



VLSI Architecture for Area Vedic Multiplier Using Common Boolean Logic

Sachin Pratap Singh Rajpoot, Vipul Agarwal
Trinity Institute of technology & Research,
Bhopal, India

Abstract-In the advanced digital technology the need is of high speed in real time system along with the improvement in implementation issue. Vedic Multipliers has been used to solve the typical and tedious engineering calculation by simple Vedic methods. Here in this paper we have proposed the Vedic multiplier with Common Boolean Logic adder to improve the propagation delay time and area on silicon chip. With this slight improve in the multiplier, great results have been achieved in signal processing tasks. The VM has been designed for the target device XC3S400 -5 PQ208.

Keywords: Common Boolean Logic (CBL), Vedic Multiplier (VM), Xilinx

I. INTRODUCTION

Progression in innovation is becoming astoundingly to maintain the need of shrewd clients. Fast correspondence and information exchange is the vital prerequisite of this computerized world. Rapid of information computation is the primary worry in computerized frameworks. It fundamentally relies on the quantity of entryways used to plan the rationale. Multiplier and adders are the building hinder in Digital sign preparing framework. Inertness and spread postponements are the territory to be worked upon to make the frameworks proficient as for time taken in preparing the sign. Advanced multipliers accomplish the most extreme usage effectiveness and check execution in computerized signal handling framework. It has genuinely substantial number of hardware and complex outline included in it. For a sample, planning the n bit multiplier the doors utilized is the square (n^2) of information bits.

Increase process includes arrangement of expansion of halfway items produced from duplicating the data bits one by one. This procedure takes extensive compass of time to compute the outcome. As the quantity of bit builds the outline of multiplier turn out to be more mind boggling to execute. To decrease the preparing time the halfway item are ascertained in parallel at the same time. At that point these halfway items have been included with the assistance of half viper and full snake. A fractional item convey is likewise created after every option which must be included the following stride of expansion. The motivation behind utilizing this full viper is to diminish the quantity of augmentations.

Vedic science is the idea utilized as a part of the season of old Indian and was rediscovered in mid twentieth century from antiquated Indian form prevalently known as Vedas. This Vedic arithmetic has been connected to advanced calculation counts of the computerized multiplier. One of the benefits of utilizing Vedic computation as a part of that it helps in improving the convoluted protracted figuring into the more straightforward structures. Numerous systems have been proposed for increase and division. One nonexclusive system is accommodated every operation alongside some further particular estimations. Urdhwa – Triyakbhyam is the general recipe relevant to all instances of augmentation and division. It implies vertically and transversal.

The paper is organized as follows: Section II proposes the related work. Section III contains architecture of proposed Vedic multiplier using carry Boolean logic Section IV provides proposed methodology for Vedic multiplier. Section V contains results and discussion. Section VI conclusions followed by future work.

II. RELATED WORK

Shahnam et al., [1], We present a method for implementing high speed Finite Impulse Response (FIR) filters using just registered adders and hardwired shifts. We extensively use a modified common sub-expression elimination algorithm to reduce the number of adders. We target our optimizations to Xilinx Vertex II devices where we compare our implementations with those produced by Xilinx Coregent using Distributed Arithmetic. We observe up to 50% reduction in the number of slices and up to 75% reduction in the number of LUTs for fully parallel implementations. We also observed up to 50% reduction in the total dynamic power consumption of the filters. Our designs perform significantly faster than the MAC filters, which use embedded multipliers.

Amina Naaz.S et al. [2], in today's world lots of research work is going in the field of communication and signal processing applications. Every application demands for a higher throughput arithmetic operation. One of the key arithmetic operations is multiplication which takes maximum execution time. The development of efficient multiplier is a subject of interest over decades. So there is a need for an efficient multiplier which obtains higher performance for real time signal processing application. This paper presents the modular design of Vedic multiplier using carry select adder. The delay of proposed multiplier is reduced due to high speed carry select adder. The proposed multiplier is applied to parallel FIR filter. It can be observed that the combinational delay reduced for the proposed multiplier compared to existing architecture.

