Energy Efficient Implementation of Parallel CMOS Multipliers with Improved Compressors

Dursun Baran
ECS Department
University of Texas at Dallas
Richardson, TX 75080, USA
dursun@acsel-lab.com

Mustafa Aktan
ECS Department
University of Texas at Dallas
Richardson, TX 75080, USA
mustafa@acsel-lab.com

Vojin G. Oklobdzija
ECS Department
University of Texas at Dallas
Richardson, TX 75080, USA
vojin@acsel-lab.com

ABSTRACT

Booth encoding is believed to yield faster multiplier designs with higher energy consumption. 16x16-bit Booth and Non-Booth multipliers are analyzed in energy and delay space under varying constraints. It is shown that Non-Booth multipliers start to become more energy efficient for strict delay targets. In addition, novel 3:2 and 4:2 compressors are presented to save energy at the same target delay. The proposed compressors provide up to 20% energy reduction depending on the target delay at 65nm CMOS technology. Non-Booth multiplier implemented with the proposed compressors provides performance advantage as the voltage is scaled from its nominal value. Further, we examined all designs in 45nm, 32nm and 22nm CMOS technology nodes.

Categories and Subject Descriptors

B.2.4 [**High-Speed Arithmetic**]: Cost/performance; B.6.1 [**Design Styles**]: Combinational logic, Parallel Circuits; B.7.1 [**Types and Design Styles**]: VLSI (very large scale integration)

General Terms

Algorithm, Design, Performance

Keywords

Arithmetic and Logic Structures, Booth Encoding, High-Speed Arithmetic, Low-Power Design, VLSI

1. INTRODUCTION

Multipliers are often found in the critical path of signal processors. The multiplication is a slow operation and the improvement in the multiplier performance usually leads to higher operating speed. Historically, technology scaling has been used to improve the performance at a reduced energy budget. Unfortunately, technology scaling cannot sustain the constant power density because of the threshold voltage limitation. Therefore, the circuit implementation techniques and energy-delay tradeoffs become critical. Energy efficient parallel multiplier design requires the exploration of multiplication algorithms, technology constraints and circuit implementation techniques.

The parallel multipliers consist of three main computational blocks namely partial product generation, partial product reduction and final addition. The simplest way of partial product generation is AND operation [6]. Booth algorithm [1],[2] has been used to reduce partial products at the expense of more complex partial product generation. In Booth algorithm, the negative partial products are also generated and they require two extra sign bits to operate correctly. The second block is partial product reduction and historically, several schemes were proposed to improve the speed of the partial product reduction. Array multiplier has a delay proportional to the number of partial products [5], [6]. Wallace [3] tree requires $log_{3/2}(N)$ levels of 3:2 compressors to reduce the N inputs to two carry-save redundant form for Non-Booth multipliers. Dadda [4] minimized the number of counters used in partial product reduction block. In addition, higher order compressors can provide delay improvements [7] such as 4:2 and 9:2 compressors. Further delay improvement is obtained from Three Dimensional Optimization Method (TDM) that connects fast output to slow input and vice versa [9]. In this method, the global optimum is guaranteed by designing as one N^{th}-order compressor instead of individual compressors. 3:2 compressors provide the finest granularity to optimally tie the inputs and outputs to take advantage of the delay difference of sum and carry outputs. The third computational block is the final addition of two carry-save redundant outputs from partial product reduction block. In [12], the optimal final adder designs were explored that considers the signal arrival profiles from partial product reduction block. The signals sitting close to the middle bit positions come later than the signals close to LSB and MSB bit positions [10]. The efficient implementation of final adder considerably increases the energy efficiency of the overall multiplier.

In this paper, single-cycle 16x16-bit Booth multipliers [13] are compared to efficient implementations of 16x16-bit Non-Booth multipliers [11]. In addition, novel 3:2 and 4:2 compressors are presented. The paper is organized as follows. Section 2 gives the implementation details of 16x16-bit Booth Multiplier. Section 3 presents 16x16-bit Non-Booth multiplier architecture. The novel 3:2 and 4:2 compressors are presented in Section 4 and results will be provided in Section 5. Section 6 concludes the work.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
ISLPED'10, August 18–20, 2010, Austin, Texas, USA.
Copyright 2010 ACM 978-1-4503-0146-6/10/08 ...$10.00.

