International Journal of Engineering Technology and Applied Science
(ISSN: 2395 3853), Vol 1 Issue 3 June 2015

A New Robust Reduced-Bit Multiplication Algorithm by using Vedic Mathematics

Miss. Deepika Vyas[1], Mrs. Neha Jain[2]
M.tech Student[1], Assistant professor[2]
Shri Ram Institute of Technology, (SIRT) Jabalpur[1, 2]
deepika.vyas1987@gmail.com[1], jain.neha71@gmail.com[2]

Abstract: In this research article we've introduced a brand new fast, robust multiplication methodology on the idea of Vedic arithmetic. Basic and also the core of all the Digital Signal Processors (DSPs) are a unit its multipliers and speed of the DSPs is especially determined by the speed of its multipliers. That multiplication is the most basic operation with intensive arithmetic computations. Nikhilam Sanskrit literature, one among the Multiplication sutra of Vedic arithmetic is economical in multiplying massive decimal numbers because it reduces multiplication of two massive decimal numbers to two smaller numbers. The planned repetitious algorithmic rule is taken from Nikhilam sutra and is more optimized by use of dropping least important zeros of the binary numbers and activity bit shifting to require the advantage of bit reduction in multiplication. The reduced bit multiplication algorithmic rule supported the Vedic multiplication formula planned during this paper that's Nikhilam sutra. We have a tendency to beinitially mentioned thoroughly. Nikhilam sutra, being a general multiplication formula, is equally applicable to all or any cases of multiplication. It's applied to the arithmetic and is shown to yield a number design that is incredibly, almost like the favored current number. Nikhilam sutra, on the opposite hand, is additional economical within the multiplication numbers because it reduces the multiplication of two large numbers to it of two smaller numbers. The framework of the planned algorithmic rule is taken from this sutra and is more optimized by use of some general arithmetic operations like enlargement and bit-shifting to require advantage of the bit-reduction in multiplication.

Keywords:Vedic multiplier architecture, Nikhilam sutra architecture,

I Introduction

The Vedic arithmetic is an extract from the four Vedas (books of wisdom). It's truly a sub module of "Sthapatya-veda" (book on engineering and architecture), that is an upya-veda (supplement) of Atharva Veda. Due to its simplicity and regularity, it finds its utility and applications within the fields of pure mathematics, trig, quadratic equations, factorization and calculus. His sanctity Jagadguru Shankaracharya Bharati Krishna Teerthaji maharajah (1884-1960) comprised all this work. He did an in depth analysis within the Vedas and came up with the simplified sort of calculations that are nonetheless thus powerful. He came up with complete explanations and bestowed them within the type sutras. He created sixteen sutras (formulae) and sixteen Upya sutras (sub formulae) once intensive analysis in Atharva sacred writing. As of these sutras were extracts from swamiji's own findings and analysis, these don't seem to be found expressly within the sacred writing. 'Vedic' may be a term derived from the word 'Veda' which implies the store-house of all information. The facility of religious writing, arithmetic isn't solely confined to its simplicity, regularity, however conjointly it's logical. Its high degree of eminence is attributed to the aforesaid facts. it's these extraordinary characteristics, that created religious writing arithmetic, become thus well-liked and therefore it's become one in every of the leading topics of analysis not solely in India however abroad similarly. Vedic writing mathematics' logics and steps will be directly applied to issues involving pure mathematics functions, plane and sphere pure mathematics, conics, the calculus, infinitesimal calculus and applied math of varied kind. The awe hanging and therefore the amazing feature of religious writing, arithmetically within the undeniable fact that it simplifies the difficult trying calculations in standard arithmetic to an easy one in an exceedingly a lot of quickness and economical manner. This is often attributed to the actual fact that the religious writing formulae are claimed to be supported the "natural principles on that the human mind works". Thus, this presents some effective algorithms which might be applied to varied branches of engineering.

