Reverse Converter Design via Parallel-Prefix Adders: Novel Components, Methodology, and Implementations

Azadeh Alsadat Emrani Zarandi, Amir Sabbagh Molahosseini, Mehdi Hosseinzadeh, Saeid Sorouri, Samuel Antão, and Leonel Sousa

***Abstract*— In this brief, the implementation of residue number system reverse converters based on well-known regular and modular parallel-prefix adders is analyzed. The VLSI implementation results show a significant delay reduction and area × time^2 improvements, all this at the cost of higher power consumption, which is the main reason preventing the use of parallel-prefix adders to achieve high-speed reverse converters in nowadays systems. Hence, to solve the high power consumption problem, novel specific hybrid parallel-prefix-based adder components that provide better tradeoff between delay and power consumption are herein presented to design reverse converters. A methodology is also described to design reverse converters based on different kinds of prefix adders. This methodology helps the designer to adjust the performance of the reverse converter based on the target application and existing constraints.**

***Index Terms*— Digital arithmetic, parallel-prefix adder, residue number system (RNS), reverse converter.**

I. Introduction

In the world of battery-based and portable devices, the residue number system (RNS) can play a significant role due to its low-power features and competitive delay. The RNS can provide carry-free and fully parallel arithmetic operations [1], [2] for several applications, including digital signal processing and cryptography [3]–[6]. However, its real usage requires forward and reverse converters to be integrated in the existing digital systems. The reverse conversion, i.e., residue to binary conversion, is a hard and time-consuming operation [7]. Hence, the problem of designing high-performance reverse converters has motivated continuous research using two main approaches to improve the performance of the converters: 1) investigate new algorithms and novel arithmetic formulations to achieve simplified conversion formulas and 2) introduce new moduli sets, which can lead to more simple formulations. Thereafter, given the final simplified conversion equations, they are computed using well-known adder architectures, such as carry-save adders (CSAs) and ripple-carry architectures, to implement carry-propagate adders (CPAs) and, more seldomly, fast and expensive adders such as the ones with carry-look ahead or parallel-prefix architectures.

In this brief, for the first time, we present a comprehensive methodology to wisely employ parallel-prefix adders in carefully selected positions in order to design fast reverse converters. The collected experimental results based on area, delay, and power consumption show that, as expected, the usage of the parallel-prefix adders to implement converters highly increases the speed at the expense of additional area and remarkable increase of power consumption. The significant growing of power consumption makes the reverse converter not competitive. Two power-efficient and low-area hybrid parallel-prefix adders are presented in this brief to tackle with these performance limitations, leading to significant reduction of the power-delay product (PDP) metric and considerable improvements in the area-time^2 product (AT^2) in comparison with the original converters without using parallel-prefix adders.

Manuscript received April 16, 2013; revised August 27, 2013 and December 1, 2013; accepted January 16, 2014. Date of publication February 26, 2014; date of current version January 30, 2015. This work was supported by the National Funds through Fundação para a Ciência e a Tecnologia under Project PEst-E/EEI/LA0021/2013.

A. A. E. Zarandi and M. Hosseinzadeh are with the Department of Computer Engineering, Tehran Science and Research Branch, Islamic Azad University, Tehran 14778-93855, Iran (e-mail: a.emrani@srbiau.ac.ir; hosseinzadeh@srbiau.ac.ir).

A.S. Molahosseini and S. Sorouri are with the Department of Computer Engineering, Kerman Branch, Islamic Azad University, Kerman 76351-31167, Iran (e-mail: sabbagh@iauk.ac.ir; sorouri@iauk.ac.ir).

