Discrete Fourier Transform (DFT)

by using

Vedic Mathematics

Shripad Kulkarni

Contents **Page No.**

ACKNOWLEDGMENT

It is a pleasure to acknowledge the contributions of all those who helped to make Seminar successful. Firstly, I would like to thank our Head of the Department, Dr.A.V. Deshmukh for giving us the opportunity to present a seminar on any topic of our interest.

I would like to thank my guide, Prof. D.K.Maghade, for giving valuable information and constant encouragement during the preparation of the seminar.

A special thanks to Dr. C.S. Warnekar, for giving me guidance as well as information during the preparation for this seminar.

I would also like to thank all fellow colleagues, who were a driving force for me in this endeavor.

PREFACE

In the present age of digital communication various audio /visual or any other perception signals are sampled on time [i.e. axis] and are quantized on amplitude [y-axis], to produce discrete version of the continuous signal .This results in the corresponding information being contained in a series of binary (0 & 1) sequence.

Hence any processing or transformation of original signal boil's down to suitable discrete mathematical operation applied to binary sequences. Different algorithm's exist to accomplish each of these tasks. The task themselves may include basic arithmetic operations like addition, subtraction, multiplication, division, matrix, squaring, exponential operations etc. While implementing these algorithms on digital computer, the prevalent VAN-NEUMAN architecture uses register operations like shift, move, compliment, add etc. to accomplish these basic arithmetic tasks.

The actual CPU implementation of these operations is through suitable amalgamation of algorithm and implementing architecture.

Though there are many algorithms for the same task only VAN-NEUMAN architectural implementation of classical method is found to be used in present day digital computers. The Vedic mathematical methods suggested by Shankaracharya Sri. Bharti Krishna Tirtha through his book offer efficient alternatives. The present seminar analyses and compares the implementation of DFT algorithm by existing and by Vedic mathematical technique . It is suggested that architectural level changes in the entire computation system to accommodate the Vedic mathematical method shall increase the overall efficiency of DFT procedure.

SURVEY OF LITRETURE

Vedic Mathematics by Shakaracharya Sri Bharati Krishna Tirtha. This book has been referred to understand modern application of Vedic mathematical algorithms. The Vedic mathematical procedures suggested by Shankaracharya Sri. Bharti Krishna Tirtha are fundamental in nature. They are very efficient in terms of memory and time usage. These techniques are equally applicable to various modern applications.

We site following examples appearing through literature in this context.

1) Dr. P.K Shrimani, professor and chair person Department of Computer Science and his colleague Mrs. Ranij of Banglore university have implemented Vedic Mathematical algorithms for Elliptic Curve Cryptography and have compared the time complexity of conventional and Vedic Maths methods.
2) Dr. Kailash and Ms. Priyanka Mishra of Department of Physics, Hamorpur, Haryana University, have developed a computer program for evaluation of recurring decimals using Vedic mathematical algorithm.
3) Dr. K. William in his book published by Motilal Banarasi Dass has exhibited application of Vedic Mathematical methods for astronomical calculation involving relative velocities.
4) Dr. S.K.S. Rai and Dr. Kailash have demonstrated the efficiency of Vedic Mathematical methods in classical factorization problem of mathematics.
5) The Computer Society of India COMMUNICATION article by Dr. Kailash appearing in the computer society of India Journals(VOL19 NO.4 PP 11 TO 14) also discuss Vedic mathematical algorithm for divisibility and recurring algorithms the same author has published simplified version through NCERT Publication for school sciences.
6) Himanshu Thapliyal and M.B Srinivas of Center for VLSI and Embedded System Technologies Hyderabad have proposes the **VLSI** hardware implementation of RSA encryption/decryption algorithm using the algorithms ofAncient Indian Vedic Mathematics that have been modified to improve performance. The results show that RSA circuitry implemented using Vedic division and multiplication is efficient in terms of area/speed compared to its implementation using conventional multiplication and division architectures.

Discrete Fourier Transform

Almost all branches of engineering and science use Fourier methods. The words *frequency*, *period*, *phase*, and *spectrum* are important parts of an engineer's vocabulary. The basic idea, the decomposition of signals into orthogonal trigonometric basis functions, is a natural and powerful tool, which is used in a vast number of applications.