The multiplier factor is one in every of the elemental hardware blocks in several Digital Signal process systems for playing different operations like frequency domain

filtering FIR, IIR, frequency transformations (FFT), etc., The multiplier factor, being the foremost important block in several such digital systems, their speed and potency is primarily dependent upon the speed, area, the output potency of the multipliers enforced in these systems. The other feature of the multiplier factor that has got to tend quantitative concern in coming up with of the systems is Power Dissipation, viz. The multiplier factor may be a supply of high power dissipation. In the next section we have focused on the "Vedic Mathematics.

II Vedic Mathematics

The word "Vedic" is a Sanskritic language word derived from the word "Veda" which suggests the gathering of all information. Veda may be a gift from ancient sages of Bharat to the present world. From the traditional time Vedas were passed from previous generation to next generation orally instead of written. Vedic writing arithmetic is principally supported sixteen Sutras (or aphorisms) managing numerous branches of arithmetic like arithmetic, algebra, pure mathematics etc. These Sutra's meanings with few words noncommissioned below alphabetically:

1. (Anurupye) Shunyamanyat – if one is in quantitative relation, the opposite is zero.
2. Chalana-Kalanabhyam – variations and similarities.
3. Ekadhikina Purvena – By an additional than the previous one.
4. Ekanyunena Purvena – By one less than the previous one.
5. Gunakasamuchyah – The factors of add is up to add of the factors.
6. Gunitasamuchyah – the product of add is up to add of the product.
7. Nikhilam Navatashchramam Dashatah – All from nine and last from ten.
8 ParvaryaVojayet – Transpose and modify.
9. Purnapurnabhyam – By the completion or no completion.
10. Sankalana- Vyavakalanabhyam – By addition and by subtraction.
11. Shesanyankena Charamena – The reminders by the last digit.
12. Shunyam Saamyasamuchyah – once add is that the same that add is zero.
13. Sopaantyadvayamantyam – the ultimate and doubly the penultimate.
14. Urdhva- Tiryagbhyam – Vertically and crosswise.
15. Vyashtisamanstih – half and whole.
16. Yaavadunam – regardless of the extent of its deficiency.

The attractor of Vedic arithmetic lies within the proven fact that is reduces the complicated wanting calculations in typical arithmetic to a really easy one. This can be therefore as a result of the religious writing formulae area unit claimed to be supported the natural principles and sutra on that the human mind works. This can be a really interesting field and presents some effective algorithms which might be applied to numerous branches of engineering like computing and digital signal process [5]. We've got to focus on sizable amount multiplication. If we would like to try and do the big variety multiplication the suable technique that's "Nikhilam sutra". [9]. Here we have focused on large number multiplication. If we want to do the large number multiplication the suable method that is "Nikhilam sutra"..

III Proposed Method

In this section we have introduced a new method that merges with Nikhilam sutra. Here first we discuss the Nikhilam sutra then we defend dsp operation with Nikhilamsuta. If we merge the Nikhilam sutra method in DSP the result is faster, rather than normal DSP operation because Vedic perform faster than normal DSP.

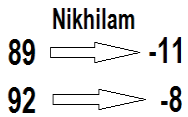
The formulae simply means: "all from 9 and the last from 10". The formula can be very effectively applied in multiplication of numbers, which are nearer to bases like 10, 100, 1000 i.e., to the power of 10. The procedure of multiplication using the Nikhilam involves a minimum number of steps, time saving and only mental calculation. The numbers taken can be either less or more than the base considered. The difference between the number and the base is termed as a deviation.

Nikhilam Dasatah literally means ―All from Nine and the last from Ten. The sutra basically means start from the left most digit and begin subtracting _9' from each of the digits; but subtract 10' from the last digit.

The following example illustrates the way in which this Sutra could reduce the number of iterations to reduce the whole Multiplication.

Explanation of Nikhilam Sutra

To multiply 92 and 89. Apply Nikhilam Sutra All from nine and last from ten‖ on both the numbers



The arrows in Figure 2 indicate the operation of the Nikhilam Sutra being performed, viz. The subtraction of

10 from the last digit and 9‘s from all the other digits starting with the leftmost digit.

