INDEX-BASED RNS DWT ARCHITECTURES FOR CUSTOM IC DESIGNS

J. Ramírez [1], P. G. Fernández [2], U. Meyer-Bäse [3], F. Taylor [4], A. García [5] and A. Lloris [1]

[1] *Dept. of Electronics and Computer Technology*
University of Granada, SPAIN

[2] *Dept. of Electrical Engineering*
University of Jaén, SPAIN

[3] *Dept. of Electrical and Computer Engineering*
FAMU-FSU College of Engineering, USA

[4] *High-Speed Digital Architecture Laboratory*
University of Florida, USA

[5] *School of Computer Engineering*
Universidad Autónoma de Madrid, SPAIN

Abstract - The design of high-performance, high-precision, real-time digital signal processing (DSP) systems, such as those associated with wavelet signal processing, is a challenging problem. This paper reports on the innovative use of the residue number system (RNS) for implementing high-end wavelet filter banks. The disclosed system uses an enhanced index-transformation defined over Galois fields to efficiently support different wavelet filter instantiations without adding any extra cost or additional look-up tables (LUT). An exhaustive comparison against existing two's complement (2C) designs for different custom IC technologies was carried out. These structures demonstrated to be well suited for field programmable logic (FPL) assimilation as well as for CBIC (cell-based integrated circuit) technologies.

INTRODUCTION

There is a growing demand for digital image processing to be performed at greater real-time bandwidths, with higher precision and lower complexity. Since these systems are intrinsically SAXPY (S=AX+Y) dominant, advanced solutions must overcome existing arithmetic limitations. An arithmetic system capable of surmounting this barrier is the residue number system, or RNS. Computer arithmeticians have long held that the RNS offers a distinct MAC (multiply and accumulate) speed-area advantage [1] in SAXPY-intensive applications. In [2], the RNS was used to design a wavelet transform using field-programmable logic (FPL). The design was compared to a two's complement (2C), and distributed arithmetic (DA) implementation. The RNS solution was found to be superior to the 2C case and compared favourably with the DA instantiation, but unlike a DA design, was fully programmable. Another demonstration of the RNS benefits is found in [3] for use in orthogonal wavelet filter bank applications. The filter banks were designed to accept 8-bit input signals, process using 10-bit coefficients, and ran 23.45% and 96.58% faster than a 2C design for one and two octaves, respectively. A weakness of the reported RNS solution was that fixed coefficient multiplication was mapped into look-up tables (LUTs). Consequently, the tables needed to be re-programmed whenever a different set of wavelet coefficients were selected. This paper explores an efficient means of obtaining efficient discrete

0-7803-7145-3/01/$10.00 ©2001 IEEE

wavelet transform (DWT) architectures defined over multiple filter coefficient sets, by means of the RNS.

INDEX-BASED ARITHMETIC

In the RNS, numbers are represented in terms of a relatively prime basis set (moduli set) $P=\{m_1, \ldots m_L\}$. Any number $X \in \boldsymbol{Z}_M=\{0, 1, \ldots, M-1\}$, where $M=\prod_{i=1}^{L} m_i$, has a unique RNS representation $X \leftrightarrow \{X_1,\ldots,X_L\}$, where $X_i=X \bmod m_i$. Like the 2C system, the RNS arithmetic is exact as long as the final result is bounded within the system's dynamic range $\boldsymbol{Z}_M$. Mapping from the RNS back to the integer domain is defined by the Chinese Remainder Theorem (CRT) [1]. RNS arithmetic is defined by pair-wise modular operations:

$$\begin{aligned} Z = X \pm Y &\leftrightarrow \left[\left| X_{m_1} \pm Y_{m_1} \right|_{m_1}, \ldots, \left| X_{m_L} \pm Y_{m_L} \right|_{m_L} \right] \\ Z = X \times Y &\leftrightarrow \left[\left| X_{m_1} \times Y_{m_1} \right|_{m_1}, \ldots, \left| X_{m_L} \times Y_{m_L} \right|_{m_L} \right] \end{aligned} \tag{1}$$

where $|Q|_{m_j}$ denotes Q mod m_j. The individual modular arithmetic operations are typically performed as LUT calls to small memories. The RNS differs from traditional weighted numbering systems in the fact that the RNS arithmetic is *carry-free* and can operate at a constant speed over a wide range of precisions.