When describing a digital system, the discrete time Fourier transform (DTFT) was introduced because it arises naturally as the frequency response function of a digital filter.

The discrete Fourier transform (DFT), occasionally called the finite Fourier transform, is a transform for Fourier analysis of finite-domain discrete-time signals. It is widely employed in signal processing and related fields to analyze the frequencies contained in a sampled signal, to solve partial differential equations, and to perform other operations such as convolutions.

Fourier transforms have many important applications in all branches of pure and applied mathematics. Given an arbitrary function f(t), the basic idea is to decompose that function (over some finite interval) optimally into a sum of pure sine waves, and to find the Frequencies, amplitudes, and phases of those waves.

DFT ANALYSIS:

The sequence of N complex numbers $x_0, ..., x_{N-1}$ is transformed into the sequence of N complex numbers $X_0, ..., X_{N-1}$ by the DFT according to the formula:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} \qquad k = 0, \dots, N-1$$

Where e is the base of the natural logarithm i is the imaginary unit($i^2 = -1$), and π is pi .

Consider the following sequence,

X{n} = {255,100,150,200,255,100,150,200}

We have,

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} \qquad k = 0, \dots, N-1$$

X(0) = x(0).1 + x(1).1 + x(2).1 + x(3).1 + x(4) .1 + x(5).1 + x(6).1 + x(7).1

X(0) = 255+ 100+ 150+ 200+255+100+150+200

X(0) = 1410

Number of multiplication 8

Number of multiplication; 7

Similarly if we compute for the X(1) ,X(2), X(3) ,X(4),X(5) ,X(6),X(7) sequences

Then total number of additions and multiplications are

Number of multiplication: 113

Number of additions: 56

Number of subtractions: 49

The number of multiplication and additions will increase as we go further for 16-bit sequence, 32- bit sequence. Of DFT.

We could see the possibility of improving the DFT computation on memory and time scale. For improving performance in DFT in memory and time scale using Vedic Mathematics.

Introduction to Vedic Mathematics (VM)

Vedic mathematics is part of four Vedas (books of wisdom). It is part of Sthapatya-Veda (book on civil engineering and architecture), which is an upa-veda (supplement) of Atharva Veda. It covers explanation of several modern mathematical terms including arithmetic, geometry (plane, co-ordinate), trigonometry, quadratic equations, factorization and even calculus. His Holiness Jagadguru Shankaracharya Bharati Krishna Teerthaji Maharaj. (1884-1960) comprised all this work together and gave its mathematical explanation while discussing it for various applications. Swamiji constructed 16 sutras (formulae) and 16 Upa sutras(sub formulae) after extensive research in Atharva Veda. Obviously these formulae are not to be found in present text of Atharva Veda because these formulae were constructed by Swamiji himself. Vedic mathematics is not only a mathematical wonder but also it is logical. That's why VM has such a degree of eminence which can not be disapproved. Due these phenomenal characteristics, VM has already crossed the boundaries of India and has become a leading topic of research abroad. VM deals with several basic as well as complex mathematical operations. Especially, methods of basic arithmetic are extremely simple and powerful. In this article, we have limited ourselves for multiplication methods and discuss application of these methods to DFT.

Some of the sutras discussed by Swamiji are listed below.

Vedic mathematics is based on sixteen sutras which serve as somewhat cryptic instructions for dealing with different mathematical problems. Below is a list of the sutras, translated from Sanskrit into English:

- By one more than the one before
- All from 9 and the last from 10
- Vertically and crosswise
- Transpose and apply
- If the Samuccaya is the same it is zero
- If one is in ratio the other is zero
- By addition and by subtraction
- By the completion or non-completion
- Differential calculus
- By the deficiency
- Specific and general
- The remainders by the last digit
- The ultimate and twice the penultimate
- By one less than the one before
- The product of the sum
- All the multipliers

Sub sutras or Corollaries

- Proportionately
- The remainder remains constant
- The first by the first and the last by the last
- For 7 the multiplicand is 143
- By osculation
- Lessen by the deficiency
- Whatever the deficiency lessen by that amount and set up the square of the deficiency
- Last Totaling 10
- Only the last terms
- The sum of the products
- By alternative elimination and retention
- By mere observation
- The product of the sum is the sum of the products
- On the flag

Multiplication in VM

Multiplication methods are extensively discussed in Vedic mathematics. Various tricks and short cuts are suggested by VM to optimize the process. These methods are based on concept of;

