Fixed-Width Multiplier Circuits Using Column Bypassing and Decompositon Logic Techniques

S. Balamurugan and P. S. Mallick

School of Electrical Engineering, VIT University, Vellore, India-632014
sbalamurugan@vit.ac.in, psmallick@vit.ac.in

Abstract: This paper presents, low power signed and unsigned fixed-width multipliers using the column bypassing technique with carry save adder array structure. We have decomposed the partial products into two parts and executed them in parallel to reduce the delay of proposed fixed-width array multiplier. The proposed multiplier reduces the power consumption by skipping the unwanted switching activity when the multiplicand operand consists of a number of zeros. This work evaluates the power, delay and area of fixed-width multipliers and shows that the proposed array multiplier consumes less power and has reduced delay compared to the conventional fixed-width array multipliers.

Keywords: Fixed-width multipliers, low power, signed array multiplier, column bypassing, parallel architecture

1. Introduction

The basic requirements of today's high-performance computing systems are high speed, lower power and lesser area. Precision multipliers [1] are promising building blocks of next generation DSP applications which are computationally expensive, consume more power and time. Performance of a system is generally determined by the performance of multipliers. There are two possible ways to speed up the multiplication, either by reducing the number of partial products [2] or by accelerating their accumulation. To accelerate the partial product accumulations, two different parallel multiplier architectures are proposed in literatures, namely array based multipliers [3-4] and tree based multipliers [5-6].

Although the array architecture is considerably slower than a tree structure, it is still a preferred design technique in industry due to various reasons. An array multiplier is regular in its structure and uses only short wires that go from one full adder to horizontally, vertically, or diagonally adjacent full adders [4]. Thus, it has a simple and efficient VLSI layout in contrast to tree structures that are highly irregular and have complex routing networks. Furthermore, it can be easily and efficiently pipelined after every CSA or after every few rows. Additionally, in general, tree architectures dissipate more power and take more area due to interconnect wires. Hence it is not recommended for low power applications [7-9]. Therefore, we have considered an array based multiplier and also compared its performance with other architectures.

In many DSP and multimedia applications, it is desirable to have a same output operand width as that of the input operand width. These techniques are called truncated output or fixed-width output. In multiplier design, the fixed-width (FW) output operand is obtained either by means of post-truncation method (PTM) or direct truncation method (DTM). In PTM, we get the best accuracy by forming all the partial products and round off to get the final product. However, the PTM multiplier consumes large power consumption and silicon area. In the DTM, n-least significant partial product columns are removed directly. By doing so, a truncation error will be introduced into the output while savings in significant power consumption, hardware complexity reduction, and timing are achieved.

Intense research has been done to develop efficient multipliers with fixed-width output [10-16], [20] which provide output at a required degree of precision without excessive hardware

Received: March 6th, 2014. Accepted: November 16th, 2015
DOI: 10.15676/ijeei.2015.7.4.9

requirements. Analysis shows that the zero average probability of input signals is over 73.8% [18] in the conventional DSP and multimedia applications. When the zero partial products are added, large number of signal transitions were generated which are unnecessary and do not contribute to the final product. Many researchers have designed full width multipliers using both row bypassing and column bypassing technique but design of fixed-width multipliers (FWM) using bypassing technique has not received much attention. Recently our group proposed a design of low power FWMs using a bypassing technique [19-20]. In [19], we have designed an unsigned low power FWM using the row bypassing technique, where the ripple carry adder (RCA) array is used. In this structure, carry out is propagated in the same row. This structure produces more delay due to full adder and multiplexer pair in the critical path. The delay produced by full adder and MUX pair in the critical path is significantly reduced by means of decomposing partial products into two dissimilar blocks and executing them in parallel [20]. This structure gives better results in terms of low power and reduced delay. In this paper, we present an unsigned as well as signed FWMs that use a column bypassing technique, in which we have used carry save adder (CSA) array, where, carry out is propagated to next row diagonally. Using column bypassing adder (CBA) cell, the entire adders of a particular column are disabled when the value of the multiplicand bit is zero. Thus we adopt the column bypassing adder [17] cell in the FWM where the distribution of 0s and 1s is not uniform. Our proposed structure using a column bypassing technique achieves significant reduction in power consumption and delay as compared to PTM in [17] as well as conventional fixed width array multipliers in [12], [14] and fixed-width multipliers using row bypassing technique [20].

