



## Radix 2/24 and 2/25 Split Radix Fast Fourier Transform Designs for Reduced Multipliers with Twiddle Factor Approximation

**G Prasanna Kumar**, Electronics and Communication Engineering, Vishnu Institute of Technology, Bhimavaram, India. [godiprasanna@gmail.com](mailto:godiprasanna@gmail.com)

**M.Aravind Kumar**, Professor & Principal, Department of ECE, West Godavari Institute of Science and Engineering, Affiliated to JNTUK, Andhrapradesh. [drmaravindkumar@gmail.com](mailto:drmaravindkumar@gmail.com)

**Satyanarayana Murty. P**, Professor, Department of ECE, Gonna Institute of Information Technology and Sciences, Anakapalle, Affiliated to JNTU Vizianagaram, Andhrapradesh.

### Abstract:

This work proposes a novel method for designing split-radix Fast Fourier Transform architecture. The new Radix- 2/24 and Radix- 2/25 data flow algorithms that are proposed in this paper have effective twiddle factor multiplier reduction. FFT with the radix of split initially modified so that all the twiddle multipliers are placed at two stages and the less important multiplier of multiplication with  $w_8^1$  replaced by two-point butterfly followed by shift register. This technique significantly reduces the twiddle factor of  $w_8^1$  multiplication, which in turn lowers the overall multipliers for higher order FFT. Three real multipliers are used to produce complex multiplication, and five real additions are used to further reduce multipliers. Comparing the proposed data flow designs to other recent existing ones will result in lower overall multipliers. The proposed designs use less power than other recent existing designs.

**Keywords**—Adders; Split-Radix; FFT; Multipliers; Shift Register; Twiddle factor

### 1.INTRODUCTION

Among the many transformations used in signal processing, the Discrete Fourier transform (DFT) has a great deal of application in the fields of communication, image processing, and signal processing. In all these fields, DFT cannot be computed directly from a mathematical formulation due to computations in the real world. FFT is one who compute DFT in a very efficient manner, in the perspective of adders and multipliers, FFT algorithms with different variations in radix [1], [2], [3] is much significant research area, this algorithms grows rapidly after invention of Cooley and turkey [4] who are behind the invention of the FFT. The discrete sequence  $x_n$  has the DFT is  $X_k$  expressed in the equation (1).

$$X_k = \sum_{n=0}^{N-1} x_n W_N^{kn} \quad (1)$$

Radix-2 algorithm is the basic for any higher radix algorithm, in DIF FFT  $X_{2k}$  and  $X_{2k+1}$  is expressed in equations (2) and (3), the flow of radix-2<sup>2</sup> DIF FFT is shown in Figure 1 in which twiddle factor multiplication is done for every two stages.



$$X_{2k} = \sum_{n=0}^{\frac{N}{2}-1} \left\{ x_n + x_{n+\frac{N}{2}} \right\} W_N^{2kn} \quad (2)$$

$$X_{2k+1} = \sum_{n=0}^{\frac{N}{2}-1} \left\{ x_n - x_{n+\frac{N}{2}} \right\} W_N^n W_N^{2kn} \quad (3)$$

Class of split radix FFT is proposed in [9], [10] research on split radix FFT for optimization solution is proposed in [11], [12]. Optimization of split -Radix FFT [13][14] for VLSI implementation of FFT is also important because such a technique reduce power consumption and reduce the time to take for generating all the required outputs that is latency, split radix FFT using CORDIC rotations [15] with very low latency is proposed. Several other designs for optimization of split-radix FFT [16] developed in recent decade. Scaled split-radix FFT [17][18] also reduce the hardware required for implementation. Split radix algorithms for Hadamard transform proposed in [19].

By the consideration given by all the previous works, this work mainly focuses on the split- radix FFT of Radix-2/2<sup>4</sup> and radix-2/2<sup>5</sup> with reduced multipliers. to begin that initially starting with radix-2/2<sup>2</sup> and extended to 2/2<sup>5</sup> radix.

