Algorithm for Convolution Operation in DFT using Vedic Multiplication

Neelisha Batham
neelishabatham@gmail.com

Prof. Shaista Anjum
shaianjum14@gmail.com

Abstract **– This paper discusses the power and area optimized digital multiplier using the vedic multiplication methodology for Fast Fourier Transform (FFT) application. This algorithm involves the concurrent addition and partial product generation which improves the computational time of multiplication. The ancient mathematic techniques of vedic multiplication uses 16 formulae which makes the *multiplication* operation easy and faster**.

Keyword **– Area and Power Optimization, Convolution, Fast Fourier Transform, Vedic Multiplier.**

I. Introduction

The digital multiplication process in fast Fourier Transform application is a complex and time consuming process. Also the area and power dissipation is the major design constraints in recent VLSI design. The design architecture is base on vedic Urdhava Tiryakbhyam multiplication. In this method the partial product of digit from two multiplicands and their sum is generated concurrently which improves the speed of multiplication. Further the use of deep submicron technology and transmission gate for CMOS layout design will improve the area and power dissipation of design as compare to conventional multiplication method [1,2].

II. Convolution

The convolution is the process of multiplication of two signals x(n) and h(n) generating the third signal that is the modified result of the original function, giving the area overlap between the two functions as a function of the amount that one of the original functions is translated. This is the mathematical operation particularly use in signal and system analysis. This operation is similar to the cross correlation . Its application is in analysis of probability, statistics, computer vision, image and signal processing, electrical engineering, and differential equations. The convolution operation is, circular convolution and linear convolution. The circular convolution can be defined for periodic functions i.e. mathematically this operation is perform on the circle. These generalizations of the convolution have applications in the field of numerical analysis and numerical linear algebra, and in the design and implementation of finite impulse response filters in signal processing. Linear convolution is a numerically intensive operation that can be efficiently implemented in the transform domain using the Discrete Fourier Transform (DFT). The analog input signal is sampled in to the finite number of discrete samples according to the sampling theorem. The number of discrete samples are assume to be L, representing a finite time record of the input signal.

The duration of the data record in seconds will be: $T_L = LT_s$

Where T_s is the sampling time,

The sampling frequency is the reciprocal of sampling time as $f_s = 1/T_s$.

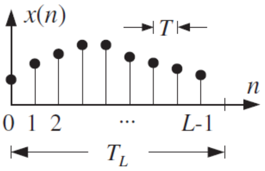


Fig 1. Discrete time signal

Thus the total number of discrete samples are for record of duration T_L seconds:
$L = T_L fs$

The L collected signal samples, say x(n), n = 0, 1, . . . , L-1, can be represented as:
X(n) = [x0, x1, . . . , xL-1]

The convolution of two signals for direct and linear time invariant system is written in equation as:

$Y(m) = \sum_{n=0}^{N-1} x(n)\, h(n-m)$

describe the filtering equation of an LTI system in general. An alternative way of writing these equations, called the convolution table form, is obtained by noting that the sum of the indices of h(m) and x(n-m) is m+(n-m) = n.

$Y(n) = \sum_{i\,j\,(i+j=n)} h(i)x(j)$

That is, the sum of all possible products h(i)x(j) with *i+j = n* [7].

III. Direct Form of Convolution

The two signal x(n) and h(n) represented in discrete time signals x(n) = {x(0), x(1), x(2),…….,x(L-1)}of length L and h(n) = {h(0), h(1), h(2),…….,h(N-1)} of length N are convolve by convolution equation. Equation shows that one signal is mathematically written in direct form and second signal is folded and delayed by m sampled.

The convolution of the length-L input X with the order-M filter h will output the sequence *Y(n).*

$$y(n) = \sum_{m}^{M} h(m)x(n-m)$$

$0 \le m \le M$ *For h(m)*
$0 \le n-m \le L-1$ *For x(n-m)*
$m \le n \le L-1+m$ or
$0 \le n \le L-1+M$ *For n*

Then output sequence y(n) is y = [y(0), y(1),.....,y(L-1+M)]. The length of convolution result signal is

$L_y = L_x + L_h - 1$. This represents the linear convolution of two signals [5,6,7].