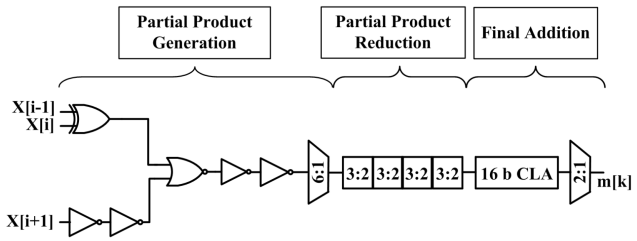

Figure 1: Critical path of a 16x16-bit Booth multiplier.

2. 16X16 BOOTH MULTIPLIER

Booth algorithm [1],[2] is widely used in current processor designs. A state-of-art implementation of Booth multiplier that is presented in [13] is selected as a reference design for the work presented in this paper. The first block performs radix-4 modified Booth encoding with sign extension. One level of 6-to-1 transmission gate multiplexer is used to select the correct partial product [13]. Then, 4 levels of 3:2 compressors are used to reduce 8 partial products to two carry-save redundant output. Hybrid adder architecture is used to exploit the signal arrival profile [13]. For the bits <7:0> ripple carry adder, the bits <23:8> variable-block carry look-ahead adder and the bits <31:24> conditional sum ripple carry adders are used [13]. The critical path of the multiplier is shown in Figure 1.

There is a broad range of the implementations for the 3:2 compressors [16]. *Complementary* CMOS (it is referred as CMOS in this paper) logic gates consist of a NMOS pull-down and a dual PMOS pull-up logic. Static mirror adder is a widely used CMOS implementation example of 3:2 compressors. Mirror adder has almost the best speed at the expense of high power dissipation with a 3.3V voltage supply [16]. In [17], it was shown that CMOS logic style will be the best choice for the future technologies. CMOS logic gates have efficient layouts and they are suitable to be used in synthesis tools. In addition, voltage scaling and the technology scaling requires the robustness and it a well known feature of CMOS logic gates. Therefore, we explored the CMOS implementations of 1-bit adder cells. The schematic of static mirror adder is given in Figure 2. The p/n ratio of 2.0 is used to have a design with equal rise and fall delays and minimal transistor sizes are given in Figure 2. In this work, the whole multiplier designs with a p/n ratio of 1.5 are also explored. Carry is the fast output that is connected to slow inputs of succeeding compressors that are A and B as shown in Figure 2. Three series PMOS stack in sum logic is active when all inputs are logic low (A=0, B=0, and C=0). Similarly, three NMOS stack in sum logic is active when all inputs are logical high (A=1, B=1, and C=1). When the inputs are not all logic high or low these stacks are not active. However, these PMOS and NMOS stacks increase the parasitic delay at node Y irrespective to input combinations (Figure 2). For a spatially and temporally uncorrelated inputs, the probability of being all logic high or all logic low is 0.25 (1/8 + 1/8). From this simple analysis, for the most of the time, PMOS and NMOS stacks will increase the parasitic delay even if they are not active. This observation inspired us to make a modification on the mirror adder to reduce the parasitic delay.

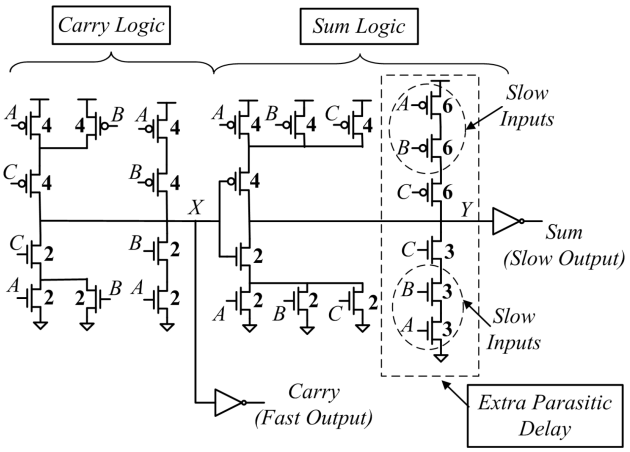

Figure 2: Static Mirror adder with a p/n ratio of 2.0 (conventional 3:2 compressor).

