

VLSI Architecture for Radix-4 Booth Complex Multiplier using Cyclic Redundant Adder

Abhinav Sahu(asahu7672@gmail.com), Associate Prof. Monika Kapoor(monikak@lnct.ac.in)

M. Tech. Scholar, Dept. of Electronics and Communication, LNCT, Bhopal¹
Associate Professor, Dept. of Electronics and Communication, LNCT, Bhopal²

Abstract- The main objective of this research paper is to design architecture for radix-4 complex Vedic multiplier by rectifying the problems in the existing method and to improve the speed by using the cyclic redundant adder. The multiplier algorithm is normally used for higher bit length applications and ordinary multiplier is good for lower order bits. These two methods are combined to produce the high speed multiplier for higher bit length applications. The problem of existing architecture is reduced by removing bits from the remainders. The proposed algorithm is implementation Xilinx software with Vertex-7 devicefamily.

Keywords – Vedic Multiplier, Complex Multiplier, Cyclic Redundant Adder, Xilinx Software

I. INTRODUCTION

In signal processing, a finite impulse response (FIR) channel is a channel whose reaction to any limited length input is of limited term, since it settles to zero in limited time. Rather than infinite impulse response (IIR) channels, which may have inward input and may keep on responding inconclusively (for the most part decaying). FIR channels are broadly utilized as a part of different DSP applications. In a few applications, the FIR channel circuit must have the capacity to work at high example rates, while in different applications, the FIR channel circuit must be a low-control circuit working at direct examplerrates. Parallel (or square) handling can be connected to computerized FIR channels to either expand the compelling throughput or lessen the power utilization of the first channel. While consecutive FIR channel usage has been given broad thought, next to no work has been done that arrangements straightforwardly with decreasing the equipment many-sided quality or power utilization of parallel FIR channels. Customarily, the use of parallel handling to a FIR channel includes the replication of the equipment units that exist in the first channel. The topology of the multiplier circuit additionally influences the resultant power utilization. Picking multipliers with more equipment expansiveness as opposed to profundity would decrease the postponement, as well as the aggregate power utilization. A considerable measure of outlinestrategies for low power computerized FIR channel have been proposed, for instance, a strategy executing FIR channel utilizing simply enrolled adders and hardwired shifts exist.

Parallel duplication is utilized to meet out the present prerequisite. Two kinds of parallel augmentations are exhibit duplication and tree increase. The fundamental multiplier is a basic cluster multiplier and it is planned in view of move and –

include task. One of the cases for exhibit increase is the Braun multiplier and is intended for unsigned paired numbers. For tree structure Wallace multiplier is outlined and it is likewise for an unsigned double numbers. In the exhibit augmentation, for marked numbers Baugh – Wooley, Booth Multiplier and Modified Booth Algorithm (MBA) are utilized. Dadda is another kind of multiplier in light of tree structure and is utilized for the increase of the marked numbers. These traditional double multipliers for unsigned numbers are considered for examination. Vedic arithmetic is the arrangement of science followed in old India and mostly manages Vedic scientific formulae and their applications to different branches of math. The word 'Vedic' is gotten fromthe word 'Veda' which implies the storage facility of all information.

Vedic science was remade from the antiquated Indian sacred writings (Vedas) by Sri Bharati Krishna Tirthaji (1884-1960), after his eight years of research on Vedas. As indicated by his examination, Vedic arithmetic is principally in light of sixteen standards or word-formulae and thirteen sub-end products which are named as Sutras. This is an exceptionally intriguing field and exhibits some viable calculations which can be connected to different branches of Engineering, for example, Computing and Digital Signal Processing. Vedic science diminishes the many-sided quality in figurings that exist in customary arithmetic. By and large there are sixteen sutras accessible in Vedic arithmetic.

Among them just two sutras are pertinent for increased activity. They are Urdhava Triyakbhyam sutra (truly implies vertically and across) and Nikhilam Sutra (truly implies All from 9 and last from 10). Urdhava-Triyakbhyam is a non- specific technique for augmentation. The rationale behind Urdhava Triyakbhyam sutra is especially like the conventionalcluster multiplier. Here the paired usage of this calculation is determined in light of a similar rationale utilized for decimal numbers. The double usage of Nikhilam Sutra isn't yet effective. In this research paper, a novel architecture of FIR filters based on complex multiplier using common Boolean logic adder.

