The CSI Journal on
Computer Science and Engineering
Vol. 13, No. 1, 2015
Pages 24-30
Regular Paper

Architecture-Level Design Space Exploration for Radix-16 Sequential Multipliers

Saba Amanollahi Ghassem Jaberipur

Department of Computer Science and Engineering, Shahid Beheshti University, Tehran, Iran

Abstract

Figures of merit of multiplier circuits are greatly influenced by the choice of partial product generation and reduction components and the final product generating adders. In this paper, we focus on radix-16 sequential 32-, 64-, and 128-bit multipliers and study the impact of the aforementioned design options, within a 3-dimensional design space, on the latency, area and energy figures on the synthesized multipliers in 45 nm Nan gate technology. A variety of five different fast final adders and three alternative carry-save adders for partial product reduction contribute to the configuration of the design space, where the delay, power, area, and energy figures for the 45 synthesized multipliers show at least31%, 42%, 20%, and 19% variation from minimum to maximum of the obtained measures. Comparison with the main reference work demonstrates at least 50% speed up, 60% energy improvements.

Keywords: Sequential Multiplier, Design Space Exploration, Adder Architectures, Energy Efficiency.

1. Introduction

Multiplication is a frequent operation in many applications that are run in digital processors. Nowadays, latency and power dissipation of multiplication circuits are more critical than other characteristics such as silicon area, which is also important in applications that the die area is very limited. Fast parallel multiplier circuits are known for considerable power and area consumption [1]. while the sequential multipliers, though slow, are usually known as low power and cost [2].

The basic sequential multiplication scheme regarding an n-bit multiplication $X \times Y$ would produce one partial product at a time (i.e., X or 0 depending on the corresponding bit of the multiplier$Y$). This is added to the previously cumulated partial product (CPP). Figure 1 depicts the general structure of such scheme, where the time required for partial product generation (PPG) and partial product reduction (PPR) to take place in each iteration is almost the same as that of the final n-bit addition.

On the other hand, it is common to keep the CPP in a redundant format (e.g., carry-save), whereby a fast and low power carry-save adder (CSA) takes care of the PPR. However, the final addition must yield a non-redundant binary result, which requires a carry-propagate addition. Therefore, to tune the time of each iteration with that of this final n-bit addition, designers have opted to use radix-2^h ($h \geq 2$) multipliers that utilize h bits of multiplier per iteration, thus reducing the number of iterations and prolonging the PPG and PPR. A typical value for h is 4, even among other arithmetic operations. For example, radix-16 multipliers [3] dividers [3], and floating point units [4] are common, where in particular radix-16 adders can be easily shared between binary and decimal units [5]. Figure 2 depicts a typical radix-16 multiplier (reproduced from [3]), where each partial product is generated as four components.

That is instead of pre-computing all the required $[0, 15] \times X$ multiples, only four easy-to-generate multiples $\{1, 2, 4, 8\} \times X$ are originally obtained and the actual four components of partial product are selected via the bits of multiplier's digit $y_{i+3}y_{i+2}y_{i+1}y_i$.

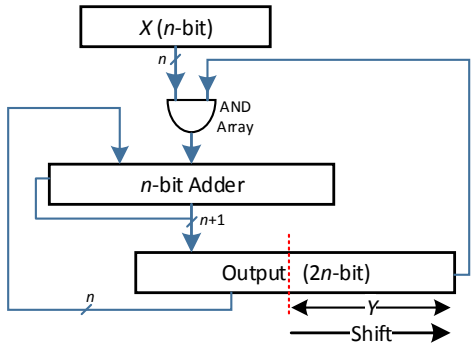


Figure 1. General Structure of the n-bit sequential multiplier

In this paper, we are particularly interested to examine and explore a design space on radix-16 multiplication, where the possible design options are to use different carry-save adders (i.e., binary, radix-4, and radix-16) for PPR, alternative carry-propagate adders (e.g., variety of parallel prefix architectures) for the final product computation and the operand widths (i.e., the actual value of n). We also aim to examine the impact of different influential design choices on speed, power, energy, and energy-delay product (EDP) measures of the multiplier. The outcome of such endeavor is meant to help the design engineers to locate the best sequential multiplier design that suits the goals of their application. It should be emphasized that in radix-16 sequential multiplication, delay of the main iterative body and the final adder could be almost balanced; hence the choice of radix-16 multiplication for this study.