Yu-Chi Tsao et al. [3], based on fast finite-impulse response (FIR) algorithms (FFAs), this paper proposes new parallel FIR filter structures, which are beneficial to symmetric coefficients in terms of the hardware cost, under the condition that the number of taps is a multiple of 2 or 3. The proposed parallel FIR structures exploit the inherent nature of symmetric coefficients reducing half the number of multipliers in sub-filter section at the expense of additional adders in preprocessing and post processing blocks. Exchanging multipliers with adders is advantageous because adders weigh less than multipliers in terms of silicon area; in addition, the overhead from the additional adders in preprocessing and post processing blocks stay fixed and do not increase along with the length of the FIR filter, whereas the number of reduced multipliers increases along with the length of the FIR filter. For example, for a four-parallel 72-tap filter, the proposed structure saves 27 multipliers at the expense of 11 adders, whereas for a four-parallel 576-tap filter, the proposed structure saves 216 multipliers at the expense of 11 adders still. Overall, the proposed parallel FIR structures can lead to significant hardware savings for symmetric convolutions from the existing FFA parallel FIR filter, especially when the length of the filter is large.

Laxman P.Thakre et al., [4] proposed Vedic algorithm for the implementation of multipliers to be used in the FFT. The conventional multiplication method requires more time & area on silicon than vedic algorithms. More importantly processing speed increases with the bit length. This will help ultimately to speed up the signal processing task, as it is well known that the multiplier is the basic building block of FFT. The delay of this Vedic Multiplier is 58.28ns and the number of slices used is 461 out of 3584. Number of four input LTUs used is 808 out of 7168. Deepshikha Bharti et al., [2] proposed the design of filter operation with delay efficient addition and multiplication architecture that reduces the bit width. Efficient parallel adder has been used to form FIR filter with 8 and 16 tap. Amina Naaz.S et al.,[3] proposed the efficient multiplier which obtains higher performance for real time signal processing application. It has designed the Vedic multiplier using carry select adder which helps in reducing the delay. This multiplier is implemented to design the parallel FIR filter. Here the propagation delay is 40.38ns and the number of slices used are 445 out of 3584. The number of 4 input LUTs are 777 out of 7186.

III. COMMON BOOLEAN LOGIC

Area and power efficient high speed data logic path are the most significant areas of research. With the help of simple modification in gate level we can achieve the improvement in the results. Speed of the adder depends on the time required to propagate the carry through the adder. These adder works in series format, that is the sum of the elementary position bit is calculated when the previous bits are summed and the carry is propagated to that next stage.

Carry select adder (CSLA) is one of the advanced adders used in data processing processors to perform fast arithmetic function. It focuses on the problem of carry propagation delay by generating the carry independently at each stage and the select the efficient one with the help of multiplexer to perform the sum. The conventional CSLA is RCA (Ripple carry adder) which generate the partial sum and carry by using the input carry condition $C_{in}=0$ and $C_{in}=1$, select one out of each pair to form final sum and final carry output. RCA is not area efficient as large number of gates circuitry is used to form the partial products and then the final sum and carry is selected.

Another form of CLSA adder uses binary to excess-1 convertor replacing ripple carry adder with $C_{in}=1$. This adder is known as CLSA along with BEC. The number of gates used has been reduced when we have to design large bit adder. This adders is more conventional as compare to RCA when deal with silicon area used but this is having marginally higher delay time.

The proposed Common Boolean Logic (CBL) adder is area-power-delay efficient. It work on the logic to remove the redundant adders and use Common Boolean Logic as compare to conventional carry select adder.

The CBL block is comprised of two parts sum generation block and carry generation block. In sum generation block the output sum is achieved using the multiplex. This multiplex is used to select the output value depending on the value of C_{in} (previous bit).

If $C_{in}=0$, then output is xor of the two input bits. If $C_{in}=1$, then output get inverted. In carry generation block, multiplexer is used to select the carry of next stage depending upon the previous carry input. If $C_{in}=0$, c_{out} is OR of two input and if $C_{in}=1$ the output carry is AND of the input bit.