II. VEDIC MULTIPLIER

Vedic multiplier furthermore, Cyclic Redundant Adder can contrast and regular strategy which is processed by Vedic multiplier, XOR entryway and half adder. Proposed procedure gives less way delay and less territory. Information grouping of Conventional strategy is significantly more than to proposed technique; however proposed technique has less

spread postponement. Region and engendering postponement can be decreased by the guide of Cyclic Redundant Adder. Rationale Diagram of Vedic Multiplier utilizing Cyclic Redundant Adder is appeared in figure 1. In the long run, all the planning levels of computerized framework or IC's Packages rely upon number of entryways in a solitary chip that

is likewise rung base approach. Cyclic Redundant Adder can be decreased in regards to the territory or number of gates. On the off chance that we expel the main XOR gates from adjusted Cyclic Redundant Adder nothing will be changed for

result however zone and proliferation deferral will be decreased.

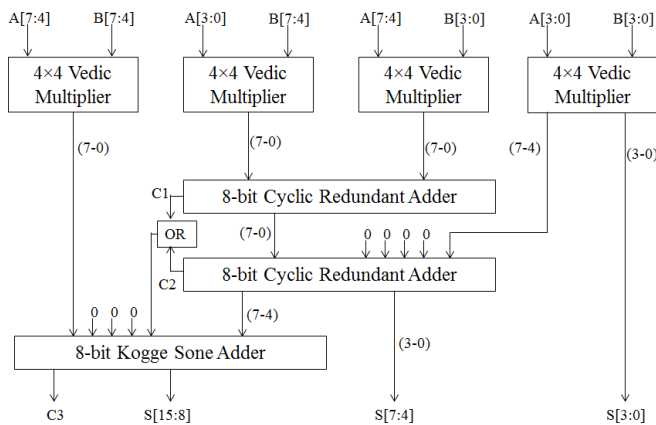


Figure 1: 8-bit Vedic Multiplier using Cyclic Redundant Adder

Cyclic Redundant Adder

In a zone effective and low power half adder based Carry select adder (CSLA) utilizing Cyclic Redundant Adder is outlined so as to upgrade the general framework execution as far as territory and power as contrast with other existing designs. Half adder is utilized to produce the incomplete entirety for cin=0 and Cyclic Redundant Adder is utilized for figuring halfway total for cin=1.

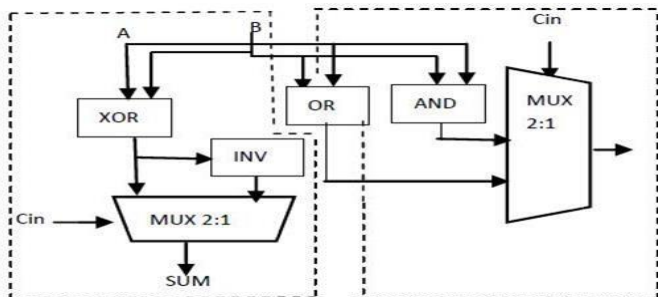


Figure 2: Block Diagram of Cyclic Redundant Adder

This engineering is utilized to expel the recreated adder cells

in the traditional CSLA, spare number of entryway tallies and accomplish a low power. Through investigating reality table of a solitary piece full snake we recommend that for producing yield summation and convey motion for cin=0, require just a single XOR gate and one AND entryway individually, the yield summation motion for cin=1 is simply the opposite as

cin=0. Summed up figure of regular Cyclic Redundant Adder is appeared in figure 2.

III. COMPLEX MULTIPLIER

Suppose two numbers are complex then

$$A = A_r + jA_i$$

$$B = B_r + jB_i$$

The product of A and B then

$$P = A \times B = A_r \times B_r - A_i \times B_i + j(A_r \times B_i + A_i \times B_r)$$

$$P_r = A_r \times B_r - A_i \times B_i$$

$$P_i = A_r \times B_i + A_i \times B_r$$

Where P_r and P_i is speaks to the genuine and nonexistent piece of the yield of the mind boggling multiplier. Ar and Ai is speaks to the genuine and fanciful piece of the principal contribution of the unpredictable multiplier. Br and Bi is speaks to the genuine and nonexistent piece of the second contribution of the unpredictable multiplier.