In radix-2/2<sup>2</sup> FFT the odd samples of X<sub>2k+1</sub> is further divided in to two 2-point radix-2 decompositions those are X<sub>4k+1</sub> and X<sub>4k+3</sub>in equations (4) and (5), the complete data flow structure is shown in **Error! Reference source not found.**

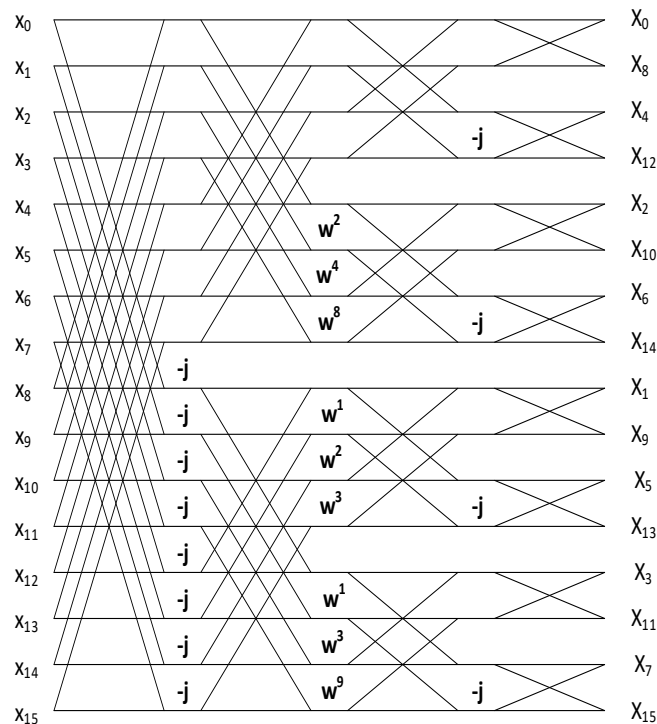


Figure 1 Radix-2<sup>2</sup> DIF FFT Data Flow



$$X_{4k+1} = \sum_{n=0}^{\frac{N}{4}-1} \left\{ x_n - x_{n+\frac{N}{2}} - j \left( x_{n+\frac{N}{4}} - x_{n+\frac{3N}{4}} \right) \right\} W^n W_N^{4kn} \quad (4)$$

$$X_{4k+3} = \sum_{n=0}^{\frac{N}{4}-1} \left\{ x_n - x_{n+\frac{N}{2}} + j \left( x_{n+\frac{N}{4}} - x_{n+\frac{3N}{4}} \right) \right\} W^{3n} W_N^{4kn} \quad (5)$$

In radix-2/2<sup>3</sup> FFT, odd samples of X<sub>2K+1</sub> those are X<sub>4K+1</sub>, X<sub>4K+3</sub> are further divided into two even and odd samples X<sub>8K+1</sub>, X<sub>8K+5</sub>, X<sub>8K+3</sub> and X<sub>8K+7</sub>, those equations are from (6) to Error! Reference source not found., Error! Reference source not found. shows the Radix-2/2<sup>3</sup> FFT in which twiddle factors are multiplied according to the equations. Similarly, Radix-2/2<sup>4</sup> structure is decomposed based on the equations from Error! Reference source not found. to Error! Reference source not found. and shown in Error! Reference source not found.

$$x1_n = x_n - x_{n+\frac{N}{2}} - j \left( x_{n+\frac{N}{4}} - x_{n+\frac{3N}{4}} \right) \quad (6)$$

$$x2_n = x_n - x_{n+\frac{N}{2}} + j \left( x_{n+\frac{N}{4}} - x_{n+\frac{3N}{4}} \right) \quad (7)$$

$$X_{8k+1} = \sum_{n=0}^{\frac{N}{8}-1} \left( x1_n + W_8^1 x1_{n+\frac{N}{8}} \right) W^n W_N^{8kn} \quad (8)$$