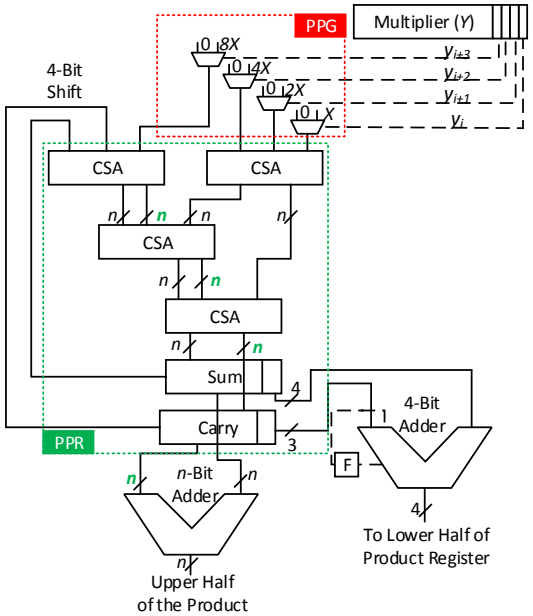


Figure 2. Typical radix-16 multiplication architecture [3]

Among the previously published sequential multiplier designs, there are few works that regard special cases. For example, that of [6] provides only the most significant half of the product. In the works of [7] [8] [9], number of cycles depend on the number of 1s in the multiplier. A radix-2 sequential multiplier has been proposed in [1] where the sparse-2 Ling adder is employed for the PPR. The CPP manipulation of radix-4 multiplier of [2] uses carry save (CS) adders. Such adders are also used in the radix-16 design of [3] (see figure 2). Furthermore, an interesting study by [1] shows that among the four designed multipliers, therein, the sequential one dissipates the lowest power.

A noteworthy characteristic of the aforementioned design choices is the utilization of higher radix CSAs in the PPR, which as far as the knowledge of the authors goes, has no precedence in the relevant literature. As will be shown in Section 3, this design option results in lowering the power and energy figures.

The rest of this paper is organized as follows. In Section 2, details of the design space is discussed. Results of the proposed exploration is provided in Section 3. Finally, the paper is concluded in Section 4.

2. The Design Space

Partial product reduction is recognized as the bulk of multiplication process. Therefore, it is influential on the design choices. For example, the CPP is often kept in a redundant format such as binary carry-save [2] [3], where the PPR block is actually a CSA, with sum (S) and carry (C) output, as in figure 2 The final redundant CPP is to be converted to its binary equivalent to yield the final product. This is normally done via a carry propagate adder, the delay of which is often more than that of one PPR operation. Since the processor's cycle time is usually equal to that of its addition latency, the common choice for cycle time of the sequential multiplier is the same. Therefore, there is enough room to prolong the PPR in favor of reducing its power and area consumption and also number of cycles.

That is probably why, contrary to typical parallel multiplier designs, no cycle is dedicated to the partial product generation (PPG), which is distributed within PPR cycles. On the other hand, a common approach to reduce the number of cycles and also lengthen the PPR is to use higher radix-2^h (e.g., h = 4) PPG. However, another possibility in order to speed up the PPR (for instance in case of h = 4), is to keep the CPP in radix-4 CSA or binary CSA; thus widening the design space. Note that the higher the radix of the CPP, the lower is the size of C register; hence lower silicon area is consumed and lower power dissipation is expected. Another possibility for expansion of the design space is to use various adder architectures (e.g., variety of parallel prefix adders, carry-skip or carry-select adders) for the final n-bit adder. This can also have considerable impact on the cycle time and power dissipation of the corresponding multiplier.

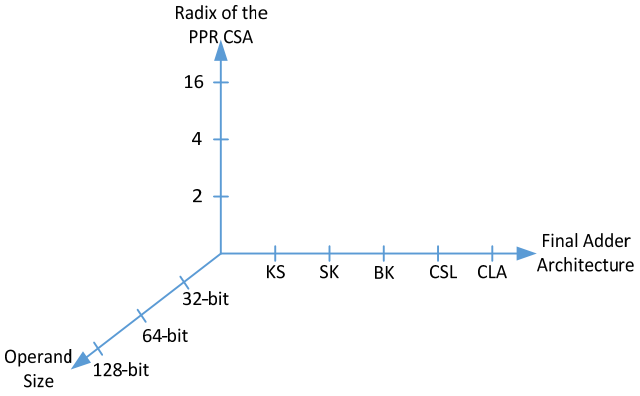


Figure 3. The considered 3-dimentional design space

The aforementioned design space, for radix-16 multipliers, is 3-dimentional (see figure 3), where the nubs are:

- The operand size (i.e., $n \in \{32, 64, 128\}$): These are chosen from the popular data path widths.
- Radix of the PPR CSA (i.e., 2, 4, or 16): Figures 2, 4, and 5, represent the required architectures, where the PPG is exactly the same, and thus its details are not repeated in figures 4 and 5 function of the required CSAs are illustrated (in the span of 10 bits, for brevity) in dot notation representation as in figures 6-a (used in all the three architectures), -b, and –c (triple use in figures 4 and 5, respectively). Regarding the impact of the CSA radix on the final adder, note that the higher the radix the more sparse is the second operand of the final adder (i.e., $n/2$ and $n/4$ bits).Accordingly, the early output 4-bit adder of figure 2 is reduced to a 2-bit adder in figure 4, and vanishes in figure 5.
- Alternative choices for the final adder architecture; namely, Kogge-Stone (KS), Sklansky (SK) and Brent-Kung (BK) parallel prefix structures, carry look-ahead (CLA) with the blocking factor of 4, and carry select (CSL) adders [10].

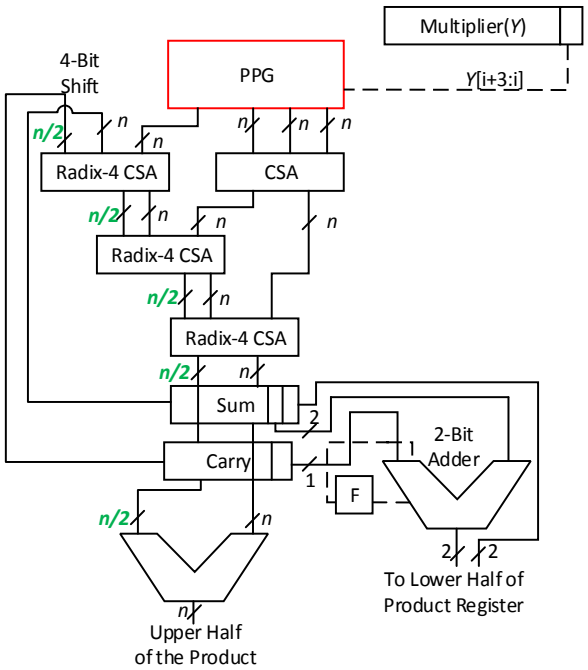


Figure 4. Radix-16 multiplier with radix-4 CSAs

3. Results and Discussion

The proposed design space exploration (see figure 3) led to 45 different designs forradix-16 sequential multiplier. All the corresponding circuitrieshave been coded in verilog hardware description language and synthesized via Synopsys Design compiler under minimum delay constraint in 45nm Nan Gate technology [11].

Before providing an overall evaluation picture of the explored design space, we compare the figures of merit of the final adder architectures and the impact of the different three CSA types on those adders.

A) Power and Delay Characteristics of the Final Adders

Impact of the different architectures regarding the final adder on the delay and dissipated power is illustrated in figure 7, for word length n = 128 corresponding to the required adder for figure 6-a, in 45nm Nan Gate technology [11]. The KS adder is the fastest at the cost of highest power dissipation, while CLA shows the longest delay at the advantage of lowest dissipating power.