Circular Convolution

Let us consider two input sequence
x(n) = [x(0), x(1), ..., x(L-1)] and
h(n) = [h(0), h(1), ... , h(M+1)]

The convolution of the length-L input X with the order-M filter h will output the sequence *Y(n).*

$$y(n) = \sum_{m}^{M} h(m)x(n-m)$$

$0 \le m \le M$ *For h(m)*

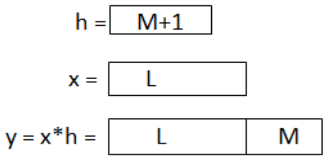
$0 \le n-m \le L-1$ *For x(n-m)*

$m \le n \le L-1+m$ or

$0 \le n \le L-1+M$ *For n*

Then output sequence y(n) is y = [y(0), y(1),.....,y(L-1+M)];

$L_y = L_x + L_h - 1$



From Equation (1.5.2(c))

$-(L-1) \le m-n \le 0$

Adding both n side $n-(L-1) \le m \le n$

m must satisfy simultaneously the inequalities

$0 \le m \le M$

$n-(L-1) \le m \le n$

From above relation *m* must be greater than the maximum of the two left- hand sides and less than the minimum of the two right- hand sides.

$$Max(0, n-L+1) \le m \le Min(n, M)$$

Then the Direct form of Convolution

$$y(n) = \sum_{m=\max(0,n-L+1)}^{\min(n,M)} h(m)x(n-m)\,;$$

n = 0,1,2, ...,L+M-1

Equation represent the linear convolution of input sequence x and *h* for n = 0, 1... L+M-1. Technically linear convolution gives an opportunity to calculate a L-point circular convolution of the two input sequence. The circular convolution of the L+M-1 point linear convolution calculate from given condition.

$$y_C(n) = y_0(n) + y_0(L+n)$$

n = 0, 1... L-2

Equations (1.5.2(f)) represent the Circular Convolution [5,6,7].

Vedic Multiplication Sutra

The vedic multiplication is base on the ancient Indian Vedic Mathematics algorithm of Urdhva Triyakbhyam. This is the vertical and crosswise multiplication method. This is called as multiplication sutras and use in all cases of multiplication operation. This algorithm involves the concurrent addition and partial product generation which improves the computational time of multiplication. This is like the parallel processing operation. The parallelism in generation of partial products and their summation is obtained using Urdhava Triyakbhyam which is shown in fig 2. For understanding of this method lets consider the multiplication of two decimal number. Consider decimal number 63 is multiplied with decimal number 56. The fig 2 shows the multiplication of decimal digits of both sides of line and added their carry from last step. The generated carry is added to the next step and process continues. If more than one line are there in one step, all the result are added to the previous carry. In each step, units place digit act as the result bit while the higher digit act as carry for the next step. Initially the carry is consider to be zero. In step 1the decimal digit 3 and 6 are multiplied. In step 2 and step 3 the crosswise multiplication is done. Carry of step 1 is added in this step 2 and step 3. In step 4 the decimal digit 6 and decimal digit 5 is multiplied and added with carry of previous step. This process is somewhat similar to the conventional multiplication process but concurrent partial product generation and addition operation of higher digit length number is makes this process faster [1,2].

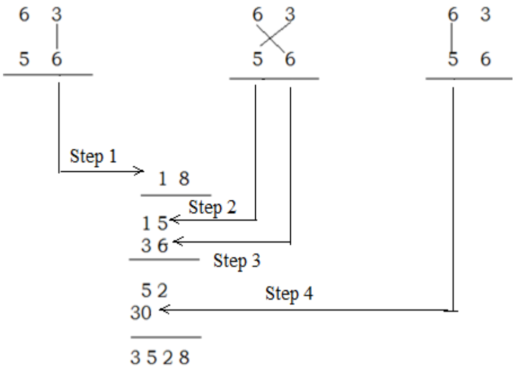


Fig. 2. Two Decimal Number multiplication by Vedic Method.

