

Comparison of Discrete Fourier Transform and Fast Fourier Transform with Reduced Number of Multiplication and Addition Operations

Bimal Pal

Department of Computer Science & Engineering,
Government College of Engineering and Ceramic Technology,
Kolkata, Pin-700010, West Bengal, India.

Pijush Kanti Bhattacharjee

Department of Electronics and Communication Engineering,
Camellia Institute of Technology and Management,
Bainchi, Hooghly, Pin-712134, West Bengal, India.

Received: November 6, 2020. Revised: December 11, 2020. Accepted: December 15, 2020.
Published: December 17, 2020.

Abstract— Fast Fourier Transform is an advanced algorithm for computing Discrete Fourier Transform efficiently. Although the results available from the operation of Discrete Fourier Transform (DFT) and Fast Fourier Transform (FFT) are same, but exploiting the periodicity and symmetry property of phase factor Fast Fourier Transform computes the Discrete Fourier Transform using reduced number of multiplication and addition operations. The basic structure used in the operations of Fast Fourier Transform is the Butterfly structure. For the implementation of Fast Fourier Transform the two methods are used such as decimation in time (DIT) and decimation in frequency (DIF). Both the methods give same result but for decimation in time of Fast Fourier Transform bit reversed inputs are applied and for decimation in frequency of Fast Fourier Transform normal order inputs are applied, and the result is reversed again. In this paper, operations for DFT and FFT have been discussed and shown with examples. It is found that generalized formula for FFT have been described same in the books, but the expressions in the intermediate computations for the first decimation and second decimation are different in the various books of Digital Signal Processing. The expressions in the intermediate computation of FFT described in different books are broadly compared in this paper.

Keywords— Butterfly, Discrete Fourier Transform, Fast Fourier Transform, Twiddle Factor.

I. INTRODUCTION

Discrete Fourier Transform gives us frequency spectrum of a time domain signal from its equally spaced sampled sequence. It is given a vector of n input amplitudes like $[f_0, f_1, f_2, \dots, f_{n-2}, f_{n-1}]$, Discrete Fourier transform yields a set of n frequency magnitudes, where $n = 0, 1, 2, \dots, N - 1$. The “ N ” is the length of the sequence to be transformed. $X(k)$ are a finite-duration sequence for $0 \leq k \leq N - 1$. Here k is used to denote the frequency domain ordinal, and n is used to represent the time domain ordinal [1]-[15].

N -point Discrete Fourier Transform (DFT) of the sampled sequence $x(n)$ can be computed using the equation,

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j2\pi kn/N}, \text{ where } j = \sqrt{-1}$$

$X(k) = \sum_{n=0}^{N-1} x(n) \cdot W_N^{kn}$, where $W_N = e^{-j2\pi/N}$ for a given value of k .

For 8-point DFT the values of W_N^{kn} will be,

$$W_8^0 = e^{-j2\pi \cdot 0/N} = e^0 = 1$$

$$W_8^1 = e^{-j2\pi \cdot 1/8} = \cos \pi/4 - j \sin \pi/4 = 0.707 - j (0.707)$$

$$W_8^2 = e^{-j\pi/2} = \cos \pi/2 - j \sin \pi/2 = -j$$

$$W_8^3 = -0.707 - j (0.707)$$

$$W_8^4 = -1$$

$$W_8^5 = -0.707 + j (0.707)$$

$$W_8^6 = j$$

$$W_8^7 = 0.707 + j (0.707)$$

$$W_8^8 = 1$$

It is seen the magnitudes of W_8^{kn} , i.e., $|W_8^{kn}|$ is always 1, only phase angle differs. Hence, $|W_N^{kn}| = 1$.

For input sequence, value of $x(n) = [1, 2, 3, 4, 5, 6, 7, 8]$, then the magnitudes of the 8-point DFT will be as,

$$X[0] = x(0).W_8^{0.0} + x(1).W_8^{0.1} + x(2).W_8^{0.2} + x(3).W_8^{0.3} + x(4).W_8^{0.4} + x(5).W_8^{0.5} + x(6).W_8^{0.6} + x(7).W_8^{0.7}$$

$$X[0] = 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 = 36.000000$$

$$X[1] = x(0).W_8^{1.0} + x(1).W_8^{1.1} + x(2).W_8^{1.2} + x(3).W_8^{1.3} + x(4).W_8^{1.4} + x(5).W_8^{1.5} + x(6).W_8^{1.6} + x(7).W_8^{1.7}$$

