

Power-Time Efficient Algorithm for Computing Reconfigurable FFT in Wireless Sensor Network

Anuj Kumar Varshney¹, Vrinda Gupta²

Department of Electronics and Communication Engineering
National Institute of Technology, Kurukshetra, Haryana
Kurukshetra 136119, India

¹anuj.varshney83@gmail.com,

²vrindag16@yahoo.com

Abstract—Wireless sensor network have several constraints, such as a short transmission range, poor processing capabilities, limited available power and slow execution time. The design of algorithms for fast processing of information in sensor network is an emerging research area nowadays. However, due to unique characteristics of the sensor network computing paradigm, the objective is to design algorithms that meet two contradicting goal of low power and fast execution time. In this work, a power-time efficient algorithm is proposed which is implemented with the help of Vedic mathematics for computing FFT (Fast Fourier Transform) over data distributed across smart sensors. The word “Vedic” is derived from the word “veda” which means the store-house of all knowledge. Vedic mathematics is mainly based on 16 Sutras (or aphorisms) dealing with various branches of mathematics like arithmetic, algebra, geometry etc. Significant reduction in power requirement and delay has been achieved based on the proposed approach.

Keywords- Wireless Sensor Network, FFT, Vedic Mathematics

I. INTRODUCTION

Wireless sensor network are gaining immense importance in today's world with respect to innumerable number of application in which they can be used. These find applications in military surveillance, environment monitoring, inventory management, habitat monitoring, health monitoring etc. One basic application of wireless sensor network is in Vehicle tracking and its localization. There has been many algorithm proposed to overcome power and execution time in the wireless sensor network. For example; suppose a vehicle is moving over a region where a network of acoustic sensing nodes has been deployed to determine the location of the vehicle then the first thing to be done is to find the LOB (Line of Bearing) or direction from which sound is being detected. Fig. 1 shows the scenario for vehicle tracking using LOB estimation.

The intersection of multiple LOB's determines the source location. To perform LOB estimation, generally beam forming algorithm is used. There are two possible ways to partition the computation for LOB estimation as shown in Fig. 2. The first is the direct technique, where each computation (FFT and Beam forming) is done at the cluster head while the second is the distributed technique in which the FFT is distributed among all sensors.

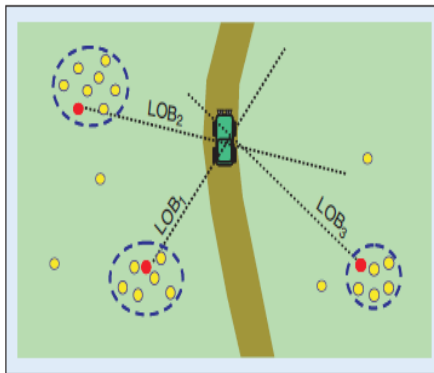


Figure 1. Line of bearing estimates to determine the vehicle's location [1]

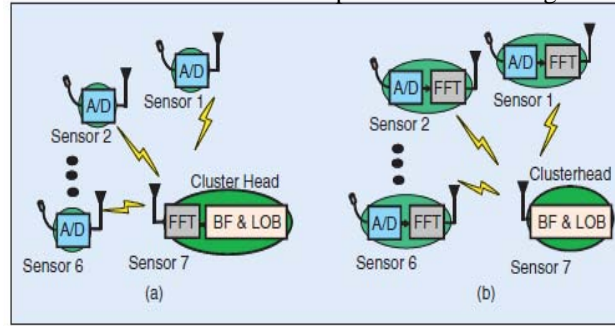


Figure 2. (a) Direct Technique, and (b) Distributed Technique [1]

The FFT has been studied extensively as a frequency analysis tool in DSP. The FFT is faster version of DFT (Discrete Fourier Transform). It utilizes clever algorithms to do the same thing as the DFT, but in much less time. The DFT is extremely important in the area of frequency (spectrum) analysis as it takes a discrete signal in the time domain and transforms that signal into its discrete frequency domain representation. Without a discrete-time to discrete-frequency transform it is not possible to compute the Fourier transform with a microprocessor or DSP based system [1]. The DFT representation of a discrete time signal $X(nT)$ is:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} \quad (1)$$

where T = Sampling Period, N = Frame Length, $K = 0, 1, 2, 3, \dots, (N-1)$, here it represents the $N \times N$ matrix. For N values of K the total number of complex multiplications is N^2 and total number of additions is $N(N-1)$.