A variety of RNS multipliers are available, including pure LUT multipliers, square root multipliers, index-transform multipliers, and array multipliers. The index-transformation multipliers [4, 5] are based on the mathematical properties associated with a Galois fields denoted GF(p), where p is prime. All the non-zero elements in a Galois field can be generated by exponentiating a primitive element denoted g_j. This property can be exploited for multiplication in GF(m_j) through the use of a well known isomorphism existing between the multiplicative group $Q=\{1, 2, \ldots, m_j-1\}$, with multiplication performed modulo m_j, and the additive group $I=\{0, 1, \ldots, m_j-2\}$, with addition performed modulo (m_j-1). The mapping is given by: $q=\Phi_j^{-1}(i)=g_j^i$, $q \in Q$, $i \in I$ and multiplication, using index arithmetic is based on $\left|q_j q_k\right|_{m_j} = g^{\left|i_j+i_k\right|_{m_j-1}}$. Thus, the multiplication of two numbers, say q_j and q_k, can be performed by adding exponents in a modular sense. The exponents, or indexes, i_j and i_k, can be pre-computed and stored in a lookup table or LUT. Adding the indexes can be performed with a modulo (m_j-1) adder, and the inverse index transformation of i_j into q_j can be performed again using a LUT.

DWT SOLUTIONS

Wavelets [6] are gaining in importance, especially for use in image coding and compression applications. The discrete wavelet transform (DWT) decomposes a signal at increasing resolution levels (multi-scale resolution). An attractive feature

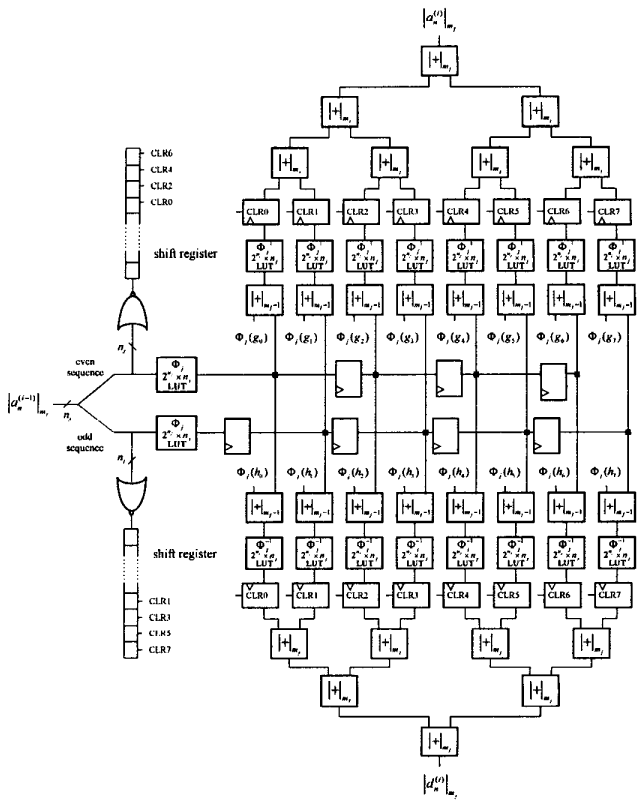

Fig. 1. Design of an RNS-based
1-D DWT architecture with index-transformation.

of the wavelet series expansion is that the underlying multi-resolution structure leads to an efficient discrete-time algorithm based on a filter bank implementation. An N^{th}-order 1-D DWT decomposition of a sequence x_n is defined by:

$$\begin{aligned} a_n^{(i)} &= \sum_{k=0}^{N-1} g_k a_{2n-k}^{(i-1)} \qquad i = 1,\ 2,\ \ldots,\ J \\ d_n^{(i)} &= \sum_{k=0}^{N-1} h_k a_{2n-k}^{(i-1)} \qquad a_n^{(0)} \equiv x_n \end{aligned} \tag{2}$$