In our previous work [20], we have implemented FWMs using row bypassing technique (RBT) and compared with post-truncated multiplier (PTM) using RBT [17] and FWMs of [12]. But as per our knowledge, the performance comparison of FWMs with all other FWMs like array based signed FWMs, unsigned FWMs, FWMs using RBT, PTMs using column bypassing technique (CBT), Tree based FWMs are not available in a single paper. In this paper, we have presented a detail comparison and it is shown that the performance of our FWMs using CBT is better in terms of power and delay as compared to all these architectures except tree based architecture, where delay is less. The previous work was done using RBT with ripple carry adder (RCA) array structure, whereas here it is implemented using CBT with carry save adder (CSA) array structure.

This paper is organized as follows. In the next section, we present the fundamentals of fixed-width parallel array multiplier design, followed by the description of the column bypassing technique for FWM design in section 3. The structure of the proposed unsigned as well as signed FWM using column by passing architecture is described in section 4. We discuss the experimental results of proposed as well as conventional architectures in section 5. Conclusions are given in section 6.

2. Fundamentals Of Fixed-Width Parallel Array Multiplier

When n-bit outputs are required, the most possible choice is FWMs as they do not have a least significant product (LP). This halves the final circuit compared to the full-width multipliers [1]. Parallel array multipliers with an unsigned n - bit inputs a and b will produce a 2n-bit product P

$$P = A.B = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1}\left(a_i.b_j\right)2^{i+j} = \sum_{k=0}^{2n-1} P_k\, 2^k \qquad (1)$$

where a_i, b_j and P_k denotes the i^{th} bit of A, the j^{th} bit of B and k^{th} bit of P. The full-length of the partial product can be divided into two subsets called most significant product (MP) and the least significant product (LP) as shown in Figure 1.

So, we can write as

$$P = MP + LP \qquad (2)$$

where,

$$MP = \sum_{j=1}^{n-1} b_j 2^j \sum_{i=n-j}^{n-1} a_i 2^i = \sum_{k=n}^{2n-1} P_k 2^k \quad (2a)$$

$$LP = \sum_{i=0}^{n-1} a_i 2^i \sum_{j=0}^{n-1-i} b_j 2^j = \sum_{k=0}^{n-1} P_k 2^k \quad (2b)$$

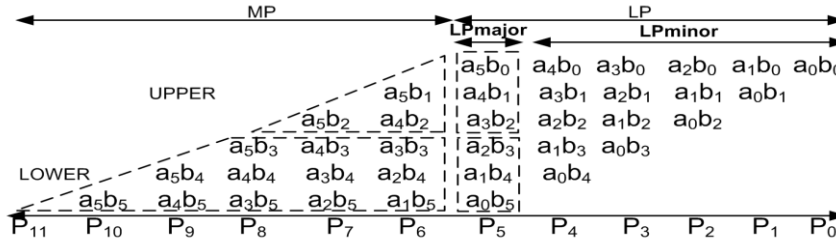

Figure 1. Separation of partial products of a 6x6 multiplier into MP and LP

By removing LP part directly, we get fixed-width output. This solution is very advantageous in terms of hardware performance. However, a direct truncation of LP part introduces a biased error in the final output. To reduce this error, we have two correction techniques in the literature, namely constant correction and variable correction techniques were implemented in [10-15], [21]. Even though the constant correction bias technique [10-11] is simpler than variable correction bias technique [12-15], the truncation error is more since the bias cannot be adjusted adaptively by the input vectors. Since the accuracy of fixed-width multiplier is a trade-off with area cost, we have the different fixed-width multipliers in the literature [21]. The accuracy of truncated multiplier was enormously sharpened by adopting the LPmajor column in the variable-correction truncated multipliers [12-13]. The number of columns considered to generate the bias value in [14] is more for better accuracy which makes a large difference in reducing power and area. Ko and Hsiao jointly considered the tree reduction, truncation and rounding of partial product bits during the design of fast parallel truncated multipliers to achieve faithful rounding [16]. In our proposed work, we have divided the LPmajor column into two parts. As we are using parallel architecture the upper and lower parts of LPmajor partial products are used to generate the bias value of upper and lower blocks of the MP part respectively. Here, we utilize the same partial products (LPmajor column) for AND-OR cell combinations, where AND gate output goes to carry input of MP part and OR gate output propagate downwards as in [12] to generate the error compensation bias value and hence we assume that our proposed architecture will also have the same accuracy [12].