S. Antão and L. Sousa are with the Instituto de Engenharia de Sistemas e Computadores (INESC-ID), and Instituto Superior Técnico (IST), Lisboa 1000-029, Portugal (e-mail: sfan@sips.inesc-id.pt; las@inesc-id.pt).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2014.2305392

II. Background

The forward converter, modulo arithmetic units, and reverse converter are the main parts of the RNS. In contrast to other parts, reverse converter consists of a complex and nonmodular structure. Therefore, more attention should be directed to its design to prevent slow operation and compromise the benefits of the RNS. Both the characteristics of the moduli set and conversion algorithm have significant effects on the reverse converter performance. Hence, distinct moduli sets have been introduced [8]–[14]. In addition to the moduli set, hardware components selection is key to the RNS performance. For instance, parallel-prefix adders are known as unsuitable structures for complex reverse converters because of their high power consumption. However, parallel-prefix adders with its high-speed feature have been used in the RNS modular arithmetic channels. This performance gain is due to parallel carry computation structures, which is based on different algorithms such as [15]–[17]. Each of these structures has distinct characteristics, such as Sklansky (SK), and Kogge–Stone (KS) have the maximum and minimum fan-out, respectively, both providing minimal logic depth. Minimum fan-out comes at the expense of more circuit area [18]. Therefore, hardware components selection should be undertaken carefully.

III. New Parallel-Prefix-Based Components

The Chinese remainder theorem, or other related improved approaches and techniques [7] underlie the RNS reverse conversion, whose formulation can be directly mapped to ripple-carry adders (RCA). However, this leads to significant speed degradation, due to the linear increase of the delay in the RCA with the number of bits. Parallel-prefix adders can be used in the RNS reverse converters to bind the delay to logarithmic growth. However, in reverse converters, several parallel-prefix adders are usually required. Even when only one adder is used, the bit length of this adder is quite large. Consequently, this results in high power consumption notwithstanding its high speed. Therefore, in this section, two approaches that take advantage of the delay properties of the parallel prefix adders with competitive power consumption are introduced.

Usually, one regular binary addition is required in reverse converter structures to achieve the final binary representation. This final addition has an important effect in the total delay of the converter due to

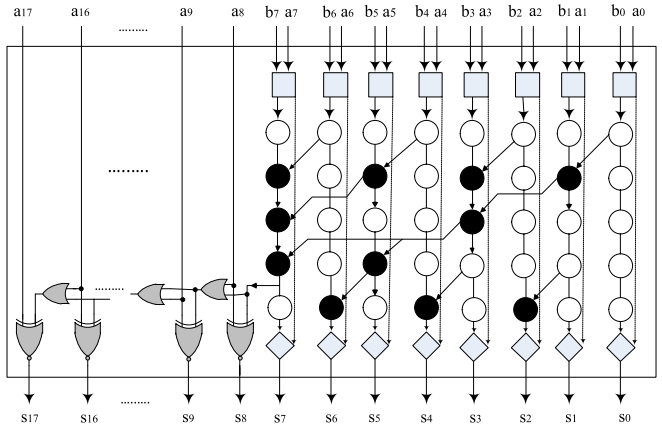

Fig. 1. HRPX structure with BK prefix network.

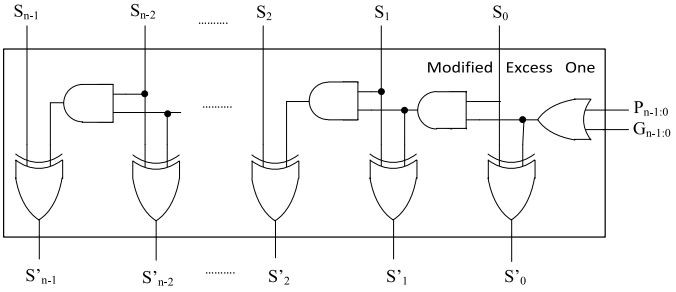

Fig. 2. Modified excess-one unit.

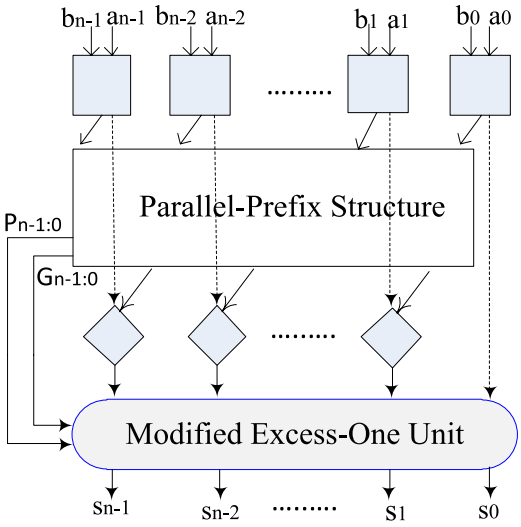

Fig. 3. HMPE structure.

the large bit-length of the operands. A thorough assessment of this final regular addition in recent converter designs shows that one of the operands has some constant bits with value 1 as highlighted by the following lemma, which applies to a class of converters described in [10].