- Now we write this down side-by-side,

92	-08
89	-11

- Multiply (-08) and (-11) to get = 88‘

92	-08	x
89	-11	
	88	

.

- Now we cross-add. This is done by both adding 92 and -11 to get 81‖or ―adding 89 and –08 to get 81.

- Note that in both operations you get the same answer that is = 81‘which is written below to get the solution

This technique works very well if the numbers to be multiplied are near a base. Upon little alteration, this also works very well for any pair of numbers.
Nikhilam Navatascharam Dashtah means all from 9 and last from 10. This algorithm works for all numbers, but it works efficiently for larger numbers. Since it finds out the complement of the large number from its nearest base to perform the multiplication operation on it, the larger is the original number, lesser the complexity of the multiplication. The procedure for Nikhilam Sutra is-

1. Take the base of calculation as a power of 10 which is nearest to the multiplicands say M and N.
2. Subtract base B from each multiplicand and note two remainders as say M and N.
3. Column 1 contains the numbers and Column 2 contain difference from the nearest base.
4. The product will have two parts. Right part (R) is obtained by multiplication of two remainders namely m and n .i.e. R = m x n

5. Left part (L) can be obtained by cross subtracting the second number of Column 2 from the first number of column 1 or vice versa,

The answer is obtained by just concatenating left and right part as LR. The improvement of these methods over conventional multiplication largely depends on closeness of multiplicands to the power of 10.
The illustration of this sutra can be given through three cases as:
Case 1: When both the deviations are positive
Let M & N are 12 and 14. These two numbers are nearer to he base 10 with deviation of 2 and 4 respectively Add one of the numbers to the deviation of the other. Place these numbers side by side: 168

Number	Deviation with Base (1000)
997	(997 – 1000) = -3
1015	(1015 – 1000) = 15
	(15 * -3) = -45

Product of

Number	Deviation from Base (10)
997	-3
1015	15
(997 +15) or (1015-3) =1012	(-3 * 15)= -45

Sum

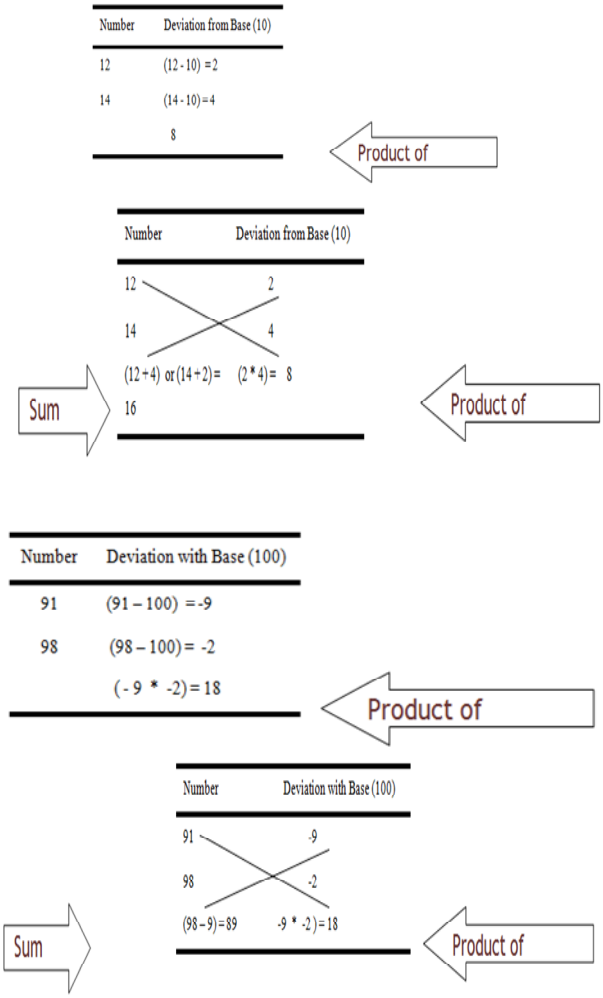
Product of

Case 2: When both the deviations are negative

Let M & N are 91 and 98. These two numbers are nearer to the base 100 with deviation of -9 and -2. Add one of the numbers to the deviation of the other. Place these numbers side by side: 8918