Same is the method use for binary number multiplication. Let us consider two four bit binary numbers on which multiplication operation is to be performing. Lets consider two 4 bit binary numbers, 1111 (decimal equivalent 15) and 1011 (decimal equivalent 11) for multiplication using vedic sutras shown in fig 3.

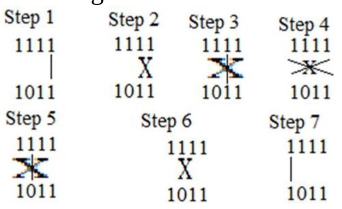


Fig. 3. Two 4 bit Binary Number multiplication by Vedic Method.

In step 1 the least significant bit of both binary numbers are multiplied nd generates the lsb bit of multiplication result. Then the LSB are multiplied crosswise with next bit in step 2 and added this product of multiplier cross wise.

The carry generates from this addition is use in next step. At step 3, the crosswise and vertical multiplication is performing for LSB and next two upper bits of multiplicand numbers. These products are added over again along with earlier step carry bit. The carry propagates to next step. At step 4 the all four bits are crosswise multiplied and added with previous carry. The sum is the corresponding product term and carry is again passes to the next step. At step 5, now the same step 3 is repeated but the bits are next three higher bits other than LSB. At step 6, the cross wise multiplication is performing for most significant bit and its previous lower bit. The product is obtain along with carry which is use in next step addition. At last step 7, the MSB bits are multiplied and product is added with previous carry. The carry may be generates in step 7 which is the MSB bit of final result[3,4].

The above method can be simplified by using two bit multiplication shown in fig 4.

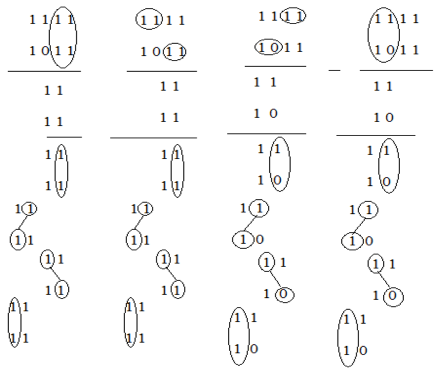


Fig. 4. Simplified implementation of 4 bit multiplication using 2 bit multiplier.

The algorithm design for multiplication of two 4 bit numbers uses the multiplication algorithm of two bit numbers. This shows that for the multiplication of N X N bit numbers a N/2 X N/2 bits multipliers are use. The processing speed of NXN number multiplication is increase due to the parallel computation of partial product generation and addition operation. This makes the multiplier in microprocessor independent of the clock frequency. The multiplier requires same time duration to calculate products and addition concurrently and hence is independent of the clock frequency. The Multiplier has the advantage that as the number of bits increases, gate delay and area increases very slowly as compared to other multipliers. Therefore it is time, space and power efficient. It is confirmed that this design is quite efficient in terms of chip area and computational speed .

The vedic mathematics Urdhva-Tiryagbhyam sutra are also applicable for all cases of multiplication and division operations of a large digit length numbers. The multiplication algorithm are also called as “vertically and crosswise” array multiplication algorithm.

Convolution Algorithm using Urdhava- Tiryagbhyam Sutras

The mathematical algorithm using vedic multiplication sutras for discrete sample multiplication in convolution operation is shown below. Here the multiplicand input sequence length is taken as two to four.

For the length of the input sequence is two, consider two discrete time signal x(n) = {x0,x1}i.e. L=2 and h(n) = {h0, h1} i.e. M=2. Then the convolution signal length is L+M-1 i.e. 3.

Length of output sequence y(n) is = L+M+1-1 = 3

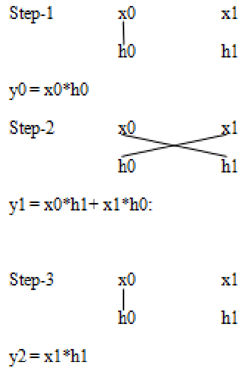


Fig. 5. Vedic multiplication of two bit multiplicands.