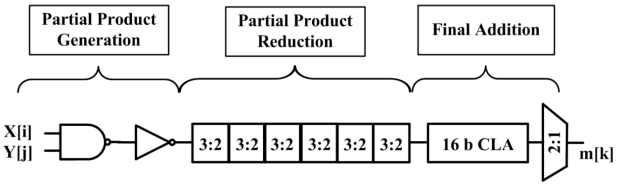

Figure 3: Critical path of a 16x16-bit Non-Booth multiplier.

3. 16X16 NON-BOOTH MULTIPLIER

Non-Booth multiplier generates all partial products. They have a more complex partial product reduction tree and a simpler partial product generation block. The critical path of a 16x16-bit Non-Booth multiplier is given in Figure 3.

As shown in Figure 3, 16x16-bit Non-Booth multiplier has two more levels of 3:2 compressors and quite simpler partial product generation block as compared to 16x16-bit Booth multiplier given in Figure 1. Non-Booth multiplier uses signed magnitude binary number representation. Therefore, there are no sign and complementation bits that are available in Booth multipliers to deal with negative partial products. It further simplifies the Non-Booth multiplier implementation. In order to have a fair comparison of Booth and Non-Booth multipliers, only unsigned inputs are assumed in all designs. Signed magnitude representation is widely used in floating point processors.

First two 3:2 compressors of partial product reduction tree are designed as one 4:2 compressor that was proposed by Weinberger [8]. The partial product generation block and the level of 4:2 compressors are given in Figure 4. Carry is the fast output and it is connected to slow input of next level of compressors. Sum is the slow output and it is connected to fast input of the next level of compressors. Internal carry (C[k]) is connected to the carry input (C[k+1]) of the next 4:2 compressor. Therefore, there is only one level of carry propagation between successive compressors [8]. After the first level of 4:2 compressors, the partial products are reduced by half. The rest of Non-Booth multiplier is implemented same as Booth multiplier. The critical path of

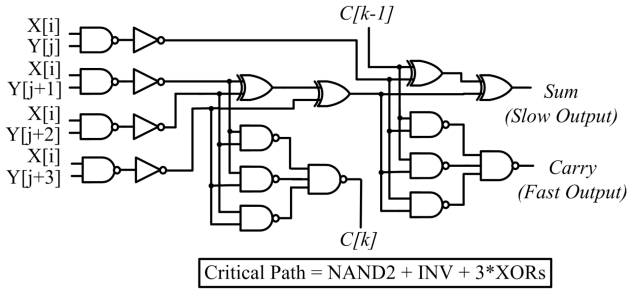

Figure 4: Partial product generation block and the level of conventional 4:2 compressor.

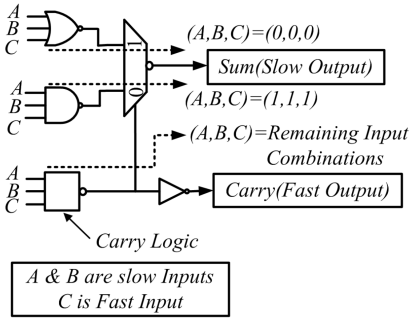

Figure 5: Improved 3:2 compressor design.

partial product generation and the level of 4:2 compressors is NAND2+INV+3XOR2. The critical path of Booth encoder and Booth selector is XOR2+NOR2+2INV+MUX61 as shown in Figure 1. By just considering the complexity of critical paths, it is not clear to prefer one scheme to other. Therefore, a comparative analysis of Booth and Non-Booth multipliers in energy-delay space is required to make a conclusion.

4. NOVEL 3:2 AND 4:2 COMPRESSORS

In this section, a novel 3:2 compressor is presented that modifies the conventional static mirror adder to reduce the parasitic delay by removing three PMOS and NMOS stacks from sum signal path. In addition, the conventional 4:2 compressor is modified to merge inverter stage of partial product generation block into the partial product reduction tree. Using proposed 4:2 compressors, n^2 inverters are removed from nxn-bit multipliers.

4.1 3:2 Compressor Design

The extra parasitic delay comes from the NMOS/PMOS stacks are addressed in Section-2. The improved 3:2 compressor that alleviates the parasitic delay of the sum signal is shown in Figure 5.