$$X[1] = -4 + 9.656j$$

$$X[1] = \sqrt{(-4)^2 + (9.656)^2} = 10.452520$$

$X[2] = 5.656862$ [when W_8^{kn} , $kn > 8$, applying periodicity and symmetry property to keep kn within $0 \sim 8$]

$$X[3] = 4.329572$$

$$X[4] = 4.000000$$

$$X[5] = 4.329563$$

$$X[6] = 5.656832$$

$$X[7] = 10.452387$$

The results of the DFT operations have been calculated and verified using the program written in the C Compiler software. Here for this 8-point DFT total 8^2 or 64 multiplications and 56 additions have been done. For a particular value of k the direct computations of $X(k)$ requires N complex multiplications and $(N - 1)$ additions. So, N values of DFT computation requires N^2 complex multiplications and $N(N - 1)$ complex additions.

II. FAST FOURIER TRANSFORM

Fast Fourier Transform (FFT) uses mathematical short-cuts to reduce the number of computations using the Symmetry and Periodicity property of phase factor which is also called twiddle factor like values of W_N^{kn} mentioned below:

$$W_N^{k+N} = W_N^k \quad [\text{Periodicity Property}]$$

$$W_N^{k+N/2} = -W_N^k \quad [\text{Symmetry Property}]$$

For 8-point Fast Fourier Transform (FFT) due to Periodicity and Symmetry Property the values of W_8^{kn} are,

$$W_8^0, W_8^8, W_8^{16}, W_8^{24}, \dots = 1;$$

$$W_8^4, W_8^{12}, W_8^{20}, W_8^{28}, \dots = -1;$$

For an N -point Fast Fourier Transform (FFT), the required twiddle factors are $\log_2 N$ and the values of twiddle factors are constant and can be stored previously.

In FFT, when the summation $\sum_{n=0}^{N-1} x(n)$, W_N^{kn} is splitted into two parts, one with the even-indexed $x(n)$, i.e., $n = 2n$ for even, and another is odd-indexed $x(n)$, i.e., $n = 2n + 1$ for odd, where $n = 0, 1, 2, \dots, \frac{N}{2} - 1$, and the first decomposition equation of FFT can be written as [1]-[21],

$$X(k) = \sum_{n=0}^{\frac{N}{2}-1} x(2n).W_N^{2kn} + \sum_{n=0}^{\frac{N}{2}-1} x(2n+1).W_N^{k(2n+1)}$$

$$\text{Since, } W_N^2 = [e^{-j(\frac{2\pi}{N})}]^2 = e^{-j(\frac{2\pi}{N/2})} = W_{N/2}$$

$$\text{So, } X(k) = \sum_{n=0}^{\frac{N}{2}-1} x(2n).W_{N/2}^{kn} + W_N^k \sum_{n=0}^{\frac{N}{2}-1} x(2n+1).W_{N/2}^{kn} \quad \dots \dots \dots (1)$$

Then, $X(k) = X_e(k) + W_N^k X_o(k)$, where e indicates even and o indicates odd.

From Symmetry property, for 8-points FFT, we can write as,

$$X(k) = X_e(k) + W_8^k X_o(k), \quad \text{for } 0 \leq k \leq 3,$$

$$X(k) = X_e(k-4) - W_8^{k-4} X_o(k-4), \quad \text{for } 4 \leq k \leq 7,$$

$$\text{Since } k \geq \frac{N}{2}, W_N^{k+N/2} = -W_N^k;$$

Then for $k = 0$ to 7 , we get,

$$X(0) = X_e(0) + W_8^0 X_o(0); X(4) = X_e(0) - W_8^0 X_o(0);$$

$$X(1) = X_e(1) + W_8^1 X_o(1); X(5) = X_e(1) - W_8^1 X_o(1);$$

$$X(2) = X_e(2) + W_8^2 X_o(2); X(6) = X_e(2) - W_8^2 X_o(2);$$

$$X(3) = X_e(3) + W_8^3 X_o(3); X(7) = X_e(3) - W_8^3 X_o(3);$$

This can be represented in butterfly structure like below:

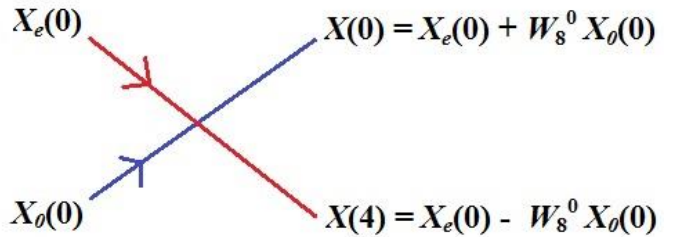


Fig. 1 Flow graph of butterfly diagram

For N -point FFT, each butterfly stage consists of $N/2$ butterflies and number of stages in the flow graph is given by $M = \log_2 N$, therefore the number of complex multiplications are required by $\frac{N}{2} \log_2 N$ and number of complex additions are given by $N \log_2 N$.

Since major discussion point in this paper is the variation of expressions in the intermediate computation of Fast Fourier Transform, here in an example, detailed operations of an 8-point decimation in time of FFT have been discussed and then comparisons with various books are shown in Table II.

The first part of the equation number (1) consists of even sequence of $x(n)$ and the second part consists of odd sequence of $x(n)$. The first part of the equation, i.e.,

$$X(k)_{\text{even}} = \sum_{n=0, \text{even}}^{\frac{N}{2}-1} x(2n).W_{N/2}^{kn}; n = 0, 1, 2, 3 \text{ for } N = 8;$$

$$X(0)_{\text{even}} = \sum_{n=0, \text{even}}^3 x(2n).W_4^{0n}, \text{ for } k = 0.$$

$$X(0)_{\text{even}} = x(0).W_4^0 + x(2).W_4^0 + x(4).W_4^0 + x(6).W_4^0;$$

$$X(1)_{\text{even}} = x(0).W_4^0 + x(2).W_4^1 + x(4).W_4^2 + x(6).W_4^3;$$

$$\text{for } k = 1.$$

$$X(2)_{\text{even}} = x(0).W_4^0 + x(2).W_4^2 + x(4).W_4^4 + x(6).W_4^6;$$

$$\text{for } k = 2.$$

$$X(3)_{\text{even}} = x(0).W_4^0 + x(2).W_4^3 + x(4).W_4^6 + x(6).W_4^9;$$

$$\text{for } k = 3.$$

The second part of the equation, i.e.,

$$X(k)_{\text{odd}} = W_N^k \sum_{n=0, \text{odd}}^{\frac{N}{2}-1} x(2n+1).W_{N/2}^{kn};$$

$$\text{Here } n = 0, 1, 2, 3 \text{ for } N = 8;$$

$$X(0)_{\text{odd}} = W_8^0 [x(1)W_4^0 + x(3)W_4^0 + x(5)W_4^0 + x(7)W_4^0];$$

$$\text{for } k = 0.$$

$$X(1)_{\text{odd}} = W_8^1 [x(1)W_4^0 + x(3)W_4^1 + x(5)W_4^2 + x(7)W_4^3];$$

$$\text{for } k = 1.$$

$$X(2)_{\text{odd}} = W_8^2 [x(1)W_4^0 + x(3)W_4^2 + x(5)W_4^4 + x(7)W_4^6];$$

$$\text{for } k = 2.$$

$$X(3)_{\text{odd}} = W_8^3 [x(1)W_4^0 + x(3)W_4^3 + x(5)W_4^6 + x(7)W_4^9];$$

$$\text{for } k = 3.$$

TABLE I. NUMBER OF OPERATIONS FOR N/2 POINTS FFT

The number of operations when N-point (N = 8) FFT is splitted into two FFTs of N/2-point size (N/2 = 4)			
Multiplication operations for first part [even sequence of x(n)]	Multiplication operations for second part [odd sequence of x(n)]	Addition operations for first part [even sequence of x(n)]	Addition operations for second part [odd sequence of x(n)]
(i)	(ii)	(iii)	(iv)
16	16 + 4	12	12
(i) + (ii), For N = 8, Total Multiplications = 36; General Formula for N-point FFT: $2(N/2)^2 + N/2 = N^2/2 + N/2$		(iii) + (iv), For N = 8, Total Additions = 24; General Formula for N-point FFT: $2[N/2 (N/2 - 1)]$	

It is seen that a FFT of size N can be computed through two FFTs of size N/2 and the computation of X(k) requires $2(N/2)^2 + N/2 = N^2/2 + N/2$ complex multiplications and $2[N/2 (N/2 - 1)]$ complex additions. These computing operations are described in the book Proakis *et al.* [1]-[15].