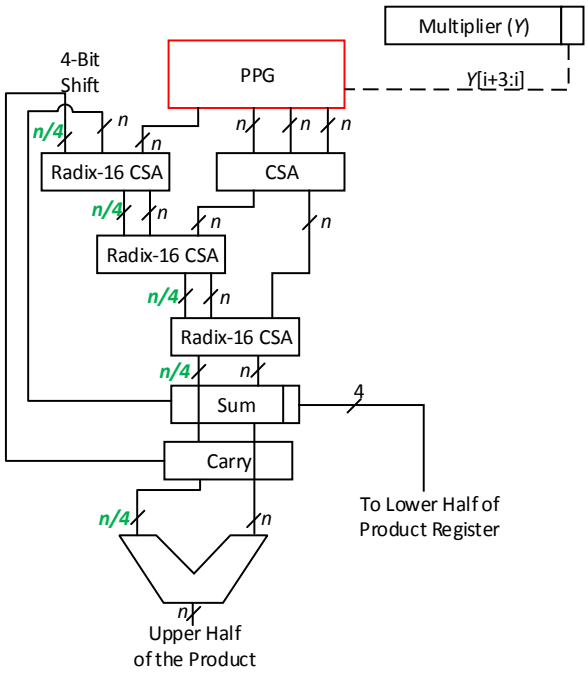


Figure 5. Radix-16 multiplier with radix-16 CSAs

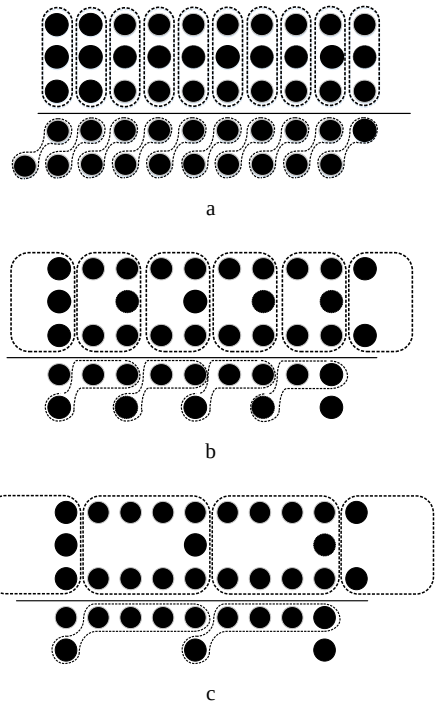


Figure 6. The dot notations of radix-2 (a), -4 (b), and -16 (c) CSAs

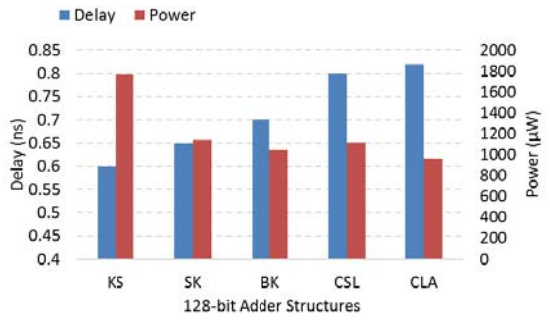


Figure 7. Delay/power trade-off for different architectures of the final adder

B) Impact of CSA Types on the Performance of Final Adders

To study the impact of different CSAs on figures of merit of the final adder, regarding the three choices of figure 6 (in case of n = 128)), and KS and CSL architectures, we provide figures 8 and 9 note that both the delay and power figures of KS (CSL) show 10% (9%) and 58% (61%) decrease, respectively, as the radix of the CSAs grow from 2 to 16.

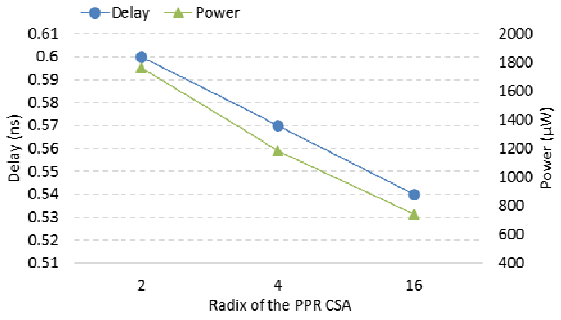


Figure 8. Impact of the PPR CSA radices on delay and power figures of the final adder with KS architecture

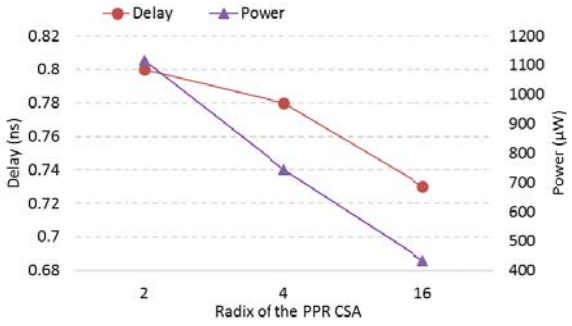


Figure 9. Impact of the PPR CSA radices on delay and power figures of the final adder with CSL architecture

C) The Overall Evaluation

Figure 10 depicts the figures of merit of 32-, 64- and 128-bit explored radix-16 sequential multipliers with regards to five different architectures for the final adder and three different CSA types. The evaluated measures are delay, power, energy energy-power product (EDP), and area, which are discussed below in separate subsections.