The different compensation functions are generally calculated from a subset of LPminor terms, thus resulting more accuracy with the penalty of more power consumption and area cost.

3. Column Bypassing Fixed-Width Array Multiplier

Power dissipation can be reduced in many DSP or other related applications at the cost of certain expenses. Power dissipation can be divided into dynamic and static power dissipation. The dynamic power consumption dominates total power consumption. The most effective way of reducing dynamic power consumption is by reducing the switching activity. To reduce this switching activity we consider the modified full adder cell [17] with bypassing logic.

In a multiplier design, if the input bit coefficient is zero, the corresponding row or column of adders need not be activated. If the multiplicand operand contains more zeros, then higher power reduction can be achieved using column bypassing multiplier [17]. If the multiplier operand contains more zeros than higher power reduction can be achieved using row bypassing multiplier [18].

We propose a fixed width array multiplier design with a CBA cell for an unsigned as well as signed numbers. We adopt CBA [17] as shown in figure 2 (a) to achieve high power

reductions. In this approach, the operations in a column can be disabled if the corresponding bit in the multiplicand is zero. This adder cell consists of two tri-state buffers and one multiplexer for achieving the bypass logic. Consider the partial product bit $a_i.b_j$ if the value of a_i is '0' then the operations in the corresponding diagonal can be disabled since all the outputs are known.

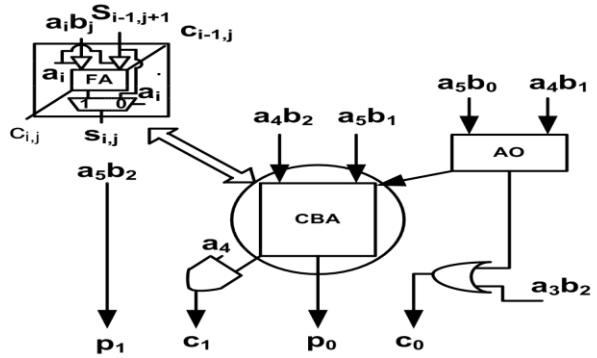

Figure 2. (a) A 6 x 3 unsigned fixed-width multiplier using CBA (Upper block)

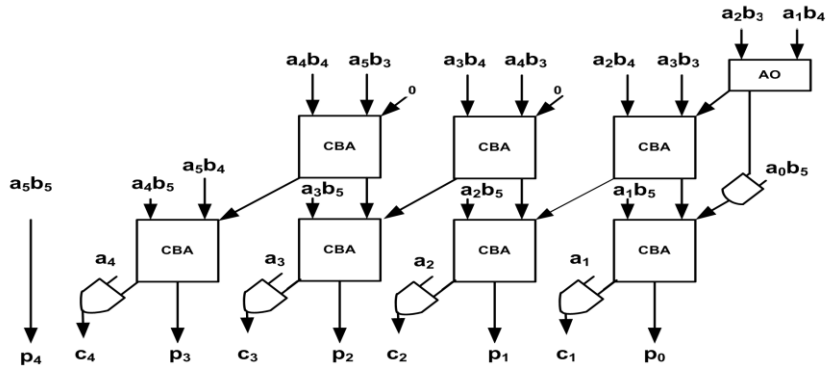

Figure 2. (b) A 6 x 3 unsigned fixed-width multiplier using CBA (Lower block)