Case 3: When both the deviations have different signs. Let m & n are 997 and 1015. These two numbers are nearer to the base 1000 with deviation of -3 and 15. Add one of the numbers to the deviation of the other. The sum and the product have different signs, so they cannot be placed side by side to get the answer. Hence, to get the answer subtract

45 from 101200 after adding two zeros to sum. i.e (101200 - 45 = 1011955)



Algorithm –

In this we have show who to implement our work on the MATLAB programming. For the simulation and other calculation we have needed a one algorithm that is shown in below-

Algorithm :NikhilamSquaring (*A*)
INPUT: *A* = Pn−1 *a xi*
OUTPUT: *B* = *A* ∗*A* = P2n−1 *b xk*
0 k
*A*1 ← *A*
i← 2, *j* ← *n* − 1
while (*i*≤ *n* and *j* ≥ 1) do if (*A*i >2j) then
*A*i ← *A*i−1 − 2j
else
*A*i ← *A*i−1
end if
i← *i*+ 1, *j* ← *j* − 1
end while
*B*1 =*A*n ∗*A*n
i← 2, *j* ← *n* − 1
while (*i*≤ *n* and *j* ≥ 1) do if *A*j 6= *A*j+1 then
*B*i ← *B*i−1 + (*A*j + *A*j+1)2i−1
Else
*B*i ← *B*i−1
end if
i← *i*+ 1, *j* ← *j* − 1
end while
return *B*← *B*n

IV Results and Simulation

The Simulation and result of our proposed method for A New Robust Reduced-Bit Multiplication Algorithm by using Vedic Mathematics is shown in this section. For simulation and results of our proposed algorithm we have to use MATLAB R 2012b (8.0.0.783) software. Here we have applied our proposed algorithm on different type of digital signal operations (DSP) they are convolutions circulation convolution. By using MATLAB. The performance of the proposed algorithm is tested for different four different operations. Basic configuration of our system is Manufacturer: Hewlett-Packard HP 4540s Processor : Intel(R) Core(TM) i5-3110M CPU @ 2.40 GHz 2.40 GHz with 4.00 GB (2.64 GB usable) RAM : System type: 64-bit Operating System. As per the above statement implementation is done is MATLAB. Here our shows in Graphical user interface (GUI) as shows in the figure. In this GUI we have compared the DSP operation timing, So we see that the two sub windows they are first one is the DSP operation window and the second one is by the Vedic operation. Here we calculate the four different operations, they are

1. Convolution
2. Cross Convolution
3. Circular Convolution
4. Auto Correlation

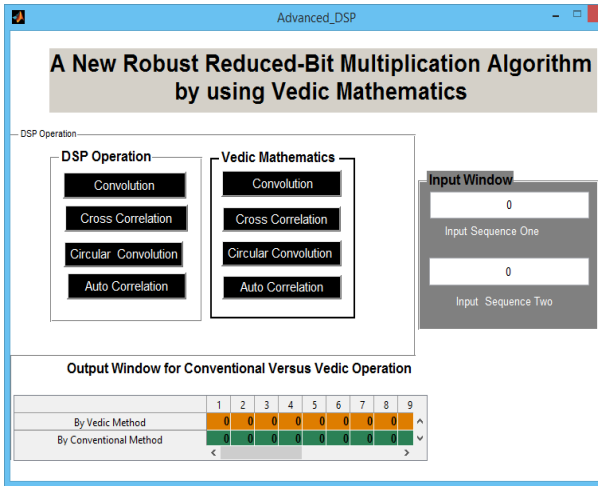


Figure 1

1. *For Convolution*

Convolution is a much easier operation in the discrete time domain than in the continuous time domain. Let's say we have two data sets, A[n] and B[n]:

A[n] = [1& 0 1 2]
B[n] = [2& 2 2 1]

We will denote convolution with the asterisk (*) in this section; it isn't "multiplication" here. Our convolution function is shown like this:

Y[n] = A[n] * B[n]

And it specifies that we will store our convolution values in the set named Y[n].