Lemma 1: $(2n + 1)$ bits of the second operand of CPA4 of the converter in [10] are always constant and equal to one's.

Proof: The [10, CPA4] is a $(4n + 1)$-bit regular RCA that performs the subtraction presented in [10, eq. (52)]. This subtraction is accomplished in [10] as follows:

$$S = P - T = P + \bar{T} + 1 \tag{1}$$

where P and T are $4n + 1$ and $2n + 1$ bits binary vectors, respectively. Hence, it is clear that

$$P = \underbrace{P_{4n} \cdots P_1 P_0}_{4n+1} \tag{2}$$

$$\bar{T} = \underbrace{1 \cdots 11}_{2n+1} \underbrace{\bar{T}_{2n-1} \cdots \bar{T}_1 \bar{T}_0}_{2n}. \tag{3}$$

Hence, $2n + 1$ bits of one of operands of [10, CPA4] are always equal to one, approximately half of the total number of bits.

Based on the Lemma 1, a regular parallel-prefix adder with the desirable prefix structure can be used to perform the first part of the addition, for which the corresponding bits of the operands are fully variable, and a RCA with simplified logic to do the second part (full adder becomes XNOR/OR gates because of the constant operand). The proposed hybrid regular parallel-prefix XOR/OR (HRPX) adder component to perform the $(4n + 1)$-bit addition of [10, CPA4] for $n = 4$ is shown in Fig. 1. It should be noticed that due to the architecture of the reverse converter, the carry output of the XNOR/OR chain is not needed and can be ignored.

Second, the modulo $2^n - 1$ addition is an essential operation in the reverse conversion for most moduli sets [8]–[12]. The regular CPA with end around carry (EAC) [19] is by default a moduli $2^n - 1$ adder with double representation of zero, but in reverse converters a single representation of zero is required. So, a one-detector circuit has to be used to correct the result, which imposes an additional delay. However, there is a binary-to-excess-one converter (BEC) [20], which can be modified to fix the double-representation of zero issue.

The main reason for the high power consumption and area overhead of these adders is the recursive effect of generating and propagating signals at each prefix level. An optimized approach is proposed in [21], which uses an extra prefix level to add the output carry. However, this method suffers from high fan-out, which can make it usable only for small width operands. However, we could address this problem by eliminating the additional prefix level and using a modified excess-one unit instead. In contrast to the BEC, this modified unit is able to perform a conditional increment based on control signals as shown in Fig. 2, and the resulted hybrid modular parallel-prefix excess-one (HMPE) adder is depicted in Fig. 3. The HMPE consists of two parts: 1) a regular prefix adder and 2) a modified excess-one unit. First, two operands are added using the prefix adder, and the result is conditionally incremented afterward based on control signals generated by the prefix section so as to assure the single zero representation.

Summarizing, the HMPE is highly flexible, since it can be used with every prefix networks. Hence, the circuit performance metrics such as area, delay, and power-consumption can be adjusted by selecting the desired prefix structure. On the other hand, the HRPX avoids the usage of a large size parallel-prefix adder with high power-consumption, and also does not have the penalty of using the long carry-propagation chain of a RCA.