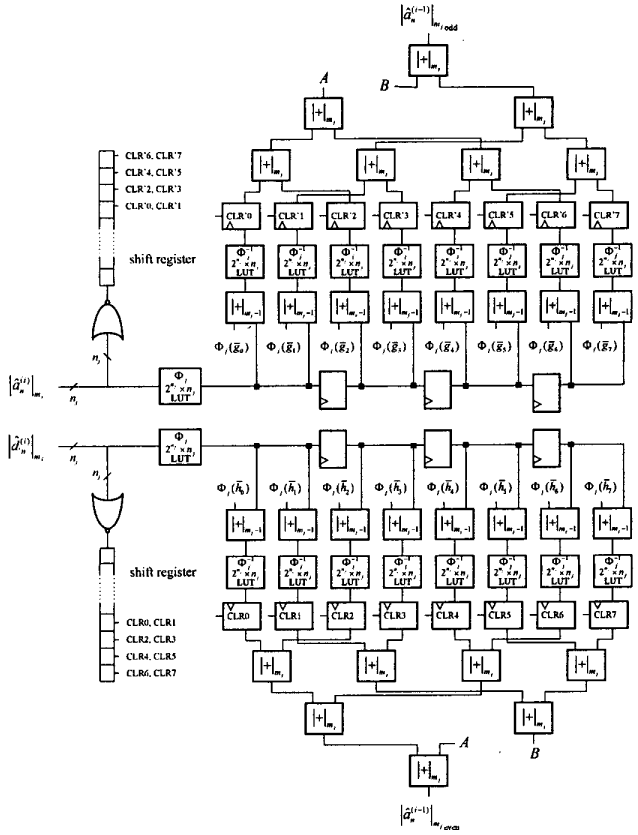

Fig. 2. Design of an RNS-based
1-D IDWT architecture with index-transformation.

where $a_n^{(i)}$ and $d_n^{(i)}$ are the octave-i approximation and detail sequences, respectively, and g_k and h_k (k= 0, 1, …, N-1) correspond to the low-pass and high-pass analysis filter coefficients. The signal x_n can be perfectly recovered through its multiresolution decomposition by iteration:

$$\hat{a}_m^{(i-1)} = \begin{cases} \sum_{k=0}^{N/2-1} \overline{g}_{2k} \hat{a}^{(i)}_{\frac{m}{2}-k} + \sum_{k=0}^{N/2-1} \overline{h}_{2k} \hat{d}^{(i)}_{\frac{m}{2}-k} & m \text{ even} \\ \sum_{k=0}^{N/2-1} \overline{g}_{2k+1} \hat{a}^{(i)}_{\frac{m-1}{2}-k} + \sum_{k=0}^{N/2-1} \overline{h}_{2k+1} \hat{d}^{(i)}_{\frac{m-1}{2}-k} & m \text{ odd} \end{cases} \quad (3)$$

where $\overline{g}_k$ and $\overline{h}_k$ represent low-pass and high-pass synthesis filter coefficients.

The design of wavelet filter banks using the RNS, presents new opportunities. If the wavelet filter coefficients are fixed *a priori*, the LUT-based modulo multiplier represents the most efficient solution to meeting low-latency and hardware efficiency. However, if the wavelet filter coefficients are to be run-time programmable, then the solution may require an unacceptably large number of LUTs to cover all coefficient instances. The use of index-transformation multipliers [4, 5] and re-timing techniques leads to DWT filterbanks designs requiring a single $2^{n_j} \times n_j$ LUT for each filter coefficient, where $n_j = \lceil \log_2(m_j) \rceil$ is the modulus wordwidth. Figure 1 shows the design based on index transformations of a modulo m_j channel, for an octave-i 8-tap decomposition filter bank. The input sequence $\left| a_n^{(i-1)} \right|_{m_j}$ is decomposed into even and odd sequences that are converted to the index-domain by means of two LUTs storing the Φ_j function. Some circuitry is added to the input to detect zero values of the input sequences. The reason for this is that multiplication by zero is not defined in the index domain and must be considered to be a special case. Notice that clearable registers have been added to make zero the filter products in case zero is detected in the even- and odd-indexed sequences. After the filter products are computed in the index-domain, the LUT storing the function Φ_j^{-1} maps the indexes back to the RNS domain, and the remaining filtering or addition stage is carried out by a modular adder tree.

