XVI IMEKO World Congress
Measurement - Supports Science - Improves Technology - Protects Environment ... and Provides Employment - Now and in the Future
Vienna, AUSTRIA, 2000, September 25-28

A SELF CORRECTING ANALOG TO RNS CONVERTER

R. Lojacono, G.C. Cardarilli and M. Re
Electronics Engineering Dept. - University of Rome "Tor Vergata"
Via di Tor Vergata, 110 – 00133 – Rome – Italy

Abstract: RNS arithmetic has received great attention in the literature [1] and its advantages have been clearly pointed out but practical applications are, until now, very poor. This fact is related to different reasons but the most important is the need of a complex conversion from analog or binary quantities to RNS representation (input conversion)[2] and to the necessary final conversion from RNS representation to the classical binary one (output conversion)[3]. Recently, some authors have faced the input conversion problem, introducing the MRNS [4] and giving a first hardware implementation for the conversion from analog quantities to RNS. Starting from this proposal, the present paper gives an alternative solution using traditional RNS representation and an architecture that can avoid, under some suitable hypotheses, the crucial problem of the conversion errors. The error conversion handling is of fundamental importance because this conversion leads to a non-positional representation.
Keywords: Analog to Digital Converter, Residue Number Systems, Fast Processing

1 INTRODUCTION

The advantages of the RNS representation of integer numbers are well known [1]: this system is carry-free and allows the possibility to decompose the calculations in parallel flows of lower complexity. However, these advantages are greatly reduced by the need of a conversion from classical binary representation to the RNS one and, finally, by the needs of an inverse conversion from RNS representation to the classical binary one. Recently, some authors have faced this last problem, giving interesting solutions [3]. Also the first problem has been considered in the literature but with minor attention. In this case, the main problem is to overcome the conventional structure based on a conventional A/D converter followed by a binary to RNS translator. However, a direct conversion from analog to RNS representation (ARNS in the following) appears very difficult to implement, particularly by the fact that the conversion errors, when the final representation is a non-positional one, can induce macroscopic effects., Particular procedures for this conversion have been proposed in order to avoid these errors [4]. These techniques are essentially based on particular type of folding and on modulus redundancy in order to control the correctness of the results.

In this paper a mixed analog/digital architecture is proposed for realizing a self-correcting ARNS converter. No modulus redundancy is introduced. The proposed architecture appears to be compact. Under reasonable hypotheses, the given conversion is always correct. The constraints, which are at the base of the proposed conversion algorithm, are given in section 2. The structure and the performance of the proposed architecture are given in section 3. Concluding remarks are given in section 4.

2 BASIC REMARKS

A RNS is based on N pairwise prime moduli m_i , i=1,…,N and can represent univocally $M = \Pi m_i$ different numbers. The representation is obtained through N residual quantities r_i , i= 1,…,N. Consequently, an integer X < M is represented by

$$r_i = \langle X \rangle_{m_i} = \langle \alpha m_i + r_i \rangle_{m_i} , \ i = 1,\dots,N \tag{1}$$

We suppose, as in most cases, that all the moduli m_i are odd. In this case, starting from the last right side of (1), we can remark the following properties:

i - $\alpha_i m_i$, being m_i an odd number , is odd or even according with α_i;

ii – if we consider the quantities X, α_i , r_i in binary form

$$LSB(X)=LSB[LSB(\alpha_i) + LSB(r_i)] = p_i \ , \ i = 1,\dots,N \tag{2}$$

with $p_i \in \{0,1\}$ and LSB (X) = p_i , i = 1,…,N (i.e. each p_i , in a correct conversion, must be equal to each other).

Now we suppose that some errors are introduced in the ARNS conversion. In this case some of the said p_i are not equal to the others. We can assume that the correct value of the p_i's (p^*) can be chosen on the basis of the majority voting rule. So we can isolate the wrong parities p_j , j =1,…,L. For these p_j we have that α_j or r_j are wrong (if α_j and r_j are together wrong, the parity p_j is equal to p^* and the presence of these errors cannot be detected).

Now it is reasonable to consider that α_j can be wrong if X is very close to a multiple of m_j. This means that the corresponding r_j will be or 0 or $m_j - 1$ (we assume that only an error less than 1 LSB is possible). Since 0 and $m_i - 1$ are both even we can conclude that α_j is wrong and must be changed in $\alpha_j + 1$, if $r_j = m_i - 1$, or in $\alpha_j - 1$, if r_j is 0. At the same time, if $r_j = m_i - 1$ must be changed in 0 and if r_j is 0 must be changed in $r_j = m_i - 1$. So the parity p_j result changed in p^*.

