Design of High Speed Single Precision Floating Point Multiplier Using Vedic Mathematics

Sneha Khobragade, Mayur Dhait,

M. Tech Student, Department of Electronics, Agnihotri College of Engineering, Nagthana Road, Wardha,

M.S, India

Professor, Department of E&C, Agnihotri College of Engineering, Nagthana Road, Wardha, M.S, India

ABSTRACT: The fundamental and the core of all the Digital Signal Processors (DSPs) are its multipliers and the speed of the DSPs is mainly determined by the speed of its multiplier. In this paper we have synthesized and verified IEEE 754 single precision Floating Point Multiplier using VHDL on Xilinx Software. The Urdhva-Tiryakbhyam sutra (method) was selected for designing of mantissa. In addition the proposed designed handled underflow, overflow and rounding condition. High speed is achieved by reducing carry propagation delay by using rippel carry adder while design of four (24 x 24 bit multiplier for single precision) .

KEYWORDS: Urdhva-Tiryakbhyam sutra,DSP.

I. INTRODUCTION

   Multipliers are key components of many high performance systems such as microprocessors, DSP processors, various FIR filters, etc. A performance of a system is generally determined by the performance of the multiplier because the multiplier is generally the slowest element in the system. Since multiplication dominates the execution time of most DSP application so there is need of high speed multiplier. Hence increasing the speed and optimizing area of the multiplier is a major design issue. As a result, a whole spectrum of multipliers with different area-speed constraints has been designed with fully parallel Multipliers at one end of the spectrum and fully serial multipliers at the other end. These multipliers have moderate performance in both area and speed. In DSP application binary floating point numbers multiplication is is one of the basic functions. The IEEE 754 standard provides for many closely related formats. Five of these formats are called basic formats and others are termed extended formats three of these are especially widely used in computer hardware and languages Single precision, Double precision, double extended precision. The Single precision consist of 32 bits, Double precision consist of 64 bits and double extended precision of 80 bits.

TABLE 1

FORMAT	SIGN	EXPONENT	MANTISSA
SINGLE PRECISION	1(31)	8 (23 TO 30)	23(0 TO 22)
DOUBLE PRECISION	1(64)	11(52 TO 63)	52(0 TO 51)

   The formats are composed of 3 units; Sign unit, Exponent unit and Mantissa unit. A typical central processing unit devotes a considerable amount of processing time in implementing arithmetic operations, particularly multiplication operation. Multiplication is an important fundamental arithmetic operation. Performance constraints can also be addressed by applying alternative technologies. A change in design implementation level by the insertion of a new technology can often make viable an existing marginal algorithm or architecture.

This work deals with the ―Design of high speed multiple precision floating point multiplier using ―Vedic algorithm ‖ . In this project, Vedic Multiplication Technique is used to implement IEEE 754 Floating point multiplier. The Vedic sutra is used for the multiplication of Mantissa. Performance constraints can also be addressed by applying alternative technologies.

II.   RELATED WORK

According to Aniruddha Kanhe [1] DSP applications is being require to the multiplication of binary floating numbers. The IEEE 754 standard endows the format for representation of Binary Floating point numbers . The Binary Floating numbers are represented in Single precision and Double precision formats. The Single precision format consist of 32 bits and the Double precision format consist of 64 bits. The both formats are composed of 3 fields; Sign field, Exponent field as well as Mantissa field. The delineated structure shows the structure of Single and Double formats of IEEE 754 standard. In case of Single, the Mantissa is represented in 23 bits and 1 bit is added to the MSB for normalization, Exponent is represented in 8 bits which is biased to 127, actually the Exponent is represented in excess 127 bit format and MSB of Single is reserved for Sign bit. When the sign bit is 1 that means the number is negative and when the sign bit is 0 that means the number is positive. In 64 bits format the Mantissa is represented in 52 bits, the Exponent is represented in 11 bits which is biased to 1023 and the MSB of Double is reserved for sign bit.The performance of Mantissa calculation Unit dominates overall performance of the Floating Point Multiplier. This unit requires unsigned multiplier for multiplication of 24x24 BITs. The Vedic Multiplication technique is chosen for the implementation of this unit. But consume more delay. Hence new technique gives promising result in terms of speed and power .