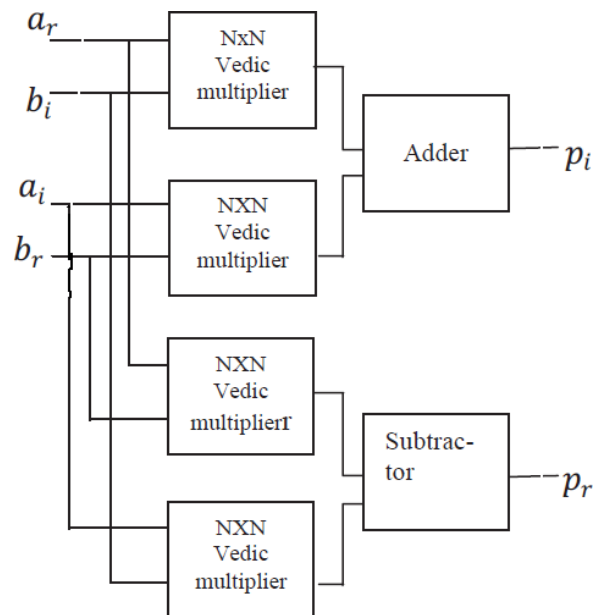


Figure 3: Block Diagram of Complex Multiplier for four Vedic Multiplier

Complex multiplier for four Vedic multipliers is shown in figure 3. In this block diagram reduce four Vedic multipliers to three Vedic multipliers is shown in below:

$$P_r = A_r \times B_r - A_i \times B_i = A_r(B_r + B_i) - B_i(A_r + A_i)$$

$$P_i = A_r \times B_i + A_i \times B_r = A_r(B_r + B_i) + B_r(A_i - A_r)$$

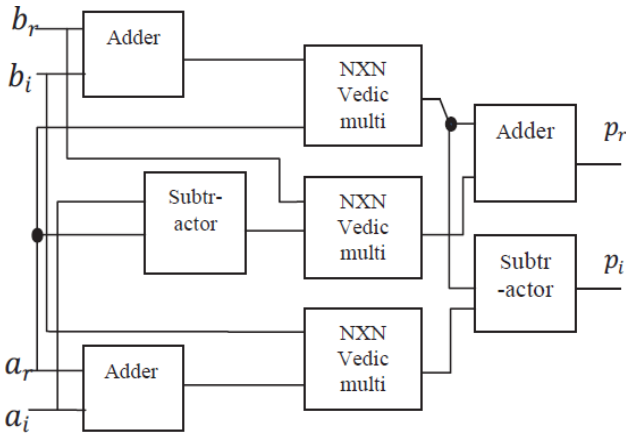


Figure 4: Block Diagram of Complex Multiplier for three Vedic Multiplier

IV. RADIX-4 ALGORITHM

To further decrease the number of partial products, algorithms with higher radix value are used. In radix-4 algorithm grouping of multiplier bits is done in such a way that each group consists of 3 bits as mentioned in table 1. Similarly the next pair is the overlapping of the first pair in which MSB of the first pair will be the LSB of the second pair and other two bits. Number of groups formed is dependent on number of multiplier bits. By applying this algorithm, the number of partial product rows to be accumulated is reduced from n in radix-2 algorithm to $n/2$ in radix-4 algorithm. The grouping of multiplier bits for 8-bit multiplication is shown in figure 5.

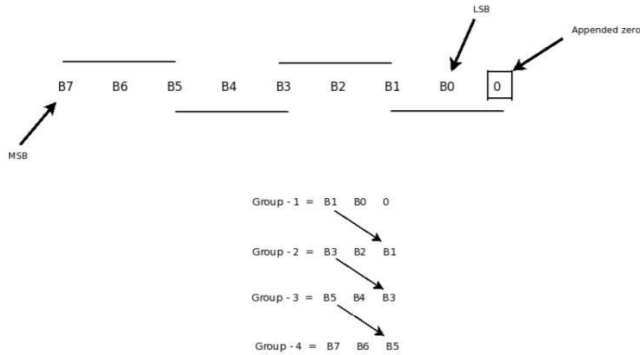


Figure 5: Grouping of multiplier bits in Radix-4 Booth algorithm

For 8-bit multiplier the number groups formed is four using radix-4 booth algorithm. Compared to radix-2 booth algorithm the number of partial products obtained in radix-4 booth algorithm is half because for 8-bit multiplier radix-2 algorithm produces eight partial products. The truth table and the respective operation is depicted in table 1. Similarly when radix-8 booth algorithm is applied to multiplier of 8-bits each group will consists of four bits and the number of groups formed is 3. For 8x8 multiplications, radix-4 uses four stages to compute the final product and radix-8 booth algorithm uses three stages to compute the product. In this thesis, radix-4

booth algorithm is used for 8x8 multiplications because number components used in radix-4 encoding style.