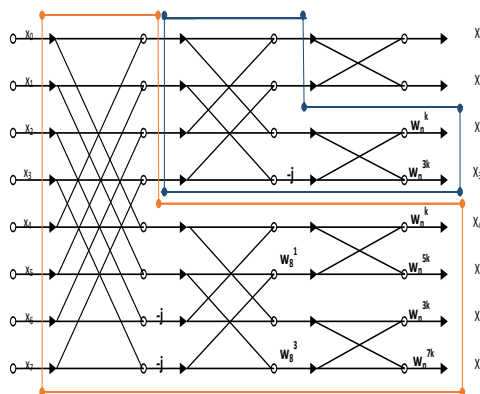


Figure 2 : Radix-2/23 split Radix FFT

Let

$$y1_n = x1_n + W_8^1 x1_{n+\frac{N}{8}}$$



$$y_{2n} = x_{1n} - W_8^1 x_{1_{n+\frac{N}{8}}}$$

$$y_{3n} = x_{2n} - jW_8^1 x_{2_{n+\frac{N}{8}}}$$

$$y_{4n} = x_{2n} + jW_8^1 x_{2_{n+\frac{N}{8}}}$$

The odd samples of  $X_{8k+1}$ ,  $X_{8k+5}$ ,  $X_{8k+3}$  and  $X_{8k+7}$  decomposed further as

$$X_{16k+1} = \sum_{n=0}^{\frac{N}{16}-1} \left( y_{1n} + W_{16}^1 y_{1_{n+\frac{N}{16}}} \right) W^n W_N^{16kn} \quad (9)$$

$$X_{16k+9} = \sum_{n=0}^{\frac{N}{16}-1} \left( y_{1n} - W_{16}^1 y_{1_{n+\frac{N}{16}}} \right) W^{9n} W_N^{16kn} \quad (10)$$

$$X_{16k+5} = \sum_{n=0}^{\frac{N}{16}-1} \left( y_{2n} - jW_{16}^1 y_{2_{n+\frac{N}{16}}} \right) W^{5n} W_N^{16kn} \quad (11)$$

$$X_{16k+13} = \sum_{n=0}^{\frac{N}{16}-1} \left( y_{2n} + jW_{16}^1 y_{2_{n+\frac{N}{16}}} \right) W^{13n} W_N^{16kn} \quad (12)$$

$$X_{16k+3} = \sum_{n=0}^{\frac{N}{16}-1} \left( y_{3n} + W_{16}^3 y_{3_{n+\frac{N}{16}}} \right) W^{3n} W_N^{16kn} \quad (13)$$

$$X_{16k+11} = \sum_{n=0}^{\frac{N}{16}-1} \left( y_{3n} - W_{16}^3 y_{3_{n+\frac{N}{16}}} \right) W^{11n} W_N^{16kn} \quad (14)$$

$$X_{16k+7} = \sum_{n=0}^{\frac{N}{16}-1} \left( y_{4n} - jW_{16}^3 y_{4_{n+\frac{N}{16}}} \right) W^{7n} W_N^{16kn} \quad (15)$$



$$\begin{aligned}
 & X_{16k+15} \\
 & \sum_{n=0}^{\frac{N}{16}-1} \left( y_{4n} + jW_{16}^3 y_{4n+\frac{N}{16}} \right) W^{15n} W_N^{16kn} \quad (16)
 \end{aligned}$$

The dataflow structure of radix-2/2<sup>5</sup> FFT is derived from the equations **Error! Reference source not found.** and shown in Figure 4. The twiddle factors at stage 3 propagates one stage back words except  $w_8^1$ , this step results reduction of complex multiplications shown in Figure 5. This is the maximum simplified design; the proposed data flow structure is shown in Figure 5.