IV. Reverse Converter Design Methodology

In this section, the methodology of reverse converter design is described. In the following, a method employing distinct components in the architecture of the reverse converter will be presented. Several reverse converters for different moduli sets have been introduced,

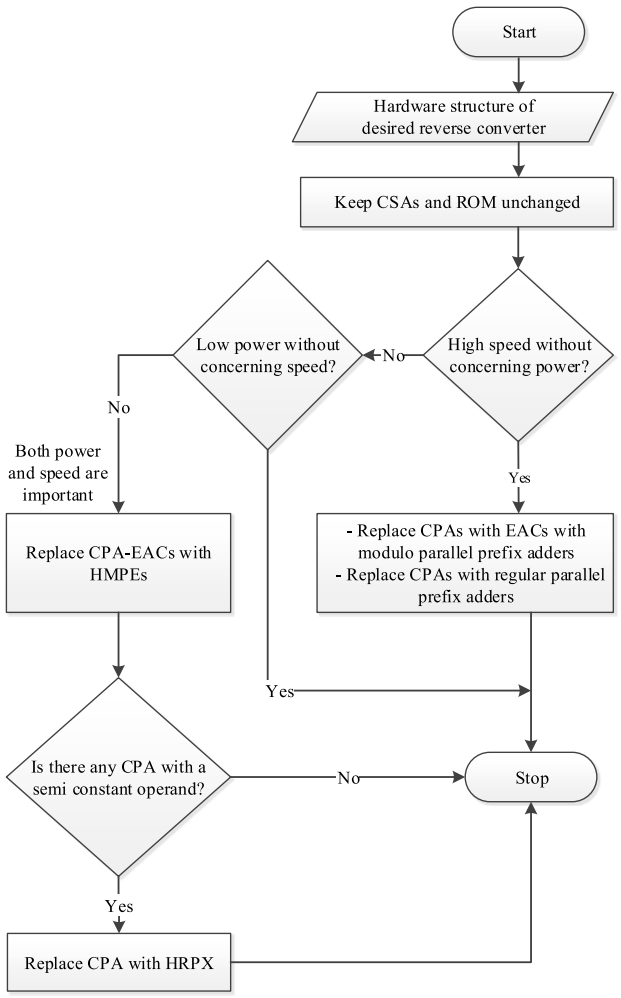

Fig. 4. Reverse converter design methodology.

which can be classified into three classes. The first class consists of converters with a tree of CSAs with EAC followed by a two-operand modulo $2^k - 1$ CPA [8], [10]. A second class includes more complex reverse converters, which have several CSAs and CPAs with EACs followed by a final regular subtractor with two operands of different size [10]–[12]. The implementation of this subtractor using regular binary-adder results in one operand with some constant bits. The third class covers the reverse converters that have been designed for moduli sets with moduli other than the popular 2^n and $2^n \pm 1$ [14]. In the following, we describe a methodology for designing reverse converters in the first and second classes. The suggested method for applying the HMPE and HRPX in the reverse converter is shown in Fig. 4.

First of all, it is relevant to decide about the required performance metrics based on the specified application. If it is just important to achieve the least power consumption and hardware cost without considering speed, no prefix adder is needed. On the other hand, if high speed is the designer goal, the CPAs with EAC and the regular CPAs should be replaced by traditional parallel prefix modulo $2^n - 1$ adders and regular parallel-prefix adders, respectively. However, for the VLSI designers, a suitable tradeoff between speed, power, and area is often more important. In this case, first, CPAs with the EAC can be replaced by the HMPEs. Then, if the converter contains a regular CPA where one of its operands has a string of constant bits with the value of one, it can be replaced with the HRPX.

V. VLSI Implementations

In order to support a thorough assessment, especially for power-consumption, the proposed method was applied to three different reverse converters and application-specific integrated circuits (ASICs) were implemented. The target reverse converters are: 1) the Converter-1 for moduli set $\{2^n - 1, 2^n, 2n+1, 2^{2n}+1 - 1\}$; 2) the Converter-2 for $\{2^n - 1, 2^n+1, 2^{2n}, 2^{2n}+1 - 1\}$; and 3) the Converter-3 for $\{2^n - 1, 2^n+1, 2^{2n}, 2^{2n}+1\}$. The architectures of these converters are proposed in [10] and [11], each of them is configured based on the Fig. 4 methodology. The implemented converters can be classified as follows: 1) cost effective designs using only the RCAs for the CPAs with the EAC and regular CPAs; 2) speed efficient designs, which substitute all the CPAs with EAC and the regular CPA by the parallel-prefix modulo $2^n - 1$ adders of [22]-Type-I, and KS regular parallel-prefix adders, respectively; and 3) designs that use both HMPE and HRPX, tradeoff between circuit parameters. Three well-known approaches for prefix network [18], i.e., Brent–Kung (BK), SK, and KS, have been considered for implementing the required prefix network in the proposed designs.