According to Mohamed Al-Ashrafy [2] for designing of floating point multiplier is efficient using Carry save multiplier. This paper presents an implementation of a floating point multiplier that supports the IEEE 754-2008 binary interchange format; the multiplier doesn't implement rounding and just presents the significant multiplication result as is (48 bits); this gives better precision if the whole 48 bits are utilized in another unit; i.e. a floating point adder to form a MAC unit. The design has three pipelining stages and after implementation on a Xilinx Virtex5 FPGA. In order to enhance the performance of the multiplier, pipelining stages are used to divide the critical path thus increasing the maximum operating frequency of the multiplier. Since a processor spends considerable amount of time in performing multiplication, an improvement in multiplication speed by using new techniques can greatly improve system performance. According to Honey Durga Tiwari [3] a Vedic multiplier and square architecture is proposed based on algorithm of ancient Indian Vedic Mathematics, for low power and high speed applications. This Paper shows how the computational complexity is reduced in the case of Vedic multipliers as compared to the conventional multipliers. The Vedic multiplication formulae, Urdhva tiryakbhyam and Nikhilam, have been investigated in detail. Due to its structure, it suffers from a high carry propagation delay in case of multiplication of large numbers. This problem has been solved by introducing Nikhilam Sutra which reduces the multiplication of two large numbers to the multiplication of two small numbers. The FPGA implementation result shows that the delay and the area required in proposed design is far less than the conventional booth and array multiplier designs making them efficient for the use in various DSP applications.

III.   FLOATING POINT MULTIPLICATION ALGORITHM

The multiplier for the floating point numbers represented in IEEE 754 format can be divided in four different units Mantissa Calculation Unit, Exponent Calculation Unit Sign Calculation Unit, Normaliser Unit.
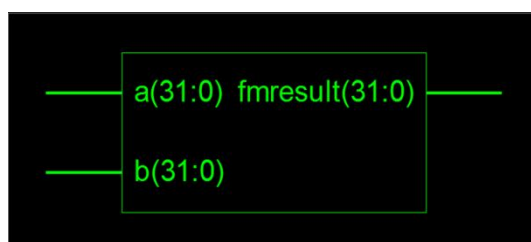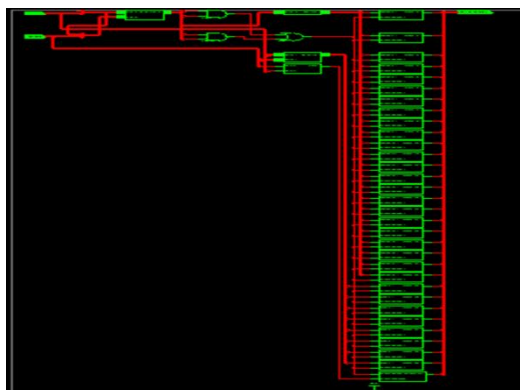
Fig.1:- Proposed architecture for single precision Floating point multiplier