The computing operations of 8-point FFT described in the various Digital Signal Processing books are narrated in the Table II below. The computing operations are described when a 8-point FFT is splitted into even and odd parts, i.e., decimated by 2 for the value of $N \geq 3$, but all values deducted in the different books are approximately same.

TABLE II. DIFFERENT BOOKS COMPARED N/2 POINTS FFT

Srl. No.	Name of Books	Multiplication Operations	Addition Operations
1.	Proakis <i>et al.</i>	$N^2/2 + N/2$	$2[N/2 (N/2 - 1)]$
2.	Diniz <i>et al.</i>	$N^2/2 + N$	$N^2/2$
3.	Rabiner <i>et al.</i>	$N^2/2 + N$	---
4.	Oppenheim <i>et al.</i>	$N + 2(N/2)^2$	$N + 2(N/2)^2$
5.	Mitra	$N + N^2/2$	$N + N^2/2$ (approx)
6.	Salivahanan <i>et al.</i>	$N + N^2/2$	$N + N^2/2$

Now, an example has been shown for 16-point FFT. So, when 16-point FFT is splitted into two parts, i.e., even and odd parts, we get the sequences x(n) after decimation (considering $N = N/2$), here $N = 16$,

$$X(k)_{\text{even}} = x(0), x(2), x(4), x(6), x(8), x(10), x(12), x(14);$$

$$X(k)_{\text{odd}} = x(1), x(3), x(5), x(7), x(9), x(11), x(13), x(15);$$

When $X(k)_{\text{even}}$ and $X(k)_{\text{odd}}$ are further splitted, i.e., decimated by two again, then the equation (1) is written,

$$X(k)_{\text{even}} = \sum_{n=0}^{\frac{N}{4}-1} x(4n). W_N^{4kn} + \sum_{n=0}^{\frac{N}{4}-1} x(4n+2). W_N^{k(4n+2)}$$

$$X(k)_{\text{even}} = \sum_{n=0}^{\frac{N}{4}-1} x(4n). W_{N/4}^{kn} + W_{N/2}^k \sum_{n=0}^{\frac{N}{4}-1} x(4n+2). W_{N/4}^{kn}$$

$$X(k)_{\text{even}} = \sum_{n=0}^3 x(4n). W_4^{kn} + W_8^k \sum_{n=0}^3 x(4n+2). W_4^{kn} \dots \dots \dots (2)$$

where $n = 0, 1, 2, 3$, and $N = 16$.

Now the x(n) sequences for the first and second part of $X(k)_{\text{even}}$ will be

$$X(k)_{\text{even,1st part}} = x(0), x(4), x(8), x(12);$$

$$X(k)_{\text{even,2nd part}} = x(2), x(6), x(10), x(14);$$

The equation for $X(k)_{\text{odd}}$ will be

$$X(k)_{\text{odd}} = \sum_{n=0}^{\frac{N}{4}-1} x(4n+1). W_N^{k(4n+1)} + \sum_{n=0}^{\frac{N}{4}-1} x(4n+3). W_N^{k(4n+3)}$$

$$X(k)_{\text{odd}} = W_N^k \sum_{n=0}^{\frac{N}{4}-1} x(4n+1). W_{N/4}^{kn} +$$

$$W_N^{3k} \sum_{n=0}^{\frac{N}{4}-1} x(4n+3). W_{N/4}^{kn}$$

$$X(k)_{\text{odd}} = W_{16}^k \sum_{n=0}^3 x(4n+1). W_4^{kn} + W_{16}^{3k} \sum_{n=0}^3 x(4n+3). W_4^{kn} \dots \dots \dots (3)$$

where $n = 0, 1, 2, 3$, and $N = 16$.