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk} \quad (2)$$

where T = Sampling Period, $W_N = e^{-j2\pi/n}$ = Twiddle constant or factor. FFT is an algorithm that efficiently computes DFT. The advantage of FFT over DFT is symmetry and periodicity property as given by (3) and (4).

By symmetry property:

$$W_N^{k+N/2} = -W_N^k \quad (3)$$

By periodicity property:

$$W_N^{k+N} = W_N^k \quad (4)$$

Radix-2 Decimation In Frequency FFT: According to divide and conquer approach, this depends on the decomposition of an N -point DFT into successively small size DFT's. If $N = r_1 \cdot r_2 \cdot r_3 \cdot \dots \cdot r_L$. But, if $r_1 = r_2 = r_3 = \dots = r_L = r$, then $N = r^L$. This number r is called radix of the FFT algorithm. Decimation in frequency FFT decomposes the DFT by recursively splitting the sequence elements $X(K)$ in the frequency domain into sets of smaller and smaller sequences.

Vedic Mathematics, The word "Vedic" is derived from the word "veda" which means the store-house of all knowledge. The hymns of the Rig Veda are considered the oldest and most important of the Vedas, having been composed between 1500 BC. The gift that the Indians gave to the world, thousands of years ago, and which is now currently employed in our global silicon chip technology, was none other than the invention of zero and decimal points. The concept of zero has been evidently described in Pingalacharya's Chanda Shastra, which dates back to 200 B.C. using Sanskrit couplets. Earliest mention in Anuyogadvaara Sutra (100 BC) is as follows "The total number of human beings in the world, when expressed in terms of denominations, occupies 29 places". Sanskrit scholars translated these vedic documents and were really surprised at the depth and breadth of the knowledge contained by them. Some major texts like "Ganita Sutras" (In Sanskrit, the ancient language of India, the word "Sutra" means "Thread of Knowledge") were devoted to mathematical knowledge, which mainly addressed the geometry of construction of sacrificial altars, geometrical figures such as straight lines, rectangles, circles and triangles. Many mathematical methods described in the Vedas were previously unknown and created great amazement among scholars. Some of the ancient texts were lost during the course of time due to various untold reasons. Thus the astonishing system of calculation born in the Vedic Age was deciphered during the start of the 20th century. During 1911 and 1918 Sri Bharathi Kirshna Tirtha (1884-1960) [6], a mathematician and the pontiff of the famous Sankara Mutt in India, carried out research by interrelating various branches of mathematics and the existing ancient Sanskrit text. After an extensive study, he proposed 16 algorithms (named as "sutras" in Sanskrit) for easier and faster calculations. The most astonishing feature in the Vedic mathematics is its

coherence. Instead of a mixture of unrelated techniques, the entire system is marvelously interconnected and fused. Vedic Mathematics a system that is far simpler and more enjoyable than modern mathematics. The simplicity of Vedic Mathematics means that calculations can be carried out mentally though the various methods. Vedic mathematics refers to the technique of calculations based on a set of 16 Sutras, or aphorisms, as algorithms and their upa-sutras or corollaries derived from these Sutras. Any mathematical problems can be solved mentally with these sutras. It offers a fresh and highly efficient approach to mathematics covering a wide range that starts with elementary multiplication and concludes with a relatively advanced topic, i.e. the solution of non-linear partial differential equations. But, the Vedic scheme is not simply a collection of rapid methods; it is a system, a unified approach. The 16 sutras of Vedic mathematics are given in the table I, but the discussion of all the algorithms is beyond the scope of this paper. The rest of paper is organized as follows. In section II, the proposed technique is described which is focused on power reduction and fast execution time of FFT over wireless sensor network. Section III covers the result and discussion part followed by conclusion in section IV.