If X is not close to a multiple of m_j , i.e. $r_j \neq 0$ and $r_j \neq m_j - 1$, it is not reasonable to suppose that α_j is wrong; then we assume that r_j is wrong and must be changed in $r_j + 1$ or $r_j - 1$. Unfortunately we haven't sufficient information to resolve this ambiguity. This problem can be overcome by introducing some additional hardware. In fact, we can perform a binary conversion of the difference

$$\alpha_i m_i - \alpha_j m_j = r_j - r_i = r_{ij} \quad (3)$$

Using this difference, it results that the correct value of r_j is given by

$$r_j = r_{ij} + r_i \quad (4)$$

3 PROPOSED ARCHITECTURE

The proposed architecture follows the scheme of the digitally corrected subranging converter as shown in Fig. 1 for the case of three moduli.

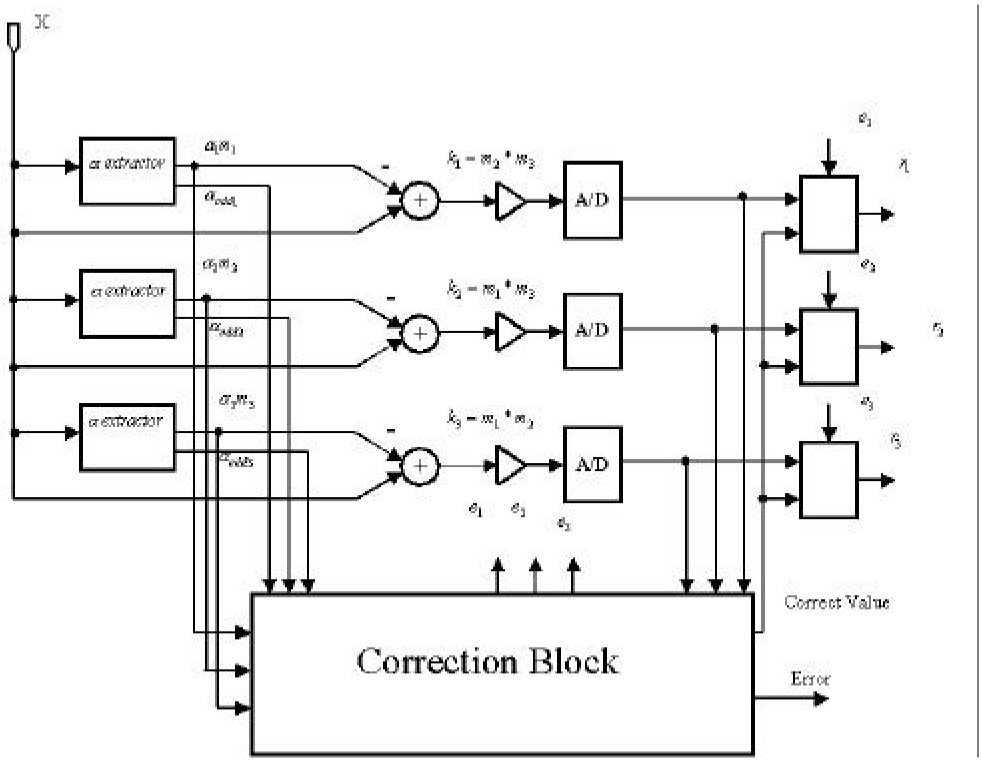

Figure 1. Whole converter architecture

However the first step is performed in analog way: only the odd (=1) or even (=0) nature of α_j is retained in digital form and the residue, in analog form, is converted in binary form by the second stage. The converter uses parallel ways to evaluate the residues belonging to each modulus. The structure of each α extractor is depicted in the Fig. 2.

The XOR gates present in the circuit are used for selecting the value of $\alpha_i m_i$, such that $\alpha_j m_j < X < (\alpha_j+1) m_j$. So $X-\alpha_i m_i$ corresponds to the analog value of the residue r_l. The OR gate presents in the figure is used to evaluate the LSB of $\alpha_i m_i$ (i.e., it evaluates if α_i is odd or even).