The proposed compressor uses a 2:1 multiplexer to select correct sum signal. When all inputs are logical low (A=0, B=0, and C=0), the signal path is NOR3+IMUX21. Similarly, when the inputs are all logical high (A=1, B=1, and C=1), the signal path is NAN3 + IMUX21. For the rest of the input combinations, the signal paths are CL(Carry Logic) + IMUX21. Therefore, the signal paths of all logical high and all logical low input combinations are not sitting on the signal path of other input combinations. This will reduce the total parasitic of the compressor. The comparison of conventional and proposed 3:2 compressor is given in Table 1 and Table 2 with a p/n ratio of 2.0 and 1.5 respectively. In this comparison, IMUX21 is implemented with complementary CMOS logic style. The logical effort and parasitic values reported in Table 1 and Table 2 are obtained from the gate characterization at 65nm CMOS technology.

Table 1: Conventional vs. proposed 3:2 compressors with a p/n ratio of 2.0 (65nm CMOS, tau = 3.615ps)

	Conventional	Proposed	Change
C_{in}(Input A)	27	24	-11.1%
C_{in}(Input B)	27	24	-11.1%
C_{in}(Input C)	21	18	-14.3%
C_{Total}	87	93	+6.9%
G_{Sum}	9.03	4.65	-48.5%
P_{Sum}	16.14	12.76	-20.9%
G_{Carry}	2.74	2.74	SAME
P_{Carry}	7.85	7.85	SAME

Table 2: Conventional vs. proposed 3:2 compressors with a p/n ratio of 1.5 (65nm CMOS, tau = 3.511ps)

	Conventional	Proposed	Change
C_{in}(Input A)	22.5	20	-11.1%
C_{in}(Input B)	22.5	20	-11.1%
C_{in}(Input C)	17.5	15	-14.3%
C_{Total}	72.5	77.5	+6.9%
G_{Sum}	9.26	4.75	-48.7%
P_{Sum}	16.89	13.16	-22.1%
G_{Carry}	2.78	2.78	SAME
P_{Carry}	8.15	8.15	SAME

As shown in Table 1, the sum signal of proposed compressor has a parasitic that is 20.9% lower than the conventional compressors at 65nm CMOS when the p/n ratio is 2.0. In addition to parasitic, logical effort of the proposed compressor is 48.5% lower than the conventional compressor by means of lower logical effort of IMUX21 as compared to the sum logic of conventional mirror adder. The input loading of the proposed compressor is 11.1% and 14.3% lower for input-A (or input-B) and input-C respectively and it relaxes the delay constraint of the previous stages. The similar results are obtained when the p/n ratio is reduced to 1.5 as given in Table 2. As shown in Figure 1 and Figure 3, the critical path delay of the multipliers is dominated by the delay of the partial product reduction tree. Therefore, faster 3:2 compressors can lead to a considerable improvement in the operating speed of overall multiplier. Therefore, the proposed compressor is expected to provide a considerable improvement in the performance of the multipliers. Further, the proposed design has a uniform delay profile with respect to different input combinations. In conventional mirror adder, the delays when all inputs are logical low or logical high are lower than proposed design. However, the delay for the rest of the input combinations is higher than the proposed design. The non-uniform delay profile of conventional static mirror adder for different input combinations increases the energy consumed in the glitches. In addition, the balanced

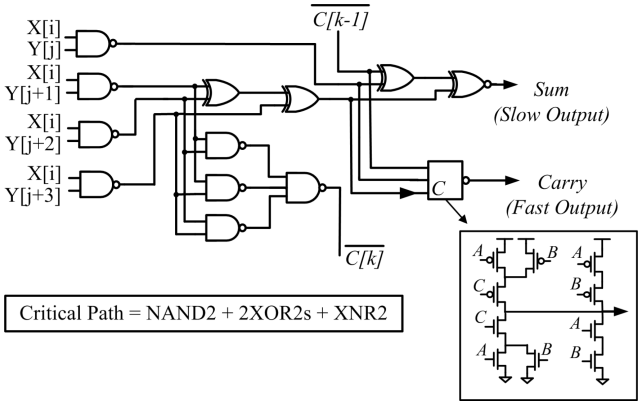

Figure 6: Partial product generation block and the level of improved 4:2 compressor.