The system exhibits symmetry for the computation of the approximation and detail sequences. The complete RNS design consists of a number of parallel channels whose combined wordwidth suffices to insure that the solution dynamic range requirements are met [4, 5].

In a similar manner, an architecture based on index-transformation may be derived for the reconstruction (synthesis) filter bank. The resulting architecture for the 1-D IDWT is shown in Figure 2.

RESULTS AND DEVELOPMENT

An 8-tap 1-D DWT filter bank was used to illustrate the design of 2C and RNS-based system. The comparison was carried out using VHDL models over Altera FLEX10KE field programmable logic (FPL) devices and two standard cell ASIC technologies. The selected ASIC reference libraries were the 0.8μm MSU SCMOS and the Chip Express 0.35μm triple-level metal CX3003 CMOS technologies. The 0.8μm MSU SCMOS cell library consists of a set of gates implementing low-level

logic functions. The Chip Express 0.35μm CMOS CX3003 technology is based on the definition of a high-level module that can be configured to operate in a very wide range of simple and complex circuit functions and combinations. The logic module is a universal function composed of three multiplexers and one AND gate. It is based on the fact that a multiplexer can implement any logic function, which may be either combinatorial or sequential.

Table 1 shows the total area and maximum sampling rate obtained for 8-tap RNS and 2C designs using 0.8μm and 0.35μm CBIC technologies. The solution adopted here for the 2C arithmetic DWT architecture was to use pipelined 2C multipliers based on Booth encoding and Wallace trees [7]. Hardware complexity and delay rapidly increase as the precision of the input and coefficients increases. These facts are shown in Table 1 and Figure 3. Note that performance is considerably higher for an RNS-based solution than for a 2C design. In order to maximize the sample rate gain, small wordwidth channels are desirable. However, only prime moduli are suitable for use in an index arithmetic system. For a 5-bit modulus set, the only admissible moduli are {17, 19, 23, 29, 31} which leads to a 22.7-bit maximum dynamic range. With a 6-bit modulus set, the dynamic range can be up to 39 bits using the moduli set {37, 41, 43, 47, 53, 59, 61}. The use of a 6-bit modulus set was found to be attractive for the designs demanding 23-, 27- and 29-bit outputs, while for the design with a 21-bit output a 5-bit modulus set is more efficient in terms of area and speed. The efficient hardware implementation of modulo multiplication by means of index transformations reveals 2C and RNS-based systems to have similar hardware complexities, while an RNS solution will take advantage of higher speed and better ASIC routability inside each channel. Table 2 shows the total resources required and maximum sampling rate obtained for a 4-tap DWT filter bank using a grade –1 Altera FLEX10KE FPL device, as well as the moduli selected to cover the dynamic range. Figure 4 shows the sampling rate as a function of the output precision. The use of 5- and 6- bit modulus set was found to be an attractive choice since performance is only limited by the LUT operation.

wordwidths modulus set	**2C**				**RNS**			
	Area (μm^2 or #modules)		F(MHz)		Area (μm^2 or #modules)		F(MHz)	
	0.8μm	0.35μm	0.8μm	0.35μm	0.8μm	0.35μm	0.8μm	0.35μm
[8,10,21] {31,29,23,19,17}	748608	19810	106.38	367.65	820360	29500	209.64	584.80
[10,10,23] {61,59,53,47}	855016	21849	105.71	353.36	910688	36436	188.32	515.46
[12,12,27] {61,59,53,47,43}	1026864	25507	86.43	293.26	1138360	45545	188.32	515.46
[14,12,29] {61,59,53,47,43,41}	1111376	28441	84.89	223.21	1366032	54654	188.32	515.46

Table 1. Total area and maximum sampling rate obtained for an 8-tap DWT filter bank. Note, [x,y,z] represents x-bit input, y-bit coefficients and z-bit output.