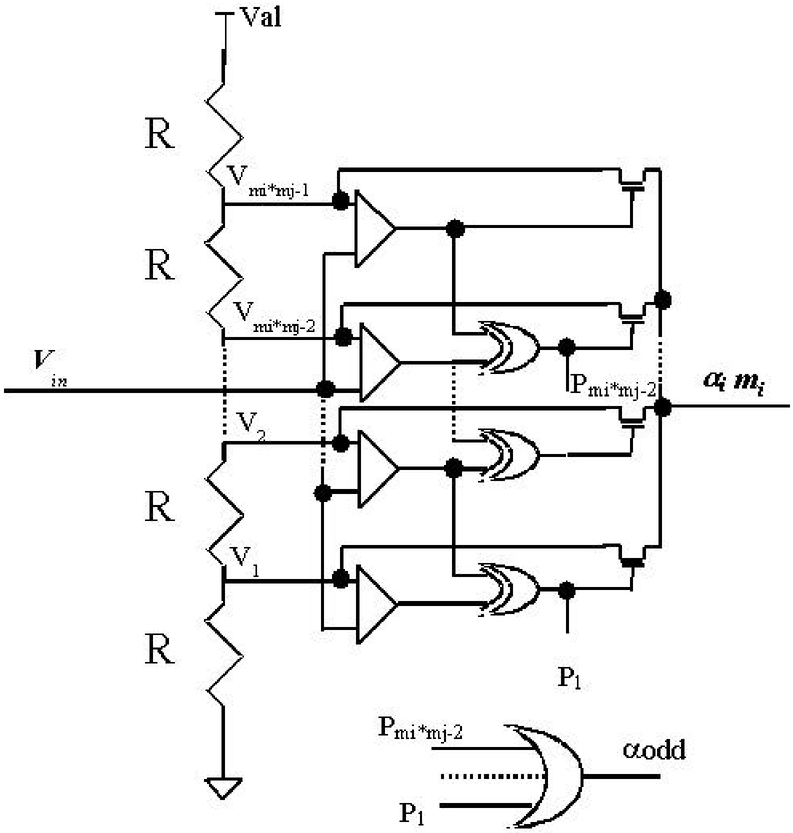

Figure 2. α extractor.

Fig. 3 shows the structure of the correction block. If the OK signal is 1 the presence of one error is detected. The error correction starts from the management of the three bits p_i I=1..3.

The XORs of these bits are sent to PLA1 which selects the proper difference $\alpha_i m_i - \alpha_j m_j$ (which is arranged as a positive voltage difference). The ADC converts this difference in a digital word r_{ij} which is added to a proper r_i selected by the MUX3. The result becomes an input of the final MUX6. This procedure performs the correction in the case of a wrong residue r_j.

At the same time the right part of the correction block manages the cases of error located in α_j. This means that, as said in the previous section, the value of the corresponding residue r_j must be changed from 0 to m_j-1 or vice versa. MUX5 selects the wrong r_j and MUX4 the corresponding m_j-1 while the NOR and the COMP furnish to the PLA2 the inputs which determine the proper output for MUX6. Using this output, MUX6 selects the right final r_j among the value given from the adder, the value 0 and the value m_j-1.

Example:

As an example let us consider the case of three moduli RNS arithmetic based on the modulus set {13, 15, 17}. With this set we obtain a wordlength that is close to 12 bits. The related converter uses three αm extractors and three 3 bit AD converters. However, the modulus 17 requires a slight modification of the DA circuit, introducing an additional voltage level and, consequently, a resistor and a voltage comparator. While the converter complexity is close to that of a three bits, the converter output uses four bits.

Another 3 bit converter is used in the correction block. It converts the analog voltage corresponding to the αm difference (3).

