NOVEL MODULO 2^n+1 SUBTRACTORS

E. Vassalos[1], *D. Bakalis*[1], *and H. T. Vergos*[2]

[1] Electronics Laboratory, Physics Department
University of Patras
Patras, Greece

[2] Computer Engineering and Informatics Department
University of Patras
Patras, Greece

ABSTRACT

Novel architectures for designing modulo 2^n+1 subtractors are introduced, for both the normal and the diminished-one number representation of the operands. Zero-handling is also considered in the diminished-one operand representation case. The modulo 2^n+1 subtractors for operands in the normal representation that are proposed are shown to be more efficient in area, delay and power dissipation than the currently most efficient ones. The proposed diminished-one modulo 2^n+1 subtractors offer similar characteristics to those of the corresponding diminished-one adders.

Index Terms— Residue number system, modulo 2^n+1 circuits, subtraction, normal and diminished-one number representation.

1. INTRODUCTION

Arithmetic modulo 2^n+1 has been used in a variety of applications, ranging from pseudorandom number generation and cryptography up to digital signal processing (DSP). A modulo 2^n+1 channel is an integral part of almost every Residue Number System (RNS) [1] [2]. RNS is an arithmetic system well-suited to applications in which the operations are limited to addition, subtraction, multiplication and squaring. Since it offers significant speedup over the binary system, RNS has been adopted in the design of digital signal processors [3] [4], FIR filters [5-7], real-time image processing units [8] [9], Discrete Cosine Transform processors [10] [11], communication components [12-14] and other units [15] [16].

Every modulo 2^n+1 arithmetic circuit adopts one of two proposed representations for its input and output operands. The normal (binary) representation of a number in modulo 2^n+1 arithmetic has the disadvantage that it requires (n+1) bits while using only the 2^n+1 combinations. The diminished-one representation [17] attacks this problem by representing each operand decreased by one compared to its normal representation. As a result, only n bits are used in the computation units, leading to more efficient modulo 2^n+1 arithmetic circuits. However, zero operands and results have to be treated separately. Several architectures of modulo 2^n+1 adders [18-22] and of modulo 2^n+1 multipliers [18] [23-25] have been recently presented for both types of operands representation.

Subtraction is another operation frequently met in digital signal processing applications [8-11] [15] [16] for operations such as mean error estimation, mean square error estimation and calculation of sum of absolute differences. Since modulo arithmetic is also frequently used in these types of applications [1-16], efficient modulo subtraction circuits are welcome. However, very little work [26] has been presented on the design of modulo 2^n+1subtractors.

In this paper, we deal with the problem of designing efficient modulo 2^n+1 subtractors, for both the normal and the diminished-one operands representation.

The rest of the paper is organized as follows: The next section presents novel architectures for designing modulo 2^n+1 subtraction circuits. Evaluation and experimental results are given in Section 3. Finally, conclusions are given in the last section.

2. MODULO 2^n +1 SUBTRACTION

In this section we present novel architectures for designing modulo 2^n+1 subtractors. The first subsection deals with the normal number representation whereas the second subsection deals with the diminished-one number representation.

2.1. Normal Modulo 2^n+1 Subtraction

Let $A = a_n \cdots a_0$ and $B = b_n \cdots b_0$ denote two (n+1)-bit modulo 2^n+1 operands, such that $0 \le A, B \le 2^n$, that follow the normal representation. Let also $|x|_m$ denote the modulo m value of a k-bit operand x. The difference of A and B taken modulo 2^n+1 ($D = |A - B|_{2^n+1}$) can be computed as follows:

$$\begin{aligned} D = |A - B|_{2^n+1} &= |A + 2(2^n + 1) - B|_{2^n+1} \\ &= |A + (2^{n+1} - 1) - B + 3|_{2^n+1} \\ &= |A + \overline{B} + 3|_{2^n+1} \end{aligned} \quad (1)$$

978-1-4244-3298-1/09/$25.00 ©2009 IEEE

where $\overline{B}$ denotes the 1's complement of operand B. Relation (1) indicates that the modulo 2^n+1 difference of A and B is equivalent to the sum of A and $\overline{B}$ taken modulo 2^n+1 as long as a correction term equal to 3 is also taken into account.

