Volume 7, Issue 5, May 2017 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

LUT Optimization Using APC and OMS Techniques

K. Vijaya Bharathi
M.Tech (DECS) SPMVV,
Tirupathi, India

K. Phanindra
M.Tech (ES) KITS,
Khammam, India

A. Sravanthi
M.Tech (DECE) GNITS,
Hyderabad, India

DOI: 10.23956/ijarcsse/SV7I5/0268

***Abstract:** Recently, we have proposed the anti symmetric product coding (APC) and odd-multiple-storage (OMS) techniques for lookup-table (LUT) design for memory-based multipliers to be used in digital signal processing applications. Each of these techniques results in the reduction of the LUT size by a factor of two. In this brief, we present a different form of APC and a modified OMS scheme, in order to combine them for efficient memory-based multiplication. The proposed combined approach provides a reduction in LUT size to one-fourth of the conventional LUT. We have also suggested a simple technique for selective sign reversal to be used in the proposed design. It is shown that the proposed LUT design for small input sizes can be used for efficient implementation of high-precision multiplication by input operand decomposition. It is found that the proposed LUT-based multiplier involves comparable area and time complexity for a word size of 8 bits, but for higher word sizes, it involves significantly less area and less multiplication time than the canonical-signed-digit (CSD)-based multipliers. For 16- and 32-bit word sizes, respectively, it offers more than 30% and 50% of saving in area–delay product over the corresponding CSD multipliers.*

***Keywords:** Digital signal processing (DSP) chip, lookup- table (LUT)-based computing, memory-based computing, very large scale integration (VLSI).*

I. INTRODUCTION

Registering with memory stages are regularly used to give the advantage of equipment reconfigurability. Reconfigurable figuring stages offer points of interest as far as lessened plan cost, early time-to-market, fast prototyping and effortlessly adaptable equipment frameworks. Duplication in twofold is like its decimal partner. Two numbers A and B can be duplicated by halfway items: for every digit in B, the result of that digit in A is computed and composed on another line, moved leftward so that its furthest right digit lines up with the digit in B that was utilized. The whole of all these fractional items gives the last outcome. Delicate multipliers area to a great degree adaptable other option to utilizing DSP squares. Rather than actualizing a combinatorial rationale multiplier, they use a novel execution in view of a fractional look-into table (LUT) usage of the increase operation, where the LUT is executed in the memory squares. Delicate multipliers increment by an element of in the vicinity of 2 and 15 the quantity of multipliers accessible. By downloading distinctive coefficient LUTs, diverse setups of multipliers and adders are created. An ordinary query table (LUT) - based multiplier is appeared in underneath figure, where A will be a settled coefficient, and X is an information word to be increased with A. Accepting X to be a positive double number of word length L, there can be 2L conceivable estimations of X, and in like manner, there can be 2L conceivable estimations of item C = A • X.

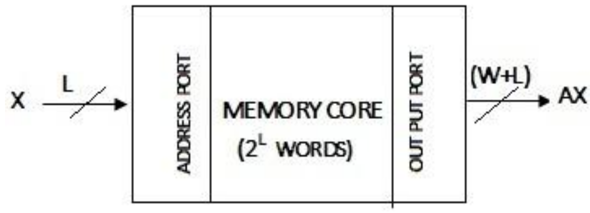

Fig1: Conventional LUT-based multiplier.

Therefore, for memory-based augmentation, a LUT of 2L words, comprising of precomputed item values relating to every conceivable estimation of X, is expectedly utilized. The item word A • Xi is put away at the area Xi for 0 ≤ Xi ≤ 2L − 1, with the end goal that if a L-bit double estimation of Xi is utilized as the address for the LUT, then the relating item esteem A • Xi is accessible as its yield.

