AN EFFICIENT RESIDUE TO ANALOG CONVERTER

Michael Lewis^[1], Jon Mellott^[1], and Fred Taylor^[1,2]

[1] The Athena Group, Inc., [2] University of Florida

ABSTRACT

The residue number system (RNS) has been applied to digital signal processing (DSP) applications that are characterized by demanding speed, power, and precision requirements. An element in a mixed-signal RNS-enabled solution is a residue to analog converter (RAC), a device that maps an RNS number into an equivalent real number. In this paper, an efficient RAC implement is presented that can be realized using existing technology.

1. INTRODUCTION

The residue number system, or RNS, is an unweighted numbering system that supports high-speed arithmetic within concurrent, non-communicating, small wordlength channels [1]. Because of this feature, computer arithmeticians have historically promoted the RNS for high-speed arithmetic-intensive applications. Engineers developing RNS-based solutions have also discovered that the RNS can also enable designs of reduce complexity, power consumption, and costs. A number of FPGA -based studies have demonstrated RNS design principles and speed/complexity advantages. Application domains include FIRs [2], discrete wavelet transform (DWT) filter banks [3], communications channelizers (digital down converter) [4], and wavelet-based image compressors [5]. RNS advantages have also been reported when the technology is reduced to ASIC implementation, including FFTs [6]. As RNS systems become more plentiful, it is natural to expect to see their use extend into many new applications. This may will require the fusion of RNS processing agents with an *analog to RNS* converter (ARC) and RNS to analog converter (RAC). Existing ARC designs are based on the use of conventional analog to digital converters (ADC) and a layer of logic that converts a digitized integer value of an analog sample, into a set of residue digits [1]. RAC design strategies have been less studied. They are historically based on the direct use of Chinese Remainder Theorem (CRT), or mixed radix conversion (MRC) algorithm-based residue to integer converters. These designs have proven to be complex, requiring a considerable investment in dedicated logic, resulting in speed, power and cost penalties. In this paper,

an efficient RAC architecture is presented that is capable of running at high real-time speeds within a low-complexity low-power architecture.

2. RESIDUE NUMBER SYSTEM

Numbers in the RNS are represented in terms of a bases set $P=\{p_1, ..., p_i, ..., p_L\}$ of relatively prime integers p_i . The *i*th *moduli*, denoted p_i , is assumed to be bounded by $p_i \le 2^b$. The dynamic range of the RNS system is given by $M = \prod p_i$, i = 1, 2,..., *L*. Any integer number $X \in [0,..., M-1] \equiv Z_M$, has a unique RNS representation given by $X \leftrightarrow \{X_1, ..., X_L\}$ where $X_i = X \mod(p_i)$. Here X_i is called the *i*th *residue* of *X* modulo p_i . Signed integers can also be represented, reserving the range [0, ..., M/2-1] for positive numbers and [M/2, ..., M-1] for negative numbers. For purposes of illustration and discussion, it will be assumed that all numbers are positive.

Arithmetic in the RNS is performed as a set of concurrent pairwise operations. If \bullet denotes modular addition, subtraction, or multiplication, then for X and $Y \in Z_M$, and $Z=(X \bullet Y) \in Z_M$, it follows that:

$$Z=(X \bullet Y) \equiv \{ (X_1 \bullet Y_1) \operatorname{mod}(p_1), \dots, (X_L \bullet Y_L) \operatorname{mod}(p_L) \} 1.$$

which can be implemented with a set of small wordlength concurrent (non-communicating) operations. It is this property that attracted computer arithmeticians with the promise of high-speed low-complexity data processing. Regardless of the details, it is now reasonably well known how to reduce the RNS to practice in silicon.

An ARC is used to map an analog sample x into an RNS L-tuple. A conventional ADC device first converts x into a quantized number X which is then translated to RNS using combinational logic lookup tables. A RAC uses the reverse multi-step process to synthesize a facsimile of x. The Chinese Remainder Theorem (CRT) or mixed radix conversion algorithm (MRC) can be used to map an RNS L-tuple { $X_1,..., X_L$ } to an integer X which is converted to an analog data sample using a conventional DAC device. At present this operation carries a rather high complexity

overhead due to intrinsic CRT inefficiencies. Specifically, the conventional CRT formula is given by:

$$X = \left\langle \sum_{i=1}^{L} \left(m_i \left\langle m_i^{-1} X_i \right\rangle \operatorname{mod}(p_i) \right) \right\rangle \operatorname{mod}(M) \qquad 2$$
$$m_i = M / p_i = \prod_{\substack{j=1\\i \neq j}}^{L} p_j, \left\langle m_i^{-1} m_i \right\rangle \operatorname{mod}(p_i) = 1$$