II. PROPOSED TECHNIQUE

A. Urdhva Tiryagbhyam

The basic sutras and upa sutras in the Vedic mathematics helps in solving mostly all the numeric computations in easy and less time. The sutra which we employ in this paper is Urdhva Tiryagbhyam (means multiplication). 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, i.e. vertically and crosswise. The Fig. 3 shows the steps involved in multiplication of two 4-bit binary numbers.

TABLE I. 16 SUTRAS OF VEDIC MATHEMATICS

No	Vedic Sutras in Sanskrit	English Meaning
1.	एकाधिकेन पूर्वेन	By one more than the one before.
2.	निखिलं नवतश्चरमं दशतः	All from 9 and the last from 10.
3.	ऊर्ध्वतिर्यग्भ्याम्	Vertically and Cross-wise
4.	परावर्त्य योजयेत्	Transpose and Apply
5.	शून्यं साम्यसामुच्चये	The summation if equal to Zero
6.	आनुष्ये शून्यं अन्यत्	If One is in Ratio the Other is Zero
7.	संकलन व्यवकलनाभ्यां	By Addition and by Subtraction
8.	पूरणापूरणाभ्यां	By the Completion or Non-Completion
9.	चलनकलनाभ्याम्	Differential Calculus
10.	यावदूर्नं	By the Deficiency
11.	व्यष्टिसमष्टिः	Specific and General
12.	शेषाण्यडेन चरमेण	The Remainders by the Last Digit
13.	सोपान्त्यद्वयमन्तर्यं	The Ultimate and Twice the Penultimate
14.	एकन्दूनेन पूर्वेन	By One Less than the One Before
15.	गुणितसमुच्चयः	The Product of the Sums
16.	गुणकसमुच्चयः	All the Multipliers

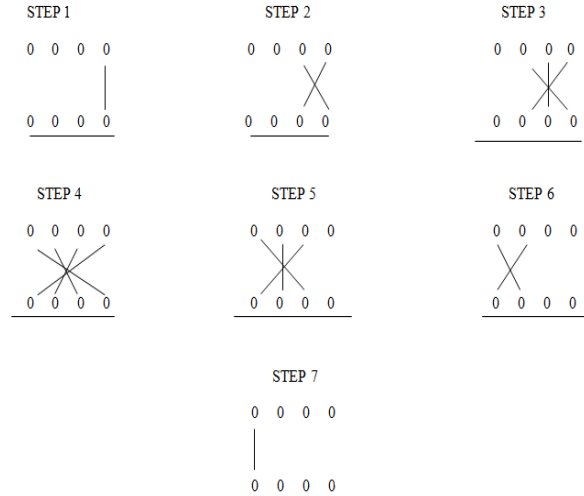


Figure 3. Multiplication of two 4- bit binary number

In the Vedic multiplication algorithm of binary number system, the multiplication of two bits a_0 and b_0 is just an AND operation and can be implemented using simple AND gate. To illustrate this multiplication scheme in binary number system, consider the multiplication of two binary numbers $a_3a_2a_1a_0$ and $b_3b_2b_1b_0$. As the result of this multiplication would be more than 4-bits like in form of $r_3r_2r_1r_0$. Line diagram for multiplication of two 4-bit numbers is shown in Fig. 3. For the sake of simplicity, each bit is represented by a circle. Least significant bit r_0 is obtained by multiplying the least significant bits of the multiplicand and the multiplier. As in the last case, the digits on the both sides of the line are multiplied and added with the carry from the previous step. This generates one of the bits of the result (r_n) and a carry (say C_n).