II. APC TECHNIQUE

A few structures have been accounted for in the writing for memory-based execution of DSP calculations including orthogonal changes and advanced channels. Be that as it may, we don't locate any critical work on LUT improvement for memory-based augmentation. As of late, we have introduced another way to deal with LUT outline, where just the odd products of the settled coefficient are required to be put away , which we have alluded to as the odd

© 2017, IJARCSSE All Rights Reserved

Multiple storage (OMS) conspire. What's more, we have demonstrated that, by the anti symmetric product coding (APC) approach, the LUT size can likewise be lessened to half, where the item words are recoded as subterranean insect symmetric sets. For straightforwardness of introduction, we accept both X and A to be certain whole numbers. The item words for various estimations of X for L = 5 are appeared in Table I. It might be seen in this Table I that the information word X on the primary segment of each line is the two's supplement of that on the third segment of a similar line. The whole of item values comparing to these two information values on a similar column is 32A. Let the item values on the second and fourth segments of a column be u and v, individually.
Since one can compose
$u = [(u + v)/2 − (v − u)/2]$ and
$v = [(u + v)/2 + (v−u)/2]$,
For $(u + v) = 32A$, we can have
$u = 16A − [(v-u)/2]$
$v = 16A + [(v-u)/2]$.

Table 1: APC words for different input values for l=5

input X	Product values	Input X	Product values	Addresses X3'X2'X1'X0'	APC words
00001	A	11111	31A	1111	15A
00010	2A	11110	30A	1110	14A
00011	3A	11101	29A	1101	13A
00100	4A	11100	28A	1100	12A
00101	5A	11011	27A	1011	11A
00110	6A	11010	26A	1010	10A
00111	7A	11001	25A	1001	9A
01000	8A	11000	24A	1000	8A
01001	9A	10111	23A	0111	7A
01010	10A	10110	22A	0110	6A
01011	11A	10101	21A	0101	5A
01100	12A	10100	20A	0100	4A
01101	13A	10011	19A	0011	3A
01110	14A	10010	18A	0010	2A
01111	15A	10001	17A	0001	1A
10000	16A	10000	16A	0000	0

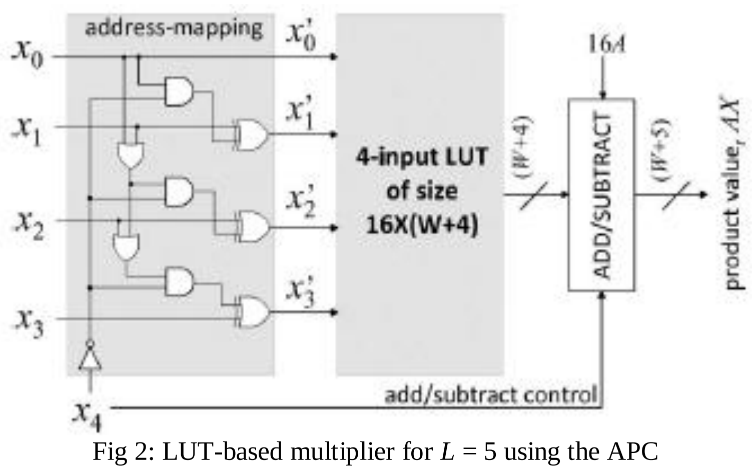

Fig 2: LUT-based multiplier for *L* = 5 using the APC

This conduct of the item words can be utilized to lessen the LUT measure, where, rather than Storing u and v, just [(v − u)/2] is put away for a couple of contribution on a given line. The 4-bit LUT addresses and relating coded words are recorded on the fifth and 6th segments of the table, individually. The item values on the second and fourth sections of Table I in this manner have negative mirror symmetry. This conduct of the item words can be utilized to lessen the LUT measure, where, rather than putting away u and v, just [(v − u)/2] is put away for a couple of contribution on a given column. The 4-bit LUT addresses and comparing coded words are recorded on the fifth and 6th sections of the table, individually.

© 2017, IJARCSSE All Rights Reserved

Since the portrayal of the item is gotten from the anti symmetric conduct of the items, we can name it as anti symmetric item code. The 4-bit address X' = (x3'x2'x1'x0') of the APC word is given by $X' = \begin{cases} X_L, & \text{if } x_4 = 1 \\ X'_L, & \text{if } x_4 = 0 \end{cases}$

Where XL = (x3x2x1x0) is the four less critical bits of X, and X'L is the two's supplement of XL. The coveted item could be acquired by including or subtracting the put away esteem (v − u) to or from the settled esteem 16A when x4 is 1or 0, separately, i.e.,

Product word = 16A + (sign esteem) × (APC word) (3)

Where sign esteem = 1 for x4 = 1 and sign esteem = −1 for x4 = 0.