4. Proposed Architecture

Unsigned fixed-width Multiplier design using column bypassing

In array based multipliers using column bypassing technique, it is observed that the critical path consists of two parts: vertical and horizontal. In the proposed architecture vertical critical path produced more delay as compared horizontal critical path, since in the vertical critical path we used CBA cell as compared to normal FA cell in the horizontal critical path. To reduce this vertical critical path delay, we decomposed the partial products into two dissimilar blocks and executed in parallel. The parallelism is implemented by adopting column-bypassing technique with an error compensator circuit that reduces the switching activity of the fixed-width multiplier when the value of the multiplicand is zero. Figure 2 (a) and figure 2 (b) show the internal structure of the upper block (upper part of the MP along with LPmajor upper part) and lower block (lower part of the MP along with *LPmajor* lower part) respectively. Thus, the partial sums and carry output from these two blocks are computed simultaneously. The output of these two blocks is given to set of full adder and half adder called vector merging adder (VMA) as shown in figure 3. Here, the addition operation in the $(i+1)^{th}$ column can be bypassed if the bit a_i is 0 which makes all partial products $a_i.b_j$ also 0.

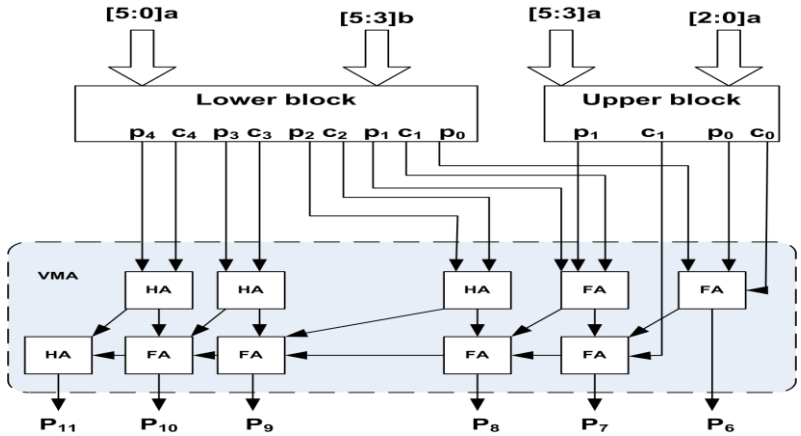

Figure 3. A 6x6 Unsigned fixed- width multiplier based on CBA

Now, we set the carry outputs of a bottom array of each block of carry save adder (CSA) as '0' because of the corresponding full adders may not produce the correct outputs when the value of a_i is zero. For this, an AND gate is added at the outputs of the last-row of the CSA array as shown in figure 2 (a) and figure 2 (b).

Signed fixed-width multiplier design using column bypassing

The fixed-width multiplier described in the previous section is used to compute the unsigned numbers. However, in most of the consumer electronics applications, we require the signed multipliers. Modified form of Baugh-Wooley two's complement signed multiplier is more preferable signed array multiplier, because it maximizes the regularity of multiplier logic and allows all the partial products to have positive sign bits. Only the AND gate to NAND gate for corresponding operand is changed and inverter is inserted at the final carry output.

For example, two signed 6-bit binary numbers $A = a_5a_4a_3a_2a_1a_0$ and $B = b_5b_4b_3b_2b_1b_0$ can generate a product P, which can be defined as follows:

$$\begin{aligned} P &= 1\times 2^{11} + a_5b_5 2^{10} + \left(\overline{a5b4} + \overline{a4b5}\right)2^9 + \left(\overline{a5b3} + \overline{a3b5}\right)2^8 \\ &+ \left(\overline{a5b2} + \overline{a2b5}\right)2^7 + \left(\overline{a5b1} + \overline{a1b5} + 1\right)2^6 + \left(\overline{a5b0} + \overline{a0b5}\right)2^5 \\ &+ \left(b_4 2^4 + b_3 2^3 + b_2 2^2 + b_1 2^1 + bo\right)\bullet\left(a_4 2^4 + a_3 2^3 + a_2 2^2 + a_1 2^1 + a_0\right) \end{aligned} \tag{3}$$