As stated in the introduction, normalized floating point numbers have the form of $Z = (-1S) * 2 (E - Bias) * (1.M)$. The below fig flowchart shows how multiply of two floating point numbers is done.
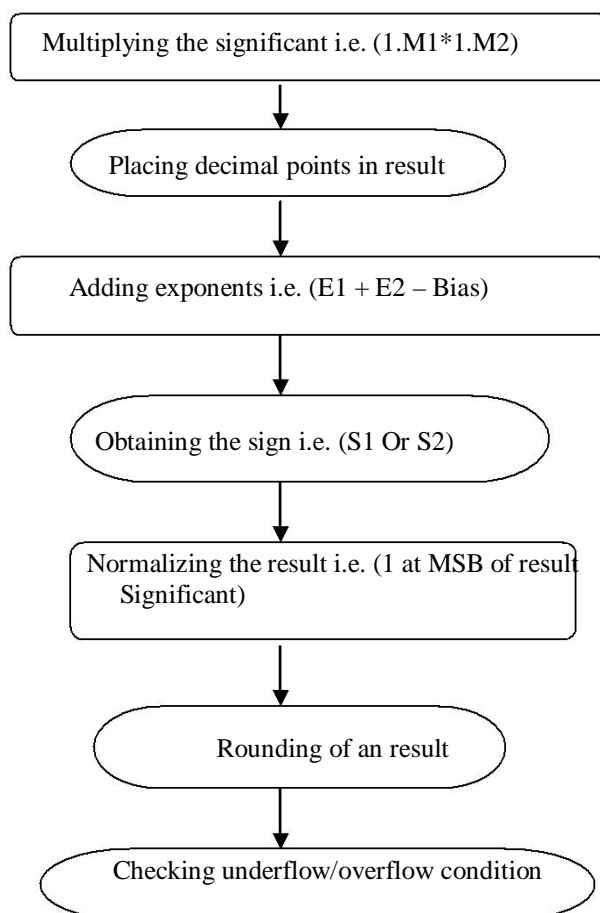

Fig.2:- Steps of multiplying Floating Point numbers.

IV.   MULTIPLIER DESIGN

A. *Vedic mathematics:*

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 .These Sutras along with their brief meanings are enlisted below alphabetically[l].

1) (Anurupye) Shunyamanyat - If one is in ratio. The other is zero.
2) Chalana-Kalanabyham Differences and Similarities.
3) Ekadhikina Purvena - By one more than the previous one.
4) Ekanyunena Purvena - By one less than the previous one.
5) Gunakasamuchyah - The factors of the sum is equal to the sum of the factors.
6) Gunitasamuchyah - The product of the sum is equal to the sum of the product.
7) Nikhilam Navatashcaramam Dashatah - All from 9 and the last from 10.
8) Paraavartya Yojayet - Transpose and adjust.
9) Puranapuranabyham - By the completion or no completion.
10) Sankalana-vyavakalanabhyam - By addition and by subtraction.
11) Shesanyankena Charamena - The remainders by the last digit.
12) Shunyam Saamyasamuccaye - When the sum is the same that sum is zero
13) Sopaantyadvayamantyam - The ultimate and twice the penultimate.
14) Urdhva-tiryakbyham - Vertically and crosswise

B. *Urdhva-tiryakbyham sutra:*

The multiplier is based on an algorithm Urdhva-Tiryakbhyam (Vertical & Crosswise) of ancient Indian Vedic Mathematics. Urdhva-Tiryakbhyam Sutra is a multiplication formula which is applicable to all cases of multiplication. It literally means "Vertically and crosswise ‖ . the generation of all partial products can be done with the concurrent addition of these partial products. The algorithm can be generalized for n x n bit number. Since the pat1ial products and their sums are calculated in parallel, the multiplier is independent of the clock frequency of the processor. Thus the multiplier will require the same amount of time to calculate the product and hence is independent of the clock frequency. The net advantage is that it reduces the need of microprocessors to operate at increasingly high clock frequencies. While a higher clock frequency generally results in increased processing power, its disadvantage is that it also increases power dissipation which results in higher device operating temperatures. By adopting the Vedic multiplier, microprocessors designers can easily circumvent these problems to avoid catastrophic device.
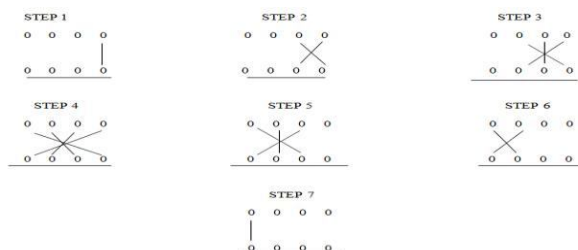

Fig.3:- Line diagram of the multiplication.