Convolution is performed by following a series of steps involving both sets of data points. First, we time invert one of the data sets. It doesn't matter which one, so we can pick the easiest choice:

A[n] = [1& 0 1 2]
B[-n] = [1 2 2 2&]

Next, we line up the data vertically so that only one data item overlaps:

A[n] -> [1& 0 1 2]
B[-n] -> [1 2 2 2&]

Now, we are going to zero-pad both sets, making them equal length, putting zeros in open positions as needed:

A[n] -> [0 0 0 1& 0 1 2]
B[-n] -> [1 2 2 2& 0 0 0]

There are some examples of DSP operation, in this paper we have to focus on the perform these operation on the on "Vedic Mathematics". With the help of different type of Vedic operations.

The better your paper looks, the better the Journal looks. Thanks for your cooperation and contribution.
Figure represent the main window for DSPs operation and performing the convolution operation for sequence one

x(n)=[1,7,2,6,9,1,4,2,5.6],
h(n)=[1,6,2,7,8,3,6,1,7,8]

Display the convolution result
y(n)=[12,19,23,39,62,66,103,122,115,140,133,81,96,95,28,44,11,16,24].
Table I and II show the time comparison between conventional versus Vedic operation for different input sequences and operations.

V. Conclusion

A fast computation of DSP operations of two finite length sequence implemented with the help of single GUI window. DSP operations are based on Robust Reduced-Bit Multiplication Algorithm by using Vedic Mathematics of Vedic mathematics, which reduces the processing time as compare to inbuilt function of Matlab. Proposed algorithm provide average processing time in micro second and conventional operation provide average time in mili second. Mathematics operation time give in Graphical User Interface window is easy to use and user friendly. In future, the Fast Fourier Transform and Filter operation is design with the help of Vedic Robust Reduced-Bit Multiplication Algorithm by using Vedic Mathematics method.

References

[1] A. K. Itawadiya, R. Mahle, V. Patel, D. Kumar, " Design a DSP operation using Vedic Mathematics" in Proc. IEEE Int. Conf. on communication and signal processing (ICCSP), 2013 India. pp.897-902.
[2] David Bbez, David Baa-Villegas "A Package For Filter Design Based On Matlab", ASEE / IEEE Frontiers in Education Conference, 10-13 Oct 2001.
[3] G. proakis, Dimitris G. Monalaskis, "Digital Signal Processing", Prentice hall Inc-1996, P.No.82
[4] H.Guilt, "Fully Iterative Fast Array for Binary Multiplication", Electronics Letters, vol. 5, p. 263, 1969.
[5] HimanshuThapliyal and Hamid R. Arabnia, "A Time-Area- Power Efficient Multiplier and Square

Architecture Based On Ancient Indian Vedic Mathematics", Department of Computer Science, The University of Georgia, 415 Graduate Studies Research Center Athens, Georgia 30602-7404, U.S.A.

[6] E. Abu-Shama, M. B. Maaz, M. A. Bayoumi, "A Fast and Low Power Multiplier Architecture", The Center for Advanced Computer Studies, The University of Southwestern Louisiana Lafayette, LA 70504.

[7] Ramalatha, M Dayalan, K D Dharani, P Priya, and S Deborah, "High speed energy efficient ALU design using Vedic multiplication techniques", ICACTEA, 2009. pp. 600-3, Jul 15-17, 2009.

[8] Harpreet Singh Dhillon and AbhijitMitra, "A Reduced-Bit Multiplication Algorithm for Digital Arithmetics", International Journal of Computational and Mathematical Sciences 2;2 © www.waset.org Spring 2008.