The product value for X = (10000) compares to APC esteem "zero," which could be determined by resetting the LUT yield, rather than putting away that in the LUT. The structure and capacity of the LUT-based multiplier for L = 5 utilizing the APC method is appeared in Fig. 2.It comprises of a four-input LUT of 16 words to store the APC estimations of item words as given in the 6th section of Table I, aside from on the last line, where 2A is put away for info X = (00000) rather than putting away a "0" for information X = (10000). In addition, it comprises of an address-mapping circuit and an include/subtract circuit. The address-mapping circuit produces the coveted address (x3'x2'x1'x0') as per (2). A clear execution of address mapping should be possible by multiplexing XL and XL' utilizing x4 as the control bit. The address-mapping circuit, be that as it may, can be improved to be acknowledged by three XOR entryways, three AND doors, two OR entryways, and a NOT entryway, as appeared in beneath figure. Take note of that the RESET can be produced by a control circuit (not appeared in this figure) as per (4). The yield of the LUT is included with or subtracted from 16A, for x4 = 1 or 0, individually, as indicated by (3) by the include/subtract cell. Thus, x4 is utilized as the control for the include/subtract cell.

III. OMS TECHNIQUE

The APC approach, in spite of the fact that giving a lessening in LUT estimate by a variable of two, fuses significant overhead of territory and time to play out the two's supplement operation of LUT yield for sign change and that of the information operand for information mapping. Be that as it may, we find that when the APC approach is joined with the OMS procedure, the two's supplement operations could be particularly streamlined since the info address and LUT yield could simply be changed into odd whole numbers. Nonetheless, the OMS method in can't be joined with the APC plot, since the APC words produced concurring are odd numbers. Besides, the OMS plot does not give a productive usage when consolidated with the APC system. In this concise, we consequently introduce an alternate type of APC and consolidated that with an adjusted type of the OMS conspire for productive memory-based augmentation.

It is demonstrated that, for the augmentation of any twofold word X of size L, with a settled coefficient A, rather than putting away all the 2L conceivable estimations of C = A ・ X, just (2L/2) words comparing to the odd products of A might be put away in the LUT, while all the even products of A could be inferred by left-move operations of one of those odd products. In light of the above suspicions, the LUT for the increase of an L-bit contribution with a W-bit coefficient could be planned by the accompanying system.

1) A memory unit of [(2L/2) + 1] expressions of (W + L) - bit width is utilized to store the item values, where the initial (2L/2) words are odd products of an, and the last word is zero.

2) A barrel shifter for delivering a most extreme of (L − 1) left moves is utilized to infer all the even products of A.

3) The L-bit input word is mapped to the (L − 1)-bit address of the LUT by an address encoder, and control bits for the barrel shifter are inferred by a control circuit.

Table 2: OMS-Based Design of the LUT of APC Words For L=5

Input X' X3'X2'X1'X0'	Product values	#number of shifts	Shifted input X"	Stored APC word	Address $d_3'd_2'd_1'd_0'$
0001	A	0	0001	P0=A	0000
0010	2XA	1			
0100	4XA	2			
1000	8XA	3			
0011	3A	0	0011	P1=3A	0001
0110	2X3A	1			
1100	4X3A	2			
0101	5A	0	0101	P2=5A	0010
1010	2X5A	1			
0111	7A	0	0111	P3=7A	0011
1110	2X7A	1			
1001	9A	0	1001	P4=9A	0100
1011	11A	0	1011	P5=11A	0101
1101	13A	0	1101	P6=13A	0110
1111	15A	0	1111	P7=15A	0111

In Table II, we have demonstrated that, at eight memory areas, the eight odd products, A × (2i + 1) are put away as Pi, for i = 0, 1, 2. . . 7. The even products 2A, 4A, and 8A are determined by left-move operations of A. So also, 6A and 12A are determined by left moving 3A, while 10A and 14A are inferred by left moving 5A and 7A, separately. A barrel shifter for creating a most extreme of three remaining movements could be utilized to determine all the even products of A.