The same algorithm is utilized to derive the proposed singed full-width and fixed-width multiplier using column bypassing technique. According to Baugh-Wooley algorithm [3], [13], [21], some AND gates of original design must be changed to NAND gates for the corresponding operands. In the proposed signed multiplier using bypassing technique, it is worth pointing out that, last row of the partial product matrix is in complemented form. For computing the last row partial products we have used full adder cell instead of CBA. The reason is, we are disabling the adder cells only when multiplicand operand is zero. To get the correct carry out of each column, we are AND ed the carry out the bypassed column cell at the end of each column with the corresponding multiplicand operands as the correction logic except last row. If we want to design the adder cell that to bypass the complemented partial products (last row), additional logic must be required. Power consumption of these additional logics may be large. In other words, adders in this row may dissipate more power in most of the time. Consequently, normal full adder will be used for this row of adders because it will dissipates less power. For comparison purpose, we derive the post truncation signed multiplier using column bypassing (PTMCB) method from the Baugh-Wooley algorithm and the corresponding results are tabulated in Table 2. To reduce the critical path, which is increased

due to bypassing logic added in critical path, the proposed fixed-width signed multiplier are also decomposed into two parts and executed in parallel. Internal structure of lower part (lower block) as well as the upper part (upper block) is shown in figure 4 (a) and figure 4 (b). The internal structure of lower part and upper part seems to be same for both the signed and unsigned proposed fixed-width multipliers. However, in proposed signed fixed-width multiplier, for computation of last row of operands of the lower block, we used normal CSA instead of the CBA cell and it is shown in figure 4 (b). The whole circuit architecture for a 6× 6 signed fixed-width multiplier is shown figure 5.

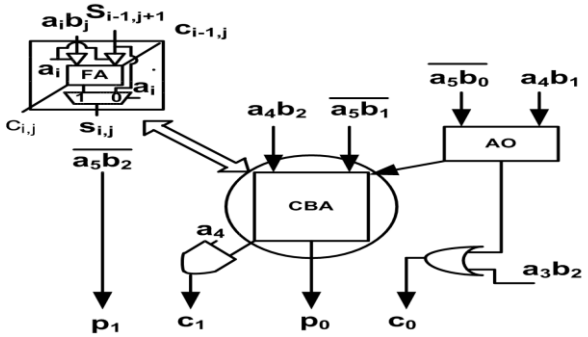

Figure 4. (a) A 6 x 3 signed fixed-width multiplier using CBA (Upper block)

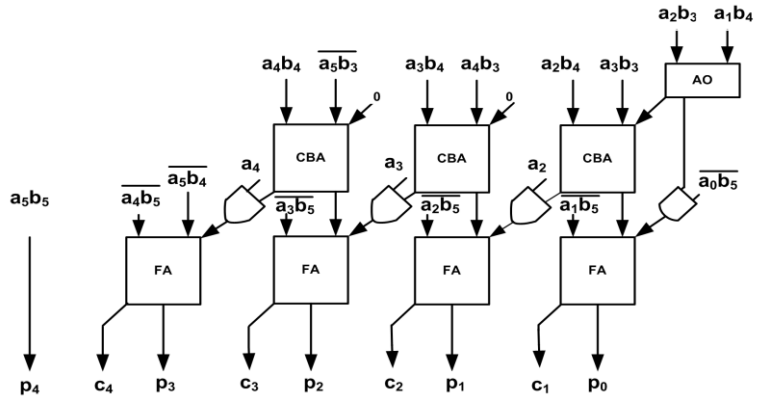

Figure 4. (b) A 6 x 3 signed fixed-width multiplier using CBA (Lower block)