This carry is added in the next step and hence the process goes on. If more than one line are there in one step, all the results are added to the previous carry. In each step, least significant bit acts as the result bit and the other entire bits act as carry. For example, if in some intermediate steps encounter 110, then 0 will act as result bit and 11 as the carry (referred to as C_n in this text). It should be clearly noted that C_n may be a multi-bit number. Thus expressions formed:

$$\text{Step1: } r_0 = a_0b_0,$$

$$\text{Step2: } c_1r_1 = a_0b_1 + a_1b_0,$$

$$\text{Step3: } c_2r_2 = c_1 + a_2b_0 + a_1b_1 + a_0b_2,$$

$$\text{Step4: } c_3r_3 = c_2 + a_3b_0 + a_2b_1 + a_1b_2 + a_0b_3,$$

$$\text{Step5: } c_4r_4 = c_3 + a_3b_1 + a_2b_2 + a_1b_3,$$

$$\text{Step6: } c_5r_5 = c_4 + a_3b_2 + a_2b_3,$$

$$\text{Step7: } c_6r_6 = c_5 + a_3b_3$$

With $c_6r_6r_5r_4r_3r_2r_1r_0$ being the final product. Partial products are calculated in parallel and hence the delay involved is just the time it takes for the signal to propagate through the gates. The hardware architecture is shown in Fig. 4.

B. Reconfigurable FFT

The basic architecture of reconfigurable FFT is to be shown in Fig. 5. It can calculate Fourier Transform either 2-point or 4-point depending upon the select line which is to be given by the programmer. Its calculate 2-point and 4-point FFT without changing the hardware of the system. This particular reconfigurable FFT is to be designed by using Vedic multiplier. Fig. 6 shows the butterfly diagram of 4-point FFT. If take 4-bit as an input of the FFT then for 4-bit multiplication, the total number of multiplier is 16 and total number of adder is 15 but in the case of Vedic method the total number of multiplier is 16 and total number of adder is 9, the gate count is less in case of Vedic method. So, the power and delay produced by the Vedic reconfigurable FFT is smaller than the power and delay produced by the conventional reconfigurable FFT.