As required by (3), the word to be put away for X = (00000) is not 0 but rather 16A, which we can get from A by four remaining movements utilizing a barrel shifter. Be that as it may, if 16A is not gotten from a, exclusive a most extreme of three remaining movements is required to get all other even products of A. A most extreme of three piece movements can be executed by a two-organize logarithmic barrel shifter, however the usage of four movements requires a three-arrange barrel shifter. In this way, it would be a more proficient system to store 2A for information X = (00000), so that the item 16A can be determined by three math left moves.

IV. APC–OMS COMBINED TECHNIQUE

The proposed APC–OMS combination technique of the LUT for L = 5 and for any coefficient width W is appeared in underneath Fig. It comprises of a LUT of nine expressions of (W + 4)- bit width, a four-to-nine-line address decoder, a barrel shifter, an address era circuit, and a control circuit for creating the RESET flag and control word (s1s0) for the barrel shifter.

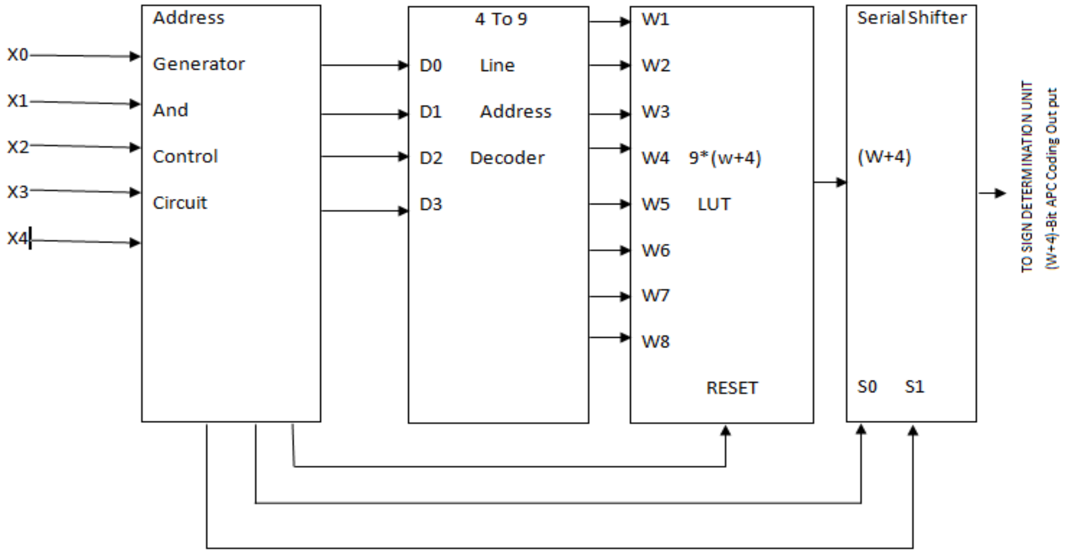

Fig 4.1: Block Diagram Of combined APC–OMS Techniques

The precomputed estimations of A × (2i + 1) are put away as Pi, for i = 0, 1, 2, . . . , 7, at the eight continuous areas of the memory cluster, as determined in Table II, while 2A is put away for information X = (00000) at LUT address "1000," as indicated in Table III. The decoder takes the 4-bit address from the address generator and produces nine word-select signs, i.e., {wi, for $0 \leq i \leq 8$}, to choose the referenced word from the LUT. The 4-to-9-line decoder is a basic change of 3-to-8-line decoder, as appeared in underneath Fig (a). The control bits s0 and s1 to be utilized by the barrel shifter to deliver the coveted number of movements of the LUT yield are produced by the control circuit, as indicated by the relations.

Take note of that (s1s0) is a 2-bit paired likeness the required number of movements indicated in Tables II and III. The RESET flag given by (4) can on the other hand be produced as (d3 AND x4). The control circuit to produce the control word and RESET is appeared in beneath Fig (b). The address-generator circuit gets the 5-bit input operand X and maps that onto the 4-bit address word (d3d2d1d0), as indicated by (5) and (6).