Now the x(n) sequences for the first and second part of $X(k)_{\text{odd}}$ will be

$$X(k)_{\text{odd,1st part}} = x(1), x(5), x(9), x(13);$$

$$X(k)_{\text{odd,2nd part}} = x(3), x(7), x(11), x(15);$$

Now putting the value of x(n) sequences in the equation (2), we get the even sequences

$$X(0)_{\text{even}} = x(0). W_4^0 + x(4). W_4^0 + x(8). W_4^0 + x(12). W_4^0 + W_8^0 [x(2). W_4^0 + x(6). W_4^0 + x(10). W_4^0 + x(14). W_4^0];$$

$$X(1)_{\text{even}} = x(0). W_4^0 + x(4). W_4^1 + x(8). W_4^2 + x(12). W_4^3 + W_8^1 [x(2). W_4^0 + x(6). W_4^1 + x(10). W_4^2 + x(14). W_4^3];$$

$$X(2)_{\text{even}} = x(0). W_4^0 + x(4). W_4^2 + x(8). W_4^4 + x(12). W_4^6 + W_8^2 [x(2). W_4^0 + x(6). W_4^2 + x(10). W_4^4 + x(14). W_4^6];$$

$$X(3)_{\text{even}} = x(0). W_4^0 + x(4). W_4^3 + x(8). W_4^6 + x(12). W_4^9 + W_8^3 [x(2). W_4^0 + x(6). W_4^3 + x(10). W_4^6 + x(14). W_4^9];$$

Now putting the value of x(n) sequences in the equation (3), we get the odd sequences

$$X(0)_{\text{odd}} = W_{16}^0 [x(1). W_4^0 + x(5). W_4^0 + x(9). W_4^0 + x(13). W_4^0] + W_{16}^0 [x(3). W_4^0 + x(7). W_4^0 + x(11). W_4^0 + x(15). W_4^0];$$

$$X(1)_{\text{odd}} = W_{16}^1 [x(1). W_4^0 + x(5). W_4^1 + x(9). W_4^2 + x(13). W_4^3] + W_{16}^3 [x(3). W_4^0 + x(7). W_4^1 + x(11). W_4^2 + x(15). W_4^3];$$

$$X(2)_{\text{odd}} = W_{16}^2 [x(1). W_4^0 + x(5). W_4^2 + x(9). W_4^4 + x(13). W_4^6] + W_{16}^6 [x(3). W_4^0 + x(7). W_4^2 + x(11). W_4^4 + x(15). W_4^6];$$

$$X(3)_{\text{odd}} = W_{16}^3 [x(1). W_4^0 + x(5). W_4^3 + x(9). W_4^6 + x(13). W_4^9] + W_{16}^9 [x(3). W_4^0 + x(7). W_4^3 + x(11). W_4^6 + x(15). W_4^9];$$

Initially N-point FFT is splitted into two N/2 point FFTs. When each N/2-point FFT is further splitted into N/4 point FFTs, then the required numbers of multiplication and addition operations have been described in the Table III below for 16-point FFTs as per described calculations.

TABLE III. NUMBER OF OPERATIONS FOR $N/4$ POINTS FFT

The number of operations when two $N/2$ -point FFTs are splitted into four $N/4$ -point FFTs			
Multiplication operations for the even sequence of $x(n)$ when $N = 16$	Multiplication operations for the odd sequence of $x(n)$ when $N = 16$	Addition operations for the even sequence of $x(n)$ when $N = 16$	Addition operations for the odd sequence of $x(n)$ when $N = 16$
(i)	(ii)	(iii)	(iv)
32 + 4 (additional)	32 + 8 (additional)	24	24
(i) + (ii), For $N = 16$, Total Multiplications = 76; General Formula for N -point FFT: $4(N/4)^2 + N/2 + N/4 = N^2/4 + N/2 + N/4$		(iii) + (iv), For $N = 16$, Total Additions = 48; General Formula for N -point FFT: $4[N/4 (N/4 - 1)]$	

So, when $N/2$ point transforms are decomposed into $N/4$ point transforms then the terms $2(N/2)^2 + N/2$ are replaced by $N^2/4 + N/2 + N/4$ for total number of multiplications.

The computing operations when $N/2$ -point FFTs are further decimated into $N/4$ -point FFTs are described in the various Digital Signal Processing books and are delineated in the Table IV below.