$$\begin{aligned}
 X_{32k+1} = \sum_{n=0}^{\frac{N}{32}-1} & \left( z_{1n} \right. \\
 & \left. + W_{32}^1 z_{1n+\frac{N}{32}} \right) W^n W_N^{32kn} \quad (17)
 \end{aligned}$$

$$\begin{aligned}
 X_{32k+17} = \sum_{n=0}^{\frac{N}{32}-1} & \left( z_{1n} \right. \\
 & \left. - W_{32}^1 z_{1n+\frac{N}{32}} \right) W^{17n} W_N^{32kn} \quad (18)
 \end{aligned}$$

$$\begin{aligned}
 X_{32k+9} = \sum_{n=0}^{\frac{N}{32}-1} & \left( z_{2n} \right. \\
 & \left. - jW_{32}^1 z_{2n+\frac{N}{32}} \right) W^{9n} W_N^{32kn} \quad (19)
 \end{aligned}$$

$$\begin{aligned}
 & X_{32k+25} \\
 & \sum_{n=0}^{\frac{N}{32}-1} \left( z_{2n} + jW_{32}^1 z_{2n+\frac{N}{32}} \right) W^{25n} W_N^{32kn} \quad (20)
 \end{aligned}$$

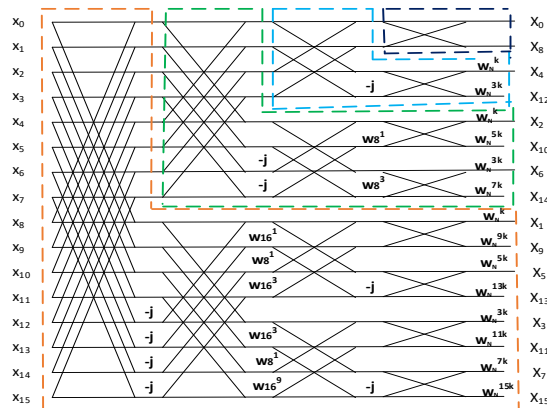


Figure 3: Radix-2/24 Split Radix FFT

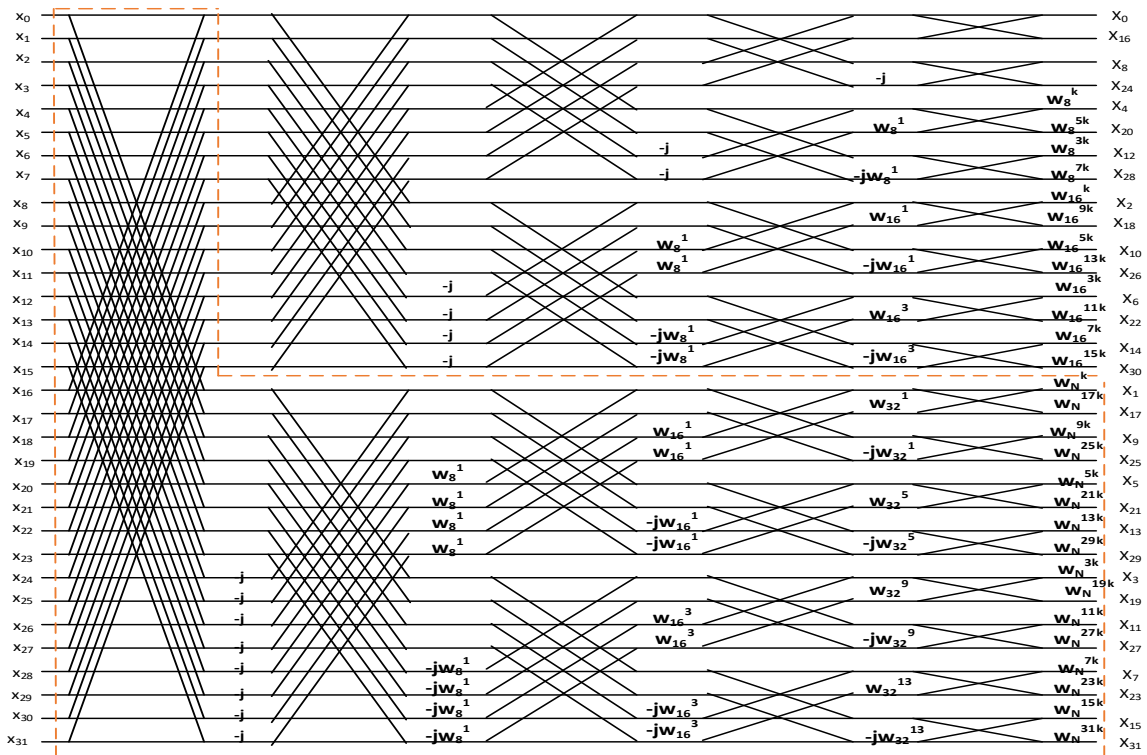


Figure 4: Radix-2/25 split radix FFT with rearranging twiddle factors

The radix-2/24 split radix FFT is depicted in Figure 3, where the twiddle factors are changed so that only the lower L-shaped portions for each of the two phases