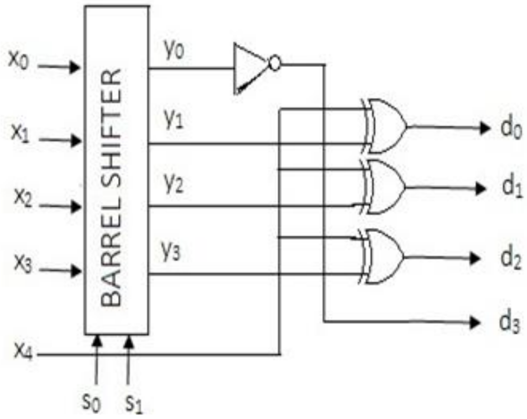

Fig 4.2: Address Generation Unit

© 2017, IJARCSSE All Rights Reserved

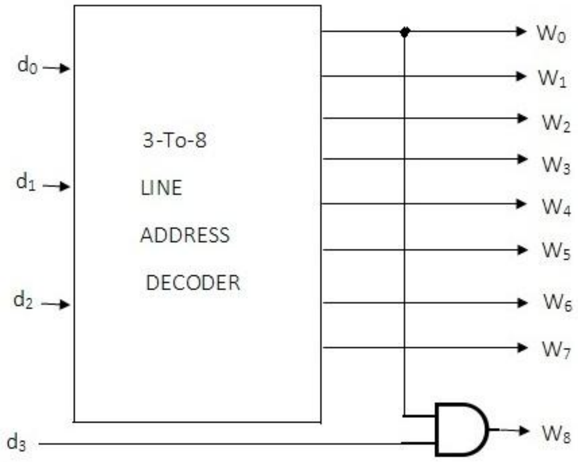

Fig 4.3: Four-to-nine-line address-decoder.

V. CONCLUSION

The proposed LUT multipliers for word measure L = W = 5 and 6 bits are coded in Verilog and combined in Xilinx ISE 10.1i. Reenactment Part is done in Modelsim 6.4b, where the LUTs are actualized as varieties of constants, and increments are executed by the Wallace tree and swell convey exhibit. The CSD-based multipliers having a similar expansion plans are likewise integrated with a similar innovation library. We have demonstrated the likelihood of utilizing LUT based multipliers to execute the consistent increase for DSP applications.