1] Multiplication using deficits and excess

2] Changing the base to simplify the operation

Various methods of multiplication proposed in VM are

1] UrdhvaTiryagBhyam – vertically and crosswise

2] Nikhilam navatashcharamam Dashatah: All from nine and last from ten

3] Anurupyena: Proportionately

4] Vinculum:

URDHVA TIRYAGBHYAM

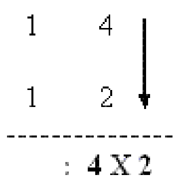
Urdhva – tiryagbhyam is the general formula applicable to all cases of multiplication and also in the division of a large number by another large number. It means Vertically and Crosswise.

We discuss multiplication of two, 2 digit numbers with this method.

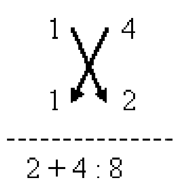
Ex.1.

Find the product of 14 and 12.

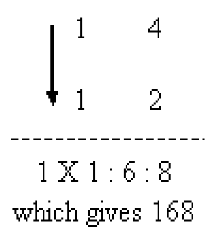
Step i)



Step ii)



Step iii)



Let us work another problem by placing the carried over digits under the first row and proceed.

```
     234
   x 316
---------------
   61724
   1222  ——— carry
----------------
   73944
```

Steps:

i) 4 X 6 = 24: 2, the carried over digit is placed below the second digit.

ii) (3 X 6) + (4 x 1) = 18 + 4 = 22; 2, the carried over digit is placed below third digit.

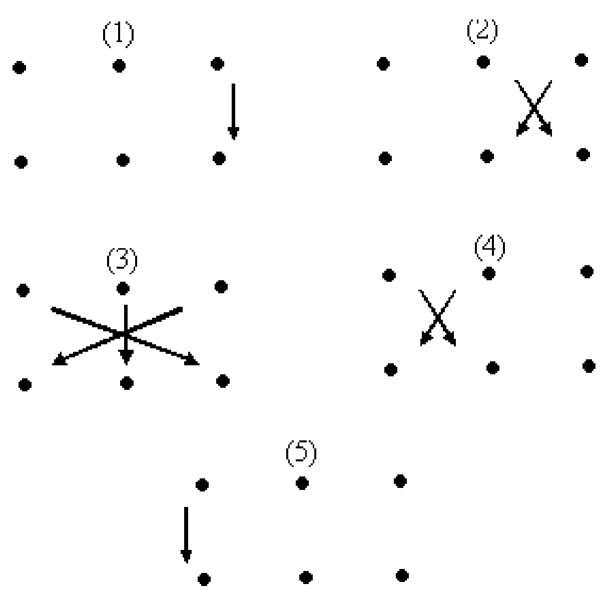
iii) (2 X 6) + (3 X 1) + (4 X 3) = 12 + 3 + 12 = 27; 2, the carried over digit is placed below fourth digit.

iv) (2 X 1) + (3 X 3) = 2 + 9 = 11; 1, the carried over digit is placed below fifth digit.

v) (2X3) =6.

vi) Respective digits are added.

General rule for a 3 digit by 3 digit multiplication:

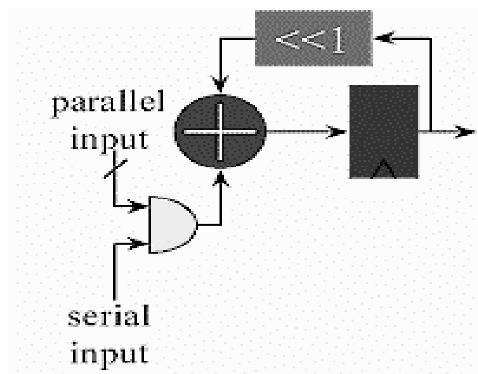


Shift and add method:

In the Van- Neumann architecture shift add algorithm is used for multiplication . Though there are many algorithm are developed for fast multiplication process Such as booth's algorithm etc .This is the most commonly used for multiplication.