For the length of the input sequence is four, consider two discrete time signal x(n) = {x0,x1, x2, x3}i.e L=4 and h(n) = {h0, h1, h2, h3} i.e M=4. Then the convolution signal length is L+M-1 i.e 7.

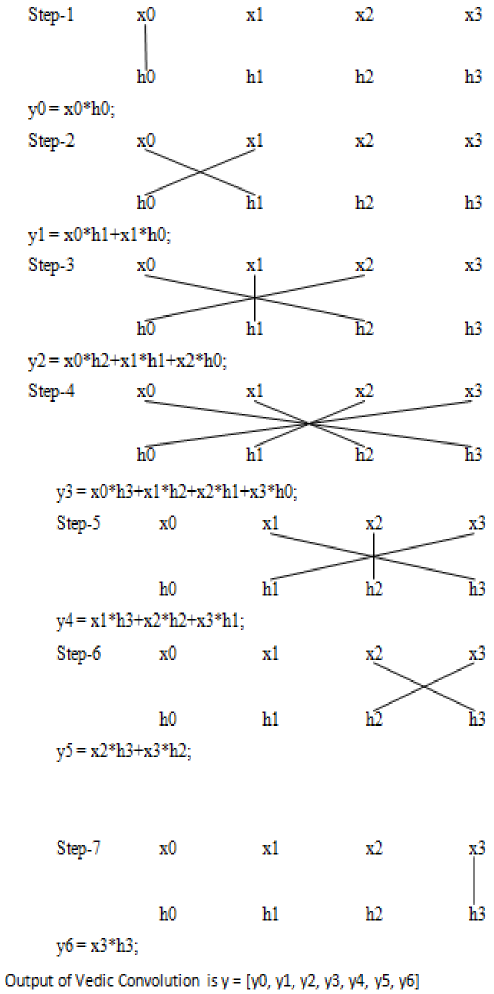


Fig. 6. Vedic multiplication of four bit multiplicands.

CONCLUSION

In this propose work, the different type multiplication operation use in convolution, circular convolution, cross correlation, auto correlation is possible using Vedic mathematics. multiplication methodology for Fast Fourier Transform (FFT) application. This algorithm involves the concurrent addition and partial product generation which improves the computational time of multiplication. The ancient mathematic techniques of vedic multiplication uses 16 formulae which makes the multiplication operation easy and faster.

REFERENCES

[1] Alex Pappachen James, Dinesh S. Kumar, and Arun Ajayan "Threshold Logic Computing: Memristive-CMOS Circuits for Fast Fourier Transform and Vedic Multiplication" IEEE Transactions On Very Large Scale Integration (Vlsi) Systems Jan 2015 Vol 10 issue 7.

[2] Yogita Bansal, Charu Madhu, Pardeep Kaur "High Speed Vedic Multiplier Designs a Review" Proceedings of 2014 RAECS UIET Panjab University Chandigarh, pp no. 06 – 08 March, 2014.

[3] D. Kayal & P. Mostafa & A. Dandapat & C. K. Sarkar "Design of High Performance 8 bit Multiplier using Vedic Multiplication Algorithm with McCMOS Technique" Springer Science+Business Media New York 28 jully 2013.

[4] Aravind E Vijayan, Arlene John, Deepak Sen "Efficient Implementation of 8-bit Vedic Multipliers for Image Processing Application" IEEE International Conference on Contemporary Computing and Informatics (IC3I) 2014 pp no. 544-550.

[5] K. K. Parhi, VLSI Digital Signal Processing Systems: Design and Implementation. New York, NY, USA: Wiley, 2007.

[6] John G Proakis, Dimitris G Manolakis : “Digital Signal Processing Principle, Algorithm, and Application” Printice Hall Third Edition.

[7] Sophocles J. Orfanidis "Introduction to Signal Processing" Rutgers University.