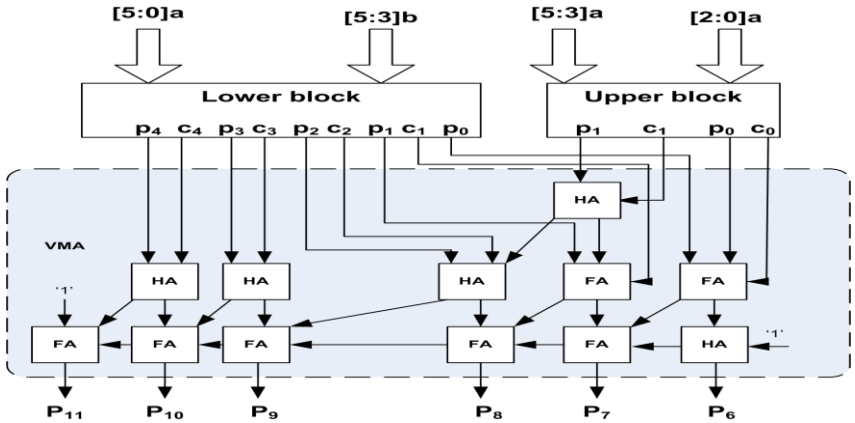

Figure 5. The proposed 6x6 signed fixed- width multiplier based on CBA

Table 1. Performance comparisons of unsigned multiplier in terms of power, delay and area

Bit Size	Architecture	Delay (ps)	Delay (%)	Power (nW)	Power (%)	Area (um^2)	Area (%)
8x8	PTM[8]	3999	100	176709	100	1259	100
	PTM[17]	5278	132	83050	47	2286	181
	[12]	3176	79	105864	60	714	57
	[14]	3454	86	121599	69	764	61
	[16]	2426	61	120069	68	859	68
	[20]	3324	83	58312	33	1093	87
	Proposed	3260	82	52434	30	1129	90
12x12	PTM[8]	4890	100	578021	100	3005	100
	PTM[17]	6970	143	188937	33	5521	184
	[12]	4827	98	313081	54	1634	54
	[14]	5105	104	356571	62	1720	57
	[16]	3262	67	345760	60	1898	63
	[20]	4808	98	114122	20	2714	90
	Proposed	4707	96	92325	16	2774	92
16x16	PTM [8]	6541	100	892018	100	5498	100
	PTM[17]	9399	144	286914	32	10090	184
	[12]	6478	99	664917	75	2941	53
	[14]	6756	103	736838	83	3049	55
	[16]	4845	74	643245	72	3263	59
	[20]	6293	96	195257	22	5015	91
	Proposed	6155	94	164370	18	5102	93

Table 2. Performance comparisons of signed multiplier in terms of power, delay and area

Bit Size	Architecture	Delay (ps)	Delay (%)	Power (nW)	Power (%)	Area (um^2)	Area (%)
8x8	PTM [3]	4221	100	176486	100	1321	100
	PTMCB#	5202	123	106424	60	2135	162
	[13]	3176	75	88793	50	672	51
	[21]*	3188	76	113481	64	840	64
	Proposed	3028	72	86497	49	1015	77
12x12	PTM [3]	4997	100	582474	100	2992	100
	PTMCB#	6896	123	215942	37	5282	177
	[13]	4872	96	296604	51	1575	53
	[21]*	4839	98	348707	60	1856	62
	Proposed	4370	87	154709	27	2570	86
16x16	PTM [3]	6628	100	1189724	100	5479	100
	PTMCB#	9326	141	340201	29	9758	178
	[13]	6522	98	607858	51	2848	52
	[21]*	6701	101	693206	58	3261	60
	Proposed	6196	93	256956	22	4811	88

*Multiplier with $\theta_{Q=0,w=1}$, #proposed post-truncated signed multiplier using column by passing

5. Experimental Results

In this section, the performance evaluation of proposed fixed-width multipliers along with the comparison to the conventional fixed-width multiplier design is presented. All these

architectures are designed and coded using Verilog HDL at the gate level and synthesized using libraries (Process: 1, Voltage: 1.2V, Temperature: 25°C) of TSMC 90nm technology. To compute the power consumption of both unsigned and signed fixed-width multipliers, we generate value change dump file for 8 x 8, 12 x 12 and 16 x 16 of proposed designs as well as previous works for 50 test patterns, which are generated randomly. The test patterns are randomly generated with uniformly distributed probability. This is then imported to Cadence RTL compiler for power, area and delay calculations.