A scaling accumulator multiplier performs multiplication using an iterative shift-add routine. One input is presented in bit parallel form while the other is in bit serial form. Each bit in the serial input multiplies the parallel input by either 0 or 1. The parallel input is held constant while each bit of the serial input is presented. Note that the one bit multiplication either passes the parallel input unchanged or substitutes zero. The result from each bit is added to an accumulated sum. That sum is shifted one bit before the result of the next bit multiplication is added to it.

```
1      1011001
0     0000000
1    1011001
1  +1011001
  10010000101
```



When number of multiplication increases then for the product of shift add routine will take time. If we have developed routine where number of multiplication are more by Vedic mathematics methods. It may be efficient which will reduce our CPU time and memory.

Comparison:

Normal method of multiplication and Vedic mathematics.

We will discuss an example in binary format.
[1] Find the product of 1 0 × 1 1

Normal Method:

```
    1   0
    ×
    1   1
    ----------
      1   0
+ 1   0   0
----------------------------
  1   1   0
```

Number of multiplication: 4

Number of additions: 2

Vedic mathematics (Urdhva – tiryagbhyam)**:**

```
    1   0
        ×
    1   1
    ----------
  1     1   0
```

Number of multiplication: 4

Number of additions: 1

[2] Find the product of 1 0 1 × 1 1 0

Normal method:

```
    1   0   1
    ×
    1   1   0
    -------------------
       0  0   0
      +
    1   0   1   0
  +
1  0   1   0   0
----------------------------------------
1  1   1   1   0
```

Number of multiplication: 9

Number of additions: 7

Vedic mathematics (Urdhva – tiryagbhyam)

```
1   0   1
×
1   1   0
-------------------
1   1   1   1   0
```

Number of multiplication: 9

Number of additions: 5

[3] Find the product of 1 0 1 0 × 1 0 0 1

Normal method:

```
              1 0   1   0

          x   1 0     0   1
   --------------------------------
              1   0    1   0
             +
            0  0   0     0   0
          +
        0  0    0    0    0    0
       +
     1   0  1    0     0    0    0
---------------------------------------------------------
     1   0   1   1     0     1  0
```

Number of multiplication: 16

Number of additions: 15

Vedic mathe matics (Urdhva – tiryagbhyam)**:**

```
        1   0   1   0

     x  1   0     0   1
   --------------------------------
  1   0   1   1    0  1  0
```

Number of multiplication: 16

Number of additions: 9

[4] Find the product of 1 0 0 0 0 0 0 0 × 1 0 0 0 0 0 0 0

Normal method;

1 0 0 0 0 0 0 0

x 1 0 0 0 0 0 0 0

100000000000000

Number of multiplication: 64

Number of additions: 77

Vedic mathe matics (Urdhva – tiryagbhyam)**:**

1 0 0 0 0 0 0 0

X 1 0 0 0 0 0 0 0

100000000000000

Number of multiplication: 64

Number of additions: 53

Comparison Table

Normal Method	Vedic method
For 2 bit multiplication Number of multiplication: 4 Number of additions: 2	For 2 bit multiplication Number of multiplication: 4 Number of additions: 1
For 3 bit multiplication: Number of multiplication: 9 Number of additions: 7	For 3 bit multiplication: Number of multiplication: 9 Number of additions: 5
For 4 bit multiplication: Number of multiplication: 16 Number of additions: 15	For 4 bit multiplication: Number of multiplication: 16 Number of additions: 9
For 8 bit multiplication Number of multiplication: 64 Number of additions: 77	For 8 bit multiplication: Number of multiplication: 64 Number of additions: 53

Conclusion

It can be easily observed from the above comparison table that Vedic mathematical multiplication process is very efficient. Implementation of Vedic multiplication will be more efficient in terms of its implementation using conventional multiplication process.

Looking ahead

1] Vedic mathematics deals with various topics of mathematics such as basic arithmetic, geometry, trigonometry, calculus etc. All these methods are very efficient as far as manual calculations are concerned.

2] If all those methods are effectively implemented in computers, it will reduce the computational speed drastically. Therefore, it could be possible to implement a complete ALU using all these methods using Vedic mathematics methods.

References

- Books:
- Shakaracharya Sri Bharati Krishna Tirtha ”Vedic Mathematics “–chapter 3 (32-34)

- S. Zakhy -Computer Organization and Architecture
    - chapter 6(271-273),edition-4 ,1996
- Prokis ,”Digital Signal Processing”-
- Sites:
- Vedic mathematics - Wikipedia, the free encyclopedia