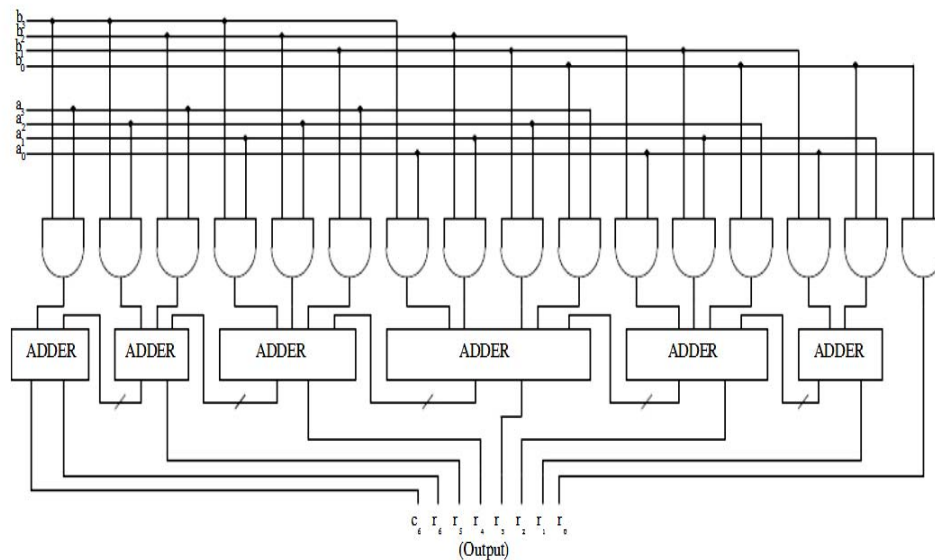


Figure 4. Hardware architecture of multiplication of two 4-bit binary number

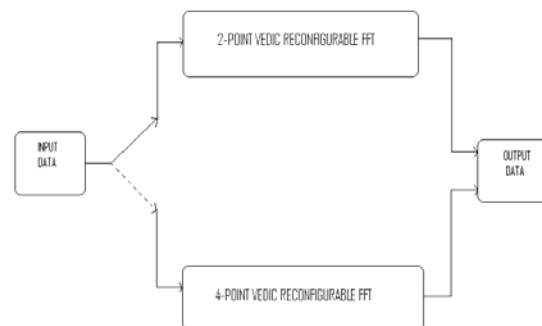


Figure 5. Reconfigurable architecture using 2-Point and 4-Point FFT

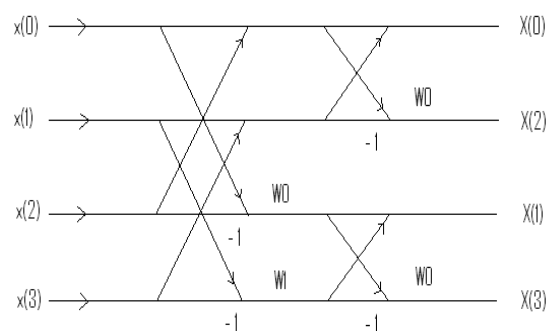


Figure 6. Butterfly diagram of 4-Point FFT

III. RESULTS AND DISCUSSION

The complete work has been carried out in Verilog HDL (Hardware Description Language). The coding is done in ISE 8.2i of Xilinx that uses XST tool for generating synthesis report while verification of code has been done using simulator, Modelsim SE and Design-Compiler. Table 2 shows the comparison between conventional reconfigurable FFT and the Vedic reconfigurable FFT for wireless sensor network.

TABLE 2: COMPARISON OF PROPOSED WORK WITH CONVENTIONAL APPROACH

Comparison Reconfigurable FFT		
	Conventional	Vedic
Power	13.248mW	9.923mW
Delay	15.568ns	14.421ns

From table 2, it is clearly observed that there is reduction in power and delay of the Vedic reconfigurable FFT in comparison with conventional reconfigurable FFT over wireless sensor network. The bar graph of corresponding result of multiplier for conventional method and proposed technique is shown in Fig. 7 (a) and (b) respectively. The simulation result corresponding to conventional and proposed technique is shown in Fig. 8 (a) and (b) respectively.