The circuits for all these configurations were designed and specified in the VHDL. Structural or behavioral descriptions can be considered. Behavioral VHDL describes just the circuit operation, and therefore the circuit's quality depends on synthesis tool [23]. However, herein our purpose is to compare the proposed architectures on a fair basis, and independently of the particular abilities of the synthesis tools. Consequently, a structural VHDL description is adopted. After a thorough verification, the ASICs were implemented using a general purpose standard cell library (TCBN65GPLUS, version 200A) tailored for the TSMC 65-nm CMOS logic salicide process (1-poly, 9-metal). The Cadence RTL Compiler tools (version v09.10-s242_1) was used for synthesizing the design and the Cadence Encounter and NanoRoute tools (versions v09.12-s159 and v09.12-s013, respectively) for placing and routing. Note that in any of the aforementioned technologies, no manual optimization of any kind was introduced. In addition, four different values of n (4, 8, 12, and 16) were considered to obtain experimental results for the different configurations of the implemented converters. The obtained results are presented in Tables I–VI. The results include: chip area (square micrometer), useful area (square micrometer), delay (nanosecond), power (milliwatt), AT^2, and PDP. The AT^2 and PDP are used to compare the circuit's area/latency and power/latency balancing.

Tables I and II show the results for the moduli set $\{2^n - 1, 2^n, 2^n+1, 2^{2n+1} - 1\}$ converters. As it was expected, the RCA-based converter [10] suggests the most competitive area and the power consumption metrics, but the highest delay. The suggested designs have considerably improved the delay, AT^2, and PDP while slightly increasing the area. Although, more power was consumed, the balance between power and delay becomes more competitive when the PDP metric is adopted. By comparing with the converter using fully parallel-prefix adders, the area, power, AT^2 (except at $n = 4$), and PDP of the proposed designs are significantly improved, but the delay increases. Experimental results for the converters with moduli-set $\{2^n - 1, 2^n+1, 2^{2n}, 2^{2n+1} - 1\}$ are presented in Tables III and IV. Similar behavior is observed except for the PDP metric. The PDP for the proposed converters are worse than for the RCA based [11] in three cases, but this improves for larger values of n, even for $n = 16$ the HMPE and HRPX-SK structure has better PDP than the RCA-based one.

Finally, the practical interest of the proposed approaches can be verified in Tables V and VI. Our main goal is to decrease the

TABLE I
EXPERIMENTAL RESULTS FOR MODULI $\{2^n - 1, 2^n, 2^n+1, 2^{2n+1} - 1\}$ CONVERTERS

Converter 1 Structure	Chip Area (μm^2)				Useful Area(μm^2)				Delay (ns)			
	4	8	12	16	4	8	12	16	4	8	12	16
RCA-based Adders	4199	8359	12137	17029	1402.6	2904.5	3918.2	5257.1	0.64	1.02	1.394	1.766
Fully Prefix Adders	5476	10774	18140	23741	1720.8	3589.9	6111.7	8435.2	0.324	0.505	0.515	0.565
HMPE & HRPX-KS	4873	10547	16348	22616	1591.6	3476.5	5797.1	7873.9	0.456	0.545	0.593	0.61
HMPE & HRPX-SK	4292	8853	12902	17739	1440.4	2842.2	4292.3	6188	0.474	0.53	0.579	0.626
HMPE & HRPX-BK	4199	8392	12746	16596	1458	2947	4367.5	5991.5	0.438	0.548	0.614	0.624