Table 1: Truth Table for Radix-4 Booth algorithm

B_{i+1}	B_i	B_{i-1}	Operation	Y_{i+1}	Y_i	Y_{i-1}
0	0	0	+0	0	0	0
0	0	1	+A	0	1	0
0	1	0	+A	0	1	0
0	1	1	+2A	0	0	1
1	0	0	-2A	1	0	1
1	0	1	-A	1	1	0
1	1	0	-A	1	1	0
1	1	1	-0	1	0	0

V. SIMULATION ANALYSIS

Simulation of these tests should be possible by utilizing Xilinx 14.2 I VHDL instrument. In this paper we are concentrating on engendering delay. Spread postpone must be less for better execution of advanced circuit.

As appeared in table I the quantity of cut, number of LUTs, delay are acquired for the complex Vedic multiplier utilizing basic Boolean rationale viper and past calculation. From the investigation of the outcomes, it is discovered that the complex Vedic multiplier utilizing basic Boolean rationale snake gives a predominant execution as contrasted and past calculation for Xilinx programming.

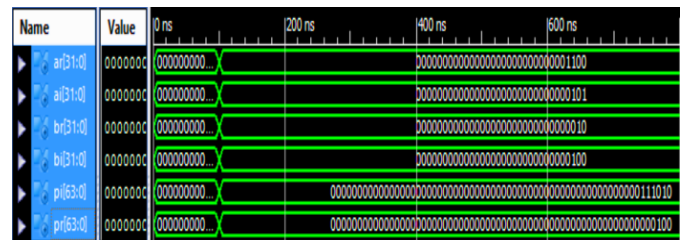


Figure 6: Output Binary Waveform of Radix-4 Complex Multiplier using Cyclic Redundant Adder

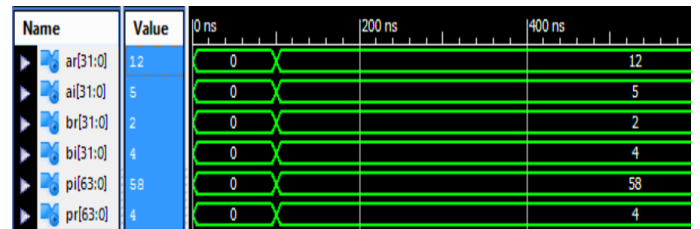


Figure 7: Output Decimal Waveform of Radix-4 Complex Multiplier using Cyclic Redundant Adder

From the analysis of the results, it is found that the complex Vedic multiplier using Cyclic Redundant adder gives a superior performance as compared with previous algorithm for Vertex-7 device family. The output waveform of the complex

multiplier using Cyclic Redundant Adder is shown in figure 6 and figure 7 respectively.

Table I: Comparison Result for 32-bit Radix-4 Complex Vedic Multiplier for four Vedic Multiplier

Design	Number of LUTs	Number of IOBs	Delay
Radix-4 Complex Vedic Multiplier [2]	10416	256	25.979 ns
Radix-4 Complex Vedic Multiplier using Ripple Carry Adder	10642	256	26.927 ns
Radix-4 Complex Vedic Multiplier using Cyclic Redundant Adder	10222	256	25.204 ns

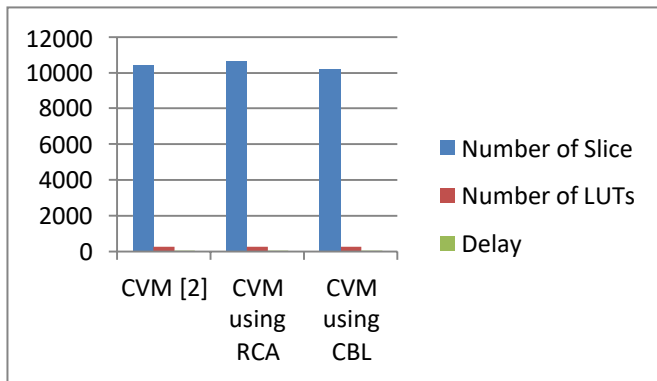


Figure 8: Bar graph of the 32-bit Radix-4 Complex Vedic multiplier

Figure 8 shows the graphical illustration of the performance of CVM using Cyclic Redundant adder discussed in this research work in term of number of slice, number of LUTs and delay. From the above graphical representation it can be inferred that the CVM using Cyclic Redundant adder gives the best performance as compared with previous algorithm.