C.1. Delay

As is expected from the content of figure 7, speed advantage of almost all the explored architectures with KS adder is evident in figure 10-a. The same advantage is generally shared by the architectures that have utilized CS-2 for PPR, to the extent that the critical delay path migrates from PPR (e.g., for CS-16) to the final addition. It is notable that the final adder architecture has no impact on the delay figures in case of CS-16. The reason is that the critical delay path in this case falls within the PPR stage. However, in case of CS-4, the critical delay path falls in PPR (for KS and SK adders), and on the final adders, otherwise.

C.2. Power Dissipation

The power figures, for the whole design space that are compiled in figure 10-b, decrease as the CSA radix grows up. The reason is twofold. First the sparsity of the higher radix CSAs, as is evident in figure 6, and second the less complexity of the final adders accordingly. For example, the architecture with radix-16 CSAs and the lowest power adder (i.e., CLA) dissipates the least power. On the other extreme, the highest power figure is experienced in the all CS-2 architecture with KS final adder (i.e., the highest power adder). Note that such differences are more evident in larger word length.

C.3. Energy Consumption

Observing the opposite orders (regarding the CSA radices) and slopes (regarding the bit widths) between figures 10-a (delay) and 10b (power), one expects small fluctuations in energy figures, which is confirmed by figure 10-c, where however, the exact figures show at most 22% energy consumption difference for the 128-bit data path.

C.4. Area

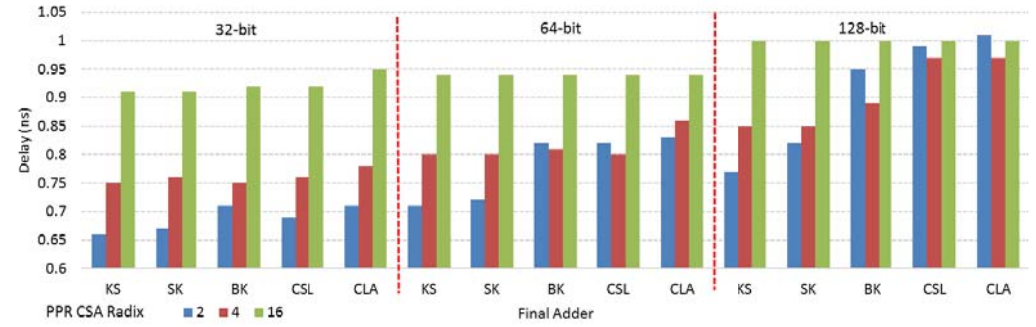
The main components of area consumption are PPR CSAs, final adder and registers. The higher the radix, the less is the register area, the sparser is the carry component of the input to final adder, which leads to less area consumption, and the longer is the CSA carry chain (i.e., 0, 2, and 4 bit for radix-2, -4, and -16, respectively). The latter, however, forces the synthesis tool to use more area consuming gates (with larger width/length ratio) to meet lower time constraints. Therefore, given that two of the aforementioned main area consuming components tend to need less area with respect to higher PPR CSA radices, and the third one otherwise, it is not easy to analyze and predict the overall area consumption behavior, which is however, captured in figure 10-d that is provided by the synthesis tool. For example, see the lowest area consumption of the radix-4 PPR CSA architectures. However, the highest and lowest area consumption of the multipliers with KS and CLA final adders, respectively, are in conformity with the normal expectation and properties of such adders.

D) Comparison with Radix-2 Sequential Multipliers

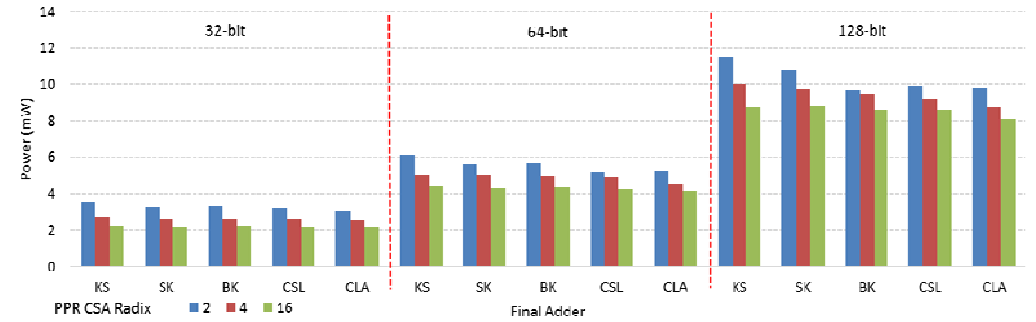
Since we have not encountered any previous realization of radix-16 sequential multipliers in the relevant literature, we use the radix-2 sequential multiplier realization of [1], for comparison purposes, which is also used therein as a