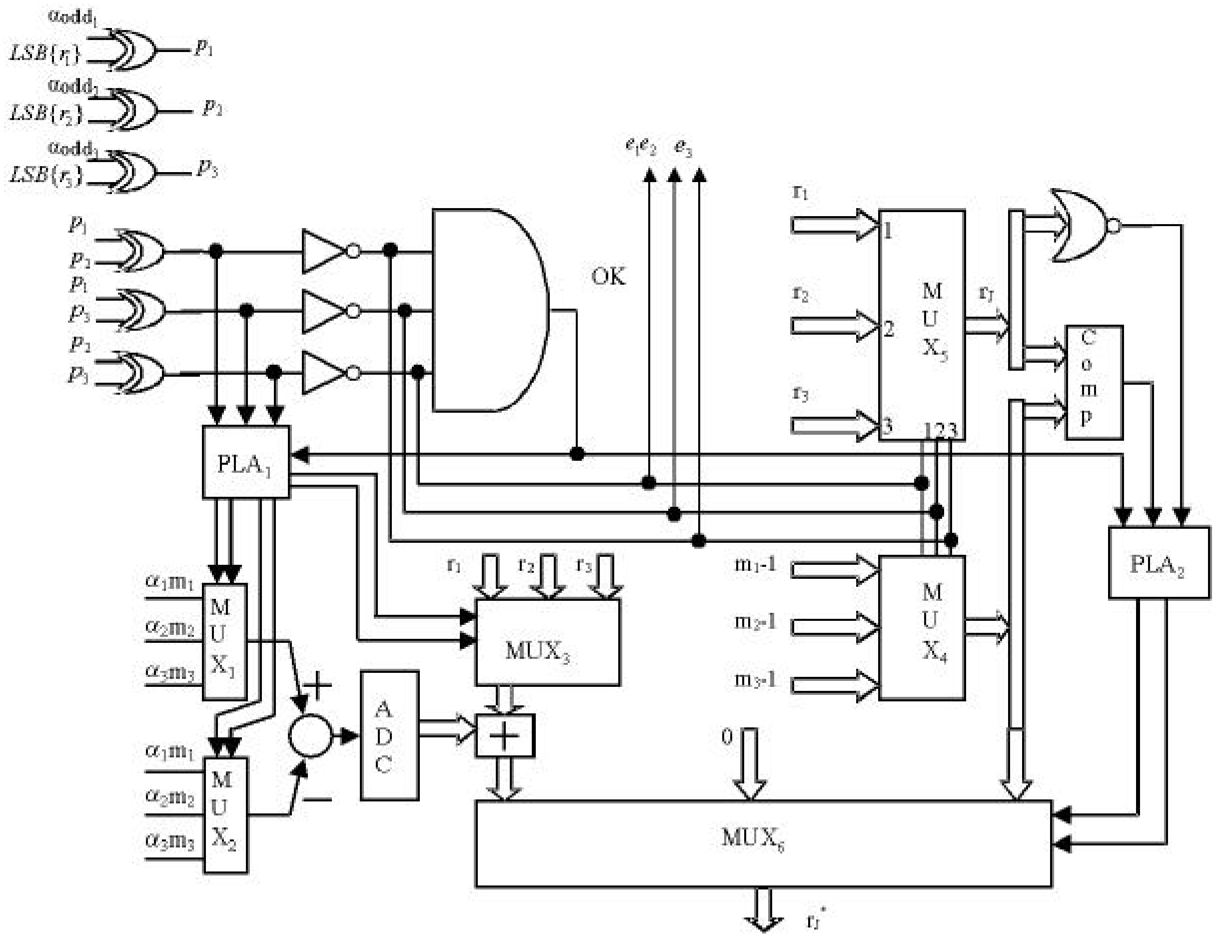

Figure 3. Architecture of the correction block.

4 CONCLUSIONS

The problem of direct conversion from analog to a n moduli RNS representation has been faced. Starting from the hypothesis that all the moduli are odd and using the majority voting rule, it is pointed out a procedure which allows the possibility of correcting the minority of wrong residues. When used in a processing system based on RNS computation, the proposed converter avoids the use of any complex binary to RNS translators, required, instead, when conventional ADCs are used.

The architecture of the converter has been developed in the case of a simple three moduli RNS,. This architecture shows that a correct conversion can be obtained with a structure that is comparable to a conventional subranging converter in terms of complexity and conversion time.

REFERENCES

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

[2] T. Stouraitis, Efficient Convertors for Residue and Quadratic Residue Number Systems, *IEE Proc. –G,* **139** (6) (1992) 626-634.

[3] G. C. Cardarilli, M. Re, and R. Lojacono, Efficient Modulo Extraction for CRT Based Residue to Binary Converters, *Proceedings of IEEE International Symposium on Circuits and Systems ,* (Hong Kong, 9-12 June), Hong Kong 1997, vol. 3, p. 2036-2039.

[4] P.E. Pace, P. A. Ramamoorthy, and D. Styer, A Preprocessing Architecture for Resolution Enhancement in High-Speed Analog-to-Digital Converters, *IEEE Trans. On Circuits and Systems – II* **41** *(6) (1994) 373-379.*

AUTHOR(S): R. Lojacono, G.C. Cardarilli and M. Re, Electronics Engineering Dept. - University of Rome "Tor Vergata", Via di Tor Vergata, 110 – 00133 – Rome – Italy, e-mail: [cardarilli, marco.re]@ieee.org, lojacono@ing.uniroma2.it