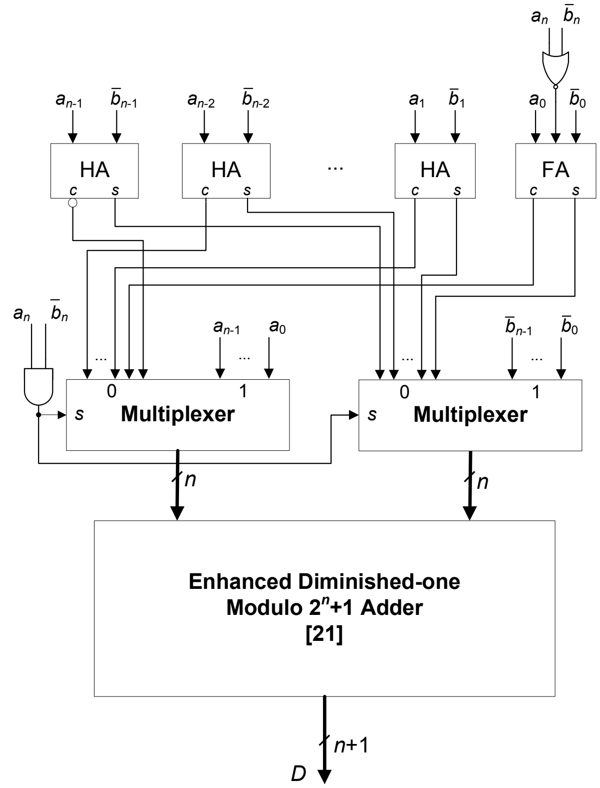
It has recently been shown [21] [22], that the modulo 2^n+1 sum of two (n+1)-bit operands X and Y that follow the normal representation can be carried out by a n-bit diminished-one modulo 2^n+1 adder. Specifically, the n least significant bits of X and Y along with a n-bit correction term, that depends on the values of the most significant bits of X and Y, are firstly added by an inverted end-around-carry carry save adder (EAC CSA) which consists of n full adders (FAs) and an inverter. The two n-bit outputs of the carry save adder are then driven to a diminished-one modulo 2^n+1 adder that produces the n least significant bits of the result. The most significant bit of the result is derived by detecting whether the two inputs of the diminished-one modulo 2^n+1 adder are complementary or not. According to [21], the area overhead for computing the most significant bit of the result is negligible when the diminished-one adder is designed using a parallel-prefix structure and at the same time the delay overhead is zero. Thus, for the modulo 2^n+1 addition of X with Y, it holds that:

$$\left|X+Y\right|_{2^n+1} = \left|X_{n-1}+Y_{n-1}+C\right|_{2^n+1} \qquad (2)$$

where $X_{n\text{-}1}$ and $Y_{n\text{-}1}$ denote the n least significant bits of X and Y respectively and C denotes the n-bit correction term which is equal to 1...1(x_n NAND y_n)(x_n XNOR y_n) while x_n and y_n denote the most significant bits of X and Y respectively.

We can use the normal modulo 2^n+1 addition architecture of [21] [22] for the normal modulo 2^n+1 subtraction as well, by substituting X and Y of equation (2) with A and $\overline{B}$ of equation (1). Furthermore, the constant value of 3 of equation (1) has to be considered along with the correction term C of equation (2). This leads us to a new correction term C' that is (n+1)-bit wide and its value is equal to (a_n AND $\overline{b}_n$)0...0($\overline{a}_n$ AND b_n). According to [21], when the correction term C' is equal to 2^n, then the carry save addition is not required and the inputs of the diminished-one adder should be driven directly by the n least significant bits of A and $\overline{B}$.