TABLE II
COMPARING POWER, AT^2 AND PDP FOR MODULI $\{2^n - 1, 2^n, 2^n+1, 2^{2n+1} - 1\}$ CONVERTERS

Converter 1 Structure	Power (mW)				AT^2				PDP			
	4	8	12	16	4	8	12	16	4	8	12	16
RCA-based Adders	3.631	4.534	5.004	5.447	574.50	3021.84	7613.99	16395.61	2.32	4.62	6.98	9.62
Fully Prefix Adders	9.522	11.64	18.54	24.18	180.64	915.51	1620.98	2692.73	3.09	5.88	9.55	13.66
HMPE & HRPX-KS	5.373	9.595	14.4	18.72	330.95	1032.61	2038.54	2929.88	2.45	5.23	8.54	11.42
HMPE & HRPX-SK	4.635	8.389	10.87	13.97	323.62	798.37	1438.95	2424.93	2.20	4.45	6.29	8.75
HMPE & HRPX-BK	5.055	8.148	10.74	14.44	279.71	885.00	1646.53	2332.95	2.21	4.47	6.59	9.01

TABLE III
EXPERIMENTAL RESULTS FOR MODULI $\{2^n - 1, 2^n+1, 2^{2n}, 2^{2n+1} - 1\}$ CONVERTERS

Converter 2 Structure	Chip Area (μm^2)				Useful Area(μm^2)				Delay (ns)			
	4	8	12	16	4	8	12	16	4	8	12	16
RCA-based Adders	4639	9858	13710	17901	1575.7	3223.8	4347.7	5638	0.693	1.056	1.33	1.736
Fully Prefix Adders	6674	12972	22350	29100	2259	4597.9	7729.6	10480.7	0.43	0.485	0.54	0.578
HMPE & HRPX-KS	5898	12655	20218	27783	1960.9	4376.5	6775.9	9753.8	0.472	0.577	0.607	0.667
HMPE & HRPX-SK	4873	10181	14639	18413	1661	3208.7	5267.5	6304.3	0.489	0.595	0.593	0.689
HMPE & HRPX-BK	4873	9999	13686	19207	1708.2	3417.8	4965.1	6684.1	0.451	0.577	0.646	0.659

TABLE IV
COMPARING POWER, AT^2, AND PDP FOR MODULI $\{2^n - 1, 2^n+1, 2^{2n}, 2^{2n+1} - 1\}$ CONVERTERS

Converter 2 Structure	Power (mW)				AT^2				PDP			
	4	8	12	16	4	8	12	16	4	8	12	16
RCA-based Adders	3.615	4.528	5.485	5.761	756.73	3594.98	7690.65	16991.22	2.51	4.78	7.30	10.00
Fully Prefix Adders	9.104	17.41	26.31	34.22	417.69	1081.54	2253.95	3501.43	3.91	8.44	14.21	19.78
HMPE & HRPX-KS	6.879	12.05	17.34	24.83	436.86	1457.06	2496.57	4339.36	3.25	6.95	10.53	16.56
HMPE & HRPX-SK	5.94	8.522	13.28	13.56	397.18	1135.96	1852.31	2992.78	2.90	5.07	7.88	9.34
HMPE & HRPX-BK	5.916	9.152	11.72	15.17	347.45	1137.88	2072.02	2902.78	2.67	5.28	7.57	10.00

TABLE V
EXPERIMENTAL RESULTS FOR MODULI $\{2^n - 1, 2^n+1, 2^{2n}, 2^{2n}+1\}$ CONVERTERS

Converter 3 Structure	Chip Area (μm^2)				Useful Area(μm^2)				Delay (ns)			
	4	8	12	16	4	8	12	16	4	8	12	16
RCA-based Adders	5040	9006	13478	17793	1591.9	2787.5	4098.2	5353.9	0.5	0.732	0.91	1.093
Fully Prefix Adders	6658	13156	20110	28192	2185.6	4641.5	7087	9800.3	0.222	0.268	0.268	0.292
HMPE-KS	7157	14255	21461	25119	2215.8	4551.8	6950.9	8987	0.203	0.268	0.294	0.442
HMPE-SK	5212	10568	14520	18656	1767.2	3422.2	4903.9	6525	0.232	0.26	0.416	0.456
HMPE-BK	5610	11310	15875	18960	1829.5	3645.7	5018.8	6478.9	0.242	0.27	0.285	0.455

cost of achieving high speed converters using parallel-prefix adders and also to provide applicable competitive tradeoff between power consumption and delay. For instance, with the HMPE and HRPX-SK converter for $n = 16\%$, 63% of the power is saved at the expense of 35% delay increase, and also 42% of improvement in the PDP is achieved when compared with fully parallel-prefix adders based