comparison reference. For fair comparison, however, we have synthesized 32-, 64, and 128-bit versions of the multiplier of this reference work with the same 45 nm technology that is used to synthesize our proposed designs. Our design of choice (among the 45 studied ones), for comparison sake, is the radix-16 multiplier with radix-16 PPR CSA and CLA final adder. Consequently, the results are compiled in table 1, where the latency and energy figures therein, clearly show the superiority of the "New" design over the reference work.

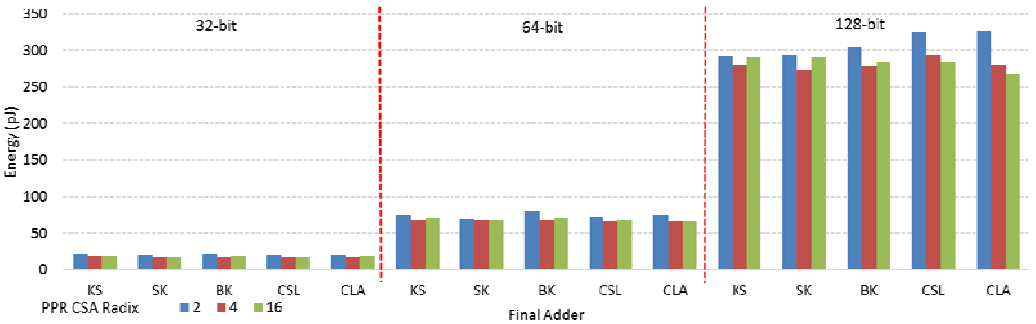
This is, however, at the cost of more silicon cost. Nevertheless the area-energy product of the proposed work is considerably inferior. Another noteworthy merit point regarding this selected designs among the proposed works is the low cycle count. For example, the 32-bit proposed multiplier produces the product in 9 cycles, while the corresponding design of [1] requires 33 cycles.



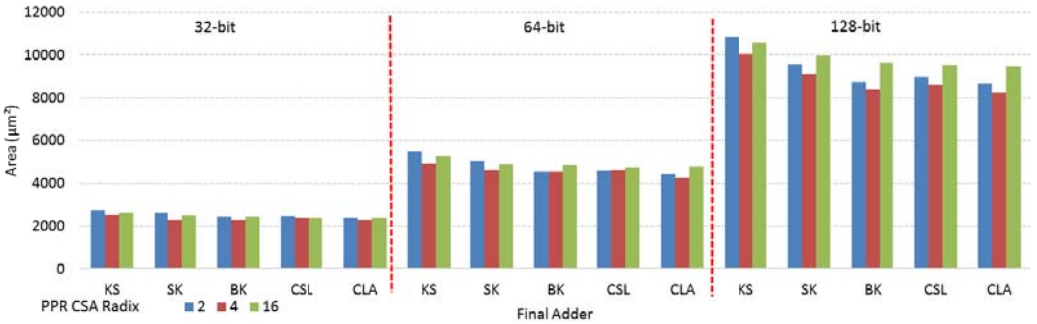
a



b



c



d

Figure 10. The radix-16 multiplier design space from point of view of a) Delay, b) power, c) energy, and d) area

Table 1. Deign parameters of the conventional radix-2 sequential multiplier [1]

Operand Width (bit)	Latency (ns)			Energy (p J)			Area (μm^2)			Energy ×Area (p J×μm^2)		
	[1]	New	Ratio (New/[1])	[1]	New	Ratio (New/[1])	[1]	New	Ratio (New/[1])	[1]	New	Ratio (New/[1])
32	17.6	8.55	0.49	48	19	0.40	1179	2356	2.00	56592	44764	0.79
64	42.88	15.98	0.37	172	67	0.39	2380	4790	2.01	409360	320930	0.78
128	103.68	33	0.32	671	268	0.40	4894	9474	1.94	3283874	2539032	0.77

4. Conclusion

A considerably large design space with 45 different radix-16 sequential multipliers was investigated. The 32-, 64-, and 128-bit data paths, use of radix-2, -4, -16 carry-save adders as the basic partial product reduction cells and a variety of carry accelerating adders to produce the final product contributed to the richness of the 3-dimensional design space. Results of the synthesis of all the 45 multipliers enable the design engineers to select the most appropriate multiplier architectures that suit the application in hand. Over 50% speedup, 60% energy saving, at the cost of double area consumption (with over 20% less area-energy product however) is experienced in comparing the proposed multipliers with a typical previous radix-2 sequential multiplier.