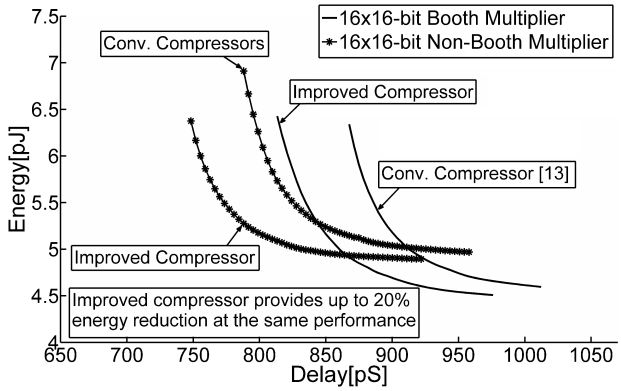

Figure 7: Energy-delay space of 16x16-bit multipliers at 65nm technology node (p/n ratio of 2.0).

delay profile of the proposed compressor for different input combinations simplifies the design process.

In the gate size optimization, the conventional mirror adder is treated as it consists of four different logic gates namely carry logic, sum logic and 2 inverters (Figure 2). This improves the optimization results further. The total capacitance of proposed compressor is 6.9% higher than the mirror adder and it leads to larger layout area. However, the consumed energy is directly related to total switched capacitance. The total switched capacitance is 21.75 units (87 $\times$ 0.25) for mirror adder with a p/n ratio of 2.0 under the totally random input vector assumptions (spatially and temporally uncorrelated). Under this assumption, the switching activity rates at the sum and carry nodes are 0.25. The total switched capacitance is 18.2 units (the switching activity at the output nodes of NAND3 and NOR3 is $7/64 = [1/8] \times [7/8]$) for the proposed compressor with the same p/n ratio and the same input vectors (Figure 5). Therefore, the proposed compressor will consume less energy.

4.2 4:2 Compressor Design

Conventional 4:2 compressor is modified to merge inversion from partial product generation block to the level of 4:2 compressors. The inversion at the inputs of the 4:2 compressors is moved to outputs. The final XOR2 gate of sum logic becomes XNR2 gate. Inverted carry is efficiently generated

Table 3: Technology characterization summary with a p/n ratio of 2.0

	65nm	45nm	32nm	22nm
Power Supply (v)	1.1	1.0	0.9	0.8
[a]W_{min}(nm)	130	110	96	66
FO4 (pS)	19.7	16.7	14.6	12.1
[b]INV Input Cap. (fF)	0.526	0.381	0.295	0.178
Wire Cap. (fF/μm)	0.19	0.18	0.17	0.16
Bit Pitch (μm)	2.00	1.69	1.47	1.01
Temperature (^{o}C)	25	25	25	25

[a]Minimal transistor width
[b]Input capacitance of minimum sized inverter

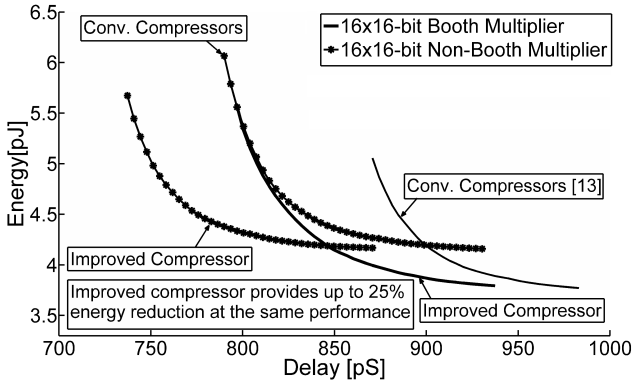

Figure 8: Energy-delay space of 16x16-bit multipliers at 65nm technology node (p/n ratio of 1.5).

using the carry logic of 3:2 compressors as given in Figure 6. Input C of carry logic has the lowest input loading and it is connected to critical path of the compressor as shown in Figure 6. Inputs A and B have large input loading and they are driven by non-critical paths. Further merging of the first level of NAND2 gates to compressor is also possible.

Using the proposed 4:2 compressors, 256 inverters are removed from the partial product generation block with a subtle change in the complexity of 4:2 compressors for a 16x16-bit Non-Booth multiplier. In addition, one inverter stage is removed from the critical path of the Non-Booth multiplier. Sum signal of the improved compressor is the slow output and it will be connected to fast input of the next level 3:2 compressors. Similarly, carry is the fast output and it will be connected to slow input of the next level 3:2 compressors.