need to be multiplied. The Radix-2/25 split radix FFT with rearranging twiddle factors is depicted in Figure 4, where the higher-level L-shaped design includes five stages and the subsequent levels are 4, 3, and 2 accordingly. The higher-level L-shaped design for twiddle factor minimization is shown by the pink dotted line. Figure 6 depicts the reduced architecture with the 2-point butterfly unit followed by the shift register (B2SR) in place of the twiddle factor  $w_{81}$ . The B2SR is used in the design in place of  $w_{81}$  to minimise total computations.

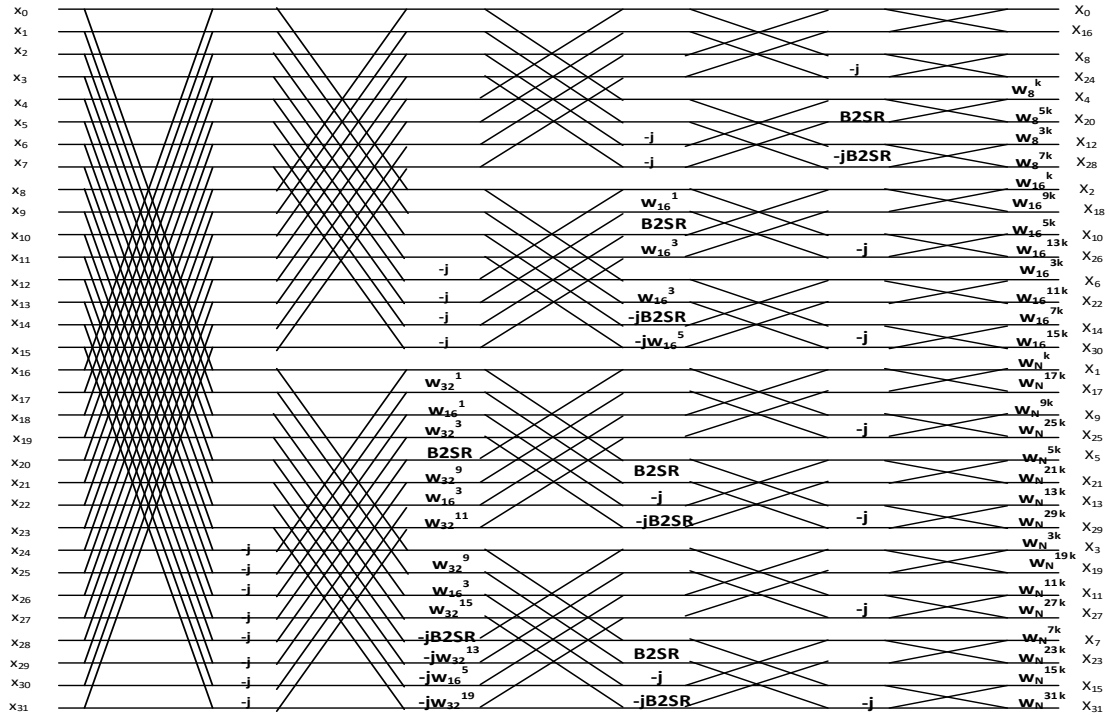


Figure 5: Radix-2/25 split radix FFT with rearranging twiddle factors and B2SR

PROPOSED RADIX-2/2<sup>4</sup> AND RADIX-2/2<sup>5</sup> DESIGNS WITH TWIDDLE FACTOR APPROXIMATION

Now we propose Radix -2/2<sup>5</sup> with twiddle factor approximation, the coefficients of the lower end of 32-point FFT is arranged in such a way to differentiate w81 coefficients shown in Figure 5. These coefficients are at stage 3 and stage 4 of FFT data flow structure.