TABLE VI
COMPARING POWER, AT^2, AND PDP FOR MODULI $\{2^n-1, 2^n+1, 2^{2n}, 2^{2n}+1\}$ CONVERTERS

Converter 3 Structure	Power (mW)				AT^2				PDP			
	4	8	12	16	4	8	12	16	4	8	12	16
RCA-based Adders	4.446	5.42	6.382	6.997	397.98	1493.61	3393.72	6396.03	2.22	3.97	5.81	7.65
Fully Prefix Adders	15.04	25.5	39.99	53.85	107.72	333.37	509.02	835.61	3.34	6.83	10.72	15.72
HMPE-KS	17.47	28.11	37.56	26.52	91.31	326.93	600.81	1755.74	3.55	7.53	11.04	11.72
HMPE-SK	11.47	21.02	16.75	19.87	95.12	231.34	848.65	1356.78	2.66	5.47	6.97	9.06
HMPE-BK	11.77	21.11	27.59	20.48	107.14	265.77	407.65	1341.29	2.85	5.70	7.86	9.32

designs. In the other hand, the proposed designs consume more power to achieve higher speed than the RCA-based ones.

Summarizing, the use of modular and regular parallel-prefix adders proposed in this brief in reverse converters highly decrease the delay at the expense of significantly more power and circuit area, whereas the proposed prefix-based adder components allows one to achieve suitable tradeoffs between speed and cost by choosing the right adders for the parts of the circuits that can benefit from them the most.

VI. CONCLUSION

This brief presents a method that can be applied to most of the current reverse converter architectures to enhance their performance and adjust the cost/performance to the application specifications. Furthermore, in order to provide the required tradeoffs between performance and cost, new parallel-prefix-based adder components were introduced. These components are specially designed for reverse converters. Implementation results show that the reverse converters based on the suggested components considerably improve the speed when compared with the original converters, which do not use any parallel-prefix adder, and reduce the power consumption compared with the converters that exclusively adopt parallel-prefix adders.

REFERENCES

[1] A. Omondi and B. Premkumar, *Residue Number Systems: Theory and Implementations*. London, U.K.: Imperial College Press, 2007.

[2] B. Parhami, *Computer Arithmetic: Algorithms and Hardware Designs*, 2nd ed., New York, NY, USA: Oxford Univ. Press, 2010.

[3] J. Chen and J. Hu, "Energy-efficient digital signal processing via voltage-over scaling-based residue number system," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 7, pp. 1322–1332, Jul. 2013.

[4] C. H. Vun, A. B. Premkumar, and W. Zhang, "A new RNS based DA approach for inner product computation," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 60, no. 8, pp. 2139–2152, Aug. 2013.

[5] S. Antão and L. Sousa, "The CRNS framework and its application to programmable and reconfigurable cryptography," *ACM Trans. Archit. Code Optim.*, vol. 9, no. 4, p. 33, Jan. 2013.

[6] A. S. Molahosseini, S. Sorouri, and A. A. E. Zarandi, "Research challenges in next-generation residue number system architectures," in *Proc. IEEE Int. Conf. Comput. Sci. Educ.*, Jul. 2012, pp. 1658–1661.

[7] K. Navi, A. S. Molahosseini, and M. Esmaeildoust, "How to teach residue number system to computer scientists and engineers," *IEEE Trans. Educ.*, vol. 54, no. 1, pp. 156–163, Feb. 2011.