V.   MANTISSA CALCULATION UNIT

The performance of Mantissa calculation Unit dominates overall performance of the Floating Point Multiplier. This unit requires unsigned multiplier for multiplication of 24*24 BITs. The Vedic Multiplication technique is chosen for the design of this unit. This technique gives promising result in terms of speed and power. The Vedic multiplication system is based on 16 Vedic sutras or aphorisms, which describes natural ways of solving a whole range of

mathematical problems. Out of these 16 Vedic Sutras the Urdhva-triyakbhyam sutra is suitable for this purpose. In this method the partial products are generated simultaneously which itself reduces delay and makes this method fast24*24 bit multiplier designed using four 12*12 bit multiplier.
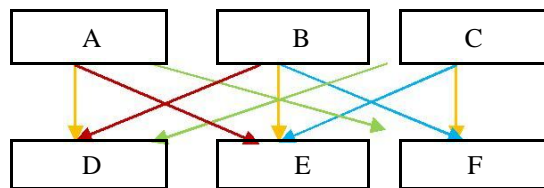

Fig.4:- 24*24 bit multiplier using rca.

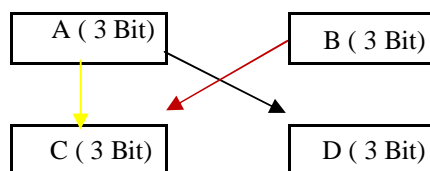Fig.5:- Basic Vedic Multiplier Logic for 3*3 Vedic multiplier.

Fig.7:- 6*6 Vedic multiplier.

VI. EXPONENT UNIT

In Exponent Unit bit 23 to 30 (8bit) are used by exponent unit. two exponent bits are added Then resultant exponent ER is subtracted with 2'complement of bias value 127 for 24 bit multiplier since exponent is of 8 bit i.e $2^8(127)$. In this we have taken 2'coplement of bias since subtractions in digital is nothing but 2'complememt addition. The two numbers E1 and E2 are already shifted by bias value and hence it is not a real exponent i.e. EA = EA-true + bias and EB = EB-true + bias. Hence ER = E1 + E2 – 127 To remove bias which was added twice for two number i.e E1 and E2. Exponent unit designed by using Rippel Carry Adder.

VII. .NORMALISER UNIT

Leading one to the left of the decimal point (i.e. 46 bit of resultant significant multiplication) can only be obtained by normalizing the result of the significant multiplication (intermediate product. Leading one is obtaine 47 as inputs are already normalized numbers. Shifting is required if leading one is at bit 47,   the intermediate product is shifted to right and   exponent is incremented by 1. But if leading one is at bit 46 i.e. left of decimal point then   no shifting is required since intermediate product is already normalized. Normaliser unit also do rounding of result number, since arithmetic operation on floating point values compute result that cannot be represented in the given amount of

precision. There are 3 methods of rounding each have correct uses and exist for different reasons. The 3 methods of Rounding are: round toward 0 (truncation) round toward + infinity round toward - infinity
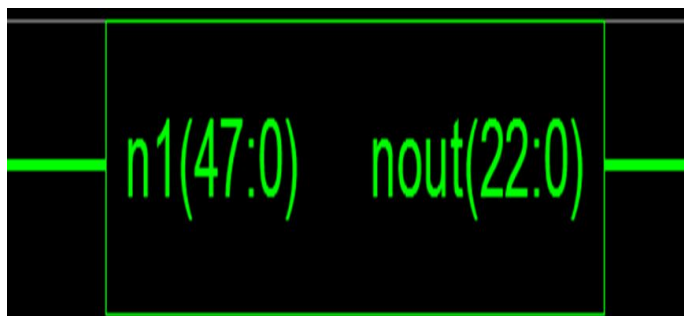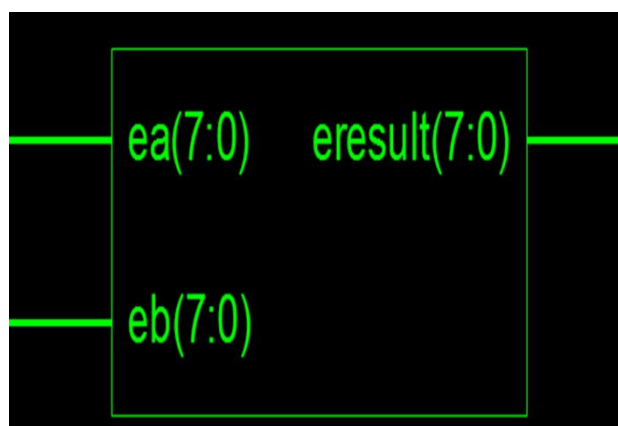

Fig.6:- Top level block of normalize.