References

[1] D. Baran, M. Aktan, and V. G. Oklobdzija, "Multiplier Structures for Low Power Applications in Deep-CMOS," IEEE International Symposium on Circuits and Systems (ISCAS), 2011, pp. 1061-1064.

[2] Mi Lu, Arithmetic and Logic in Computer Systems, Wiley, 2004.

[3] B. Parhami, Computer Arithmetic: Algorithms and Hardware Designs, 2nd Edition, Oxford University Press, USA, 2010.S. Amanollahi and G. Jaberipur, "Energy Efficient VLSI Realization of Binary64 Division with Redundant Number Systems," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2016.

[4] H. A. H. Fahmy, and M. J. Flynn, "The Case for a Redundant Format in Floating-Point Arithmetic," Proc. 16th IEEE Symp. Computer Arithmetic, pp. 95-102, 2003.

[5] F. Y. Busaba, C. A. Krygowski, W. H. Li, E. M. Schwarz, and S. R. Carlough, "The IBM z900 decimal arithmetic unit, in: Proceedings of the 35th Asilomar Conference on Signals, Systems and Computers," vol. 2, pp. 1335–1339, November 2001.

[6] M. D. Ercegovac, and T. Lang, "Fast Multiplication without Carry-Propagate Addition," IEEE Transactions on Computer, vol. 39, no. 11, 1385-1390, 1990.

[7] N. Honarmand, M. R. Javaheri, N. Sedaghati-Mokhtari, and A. Afzali-Kusha, "Power Efficient Sequential Multiplication Using Pre-computation," Proceedings of IEEE International Symposium on Circuits and Systems, 2006, pp. 2709-2712.2.

[8] M. Mottaghi-Dastjerdi, A. Afzali-Kusha, and M. Pedram, "BZ-FAD: A Low-Power Low-Area Multiplier based on Shift-and-Add Architecture," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 17, no. 2, pp. 302-306, 2009.

[9] G. R. Deepthy, l. Arathy, and G. N. Nishi, "Modified Low-Power Multiplier Architecture," Advances in Communication, Network, and Computing, pp. 172-178, 2012.

[10] M. Bahadori, M. Kamal, A. Afzali-Kusha, and, M. Pedram, "A comparative study on performance and reliability of 32-bit binary adders," Integration, the VLSI journal, vole (53), pp. 54-67, 2016.

[11] www.nangate.com, "NanGate-The Standard Cell Library Optimization Company," 2016. [On-line]. Available: http://www.nangate.com/.

Saba Amanollahi received her B.Sc. degree in Computer Engineering from University of Tehran, Tehran, Iran, in 2007, and her M.Sc. in Computer Architecture from Shahid Beheshti University, Tehran, Iran, in 2011. Now, she is working on her Ph.D. in Computer Architecture at Shahid Beheshti University. Her research interests include Computer Arithmetic, Low Power Design, and Embedded System Design.
E-mail: s_amanollahi@sbu.ac.ir

Ghassem Jaberipur is an Associate Professor of Computer Engineering in the Department of Computer Science and Engineering of Shahid Beheshti University, Tehran, Iran. He received his BS in electrical engineering and PhD in computer engineering from Sharif University of Technology in 1974 and 2004, respectively, (where he is recognized as one of the 50 distinguished graduates for years 1966-2016), MS in engineering from UCLA in 1976, and MS in computer science from University of Wisconsin,

Madison, in 1979. His main research interest is in computer arithmetic. Dr. Jaberipur is also affiliated with the School of Computer Science, Institute for Research in Fundamental Sciences (IPM), in Tehran, Iran
E-mail: jaberipur@sbu.ac.ir

Paper Handling Data:

Submitted: 23.08.2016
Received in revised form: 02.11.2016
Accepted: 21.11.2016
Corresponding author: Dr. Ghassem Jaberipur,
Department of Computer Science and Engineering, Shahid Beheshti University, Tehran, Iran.