Table 1 shows the performance comparisons of the proposed unsigned fixed-width array multipliers using the column bypassing technique with the existing unsigned multipliers work in terms of delay, power and area. The PTM [17] using the column bypassing technique has the largest area in the comparison table, which is generally acknowledged, when we want to reduce power consumption by means of reducing the switching activity. However, the delay produced by this architecture should be reduced. The percentages of delay, power and area values are normalized to PTM [8], which is an unsigned Braun array multiplier. Our proposed unsigned fixed-width multiplier using column-bypassing technique achieves power reduction by more than 70% for different bit sizes as compared to a PTM [8] and more than 3% when compared [20]. As we can see that the delay of the proposed architecture is lower when compared to the array based architecture in [8], [17], [12], [14] and [20]. But, when compared to tree based architecture [16], all the array based architectures including proposed architecture consume more delay. However, the power consumed by tree architecture [16] is more, when compared to our proposed array architectures; hence it is not recommended for low power applications. It is evident that our design outperforms in terms of both power and delay in low power applications.

Table 2 shows the comparison of the proposed signed fixed-width array multiplier using the column bypassing technique with the existing fixed-width signed array multiplier work in terms of delay, power and area. The proposed post truncated signed multiplier using the column bypassing (PTMCB) technique achieves significant power saving. However, additional hardware of adopting bypassing method increases the delay of the multiplier in the critical path. This critical path delay is reduced by adopting a parallel architecture method in the proposed signed array FWM using the column bypassing technique. As compared to existing architecture in the Table 2, the proposed fixed-width signed array multiplier using the column bypassing technique achieves significant power savings as well lesser delay for all bit sizes.

Thus, the proposed unsigned as well as signed fixed-width array multipliers consume lower power with reduced delay as compared to the conventional fixed-width array multipliers with the same or lower accuracy. In application specific processors, as discussed in section I, it can be possible to achieve more power savings if the probability of zero is more than the probability of one in the multiplicand operand by using the proposed fixed-width multiplier with the column bypassing technique.

6. Conclusion

We have presented a new unsigned and signed fixed-width array multipliers using column-bypassing technique with low power and reduced delay. Thus, the proposed architecture is more suitable for low power error-tolerant application specific processors, where the multiplicand operand contains a probability of more zero than the probability of one. Moreover, the delay of the proposed method can be further reduced by using pipelining technique.

7. Acknowledgment

Authors would like to thank Marimuthu R and Uma Sathyakam for their help and advices.

8. References

[1]. Ercegovac, M. D., and Lang, T.: Digital arithmetic. Access Online via Elsevier. 2003.