Fig.7:- Top level block of 8 bit exponent.

VIII.   SIMULATION RESULT

In behavioural simulation we have tested for the following inputs bits
a)   For Single precision FP input The multiplier a =11000000110000000000000000000000 (decimal number system -6) and the multiplicand b=00111111101101001111110111110011 (decimal number system 1.484) & we get 32 bit output1 =11000000100001110111110011110110
b)   For single precision FP input The multiplier a=01000011000001100001000000000000 (decimal number system 134.0625) And the multiplicand b=11000000000010000000000 (decimal number system -2.25) & we get 32 bit output2=11000011100101101101001000000000(decimal number system -301.
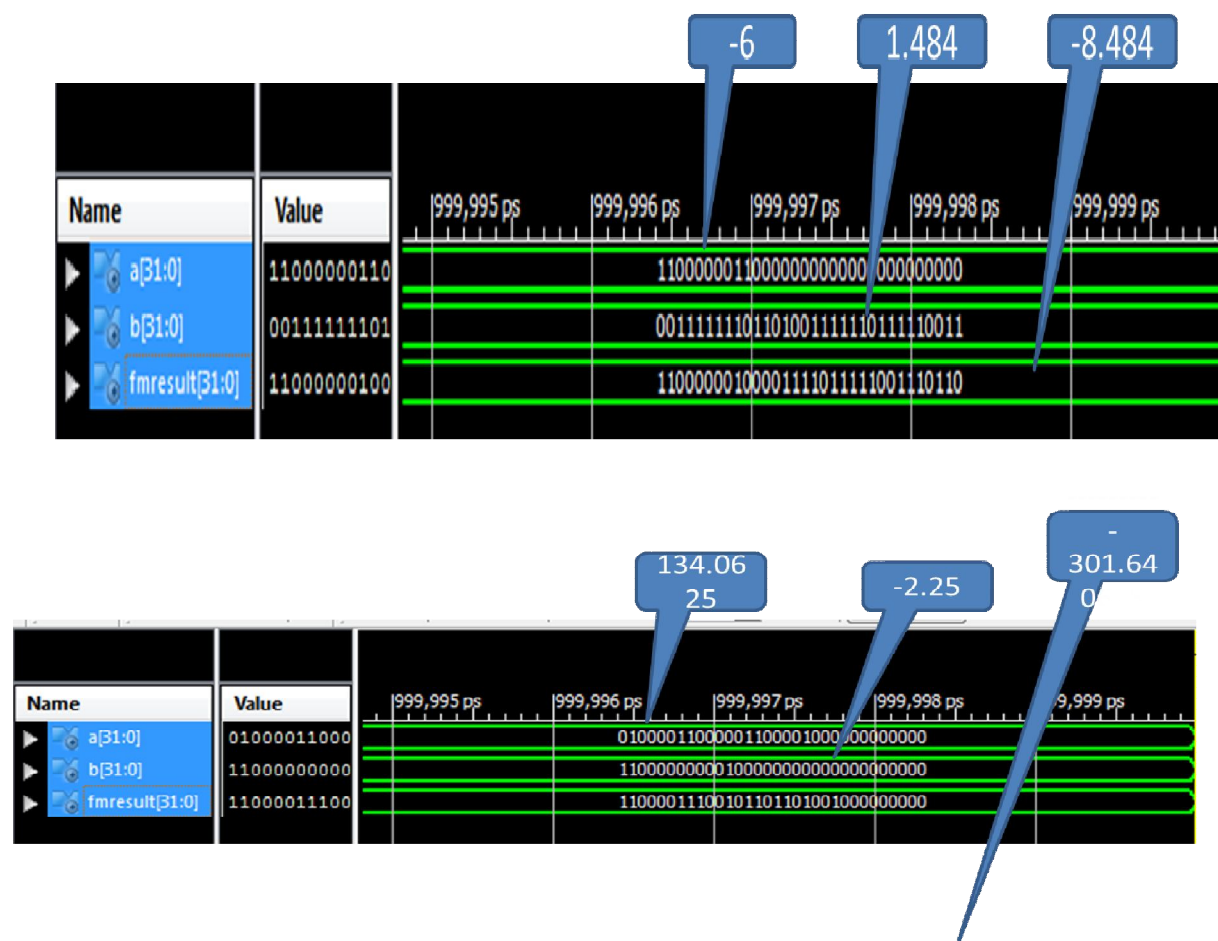
Fig:- Simulation results of Single precision floating point multiplier.