The above analysis leads to the proposed architecture for the modulo 2^n+1 subtractor that is presented in Figure 1. Two n-bit 2-to-1 multiplexers are used between the carry save adder and the diminished-one adder with a common select signal which is equal to (a_n AND $\overline{b}_n$). Furthermore, the (n-1)-bit leftmost FAs of the carry save adder can be simplified to half adders (HAs) since C' consists of (n-1) zeros and the rightmost FA along with the 2-input NOR logic gate can also be simplified since a_n (b_n) and a_0 (b_0) cannot be simultaneously at value 1 in modulo 2^n+1 arithmetic.



a_n $\overline{b}_n$
$a_{n\text{-}1}$ $\overline{b}_{n\text{-}1}$
$a_{n\text{-}2}$ $\overline{b}_{n\text{-}2}$
a_1 $\overline{b}_1$
a_0 $\overline{b}_0$
HA
HA
...
HA
FA
c s
a_n $\overline{b}_n$
$a_{n\text{-}1}$ a_0
$\overline{b}_{n\text{-}1}$ $\overline{b}_0$
0 1
s Multiplexer
0 1
s Multiplexer
n
n
Enhanced Diminished-one Modulo 2^n+1 Adder [21]
n+1
D

Fig. 1. Proposed modulo 2^n+1 subtractor for operands in the normal representation.

2.2. Diminished-one Modulo 2^n+1 Subtraction

Let A^* and B^* denote the diminished-one representation of A and B, respectively, with $0 < A, B \le 2^n$. Both A^* and B^* are n-bits wide, while A^*=A–1 and B^*=B–1. A diminished-one subtractor of A and B (that is, a circuit that accepts A^* and B^* and produces the diminished-one representation D^* of the difference $D = \left|A-B\right|_{2^n+1}$) can be computed as follows:

$$\begin{aligned} D^* &= \left|A-B-1\right|_{2^n+1} = \left|(A^*+1)-(B^*+1)-1\right|_{2^n+1} \\ &= \left|A^*-B^*-1\right|_{2^n+1} \\ &= \left|A^*+(2^n-1)-B^*+1\right|_{2^n+1} \\ &= \left|A^*+\overline{B}^*+1\right|_{2^n+1} \end{aligned} \qquad (3)$$

It is well known [18] [19], that the operation of a diminished-1 modulo 2^n+1 adder is equivalent to that of an integer adder with inverted end-around-carry. Therefore, if it is driven by two n-bit operands X and Y it computes $\left|X+Y+1\right|_{2^n+1}$. Hence, the diminished-one modulo 2^n+1 subtraction indicated in equation (3) can be achieved by a diminished-one modulo 2^n+1 adder driven by A^* and $\overline{B}^*$.

We further need to consider the cases where A or B or the result are equal to zero. Arithmetic circuits that deal with operands in the diminished-one representation usually utilize an extra bit for every operand, along with the n bits of its diminished-one representation, that indicates the case that the operand has a value equal to zero [20]. Let us denote as A_z and B_z the zero indication bits of A and B respectively, and as D_z the zero indication bit of D. The values of D_z and D^* for the four different combinations of A_z and B_z are given in Table I. The third line of Table I is justified as follows: When A=0 and B≠0, then $D = \left|A - B\right|_{2^n+1} = \left|-B\right|_{2^n+1} \neq 0$. Hence, D_z=0 and

$$D^* = \left|-B-1\right|_{2^n+1} = \left|(2^n+1)-(B^*+1)-1\right|_{2^n+1}$$
$$= \left|(2^n-1)-B^*\right|_{2^n+1} = \left|\overline{B}^*\right|_{2^n+1} = \overline{B}^* \tag{4}$$

Figure 2 presents the proposed architecture for a diminished-one modulo 2^n+1 subtractor capable of zero-handling. It is based on a diminished-one modulo 2^n+1 adder and a n-bit 4-to-1 multiplexer. A_z and B_z are used as the select signals of the multiplexer. The zero indication of the result is equal to 1 when: (a) A=B=0, or (b) A=B and A,B≠0. The first case can be detected by a 2-input AND gate whereas the second case can be detected by checking whether A^* and $\overline{B}^*$ have complementary values, or equivalently, by utilizing the enhanced diminished-one modulo 2^n+1 adder used in the previous subsection for the normal operands.