Twiddle factor simplification procedure:

$$W_8^1 = e^{\frac{-j2\pi}{8}}$$

$$W_8^1 = \frac{1-j}{\sqrt{2}}$$

a complex number of the form r1+ji1 multiplying with twiddle factor  $W_8^1 = \frac{1-j}{\sqrt{2}}$  then the result of the form r1+i1 +j(i1-r1). Two real adders and multiplication with 0.707 meets our desired multiplication output with  $W_8^1$ , Figure 5 shows the adder and subtractors units, that is two-point butterfly followed by multiplication with the value 0.707. The multiplication is performed by shift registers and adders shown in **Error! Reference source not found.**, input data is right shift by one, two and three

parallelly and perform summation of these three results gives our multiplication output.

The proposed design with approximating the twiddle factor coefficient is obtained by eliminating multipliers from radix-2/2<sup>5</sup> shown in the Figure 4. The complete design is shown in Figure 5, in which B2SR replaces the w81 in Figure 4. The symbol B2SR in the diagram represents the 2-point butterfly unit followed by shift register arrangement, so by this design multipliers are reduced and in place adders increased.

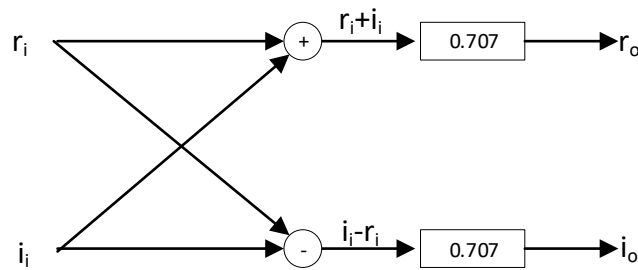


Figure 6: Simplified Design B2SR

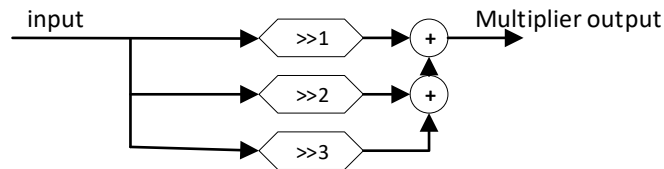


Figure 7 : Multiplication with 0.707 by using shift register technique

### COMPLEXITY OF THE PROPOSED DESIGNS

The complexity of the design majorly dependent on multipliers, the multipliers count in proposed radix-2/24 design is analyzed by the following way, N=8 no multipliers are required because of approximation of  $w_8^1$  coefficient in the design. The complex multipliers for implementing 16-point

FFT is 4 and the real multiplication are  $4*3= 12$ .

Figure 8 summarizes the multipliers needed to implement the suggested Radix-2/24 design:  $3N/4$  real multipliers are needed for each first "L" shape and  $3N/2(2m-1)$  multipliers are needed at the end of each first "L" shape where  $m = 0, 1, \text{ and } 2$ . There are multipliers for the  $N/2$  butterfly stage for  $N=16, 32, 64, \text{ etc.}$ , and there are always 8 butterfly stages for radix-2/24 starting at  $N=32$



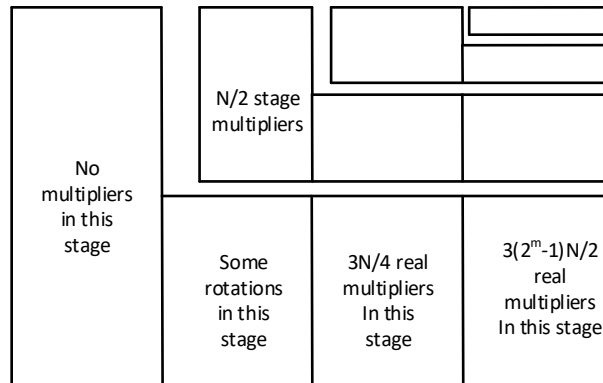


Figure 8 : Number of Multipliers Analysis of the proposed designs

$$M_N = 3 \frac{N}{4} + 3(2^m - 1)8 + M_{\frac{N}{2}} + 8M_{\frac{N}{2^4}}$$

Example: N=16, m=1 so

$$M_{16} = 3 \frac{16}{4} + 3(2^0 - 1)8 + M_{\frac{16}{2}} + 8M_{\frac{16}{2^4}}$$