IX.   CONCLUSION

Thus by the aid of Vedic mathematics the design of the single and precision floating point multiplier has obtained. The designed multiplier can be work for any floating point numbers within range of single precision format and precision format. Use of RCA unit can save the efforts of minimizing delay. The different multiplier designs can aid to heeds on a performance of Vedic subunits for better response. The analysis of different design considerations has the efficient as well as accuracy of design is improve.

REFERENCES

1. Aniruddha Kanhe, Shishir Kumar Das, Ankit Kumar Singh,‖ Design and Implementation of Floating Point Multiplier base On Vedic Multiplication Technique‖ 2012 International Conference on Communication, Information & Computing Technology (ICCICT), Oct. 19-20, Mumbai, India.
2. Mohamed Al-Ashrafy, Ashraf Salem, Wagdy Anis.‖An Efficient Implementation of Floating Point Multiplier‖.
3. Honey Durga Tiwari, Ganzorig Gankhuyag, Chan Mo Kim, Yong Beom Cho, "Multiplier design based on ancient Indian Vedic Mathematics‖, 2008 International SoC Design Conference, PP 65-68.
4. Anvesh Kumar, Ashish Raman, Dr.R.K.Sarin, Dr.Arun Khosla, Small area Reconfigurable FFT Design by Vedic Mathematics‖ 978-1-4244-5586-7/10C 2010 IEEE, Volume 5.
5. Brian Hickmann, Andrew Krioukov, and Michael Schulte, Mark Erle,‖A Parallel IEEE 754 Decimal Floating-Point Multiplier,‖ In 25th International Conference on Computer Design ICCD, Oct. 2007
6.   Jagadguru Swami Sri Bharati Krisna Tirthaji Maharaja, ―Vedic Mathematics Sixteen Simple Mathematical Formulae from the Veda,‖1965.

7. Shaifali and Ms.Sakshi, Comparison of IEEE-754 Standard Single Precision Floating Point Multiplier's. International   Journal of Emerging Trends in Electrical and Electronics (IJETEE). Vol. 1, Issue. 3, March-2013J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
8. Rekha K James, Poulose K Jacob, Sreela Sasi, ―Decimal Floating Point Multiplication using RPS Algorithm,‖ IJCA Proceedings on International Conference on VLSI, Communications and Instrumentation (ICVCI): 2011.
9. Aniruddha Kanhe, Shishir Kumar Das and Ankit Kumar Singh,Design And Implementation Of Low Power Multiplier Using Vedic Using Vedic Multiplication Technique,‖ International Journal of Computer Science and Communication (IJCSC) Vol. 3, No. 1, January-June 2012, pp. 131- November 2006.132.
10. Kavita Khare, R.P.Singh, Nilay Khare,‖Comparison of pipelined IEEE-754 standard floating point multiplier with   unpipelined multiplier ―Journal of Scientific & Industrial Research I. S. Jacobs and C. P. Bean, ―Fine particles, thin films and exchange anisotropy,‖ in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–35