Unfortunately, the 4-to-1 multiplexer resides on the critical path of the circuit and therefore contributes to the delay of the modulo subtraction operation. A similar problem appears in the diminished-one modulo 2^n+1 adders case. To remove this additional delay, [20] presented a new diminished-one adder architecture that embeds the treatment of the zero operands within the parallel prefix structure of the adder and cancels the need for the 4-to-1 multiplexer. Since the proposed diminished-one modulo 2^n+1 subtractor is built around a diminished modulo 2^n+1 adder, we can also eliminate the 4-to-1 multiplexer by using as the diminished-one modulo 2^n+1 adder the one proposed in [20]. Then the resulting subtractor circuit takes the form of Figure 3.

TABLE I
TRUTH TABLE FOR DIMINISHED-ONE MODULO 2^N+1 SUBTRACTION

A_z	B_z	D	D_z	D^*
0	0	$\left\|A-B\right\|_{2^n+1}$	♦	$\left\|A^*-B^*-1\right\|_{2^n+1}$
0	1	$\left\|A\right\|_{2^n+1}$	0	A^*
1	0	$\left\|-B\right\|_{2^n+1}$	0	$\overline{B}^*$
1	1	0	1	0

♦ Depends on the values of A^* and B^*.

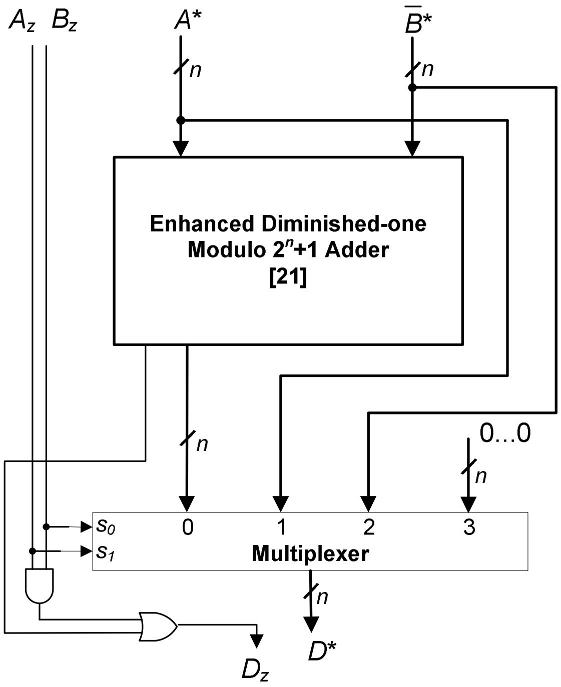


Fig. 2. Initial proposal for a diminished-one modulo 2^n+1 subtractor with zero-handling capability.

The adder's inputs are the A^* and $\overline{B}^*$ n-bit operands, along with the A_z and B_z zero indications. However, when B_z=1 (B=0), the second input of the adder should be driven with the all zero value and not with the all one value, in order to get the correct result. Hence, n 2-input NOR gates have to be used. The first input of every NOR gate is connected to B_z, while the second input is connected to B_i^*, i=0...n-1.

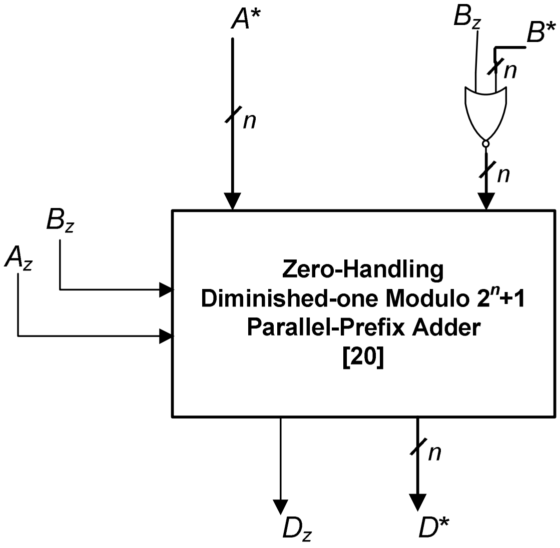

A^*
B_z
B^*
n
n
n
B_z
A_z
Zero-Handling Diminished-one Modulo 2^n+1 Parallel-Prefix Adder [20]
n
D_z
D^*

Fig. 3. Proposed diminished-one modulo 2^n+1 subtractor with zero-handling capability.