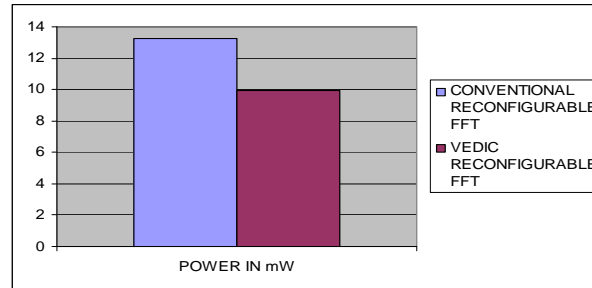


Figure 7. (a) Bar graph of conventional method

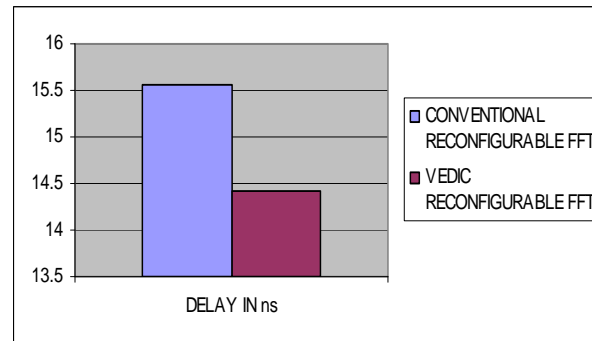


Figure 7. (b) Bar graph of proposed technique

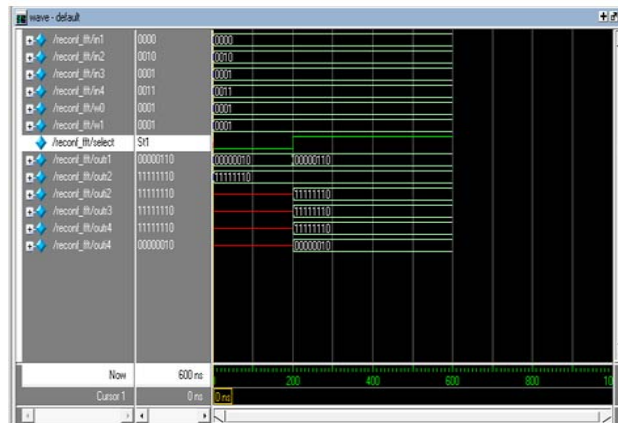


Figure 8. (a) Simulation result of conventional method

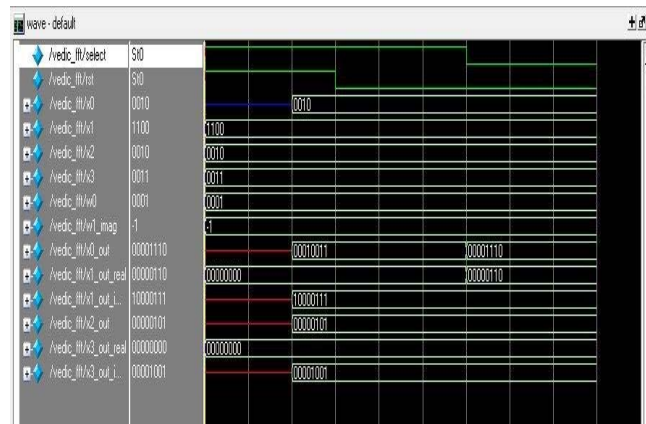


Figure 8. (b) Simulation result of proposed technique

IV. CONCLUSION

The proposed work concentrates on the study of architecture of reconfigurable FFT using conventional multiplier and Vedic multiplier for achieving high performance of the Vedic reconfigurable FFT in wireless sensor network. The comparison has made in terms of power and delay of architecture of reconfigurable FFT. Finally, the result shows that the proposed reconfigurable FFT with low power and small delay in comparison with the conventional reconfigurable FFT. The proposed architecture can find applications in various low power demanding products like cell phones, any Wi-Fi communication products etc.

REFERENCES

- [1] Alice Wang, Anantha Chandrakasan, "Energy-Efficient DSPs for wireless sensor network", IEEE signal processing magazine, jul.2002, pp.68-78.
- [2] Turkmen Canli, Mark Terwilliger, Ajay Gupta, Ashfaq Khokhar, "Power - Time Optimal Algorithm for Computing FFT over Sensor Networks", Proceeding International Conference, SenSys'04, Nov.2004,
- [3] Tze yun sung, Hsi chin hsin, Lu tingko "Reconfigurable VLSI architecture for FFT Processor" IEEE Trans. on circuit and system, jun. 2009. issue 6, vol. 8, pp. 44-46.
- [4] P. Mehta, D. Gawali, "Conventional versus Vedic Mathematical Method for Hardware Implementation of a Multiplier," Proceeding International Conference on Advances in Computing, Control, & Telecommunication Technologies, 2009. ACT '09, pp-640-642.
- [5] Ali Al Ghouwayel, Y. Louet, J. Palicot, "A reconfigurable architecture for the FFT operator in a software radio context," Proceedings. IEEE International Symposium on Circuits and Systems, ISCAS 2006, pp.181-184.
- [6] H. Thapliyal and M. B. Srinivas, "High Speed Efficient N×N Bit Parallel Hierarchical Overlay Multiplier Architecture Based on Ancient Indian Vedic Mathematics", Enformatika Trans., Dec. 2004. vol. 2, pp. 225-228,
- [7] J. Palicot, C. Roland, "FFT: a basic Function for a Reconfigurable Receiver", ICT' 2003, Feb. 2003, Papeete, Tahiti.
- [8] Jagadguru Swami Sri Bharati Krsna Tirthji Maharaja, "Vedic Mathematics", Motilal Banarsidas, Varanasi, India, 1986.
- [9] Vedic Mathematics [Online]. Available: <http://www.hinduism.co.za/vedic.htm>.
- [10] Anvesh kumar, Ashish raman, Dr.R.K.sarin, Dr.Arun Khosla, "Small area Reconfigurable FFT Design by Vedic Mathematics", IEEE 2010, vol. 5, pp. 836-838.
- [11] L. Liu and J.-F. Chamberland. "Cross-Layer Optimization and Information Assurance in Decentralized Detection over Wireless Sensor Networks." Asilomar Conference on Signals, Systems and Computers, Pacific Grove, California, Oct.2006.