TABLE IV. DIFFERENT BOOKS COMPARED $N/4$ POINTS FFT

Srl. No.	Name of Books	Multiplication Operations	Addition Operations
1.	Proakis <i>et al.</i>	$N^2/4 + N/2$	$4[N/4 (N/4 - 1)]$
2.	Oppenheim <i>et al.</i>	$N + N + 4(N/4)^2$	$N + N + 4(N/4)^2$
3.	Salivahanan <i>et al.</i>	$N + N + 2(N/4)^2$	$N + N + 2(N/4)^2$

The decimation of data sequence can be repeated again from $N/4$ points until the resulting sequences are reduced to one point sequence. The decimation can be performed up to $\log_2 N$ times. Therefore, the total number of complex multiplications are reduced to $(N/2) \log_2 N$. The Number of complex additions are $N \log_2 N$. For decimation in time (DIT) of FFT bit reversal input sequences are used and in the output normal order bit sequences are available.

A. Application of Fast Fourier Transform

Fast Fourier Transform is applied in digital signal processing; digital filtering especially noise filtering; frequency information of a signal; the correlation of two time series; image filtering, image analysis, image compression, image reconstruction and image matching; realization of hardware structure in very large scale integration (VLSI) chips; communications field like Orthogonal Frequency Division Multiplexing or Multiple Access (OFDM or OFDMA), Digital Broadcasting, Worldwide Interoperability for Microwave Access (WiMAX), Long Term Evolution (LTU in 4G Mobile System); in security for long integer

multiplication in public key cryptography algorithms like RSA (Rivest, Shamir, and Adelman); spectrogram etc. [1]-[16].

B. Comparison of DFT and FFT

A comparison format of DFT and FFT is shown below for different values of N in Table V [1]-[21].

TABLE V. DFT & FFT COMPARISON FOR VARIOUS N

Value of N	DFT		FFT	
	Complex Multiplications N^2	Complex Additions $N(N - 1)$	Complex Multiplications $(N/2) \log_2 N$	Complex Additions $N \log_2 N$
8	64	56	12	24
16	256	240	32	64
32	1024	992	80	160
64	4096	4032	192	384

III. CONCLUSION

It is found that for decimation in time operation for $N/2$ point FFT and $N/4$ point FFT, different results have been described in different books at the intermediate stages as described in the TABLE II and TABLE IV but the result available in the final stage, i.e., final calculations are same value in all the books and the literature as shown in the TABLE V. The ambiguity is mainly related with the multiplication operation of twiddle factors and to be studied in detail further. Therefore, the direct computation of Discrete Fourier Transform requires computation of the order of $O(N^2)$, whereas Fast Fourier Transform entails only the order of $O(N \log_2 N)$. Thus FFT minimizes time and space complexities having accurate result within real time.

References

- [1] J. G. Proakis and D. G. Manolakis, Digital Signal Processing: Principles, Algorithms, and Applications, Fourth Edition, Pearson Education, India, 2007.
- [2] P. S. R. Diniz, E. A. B. Da Silva, and S. L. Netto, Digital Signal Processing: System Analysis and Design, Second Edition, Cambridge University Press, Delhi, 2017.
- [3] L. R. Rabiner and B. Gold, Theory and Application of Digital Signal Processing, Prentice Hall of India Private Limited, New Delhi, 2003.
- [4] A. V. Oppenheim and R. W. Schaffer, Digital Signal Processing, Prentice Hall India Private Limited, New Delhi, 1999.
- [5] S. K. Mitra, Digital Signal Processing: A Computer – Based Approach, Tata McGraw-Hill Publishing Company Limited, New Delhi, 2000.
- [6] S. Salivahanan, A. Vallavaraj, and C. Gnanapriya, Digital Signal Processing, Tata McGraw-Hill Publishing Company Limited, New Delhi, 2005.
- [7] P. Ramesh Babu, Digital Signal Processing, Third Edition, Scitech Publications (India) Pvt. Ltd., 2001.
- [8] G. D. Bergland, "A Guided Tour of the Fast Fourier Transform", IEEE Spectrum, vol. 6, pp. 41-52, July 1969.