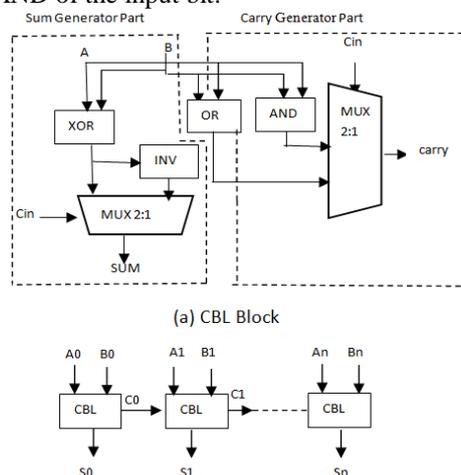


Figure 1: Block Diagram of n-bit CBL

If $C_{in} = 0$
 $Sum = A XOR B$
 $Carry = A OR B$
 else
 $Sum = NOT (A XOR B)$
 $Carry = A AND B$

This same process is used for the n number of bits and thus we get the final sum and carry as output.

IV. PROPOSED ARCHITECTURE OF 16X16 BIT VEDIC MULTIPLIER

The multiplication of two numbers is done by using Urdhwa Triyakbhyam. Here first the least significant bits of the two digits are multiplied. Then the intermediate digits are cross multiplied and added together. After this the most significant digits are multiplied.

For the 16X16 bit multiplication small block of 2X2 or 4X4 or 8X8 multiplier were used in parallel to make the process easy and efficient.

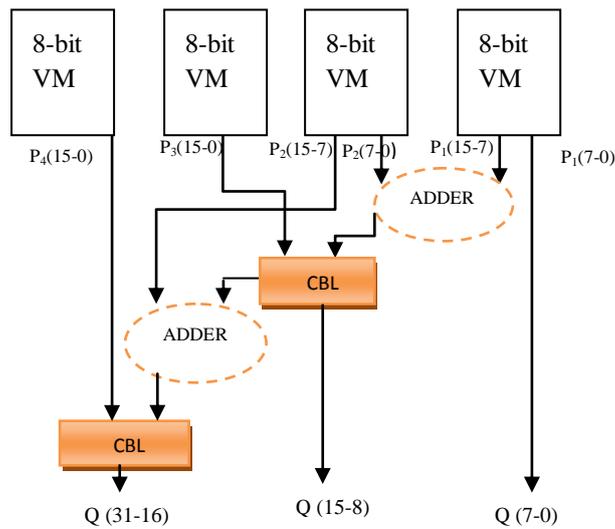


Figure 2: Proposed 16x16 Vedic Multiplier

In our proposed method the high speed carry select adder is replaced by the carry select adder along with Common Boolean logic which claims to provide a better speed and less propagation delay. Here we have used four multiplier of 8 bit to perform 16 bit multiplication. The method used is the addition of all partial product formed by the cross multiplication of one bit with another. The LSB bits of first multiplier P_1 (7-0) gives the LSB bits Q (7-0) of the final output. Another bits of first multiplier P_1 (15-8) are added in series with LSB 8 bits of second multiplier to form the 16 bits, which in turn get added with 16 bits of third multiplier by using CBL 1 Adder. The LSB bits of the output of CBL 1 adder forms the Q (15-8) bits of the final output. The remaining 8 bit P_2 (15-8) is then added with the left 8 bits of CBL 1 output to form 16 bits, which is then added with 16 bits of the fourth multiplier by using CBL 2 adder. The output from CBL 2 adder forms the Q (31-16) bits. This is how the 32bit output is achieved in the less possible time.

V. RESULTS AND DISCUSSION

All the designing and experiment regarding algorithm that we have mentioned in this paper is being developed on Xilinx 14.1i updated version. Xilinx 9.2i has couple of the striking features such as low memory requirement, fast debugging, and low cost. The latest release of ISE™ (Integrated Software Environment) design tool provides the low memory requirement approximate 27 percentage low. ISE 14.1i that provides advanced tools like smart compile technology with better usage of their computing hardware provides faster timing closure and higher quality of results for a better time to designing solution. ISE 14.1i Xilinx tools permits greater flexibility for designs which leverage embedded processors. The ISE 14.1i Design suite is accompanied by the release of chip scope Pro™ 14.1i debug and verification software. By the aid of that software we debug the program easily. Also included is the newest release of the chip scope Pro Serial IO Tool kit, providing simplified debugging of high-speed serial IO designs for Virtex-4 FX and Virtex-5 LXT and SXT FPGAs. With the help of this tool we can develop in the area of communication as well as in the area of signal processing and VLSI low power designing.