3. EVALUATION AND COMPARISONS

In this section we evaluate the circuits that result from the architectures that were proposed in the previous section and compare them with previously-presented architectures.

3.1. Normal Modulo 2^n+1 Subtraction

The proposed circuits for normal modulo 2^n+1 subtraction consist of: (a) (n-1) half adders and a simplified full adder for the inverted EAC CSA, (b) an enhanced diminished-one modulo 2^n+1 adder for the final addition [21], and (c) two n-bit 2-to-1 multiplexers for driving the appropriate inputs to the diminished-one modulo adder. We will compare our proposal against another recently proposed architecture [26] for modulo 2^n+1 subtraction of operands that follow the normal representation.

The architecture of [26] has been derived based on the equation:

$$|A-B|_{2^n+1} = \begin{cases} \left||A-B|_{2^n+1}\right| & if \quad A-B \geq 0 \\ \left||A-B+2^n+1|_{2^n+1}\right| & if \quad A-B < 0 \end{cases} \qquad (5)$$

To implement equation (5), [26] first converts the input operands A and B from unsigned numbers to signed numbers. It then computes both terms of equation (5) in parallel, using one 2-input binary adder and one 3-input binary adder. Finally, one (n+1)-bit 2-to-1 multiplexer is used to select between the outputs of the two binary adders and derive the correct result. Some further logic is also required that generates the selection signal of the multiplexer.

Comparing the architecture of [26] with the proposed architecture, we can notice that the 2-input binary adder of [26] and the diminished-one modulo 2^n+1 adder have similar area and delay requirements. The architecture of [26] further requires a 3-input binary adder and a (n+1)-bit 2-to-1 multiplexer, whereas the proposed architecture, besides the diminished-one adder, requires an inverted EAC CSA which mainly consists of half-adders, and two n-bit 2-to-1 multiplexers. Hence, we expect that the two architectures result in circuits with similar delays but the circuits of the proposed architecture require less area than that of [26].

In order to compare the area, delay and average power of the two architectures, we described in HDL modulo 2^n+1 subtractors assuming the proposed architecture as well as the architecture of [26]. We considered three different values of n, that is, n=4, 8 and 16. All binary adders follow the Kogge-Stone [27] parallel-prefix carry computation. The diminished-one modulo 2^n+1 adders follow the parallel-prefix carry computation of [19]. After validating the correct operation of the HDL descriptions, each design was synthesized and mapped to a 90nm power-characterized CMOS standard-cell library, assuming typical process parameters. Then, each design was optimized for area and delay using a standard optimization script. Finally, area and delay estimates were derived. For obtaining average dynamic power estimations, we followed a simulation-driven approach. We applied 50,000 random input vectors at a 500 MHz frequency at each design netlist and measured the average power dissipation using a commercial power estimator. The same vectors were applied to the corresponding netlists of the architectures under comparison. The attained area, delay and power estimates are given in Tables II, III, and IV, respectively.

The derived results indicate that the proposed subtractors offer significant savings in area and average power dissipation compared to the circuits of [26]. Savings up to 33% and 40% in the required implementation area and the average power consumed are observed in Tables II and IV, respectively. Furthermore, the proposed subtractors achieve higher operation frequencies.

3.2. Diminished-one Modulo 2^n+1 Subtraction

The proposed circuits for diminished-one modulo 2^n+1 subtraction, for a specific value of n, consist of: (a) n NOT logic gates for inverting the B^* operand, and (b) a diminished-one modulo 2^n+1 adder. If zero-handling is also required, then the proposed corresponding circuit consists of: (a) n 2-input NOR gates for controlling the B^* operand, and (b) a diminished-one modulo 2^n+1 adder with embedded zero-handling capability.