$$M_{\frac{16}{2}} = 0$$

$$8M_1 = 0$$

So that

$$M_{16} = 12$$

$$M_{32} = 3 \frac{32}{4} + 3(2^1 - 1)8 + M_{\frac{32}{2}} + 8M_{\frac{32}{2^4}}$$

$$M_{32} = 24 + 24 + 12$$

$$M_{32} = 60$$

Similarly, for radix  $2/2^5$  proposed design requires the multipliers shown in equation.



$$M_N = 3 \frac{N}{4} + 3(2^m - 1)16 + M_{\frac{N}{2}} + 16M_{\frac{N}{2^5}}$$

The number of Adders required increases because of approximation of multipliers with shift register and adder unit B2SR that is two-point butterfly followed by shift register unit. B2SR takes 2 real adders and shift register unit takes another 2 real adders.

### DISCUSSIONS AND COMPARISONS

As shown in Table 1 of the calculations of the proposed designs and other preceding radix-2/24 designs to date, the suggested design decreases the number of multipliers and increases the adders due to B2SR units, but overall resource utilization is low by observing multiple defined designs. Figures 9 and 10 depict, respectively, the growth of adders and multipliers as the order of the FFT grows. The growth of multipliers in the suggested designs is less than that of all other well-known designs, whereas the increase of adders is slightly more than that of several other designs.

The power consumption of the proposed designs with several other defined designs with increase of the order of the filter shown in Figure 11, which shows that the overall power consumption of the proposed designs is lower than other

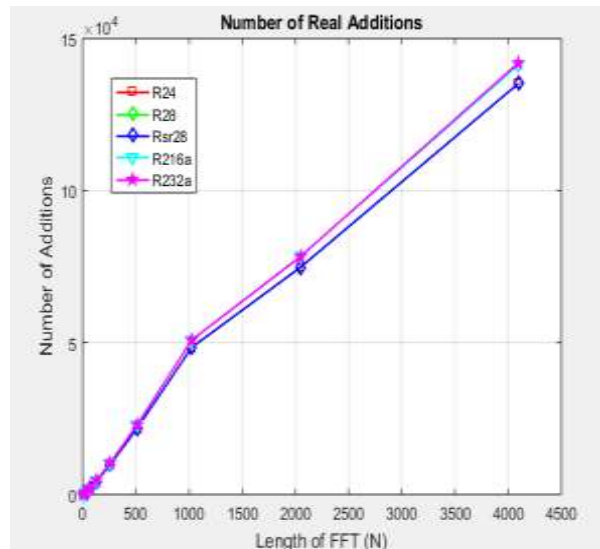


Figure 9: Number of real Adders versus length of FFT

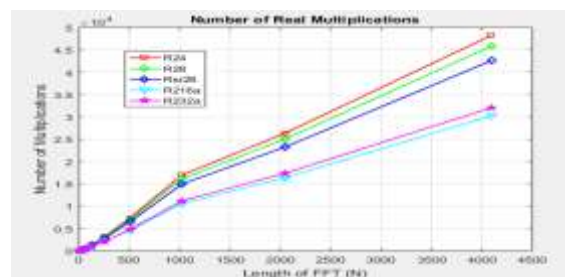


Figure 10 : Number of real Multipliers versus length of FFT,

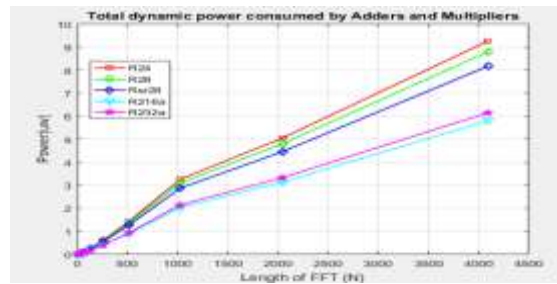


Figure 11: comparison of both designs

Table 1: Computational complexity comparison of the proposed design with several defined design