5. RESULTS

16x16-bit Booth and Non-Booth multipliers with the conventional and the proposed compressors are implemented in 65nm, 45nm, 32nm and 22nm CMOS technology nodes. The energy and delay estimation technique presented in [14] is used to estimate the energy and delay of each design. Gate characterization is done under the worst case single input switching condition. The bit-pitch of the register file is used to estimate the length of the wires and the lengths between successive PPRT stages. The summary of technology characterization is given in Table 3. The wire capacitance per length presented in Table 3 is obtained from the interconnect report of International Technology Roadmap for Semiconductors (ITRS) [15].

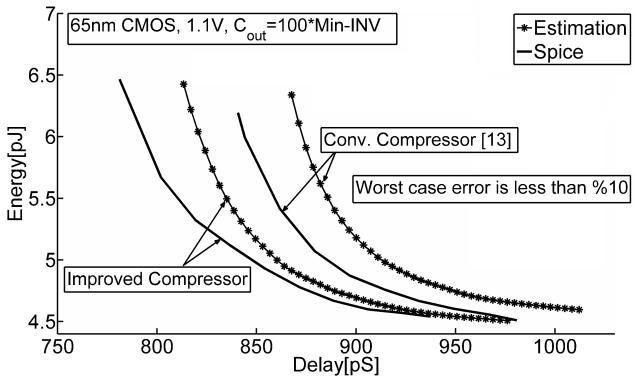

Figure 9: 16x16-bit Booth multipliers. Estimation vs. H-Spice (p/n ratio of 2.0).

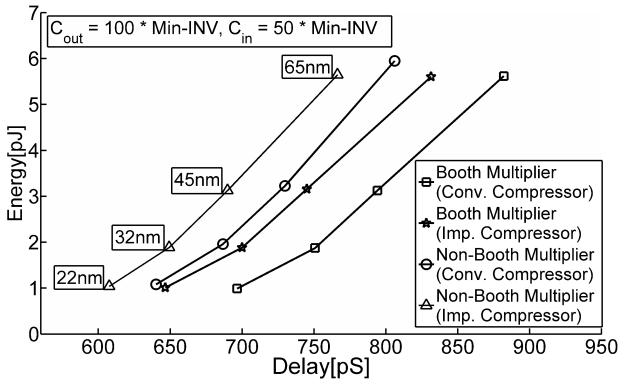

Figure 10: 16x16-bit multipliers across CMOS technology nodes (p/n ratio of 2.0).

The delay estimation is done using the logical effort delay model. For energy estimation, spatially and temporally uncorrelated inputs are assumed. It is a fair assumption to compare different topologies. An average of 25% switching activity is used to estimate the energy consumption of each design under the assumption of totally random input vectors. The average switching activity of 25% is obtained after comparison of estimation and H-Spice results. Totally random input vectors are used in H-Spice energy simulations.

The gate sizing is an iterative process and it is implemented in MATLAB. The sizes of gates are optimized under different operating conditions (delay and input/output loading). Each multiplier is loaded with 100*Min-INV and delay is optimized for an input capacitance that is swept from 20*Min-INV to 100*Min-INV. Then, minimal energy optimization algorithm is used to size the gates for minimal energy at the delay targets obtained from delay optimization.

The energy and delay space of 16x16-bit multipliers with the p/n ratio of 2.0 at 65nm CMOS technology are given in Figure 7. Conventional 16x16-bit Non-Booth multiplier provides a performance improvement of up to 10% compared to conventional 16x16-bit Booth multiplier. 16x16-bit Non-Booth multiplier is more convenient for the applications that require high performance. As the performance demand decreases, Booth multiplier becomes more energy efficient by means of lower gate count. Using the improved compressors, it is possible to reduce the energy consumption by 20% at the same speed. The proposed compressor loses advantage as the performance demand of the application is reduced as shown in Figure 7. Similar results are obtained when the p/n ratio is 1.5 and it is presented in Figure 8.

