier-less architecture can also be utilised in the long FFTs, but where the area reduction is a major constraint, with a slight expense of power Scheme I or NBW type conventional multiplier can be used.

References:

[1]. Wei Han; Arslan, T.; Erdogan, A.T.; Hasan, M., “Low Power Commutator for Pipelined FFT Processors”, *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on 23-26 May 2005 Page(s):5274 - 5277.*

[2]. Weidong Li,Lars Wanhammar,” A Pipelined FFT processor”,IEEE Transactions on consumer electronics,1999.

[3]. John G. Proakis, Dimitris G. Manolakis, “Digital Signal Processing”, *Third Edition. Principles, Algorithm and Applications.*

[4]. Johansson, S.; Shousheng He; Nilsson, P., “*Wordlength optimization of a pipelined FFT processor,* Circuits and Systems, 1999, Volume 1, Aug. 1999 Page: 501 –503.

[5]. Baas, B.M., Student member, IEEE, “A Low-Power, High-Performance, 1024-point FFT processor*, Solid-State Circuits, IEEE Journal,* Volume 34, Issue 3, March 1999 Page: 380 – 387.

[6]. Schoushheng He; Mats Torkelson,IEEE, “A New approach to pipeline FFT processor”, Applied electronics,IEEE journal,proceedings ofIPPS,1996

[7]. Jen Ming Wu and Yang Chun Fan “coefficient ordering based pipelined FFT/IFFT with minimum switching activity for a low power OFDM communications”, Institute of communications Engineering.

[8]. B.Guoan and E. Jones,”A pipelined FFT processor for word sequential data”,IEEE Transactions on Acoustics, Speech and Signal Processing,vol.37,pp.1982-1985,December 1989.

[9]. Wei Han; Arslan, T.; Erdogan, A.T.; Hasan, M., “Low Power Commutator for Pipelined FFT Processors”, Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on 23-26 May 2005 Page(s):5274 - 5277