VI. CONCLUSION

In this paper design of Cyclic Redundant Adder, Vedic multiplier, complex Vedic multiplier is presented. From implementation results it is observed that the complex booth multiplier consumes less delay compare to previous design. The architecture designs of 32 x32-bit; Modified Radix-4 Booth Encoder Multiplier is done.

REFERENCES

[1] Basant Kumar Mohanty, "Efficient Design for Fixed-Width Adder-Tree", IEEE Transactions on Circuits and Systems-II, IEEE 2018.

[2] Alberto A. Del Barrio , Román Hermida and Seda Ogrenci-Memik, "A Combined Arithmetic-High-Level Synthesis Solution to Deploy Partial Carry-Save Radix-8 Booth Multipliers in Data paths", IEEE Transactions On Circuits and Systems-I: Regular Papers, 2018.

[3] D. Kalaiyarasi and M. Saraswathi, "Design of an Efficient High Speed Radix-4 Booth Multiplier for both Signed and Unsigned Numbers", 4th International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), IEEE 2018.

[4] G. D. Licciardo, C. Cappetta, L. Di Benedetto, and M. Vigliar, "Weighted partitioning for fast multiplierless multiple-constant convolution circuit," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 64, no. 1, pp. 66–70, Jan. 2017.

[5] Elisardo Antelo Paolo Montuschi Alberto Nannarelli "Improved 64-bit Radix-16 Booth Multiplier Based on Partial Product Array Height Reduction" IEEE Transactions on Circuits And Systems-I: Regular Papers vol. 64 no. 2 February 2017.

[6] T. Adiono H. Herdian S. Harimurti "Full-Custom Design Implementation of Serial Radix-4 8-bit Booth Multiplier" The 2nd International Conference on Electrical Engineering and Computer Science (ICEECS 2016) 2016.

[7] H. Jiang J. Han F. Qiao F. Lombardi "Approximate Radix-8 Booth Multipliers for Low-Power and High-Performance Operation" IEEE Transactions on Computers vol. 65 no. 8 pp. 2638-2644 Aug. 2016.

[8] Basant Kumar Mohanty, and Pramod Kumar Meher, High-Performance FIR Filter Architecture for Fixed and Reconfigurable Applications", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 78, No.06, April 2016.

[9] K. Deerga Rao, Ch. Gangadhar and Praveen K Korrai, "FPGA Implementation of Complex Multiplier Using Minimum Delay Vedic Real Multiplier Architecture", 2016 IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering (UPCON), IEEE 2016.

[10] Indranil Hatai, Indrajit Chakrabarti, and Swapna Banerjee, "An Efficient VLSI Architecture of a Reconfigurable Pulse-Shaping FIR Interpolation Filter for Multi-standard DUC", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 23, No. 6, June 2015.

[11] Sang Yoon Park and Pramod Kumar Meher, "Efficient FPGA and ASIC Realizations of DA-Based Reconfigurable FIR Digital Filter", IEEE Transactions on Circuits And Systems-II: Express Briefs, 2014.

[12] B. K. Mohanty, P. K. Meher, S. Al-Maadeed, and A. Amira, "Memory footprint reduction for power-efficient realization of 2-D finite impulse response filters," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 61, no. 1, pp. 120–133, Jan. 2014.

[13] B. K. Mohanty and P. K. Meher, "A high-performance energy-efficient architecture for FIR adaptive filter based on new distributed arithmetic formulation of block LMS algorithm," IEEE Trans. Signal Process., vol. 61, no. 4, pp. 921–932, Feb. 2013.

[14] K. Deerga Rao, Ch. Gangadhar and Praveen K Korrai, "FPGA Implementation of Complex Multiplier Using Minimum Delay Vedic Real Multiplier Architecture", 2016 IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering (UPCON), IEEE 2016.