- [9] P. Duhamel and M. Vetterli, "Fast Fourier Transforms: A Tutorial Review", *Signal Processing*, vol. 19, pp. 259-299, 1990.
- [10] H. V. Sorensen, M. T. Heideman, and C. S. Burrus, "On Computing the Split-Radix FFT", *IEEE Transactions Acoustic Speech Signal Processing*, vol. ASSP-34, no. 1, pp. 152-156, February 1986.
- [11] H. S. Hou, "A Fast Recursive Algorithm for Computing the Discrete Fourier Transform", *IEEE Transactions Acoustic Speech Signal Processing*, vol. ASSP-35, no. 10, pp. 1455-1461, October 1987.
- [12] S. C. Pei and J. L. Wu, "Split-vector Radix 2-D Fast Fourier Transform", *IEEE Transactions Circuits Systems*, vol. 34, no. 1, pp. 978-980, August 1987.
- [13] C. F. Wu, C. H. Chen, and M. T. Shiue, "A Design of Input-Decimation Technique for Recursive DFT/IDFT Algorithm", *IEEE Transactions on Circuits and Systems-I*, vol. 66, no. 12, pp. 4713-4726, December 2019.
- [14] S. Qadeer, M. Z. A. Khan, and S. A. Sattar, "A Radix-2 DIT FFT with Reduced Arithmetic Complexity", *IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI 2014)*, New Delhi, India, 24-27th September 2014.
- [15] G. Ganesh Kumar, S. K. Sahoo, and P. K. Meher, "50 Years of FFT Algorithms and Applications", *Circuits, Systems, and Signal Processing*, vol. 38, pp. 5665-5698, 2019.
- [16] P. Jain, D. Mallik, S. S. Gill, and P. Chawla, "ASIC Design of 4K-point RADIX-2 FFT in 0.18 μm CMOS Technology", *International Journal of Information Technology*, vol. 12, pp. 1427-1433, 2020.
- [17] R. Neuenfeld, M. Fonseca, and E. Costa, "Design of Optimized Radix-2 and Radix-4 Butterflies from FFT with Decimation in Time", *IEEE 7th Latin American Symposium on Circuits & Systems (LASCAS)*, 28 Feb – 2 March, Florianopolis Brazil, 2016.
- [18] M. Medina-Melendrez, M. Arias-Estrada, and A. Castro, "Input and/or Output Pruning of Composite Length FFTs Using a DIF-DIT Transform Decomposition", *IEEE Transactions on Signal Processing*, vol. 57, no. 10, pp. 4124-4128, June 2009.
- [19] F. Douskas and K. Pekmestzi, "On the Design of the FFT Butterfly Units", *IEEE 6th International Conference on Modern Circuits and Systems (MOCAS)*, Thessaloniki, Greece, 4-6 May 2017.
- [20] S. Ghissoni, E. Costa, C. Lazzari, and J. Monteiro, "Radix-2 Decimation in Time (DIT) FFT Implementation Based on a Matrix-Multiple Constant Multiplication Approach", *IEEE 17th International Conference on Electronics, Circuits and Systems*, Athens, Greece, 12-15 December 2010.
- [21] H. Murakami, "Real-valued Decimation-in-Time and Decimation-in-Frequency Algorithms", *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 41, no. 12, pp. 808-816, December 1994.

Mr. Bimal Pal is working as an Assistant Professor in Government College of Engineering and Ceramic Technology, Kolkata, India last 14 years, and he is having qualifications M.Tech, A.M.I.E, BSc. He is doing research in Digital Signal Processing, Noise Filtering, and Communications etc.



Dr. Pijush Kanti Bhattacharjee is associated with the study in Engineering, Management, Law, Indo-Allopathy, Herbal, Homeopathic, and Yogic Medicines. He is having qualifications Ph.D (Engg.), M.E, MBA, MDCTech, A.M.I.E (B.E or B.Tech), LLB, B.Sc, B.A, BIASM, CMS, PET, EDT, FWT, DATHRY, KOVID, DH, ACE, FDCI etc. He worked in Department of Telecommunications (DoT), Government of India as a Telecom Engineer from 1981 to 2007, lastly Assistant Director. Then he worked in different Engineering Colleges, India, and Assam University [Central University], Silchar, India, as an Assistant and Associate Professor from 2007 to 2020. He has written fourteen books and more than hundred research papers. He is a Member of IACSIT, Singapore; CSTA, UACEE, USA; IAENG, IETI, Hongkong; and IE, ISTE, IAPQR, IIM, India. His research interests are in Telecommunications including Mobile Communications, Image Processing, VLSI, Nanotechnology, Electrical Power Systems, Power Electronics Circuits, Environmental Pollution and Medicine.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)

This article is published under the terms of the Creative Commons Attribution License 4.0

https://creativecommons.org/licenses/by/4.0/deed.en_US