To the best of our knowledge, the proposed diminished-one modulo 2^n+1 subtractors of subsection 2.2 are the first ones reported in the open literature. Therefore, a direct comparison against some other architecture is not currently possible. However, it is obvious that the proposed modulo 2^n+1 subtractors offer comparable area, delay and power characteristics with those of the corresponding adders, since their only overhead against the latter is few logic gates.

TABLE II
AREA ESTIMATES OF MODULO 2^N+1 SUBTRACTORS FOR OPERANDS FOLLOWING THE NORMAL REPRESENTATION

n	[26] *(μm²)*	*Proposed* *(μm²)*	*Savings* *(%)*
4	2105	1401	33.4
8	3885	3110	19.9
16	8248	6827	17.2

TABLE III
DELAY ESTIMATES OF MODULO 2^N+1 SUBTRACTORS FOR OPERANDS FOLLOWING THE NORMAL REPRESENTATION

n	[26] *(ns)*	*Proposed* *(ns)*	*Savings* *(%)*
4	0.38	0.35	7.9
8	0.44	0.41	6.8
16	0.52	0.51	1.9

TABLE IV
POWER ESTIMATES OF MODULO 2^N+1 SUBTRACTORS FOR OPERANDS FOLLOWING THE NORMAL REPRESENTATION

n	[26] *(mW)*	*Proposed* *(mW)*	*Savings* *(%)*
4	0.87	0.52	40.6
8	1.63	1.11	32.0
16	3.35	2.69	19.7

4. CONCLUSIONS

Modulo 2^n+1 arithmetic either independently or as a part of an RNS has found applicability in a variety of DSP algorithms implementations and in the design of DSP processors. Therefore, efficient implementations of modulo 2^n+1 arithmetic units are always welcome.

In this paper we have presented novel architectures for modulo 2^n+1 subtraction circuits, for operands in either the normal or the diminished-one representation. Experimental results indicate that the proposed subtractors for operands in the normal representation offer significantly less area and consume significantly less power than those previously reported [26], while also being faster. The proposed subtractors for operands in the diminished-one representation are capable of handling zero-operands and stem from adding few logic gates over the corresponding diminished-one modulo 2^n+1 adders. As a result, their delay, area and power characteristics are expected to be very close to those of the adders and therefore of high efficiency.

5. REFERENCES

[1] P. V. Ananda Mohan, *Residue Number Systems: Algorithms and Architectures*, Kluwer Academic Publishers, 2002.

[2] A. Omondi, and B. Premkumar, *Residue Number Systems: Theory and Implementation*, Imperial College Press, 2007.

[3] R. Chaves, and L. Sousa, "RDSP: A RISC DSP based on Residue Number System," In *Proc. Euromicro Symposium on Digital System Design*, pp. 128–135, 2003.

[4] J. Ramirez, A. Garcia, S. Lopez-Buedo, and A. Lloris, "RNS-enabled digital signal processor design," *Electronics Letters*, vol. 38, no. 6, pp. 266–268, March 2002.

[5] G. Cardarilli, A. Nannarelli, and M. Re, "Reducing power dissipation in FIR filters, using the residue number system," In *Proc. IEEE Midwest Symposium on Circuits and Systems*, pp. 320-323, 2000.

[6] J. Ramirez, and U. Meyer-Baese, "High performance, reduced complexity programmable RNS-FPL merged FIR filters," *Electronics Letters*, vol. 38, no. 4, pp. 199-200, February 2002.

[7] Y. Liu, and E. Lai, "Moduli set selection and cost estimation for RNS-based FIR filter and filter bank design," *Design Automation for Embedded Systems*, vol. 9, no. 2, Springer, pp. 123-139, June 2004.

[8] F. Marino, E. Stella, A. Branca, N. Veneziani, and A. Distante**,** "Specialized Hardware for Real-Time Navigation," *Real-Time Imaging*, vol. 7, no. 1, pp. 91-108, February 2001.

[9] B. Rejeb, and W. Anheier, "Real-time implementation of fractal image encoder," In *Proc. Mediterranean Electrotechnical Conference*, vol.2, pp. 612- 615, 2000.