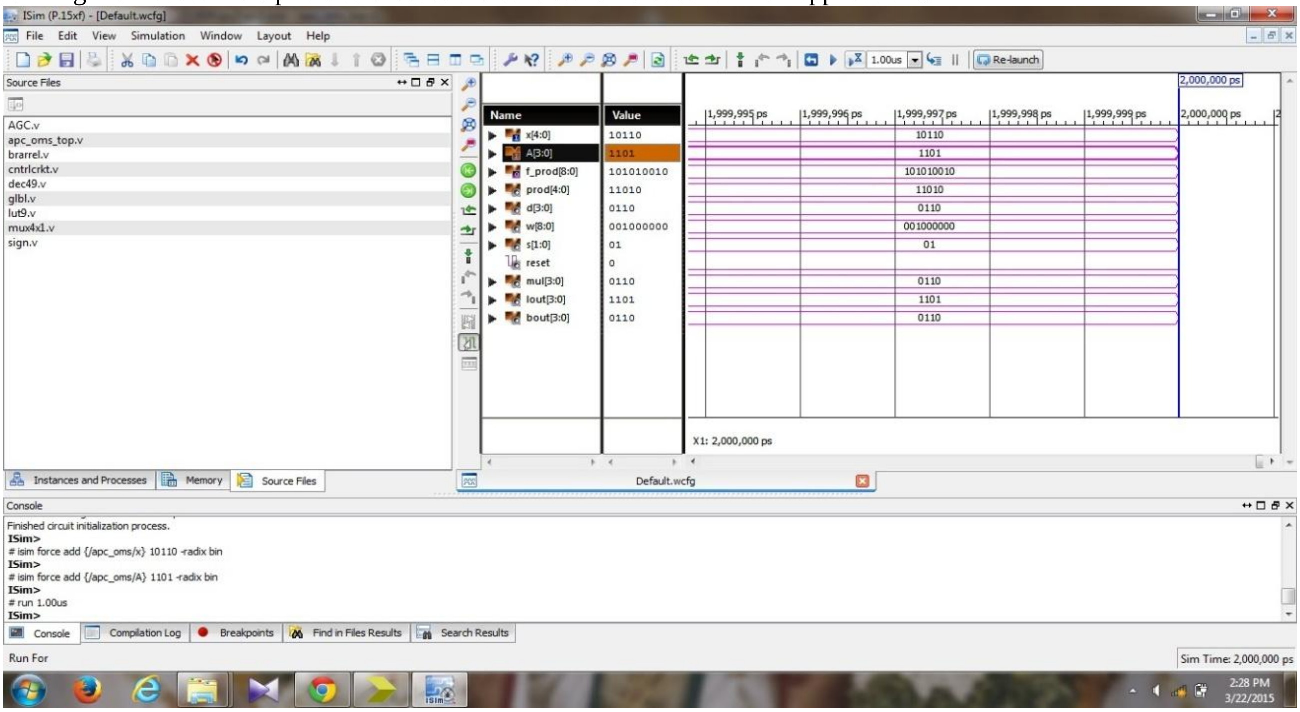

Fig 5: simulation results of APC & OMS technique.

VI. FUTURESCOPE

FPGAs and other programmable rationale exhibits are exceedingly configurable. Additionally work could even now be done to determine such adjusted OMS based LUTs for higher info sizes with various disintegration shapes. Other parallel and pipelined expansion plans for appropriate zone postpone tradeoffs. The LUT multipliers for word estimate L = W = 8, 16, and 32 bits can be coded and orchestrating utilizing Xilinx ISE 12.2i. For the Simulation Part we will utilize Modelsim 6.4b for More Less Area and Less Multiplication Time.

REFERENCES

[1] International Technology Roadmap for Semiconductors. [Online].Available: http://public.itrs.net/

[2] J.-I. Guo, C.-M. Liu, and C.-W. Jen, "The efficient memory-based VLSI array design for DFT and DCT," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process*, vol. 39, no. 10, pp. 723–733, Oct. 1992.

[3] H.-R. Lee, C.-W. Jen, and C.-M. Liu, "On the design automation of the memory-based VLSI architectures for FIR filters," *IEEE Trans. Consum. Electron.* vol. 39, no. 3, pp. 619–629, Aug. 1993.

[4] D. F. Chiper, M. N. S. Swamy, M. O. Ahmad, and T. Stouraitis, "A systolic array architecture for the discrete sine transform," *IEEE Trans. Signal Process.*, vol. 50, no. 9, pp. 2347–2354, Sep. 2002.

© 2017, IJARCSSE All Rights Reserved

[5] H.-C. Chen, J.-I. Guo, T.-S. Chang, and C.-W. Jen, "A memory-efficient realization of cyclic convolution and its application to discrete cosine transform," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 3, pp. 445–453, Mar. 2005.

[6] D. F. Chiper, M. N. S. Swamy, M. O. Ahmad, and T. Stouraitis, "Systolic algorithms and a memory-based design approach for a unified architecture for the computation of DCT/DST/IDCT/IDST," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 52, no. 6, pp. 1125–1137, Jun. 2005.

[7] P. K. Meher, "Systolic designs for DCT using a low-complexity concurrent convolutional formulation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 9, pp. 1041–1050, Sep. 2006.

[8] P. K. Meher, "Memory-based hardware for resource-constrained digital signal processing systems," in *Proc. 6th Int. Conf. ICICS*, Dec. 2007, pp. 1–4.

[9] P. K. Meher, "New approach to LUT implementation and accumulation for memory-based multiplication," in *Proc. IEEE ISCAS*, May 2009, pp. 453–456.

[10] P. K. Meher, "New look-up-table optimizations for memory-based multiplication, "in *Proc. ISIC*, Dec. 2009, pp. 663–666.

[11] A. K. Sharma, Advanced Semiconductor Memories: Architectures, Designs, and Applications. Piscataway, NJ: IEEE Press, 2003.

© 2017, IJARCSSE All Rights Reserved