[2]. Booth, A. D, "A signed binary multiplication technique." *The Quarterly Journal of Mechanics and Applied Mathematics*, Vol. 4, no. 2, pp- 236-240, 1951.
[3]. Baugh, Charles R., and Bruce A. Wooley, "A two's complement parallel array multiplication algorithm," *Computers, IEEE Transactions on*, Vol-100, no. 12, pp. 1045-1047, 1973.
[4]. Parhami, Behrooz. Computer arithmetic: algorithms and hardware designs. Oxford University Press, Inc., 2009.
[5]. Wallace, C. S, "A suggestion for a fast multiplier," *Electronic Computers, IEEE Transactions on*, Vol. 1, pp-14-17, 1964.
[6]. Dadda, L. "Some schemes for parallel multipliers," *Alta Frequenza*, Vol. 34, no. 5, pp-349-356, 1965.
[7]. Elrabaa, M. S., I. S. Abu-Khater, and M. I. Elmasry. Advanced low-power digital circuit techniques. Kluwer academic publishers, 1997.
[8]. Koren, I. Computer arithmetic algorithms. Universities Press, 2002.
[9]. Han, C-Y, H-J Park, and L-S Kim. "A low-power array multiplier using separated multiplication technique." *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, Vol- 48, no. 9, 866-871, 2001
[10]. Lim, Y.C., "Single-precision multiplier with reduced circuit complexity for signal processing applications," *Computers, IEEE Transactions on* , Vol. 41, no. 10, pp. 1333,1336, Oct 1992.
[11]. Schulte, M.J.; Swartzlander, E.E., "Truncated multiplication with correction constant [for DSP]," *VLSI Signal Processing, VI, [Workshop on]* , pp.388,396, 20-22 Oct 1993.
[12]. Jou, J- M, and Kuang, S-R. "Design of low-error fixed-width multiplier for DSP applications." *Electronics Letters*, Vol. 33, no. 19, 1597-1598, 1997.
[13]. Jou, J. M., Kuang, S. R., and Der Chen, R., "Design of low-error fixed-width multipliers for DSP applications," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, vol. 46, no. 6, pp. 836-842, 1999.
[14]. Wey, I. C., and Wang, C. C., "Low-error and hardware-efficient fixed-width multiplier by using the dual-group minor input correction vector to lower input correction vector compensation error," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, Vol. 20, no. 10, pp. 1923-1928, 2012.
[15]. Wang, J. S., Kuo, C. N., and Yang, T. H., "Low-power fixed-width array multipliers," in *Proc. of the Int. Symposium on Low power electronics and design*, August, 2004, pp. 307-312.
[16]. Hou-Jen Ko, Shen-Fu Hsiao, "Design and Application of Faithfully Rounded and Truncated Multipliers With Combined Deletion, Reduction, Truncation, and Rounding," *Circuits and Systems II: Express Briefs, IEEE Transactions on* , vol.58, no.5, pp.304,308, May 2011
[17]. Wen, M. C., Wang, S. J., and Lin, Y. N., "Low-power parallel multiplier with column bypassing," *Electronics Letters*, Vol. 41, no. 10, pp. 581-583, 2005.
[18]. Kuo, K. C., and Chou, C. W., "Low power and high speed multiplier design with row bypassing and parallel architecture," *Microelectronics Journal*, Vol. 41, no. 10, pp. 639-650, 2010.
[19]. Balamurugan, S., Balakumaran Srirangaswamy, R. Marimuthu, and P. S. Mallick, "FPGA design and implementation of truncated multipliers using bypassing technique," In *Proceedings of the International Conference on Advances in Computing, Communications and Informatics*, pp. 1111-1117. ACM, 2012.
[20]. Balamurugan, S., Ghosh, S., Balakumaran, S., Marimuthu, R., and Mallick, P. S., "Design of low power fixed-width multiplier with row bypassing," *IEICE Electronics Express*, Vol. 9, no. 20, pp. 1568-1575. 2012.
[21]. Van, L-D., and C-C. Yang. "Generalized low-error area-efficient fixed-width multipliers." *Circuits and Systems I: Regular Papers, IEEE Transactions on*, Vol- 52, no. 8, pp-1608-1619, 2005.

S. Balamurugan was born in Tamil Nadu, India, on May 9, 1978. He received his Bachelor degree in Electrical and Electronics Engineering from Bharathiar University, India, in 2001 and his M.E. degree in Applied Electronics from Anna University, India, in 2005. He has been working with Microelectronics division at School of Electrical Engineering, VIT University, Vellore, as a Assistant Professor (Senior) and working toward Ph.D. degree. His current research interest includes low power and high speed architectures. He is a member of IEEE, IEICE, and Life member of IETE.

P. S. Mallick, received Ph.D and MSc degree from Jadavpur University, India and University of Chittagong, Bangladesh respectively. He worked 4.5 years in a Sweden based industry *IAAB Electronics* as the Technical Head. He has 16 years of Teaching experience where he led various research teams and developed "Online Lab in Microelectronics", "Monte Carlo Simulator for Compound Semiconductors", "Nanostructured MIM Capacitor" and "Low cost Electric Fencer". His current area of research interest includes Nanoscale CMOS, Interconnects for VLSI circuits, optoelectronics and new materials for devices. He has published 55 research papers in different Journals and Conferences of International repute. He has authored a book on *Matlab and Simulink* and a book chapter on *CMOS Technology*, published by IET, UK. At Present Dr Mallick is working for School of Electrical Engineering, VIT University, Tamilnadu, India, as a Professor and the Dean. Dr Mallick has received the prestigious *Jawaharlal Nehru Scholarship, India* in 1998. He is one of the enlisted technical innovators of India in 2007. He is a Sr. member of IEEE, IEEE-EDS, Life member of IACS, ISTE, and Indian Laser Association. He was the Chair and present Chapter Adviser of IEEE-EDS VIT of Region 10 Asia pacific. He has successfully completed many research projects funded by the Govt. of India and organized several International Conferences and workshops.