Table I: Comparison Results of Existing and Proposed Vedic Multiplier

XC3S400 -5 PQ208	Existing Vedic multiplier	Proposed Vedic multiplier
Delay	40.83ns	39.10ns
Number of Slices	445 out of 3584 (12%)	371 out of 3584 (10%)
Number of 4input LUTs	777 out of 7168(10%)	659 out of 7168 (9%)
Number of bonded IOBs	64 out of 141 (45%)	64 out of 141 (45%)

VI. CONCLUSION

The proposed 16x16 Vedic multiplier architecture has been designed and synthesized using on Spartan 3 XC3S400 board and is used in parallel FIR filter design. The proposed Vedic Multiplier with carry select adder is compared with the existing Vedic multiplier using Carry select adder along with Common Boolean Logic and can be inferred that proposed architecture is faster compared to existing Vedic multiplier. In future the proposed multiplier performance parameters can be improved by high level pipelining operations and applied in signal processing applications like image processing and video processing.

REFERENCE

- [1] Shahnam Mirzaei, Anup Hosangadi, Ryan Kastner, "FPGA Implementation of High Speed FIR Filters Using Add and Shift Method", 1-4244-9707-X/06/\$20.00@2006 IEEE.
- [2] Amina Naaz.S, Mr.Pradeep M.N, Satish Bhairannawar and Srinivas halvi, "FPGA Implementation Of High Speed Vedic Multiplier using CSLA For Parallel Fir Architecture", 2014 2nd International Conference on Devices, Circuits and Systems (ICDCS).
- [3] Laxman P.Thakre, Suresh Balpande, Umesh Akare, Sudhir Lande, "Performance Evaluation and Synthesis of Multiplier used in FFT operation using Conventional and Vedic algorithms," Third international conference on emerging trends in Engineering and Technology , IEEE, 2010.
- [4] S. S. Kerur, Prakash Narchi, Jayashree C N, Harish M Kittur and Girish V. A., "Implementation of Vedic Multiplier for Digital Signal Processing," International Conference on VLSI ,Communication & Instrumentation (ICVCI), 2011.
- [5] G.Vaithyanathan, K.Venkatesan, S.Sivaramkrishnan, S.Sivaand, S.Jayakumar, "Simulation and implementation of Vedic multiplier usingVHDL code," International Journal of Scientific & Engineering Research, vol.4, 2013.
- [6] Pushpalata Verma and K. K. Mehta, "Implementation of an Efficient Multiplier based on Vedic Mathematics Using EDA Tool," International Journal of Engineering and Advanced Technolog(IJEAT), vol.1, June 2012.
- [7] C. Cheng and K. K. Parhi, "Furthur complexity reduction of parallel FIR filters," in Proc. IEEE ISCAS, May 2005, vol. 2, pp. 1835–1838.
- [8] C. Cheng and K. K. Parhi, "Low-cost parallel FIR structures with 2-stage parallelism," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 54, no. 2, pp. 280–290, Feb. 2007.
- [9] J. G. Chung and K. K. Parhi, "Frequency-spectrum-based low-area low-power parallel FIR filter design," EURASIP J. Appl. Signal Process., vol. 2002, no. 9, pp. 444–453, Jan. 2002.
- [10] K. K. Parhi, VLSI Digital Signal Processing systems: Design and Implementation. New York: Wiley, 1999.
- [11] Nivedita A. Pande, Vaishali Niranjane, Anagha V. Choudhari, " Vedic Mathematics for Fast Multiplication in DSP," International Journal of Engineering and Innovative Technology (IJEIT) ,vol.2, 2013.
- [12] Krishnaveni D. and Umarani.T.G, "Vlsi implementation of Vedic multiplier with reduced delay," International Journal of Scientific & Engineering Research, vol.2, May-2011.