[9] Jagadguru Swami Sri Bharath, Krishna Tirathji, "Vedic Mathematics or Sixteen Simple Sutras from The Vedas", MotilalBanarsidas, Varanasi (India), 1992.

[10] Honey Durga Tiwari, GanzorigGankhuyag, Chan Mo Kim, Yong Beom Cho, "Multiplier design based on ancient Indian Vedic Mathematics", Dept. of Electronics Engineering Konkuk University Seoul, South Korea.

[11] Kavita, UmeshGoyal, "Performance analysis of various Vedic techniques for multiplication", International Journal of Engineering trends & technology- volume 4 Issue 3-2013.

[12] Ch. Harish Kumar, "Implementation and Analysis of Power, Area and Delay of Array, Urdhava, Nikhilam Vedic Multipliers", International Journal of Scientific and Research Publications, Volume 3, Issue 1, January 2013.

[13] Harpreet Singh DhillonAbhijitMitra, "A Digital Multiplier Architecture using UrdhavaTiryabhyam Sutra of Vedic Mathematics", Indian Institue of Technology, Guwahatti.

[14] J. Park, W. Jeong, H. Choo, H. Meimand, Y. Wang, K. Roy, "High Performance and Low Power FIR Filter Design Based on Sharing Multiplication", 2002, ISLPED '02. pp.295-300.

[15] HimanshuThapliyal, SaurabhKotiyal and M. B Srinivas, "Design and Analysis of A Novel Parallel Square and Cube Architecture Based On Ancient Indian Vedic Mathematics", Centre for VLSI and Embedded System Technologies, International Institute of Information Technology, Hyderabad, 500019, India, 2005 IEEE.

[16] Harpreet Singh Dhillon, AbhijitMitra, "A Reduced-Bit Multiplication Algorithm for Digital Arithmetics," International Journal of Computational and Mathematical Sciences, Spring 2008, pp.64-69.

[17] VaijyanathKunchigi, Linganagouda Kulkarni, Subhash Kulkarni, "High Speed and Area Efficient Vedic Multiplier", Jawaharlal Nehru Technological University Hyderabad, AP, India.

[18] T.W. Parks, C.S. Burus, "Digital Filter Design – Introduction to Digital Filters", John Wiley Sons, Inc-1987, P.No.04.

[19] Monson H. Hayes "Schaum's Outline of Theory and Problems of DIGITAL SIGNAL PROCESSING", The McGraw-Hill Companies, Inc.-1999, P.No.01.

[19] Sophocles J. Orfanidis, "Introduction to Signal Processing", Prentice Hall, Inc.-2009, P.No.100.

[20] Koji Nakamura, KazouToraichi, KazukiKatagishi, Yukio Koyanagi, Akira Okamoto, "Design of FIR Filter with Small Number of Coefficient based on compactly Supported Fluency".

[21]MadhuraTilak, "An Area Efficient, High Speed Novel VHDL Implementation of Linear Convolution of Two Finite Length Sequences Using Vedic Mathematics", International Journal of Innovative Technology and Exploring Engineering (IJITEE), Volume-2, Issue-1, December 2012.

[22] A. Venkatesan, S. VenkatKumar,"Efficient Implementation of Fast Convolution in ASIP", Anna University, 2010 IEEE.

[23] ArojitRoychowdhury, "FIR Filter Design Techniques", Electronic Systems Group, EE Dept, IIT Bombay. 2002.

[24] David Bbez-Ldpez, David Baa-Villegas, Rent Alcantara, Juan Jost Romero, and Tombs Escalantd, "A Package for Filter Design Based on MATLAB", 31st ASEE/IEEE Frontiers in Education Conference,2001.

[25] Isabella1and Emi Retna, "Study paper on test case generation for GUI based testing", in proc. Int. Journal Software Engineering & Applications,Vol.3, Jan 2012, pp. 3-8.