[8] Y. Wang, X. Song, M. Aboulhamid, and H. Shen, "Adder based residue to binary numbers converters for $(2^n-1, 2^n, 2^n+1)$," *IEEE Trans. Signal Process.*, vol. 50, no. 7, pp. 1772–1779, Jul. 2002.

[9] B. Cao, C. H. Chang, and T. Srikanthan, "An efficient reverse converter for the 4-moduli set $\{2^n-1, 2^n, 2^n+1, 2^{2n}+1\}$ based on the new Chinese remainder theorem," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 50, no. 10, pp. 1296–1303, Oct. 2003.

[10] A. S. Molahosseini, K. Navi, C. Dadkhah, O. Kavehei, and S. Timarchi, "Efficient reverse converter designs for the new 4-moduli sets $\{2^n-1, 2^n, 2^n+1, 2^{2n+1}-1\}$ and $\{2^n-1, 2^n+1, 2^{2n}, 2^{2n}+1\}$ based on new CRTs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 4, pp. 823–835, Apr. 2010.

[11] A. S. Molahosseini and K. Navi, "A reverse converter for the enhanced moduli set $\{2^n-1, 2^n+1, 2^{2n}, 2^{2n+1}-1\}$ using CRT and MRC," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI*, Jul. 2010, pp. 456–457.

[12] L. Sousa and S. Antao, "MRC-based RNS reverse converters for the four-moduli sets $\{2^n+1, 2^n-1, 2^n, 2^{2n+1}-1\}$ and $\{2^n+1, 2^n-1, 2^{2n}, 2^{2n+1}-1\}$," *IEEE Trans. Circuits Syst. II*, vol. 59, no. 4, pp. 244–248, Apr. 2012.

[13] L. Sousa and S. Antão, "On the design of RNS reverse converters for the four-moduli set $\{2^n+1, 2^n-1, 2^n, 2^{n+1}+1\}$," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 10, pp. 1945–1949, Oct. 2013.

[14] M. H. Sheu, S. H. Lin, C. Chen, and S. W. Yang, "An efficient VLSI design for a residue to binary converter for general balance moduli $(2^n-3, 2^n+1, 2^n-1, 2^n+3)$," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 51, no. 3, pp. 152–155, Mar. 2004.

[15] R. P. Brent and H. T. Kung, "A regular layout for parallel adders," *IEEE Trans. Comput.*, vol. 31, no. 3, pp. 260–264, Mar. 1982.

[16] J. Sklansky, "Conditional sum addition logic," *IRE Trans. Electron. Comput.*, vol. 9, no. 6, pp. 226–231, Jun. 1960.

[17] P. M. Kogge and H. S. Stone, "A parallel algorithm for the efficient solution of a general class of recurrence equations," *IEEE Trans. Comput.*, vol. 22, no. 8, pp. 783–791, Aug. 1973.

[18] R. Zimmermann, "Binary adder architectures for cell-based VLSI and their synthesis," Ph.D. dissertation, Integr. Syst. Labor., Dept. Inf. Technol. Electr. Eng., Swiss Federal Inst. Technol., Zurich, Switzerland, 1997.

[19] S. J. Piestrak, "A high speed realization of a residue to binary converter," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 42, no. 10, pp. 661–663, Oct. 1995.

[20] B. Ramkumar and H. M. Kittur, "Low power and area efficient carry select adder," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 2, pp. 371–375, Feb. 2012.

[21] R. Zimmermann, "Efficient VLSI implementation of modulo $(2^n \pm 1)$ addition and multiplication," in *Proc. 14th IEEE Int. Symp. Comput. Arithmetic*, Apr. 1999, pp. 158–167.

[22] R. A. Patel, M. Benaissa, and S. Boussakta, "Fast parallel-prefix architectures for modulo 2^n-1 addition with a single representation of zero," *IEEE Trans. Comput.*, vol. 56, no. 11, pp. 1484–1492, Nov. 2007.

[23] H. Kunz and R. Zimmermann, "High-performance adder circuit generators in parameterized structural VHDL," Integr. Syst. Lab., ETH Zürich Univ., Zürich, Switzerland, Tech. Rep. 96/7, 1996.