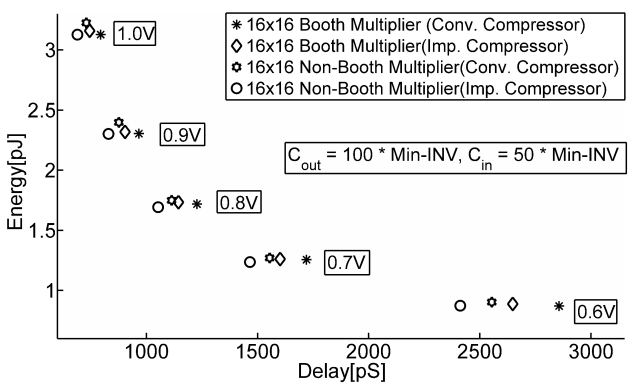

Figure 11: 16x16-bit multipliers with reduced voltages at 45nm CMOS technology (p/n ratio of 2.0).

In order to show the accuracy of the energy and delay estimation results, the 16x16-bit Booth multipliers with conventional and improved compressors are simulated in H-Spice. The results are shown in Figure 9. The worst case error of 10% is obtained from the comparison of estimation and H-Spice results at the same input loading design points. The worst case input vector delays are reported. For energy estimation, sufficient numbers of totally random input vectors are generated and applied to the multipliers. Then, energy consumption for each input vector is measured and all of them are averaged. As shown in Figure 9, the estimation accurately reflects the architectural trade-offs for the purpose of comparison.

The energy and delay results of each implementation are explored in 45nm, 32nm and 22nm CMOS technology nodes. One design point (Cout = 100*Min-INV and Cin =50*Min-INV) is selected and the designs are optimized at each technology node. Results are reported in Figure 10. In addition to technology scaling, we explored the characteristics of each multiplier with reduced voltage supplies. For this purpose, voltage is scaled from 1.0v to 0.6v at 45nm CMOS technology. The gate characterization is repeated for each voltage supply to accurately reflect the changes into the energy and delay optimization. The results are given in Figure 11. 16x16-bit Non-Booth multiplier implemented with improved compressors has a delay of 2411ps at 0.6v voltage supply. At the same voltage supply, 16x16-bit Booth multiplier implemented with conventional mirror compressors has a delay of 2856ps that is 18% higher.

The logical effort and total parasitic of conventional and improved 3:2 compressors when the voltage is scaled from 1.0v to 0.6v at 45nm CMOS technology are given in Table 4 and Table 5 respectively. As the voltage is scaled, the improved 3:2 compressor provides higher reduction in logical effort and total parasitic. It explains the delay improvement of proposed compressors at scaled voltages.

Table 4: Logical effort parameters (G) at 45nm CMOS (p/n ratio of 2.0)

Voltage Supply	1.0v	0.9v	0.8v	0.7v	0.6v	0.5v
FO4 (pS)	16.7	19.6	24.1	32.6	52.0	114
Conv. 3:2 Compressor	10.6	11.3	12.1	13.1	14.3	15.0
Impr. 3:2 Compressor	5.3	5.6	5.9	6.3	6.8	7.2
Reduction (%)	49.9	50.3	50.9	51.6	52.3	51.9

Table 5: Parasitic parameters (P) at 45nm CMOS (p/n ratio of 2.0)

Voltage Supply	1.0v	0.9v	0.8v	0.7v	0.6v	0.5v
Conv. 3:2 Compressor	16.1	16.9	17.7	18.5	19.4	20.5
Impr. 3:2 Compressor	13.9	14.5	15.0	15.6	16.2	16.8
Reduction (%)	13.7	14.2	15.3	15.7	16.5	18.0

6. CONCLUSION

Booth encoding reduces two levels of 3:2 compressors from the partial product reduction tree of a multiplier by moving some complexity to partial product generation block. Non-Booth multipliers generate all partial products. 16x16-bit Booth and Non-Booth multipliers are comparatively analyzed in energy and delay space at 65nm, 45nm, 32nm and 22nm technology nodes. Non-Booth multiplier is 10% faster than Booth multiplier at 65nm CMOS. At the same input loading, Non-Booth multipliers provide superior performance at the same energy budget at all technology nodes. In addition, conventional 3:2 and 4:2 compressors are modified to improve the performance of the multiplier. In performance critical applications, it is possible reduce the energy consumption by 20% by using the improved compressors. At the same input loading, the multipliers designed with improved compressors are faster than the multipliers designed with conventional compressors with the same energy budget. Non-Booth multiplier with improved compressors provides better energy and delay characteristics as voltage is scaled from 1.0v to 0.6v at 45nm CMOS technology.