N	R2/4		R2/8		RSR		Proposed R2/2 <sup>4</sup>		Proposed R2/2 <sup>5</sup>	
	Multiplie r	Adder s	Multiplie r	Adder s	Multiplie r	Adder s	Multiplie r	Adder s	Multiplie r	Adder s
8	4	52	4	52	0	52	0	52	0	52
16	24	148	24	148	20	148	12	148	12	148
32	84	388	84	388	76	388	60	388	48	388
64	248	964	240	964	224	964	180	964	168	964
128	660	2308	636	2308	584	2308	444	2308	456	2308
256	1656	5308	1592	5308	1468	5308	1092	5308	1110	5308
512	3988	12292	3812	12292	3540	12292	2700	12292	2598	12292
1024	9336	27652	8896	27652	8248	27652	6420	27652	6006	27652
2048	21396	61444	20364	61444	18880	61444	12336	61444	14022	61444
4096	48248	135172	45832	135172	42580	135172	30264	135172	32022	135172

**CONCLUSIONS:**

To implement FFT based on split Radix, two innovative and effective designs—Radix-2/24 and Radix-2/25 with twiddle factor approximation—are proposed. The two-point butterfly unit and shift register unit in the suggested designs streamline multiplication using the factor W81. The proposed design is mostly for reducing twiddle factor multipliers in the intermediate stages. The suggested design reduces the multipliers and boosts the adders to execute the overall N point architecture. The overall power of the suggested designs is 40% less than that of earlier designs, which is a substantial reduction.



**Reference:**

- [1] Y. Suzuki, T. Sone, and K. Kido, "A new FFT algorithm of radix 3, 6, and 12," *IEEE Trans. Acoust.*, vol. 34, no. 2, pp. 380–383, 1986.s:
- [2] W. Zheng and K. Li, "Split radix algorithm for length  $6^m$  DFT," *IEEE Signal Process. Lett.*, vol. 20, no. 7, pp. 713–716, 2013.
- [3] R. Yavne, "An economical method for calculating the discrete Fourier transform," in *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, 1968, pp. 115–125.
- [4] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," *Math. Comput.*, vol. 19, no. 90, pp. 297–301, 1965.
- [5] T. Sansaloni, A. Perez-Pascual, V. Torres, and J. Valls, "Efficient pipeline FFT processors for WLAN MIMO-OFDM systems," *Electron. Lett.*, vol. 41, no. 19, pp. 1043–1044, 2005.
- [6] M. Prasad, K. Srinivas, and G. P. Kumar, "A New Weighted Average Filter for Removing Camera Shake," *Int. J. Comput. Appl.*, vol. 975, p. 8887.
- [7] D. Takahashi, "An extended split-radix FFT algorithm," *IEEE Signal Process. Lett.*, vol. 8, no. 5, pp. 145–147, 2001.
- [8] S. Bouguezel, M. O. Ahmad, and M. N. S. Swamy, "A new radix-2/8 FFT algorithm for length- $q$ /split times/2/sup  $m$ /DFTs," *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 51, no. 9, pp. 1723–1732, 2004.
- [9] W.-C. Yeh and C.-W. Jen, "High-speed and low-power split-radix FFT," *IEEE Trans. Signal Process.*, vol. 51, no. 3, pp. 864–874, 2003.
- [10] G. P. Kumar, B. T. Krishna, and P. Kotipalli, "Split Radix Fast Fourier Transform Design with Reduced Multipliers," 2020.
- [10] T.-Y. Sung, "Memory-efficient and high-speed split-radix FFT/IFFT processor based on pipelined CORDIC rotations," *IEE Proceedings-Vision, Image Signal Process.*, vol. 153, no. 4, pp. 405–410, 2006.