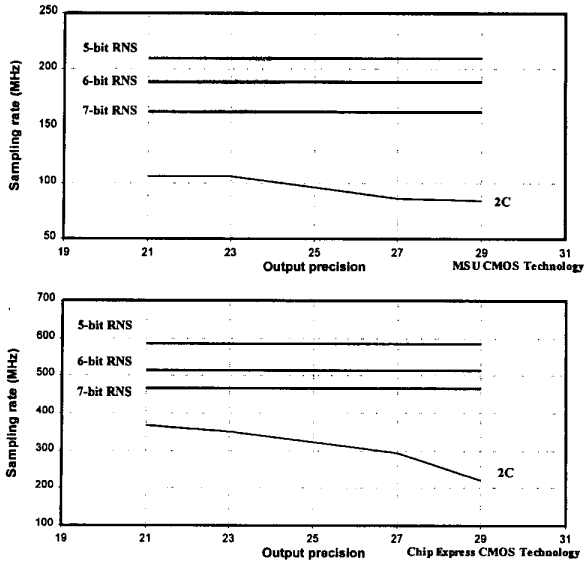

Fig. 3. Sampling rate as a function of the output precision for index-based and 2C arithmetic 1-D DWT filter banks implemented by means of CBIC technologies.

wordwidths modulus set	2C			RNS		
	#LEs	#EABs (Memory bits)	F(MHz)	#LEs	#EABs (Memory bits)	F(MHz)
[8,9,19] {61,59,53,47}	3470	0	39.06	4×314	4×10 (15360)	135.13
[8,10,20] {61,59,53,47}	3440	0	38.16	4×314	4×10 (15360)	135.13
[9,10,21] {61,59,53,47}	3647	0	34.24	4×314	4×10 (15360)	135.13
[10,10,22] {61,59,53,47}	4354	0	30.67	4×314	4×10 (15360)	135.13
[12,12,26] {61,59,53,47,43}	5446	0	27.93	5×314	5×10 (19200)	135.13
[14,12,28] {61,59,53,47,43}	7972	0	26.95	5×314	5×10 (19200)	135.13

Table 2. Total resources required and maximum sampling rate obtained for a 4-tap DWT filter bank on an Altera FLEX10KE device (grade –1). Note, $[x,y,z]$ represents x-bit input, y-bit coefficients and z-bit output.

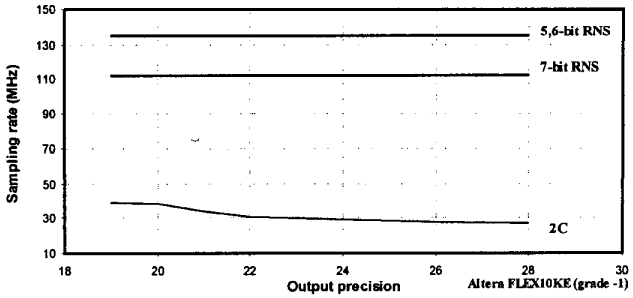

Fig. 4. Sampling rate as a function of the output precision for index-based and 2C arithmetic 1-D DWT filter banks implemented with FPL devices.

BINARY-TO-RNS AND RNS-TO-BINARY CONVERTERS

Binary-to-RNS conversion can be carried out efficiently by decomposing the B-bit 2C word, say x, into a weighted sum of smaller words $\overline{x}_i$ (e.g., 4-bit words). Equation (4) exemplifies the case where a 4-bit decomposition is considered:

$$|x|_{m_j} = \left| -2^{B-1} x_{B-1} + \sum_{l=0}^{B-2} 2^l x_l \right|_{m_j} = \left| -2^{B-1} x_{B-1} + \sum_{i=0}^{p-1} \overline{x}_i 2^{4i} \right|_{m_j} \tag{4}$$

The implementation of Equation (4) requires only $2^4 \times n_j$ LUTs.

RNS-to-binary conversion implies the use of a CRT (Chinese Remainder Theorem)-based converter. However, CRT conversion can often be a barrier in certain applications. The auto-scaling RNS-to-binary converter (ε-CRT) proposed by Griffin *et al.* [8] can overcome these drawbacks by using a few LUTs and binary (modulo 2^n) adders. For a scaled n-bit binary output, and a n_j-bit modulus set, this converter needs one $2^{nj} \times n$ LUT for each modulus of the RNS and a n-bit adder tree. This solution results more appropriate for most applications demanding high data rates [9]. Table 3 shows the area requirements for the binary-to-RNS and RNS-to-binary converters. As a result, the high throughput advantage of the presented RNS architectures over 2C designs is not degraded when converters are inserted in the system.