7. ACKNOWLEDGMENTS

This work is supported by SRC Research Grant 2009-HJ-1836, Intel Corporation, and IBM Corporation.

8. REFERENCES

[1] Booth, A. D., "A Signed Binary Multiplication Technique," *Quaterly J. Mechanical Applications in Math.,* vol. 4, part 2, pp. 236-240, 1951.

[2] Macsorley, Q. L., "High Speed Arithmetic in Binary Computers," *IRE Proc.,* vol. 49, pp. 67-91, Jan. 1961.

[3] Wallace, C. S., "A Suggestion for a Fast Multiplier," *IEEE Trans. Computers,* vol. 13, no. 2, pp. 14-17, Feb. 1964.

[4] Dadda, L., "Some schemes for Parallel Multipliers," *Alta Frequenza,* vol. 34, pp. 349-356, Mar. 1965.

[5] Pezaris, S. D., "A 40ns 17-bit Array Multiplier," *IEEE Trans. Computers,* vol. 20, no. 4, pp. 442-447, Apr. 1971.

[6] Santoro, M. R.; Horowitz, M. A., "SPIM: A Pipelined 64x64-bit Iterative Multiplier," *IEEE J. Solid State Circuits,* vol. 24, no. 2, pp. 487-493, April 1989.

[7] Song, P. J.; De Michelli, G., "Circuit and Architecture Trade-offs for High-Speed Multiplication," *IEEE J. Solid State Circuits,* vol. 26, no. 9, pp. 1184-1198, Sept. 1991.

[8] Weinberger, A., "4:2 Carry-Save Adder Module," *IBM Technical Disclosure Bull.,* vol. 23, Jan. 1981.

[9] Oklobdzija, V. G.; Villeger, D.; Liu, S. S., "A Method for Speed Optimized Partial Product Reduction and Generation of Fast Parallel Multipliers Using an Algorithmic Approach," *IEEE Trans. Computers,* vol. 45, no. 3, pp. 294-306, Mar. 1996.

[10] Stelling, P. F.; Martel, C. U.; Oklobdzija, V. G.; Ravi, R., "Optimal Circuits for Parallel Multipliers," *IEEE Trans. Computers,* vol. 47, no. 3, pp. 273-285, Mar. 1998.

[11] Bonatto, P.; Oklobdzija, V. G., "Evaluation of Booth's Algorithm for Implementation in Parallel Multipliers," *Proceedings of Asimolar-29,* vol. 1, pp. 608-610, 1996.

[12] Oklobdzija, V. G.; Villeger, D., "Improving Multiplier Design Using Improved Column Compression Tree and Optimized Final Adder in CMOS Technology," *IEEE Trans. VLSI Syst.,* vol. 3, no. 2, pp. 292-301, June 1995.

[13] Hsu, S. K.; Mathew, S. K.; Anders, M. A.; Zeydel, B. R.; Oklobdzija, V. G. ; Krishnamurthy, R. K. ; Borkar, S. Y., "A 110 GOPS/W 16-bit Multiplier and Reconfigurable PLA Loop in 90-nm CMOS," *IEEE Journal of Solid-State Circuits,* vol. 41, no. 1, pp. 256-264, January 2006.

[14] Oklobdzija, V. G.; Zeydel, B. R. ; Dao, H. Q.; Mathew, S.; Krishnamurthy, R., "Comparison of High-Performance VLSI Adders in the Energy-Delay Space," *IEEE Trans. VLSI Syst.,* vol. 13, no. 6, pp. 754-758, June 2005.

[15] ITRS (International Technology Roadmap for Semiconductors) [Online]. Available: http://www.itrs.net/Links/2007ITRS/2007Chapters/2007Interconnect.pdf

[16] Shams, A. M.; Darwish, T. K.; Bayoumi, M. A., "Performance Analysis of Low-Power 1-Bit CMOS Full Adder Cells," *IEEE Trans. VLSI Syst.,* vol. 10, no. 1, pp. 20-29, Feb. 2002.

[17] Zimmermann, R.; Fichtner, W., "Low-Power Logic Styles: CMOS Versus Pass-Transistor Logic," *IEEE Journal of Solid-State Circuits,* vol. 32, no. 7, pp. 1079-1090, July 1997.