[10] P. G. Fernandez, and A. Lloris, "RNS-based implementation of 8x8 point 2D-DCT over field-programmable devices," *Electronics Letters*, vol. 39, no. 1, pp. 21-23, January 2003.

[11] P.G. Fernandez, A.Garcia, J.Ramirez, and A. Lloris, "Fast RNS-based 2D-DCT computation on Field-Programmable Devices," In *Proc. IEEE Workshop on Signal Processing Systems*, pp. 365-373, 2000.

[12] U. Meyer-Baese, A. Garcia, and F. Taylor, "Implementation of a communications channelizer using FPGAs and RNS arithmetic," *Journal of VLSI Signal Processing*, vol. 28, no. 1-2, pp. 115-128, June 2001.

[13] J. Ramírez, A. García, U. Meyer-Baese, and A. Lloris, "Fast RNS FPL-based communications receiver design and implementation", In *Proc. Int. Conference on Field Programmable Logic*, pp. 472-481, 2002.

[14] M. Panella, and G. Martinelli, "An RNS Architecture for Quasi-Chaotic Oscillators," *Journal of VLSI Signal Processing*, vol. 33, no. 1-2, pp. 199-220, February 2002.

[15] Y. Liu, and E.M.-K Lai, "Design and implementation of an RNS-based 2-D DWT processor," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp. 376-385, February 2004.

[16] P.M. Matutino, and L. Sousa, "An RNS based Specific Processor for Computing the Minimum Sum-of-Absolute-Differences," In *Proc. Euromicro Conference on Digital System Design*, pp. 768-775, 2008.

[17] L. M. Leibowitz, "A simplified binary arithmetic for the Fermat number transform," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 24, no.5, pp. 356–359, October 1976.

[18] R. Zimmermann, "Efficient VLSI implementation of modulo $2^n\pm1$ addition and multiplication," In *Proc. Symposium on Computer Arithmetic*, pp. 158-167, 1999.

[19] H. T. Vergos, C. Efstathiou, and D. Nikolos, "Diminished-One Modulo 2^n+1 Adder Design," *IEEE Transactions on Computers*, vol. 51, no. 12, pp 1389-1399, December 2002.

[20] C. Efstathiou, H.T. Vergos, D. Nikolos, "Handling zero in diminished-one modulo $2^n + 1$ adders," *International Journal of Electronics*, vol. 90, no 2, pp. 133-144, February 2003.

[21] H.T. Vergos, and D. Bakalis, "On the Use of Diminished-1 Adders for Weighted Modulo 2^n+1 Arithmetic Components," In *Proc. Euromicro Conference on Digital System Design*, pp. 752-759, 2008.

[22] H. T. Vergos, C. Efstathiou, "A Unifying Approach for Weighted and Diminished-1 Modulo 2^n+1 Addition," *IEEE Transactions on Circuits and Systems II*, vol. 55, no. 10, pp. 1041-1045, October 2008.

[23] C. Efstathiou, H. T. Vergos, G. Dimitrakopoulos and D. Nikolos, "Efficient diminished-1 modulo 2^n+1 multipliers," *IEEE Transactions on Computers*, vol. 54, no. 4, pp. 491–496, April 2005.

[24] L. Sousa, and R. Chaves, "A universal architecture for designing efficient modulo 2^n+1 multipliers," *IEEE Transactions on Circuits and Systems I*, vol. 52, no. 6, pp. 1166-1178, June 2005.

[25] H. T. Vergos, and C. Efstathiou, "Design of efficient modulo 2^n+1 multipliers," *IET Computers and Digital Techniques*, vol. 1, no. 1, pp. 49-57, January 2007.

[26] S. Timarchi, K. Navi, and M. Hosseinzade, "New Design of RNS Subtractor for modulo 2^n+1," In *Proc. Int. Conference on Information and Communication Technologies*, pp 2803-2808, 2006.

[27] P. M. Kogge, and H. S. Stone, "A Parallel Algorithm for the Efficient Solution of a General Class of Recurrence Equations," *IEEE Transactions on Computers*, vol. 22, no. 8, pp 786-792, August 1973.