Filter bank precisions and modulus set	Binary-to-RNS / RNS-to-binary converters Area requirements			
	0.8 μm MSU		0.35 μm CX3003	
	CA/NCA (μm^2)		CA/NCA (#modules)	
	BIN→RNS (#stages)	RNS→BIN 16-bit output (#stages)	BIN→RNS (#stages)	RNS→BIN 16-bit output (#stages)
[8, 10, 21] {31, 29, 23, 19, 17}	8696/4200 (2)	24252/15845 (4)	180/85 (2)	502/361 (4)
[10, 10, 23] {61, 59, 53, 47}	14893/6045 (2)	43378/23457 (4)	314/119 (2)	910/534 (4)
[12, 12, 27] {61, 59, 53, 47, 43}	19545/7582 (2)	54878/29345 (4)	412/152 (2)	1137/668 (4)
[14, 12, 29] {61, 59, 53, 47, 43, 41}	23587/9280 (2)	64897/34920 (4)	490/186 (2)	1364/795 (4)

Table 3. Area required for binary-to-RNS and ε-CRT RNS-to-binary converters.

CONCLUSION

This paper reports on the design and implementation using FPL and CBIC technologies of forward and inverse wavelet filter banks by means of the RNS. The architecture is based on index-transformation over Galois fields, and requires a single LUT for each filter coefficient multiplication. Efficient circuitry is used to detect a zero value in the input sequence, a requirement of the design paradigm. The RNS design was compared to 2C architectures of comparable size. The reported methodology demonstrated a sustained performance improvement over 2C designs.

ACKNOWLEDGEMENTS

J. Ramírez, P. G. Fernández, A. García and A. Lloris were supported by the Comisión Interministerial de Ciencia y Tecnología (CICYT, Spain) under project PB98-1354. CAD tools and supporting material were provided by Altera Corp., San Jose CA, under Altera University Program agreements.

References

[1] M. A. Sodersterand, W. K. Jenkins, G. A. Jullien, F. J. Taylor, *Residue Number System Arithmetic: Modern Applications in Digital Signal Processing*. IEEE Press, New York, 1986.

[2] U. Meyer-Baese, J. Buros, W. Trautmann, F. Taylor, "Fast Implementation of Orthogonal Wavelet Filterbanks Using Field-Programmable Logic", *Proc. of the 1999 IEEE International Conference on Acoustics, Speech and*

Signal Processing, vol. 4, pp. 2119-2122, 1999.

[3] J. Ramírez, A. García, P. G. Fernández, L. Parrilla, A. Lloris, "RNS-FPL Merged Architectures for the Orthogonal DWT", *Electronics Letters*, vol. 36, no. 14, pp. 1198-1199, 2000.

[4] G. A. Jullien, "Implementation of Multiplication, Modulo a Prime Number, with Applications to Number Theoretic Transforms", *IEEE Trans. on Computer*, vol. C-29, no. 10, pp. 899-905, 1980.

[5] D. Radhakrishnan, Y. Yuan, "Fast and Highly Compact RNS Multipliers", *International Journal of Electronics*, vol. 70, nº 2, pp. 281-293, 1991.

[6] M. Vetterli, J. Kovacevic, *Wavelets and Subband Coding*. Prentice Hall, 1995.

[7] J. Pihl, E. J. Aas, "A Multiplier and Squared Generator for High Performance DSP Applications", *Proc. of the 39th Midwest Symposium on Circuits and Systems*, 1996.

[8] M. Griffin, F. J. Taylor, and M. Sousa, "New scaling algorithms for the Chinese Remainder Theorem," *Proc. of the 22nd Asilomar Conf. on Signals, Syst. and Comp.*, CA, 1988.

[9] J. Ramírez, A. García, P. G. Fernández, L. Parrilla, A. Lloris, "A New Architecture to Compute the Discrete Cosine Transform using the Quadratic Residue Number System," *Proc. of the 2000 International Symposium on Circuits and Systems*, vol. 5, May 2000, pp. 321-324.