The historical problem with implementing the CRT algorithm, shown in Equation 2, has been its dependence on a large wordlength modulo(M) adder. The ε -CRT provides a path around this problem [7]. The ε -CRT is actually a scaled CRT where the scale factor is chosen to convert the mod(M) dependence into a far simpler mod(2^n) mapping where $M > 2^n$. Such a converter can be implemented using a conventional *n*-bit adder. This is a logical approach to the RAC problem since signals leaving the system are normally quantized to 4-16 bits of precision. Specifically, for a given M, one chooses a scale factor k such that $k=M/2^n$ ($2^n=M/k$). The ε -CRT algorithm converts an RNS *L*-tuple { $X_1,...,X_L$ } into a scaled number X^n where:

$$X' = \left\langle \sum_{i=1}^{L} \left(m'_i \left\langle m^{-1}_i X_i \right\rangle \mod(p_i) \right) \right\rangle \mod(2^n) \quad 3$$
$$m'_i = \operatorname{round}(m_i / k); \; m'_i \le m_i$$

where "round" denotes rounding. Whereas the dynamic range of X was originally bounded by M, X' is bounded by 2^n in the scaled system. The precision of the reconstructed number X' is determined by the number of fractional bits "f" retained in the rounding operation ($f \le b$). An obvious choice is to have no fractional precision assigned to m_i ' (f=0), causing m_i ' to take the nearest integer value. These concepts are illustrated in the following example.

3. EXAMPLE CRT AND E-CRT

Suppose $P=\{3, 4, 5\}$, resulting in $M=60>2^5$. The integer X=31 has the RNS representation $31 \leftarrow \text{RNS} \rightarrow \{1, 3, 1\}$. For the traditional CRT inversions:

$$m_1=20; m_1^{-1}=2; m_2=15; m_2^{-1}=3; m_3=12; m_3^{-1}=3$$

Inverting, one obtains:

 $X = \langle (20\langle 1*2\rangle \text{mod}(3) + 15\langle 3*3\rangle \text{mod}(4) + 12\langle 1*3\rangle \text{mod}(5)) \rangle$ mod(60) = $\langle 40 + 15 + 36\rangle$ mod(60) = 31

An ϵ -CRT, for $k=M/2^5=60/32=15/8$, and no fractional bits of precision are retained by each m_i ' (f=0), produces:

$$m_1=20; m_1=round(20/(15/8)=round(10.66) \rightarrow 11$$

 $m_1=15; m_2 = round(15/(15/8) = round(8) \rightarrow 8$ $m_1=12; m_3 = round(12/(15/8) = round(6.4) \rightarrow 6$

Inverting, one obtains:

 $X = \langle (11(1*2) \mod(3) + 8(3*3) \mod(4) + 6(1*3) \mod(5)) \rangle$ $\mod(32) = \langle 22 + 8 + 18 \rangle \mod(32) = 16$

The scaled system interprets X=31 as $X_s=31/k=16.533$, which the ε -CRT interprets to be X'=16. If full fractional precision of each of the m_i 's had been retained {i.e., m_i =(10.66, 8, 6.4)}, then $X_s=X'=16.533$ which is error-free after rescaling by k=15/8.

4. RESIDUE TO ANALOG CONVERSION

The CRT and MRC algorithms have been historically been the enablers of RACs. Their primary limitation has been complexity. The complexity of a RAC can be significantly reduced with the use of an ε -CRT to replace a standard CRT or MRC. The RAC, shown in Figure 1 represents a design that directly employs an ε -CRT and is based on the use of small wordlength lookup tables (LUT), typically having a 6-bit or less address space. The LUTs are used to replace the computation of $\phi(X_i) = (m_i \le m_i^{-1}X_i \ge mod(p_i))mod(2^n)$. The *n*'-bit (*n*'=*n*+*f*) wide lookup outcome is addressed by the *b*-bit word X_i . The precision of an ε -CRT converter is controlled by the number of fractional bits of precision retained in each m_i ' which also control the size of the LUTs.

The bxn '-bit LUT responses are combined with an n 'bit digital adder as shown in Figure 1. The n '-bit outcome is then converted into an analog level with an n'-bit, or less, DAC. Nothing in the processes is technologically unusual. The DAC speed and resolution requirements are assumed to be application dependent. Suppose that a system's real-time data rate requirements can be met by 16to 14-bit DAC. The error variance introduced by an Lmoduli ε -CRT is on the order of $\sigma^2 = LO^2/12$, where $O = 2^{-1}$ for f denoting the fractional bit resolution. For the L=3-moduli demonstration system, with f=0, the statistical loss of precision resolution is far less than 1 LSB. As a result, the precision of a system implemented with a 16- to 14-bit DAC would have essentially that of the DAC itself. The significance of this is found in design choices allowed by setting $2^n = M/k$.

5. THREE MODULI EXAMPLE

The behavior of the system described in Figure 1, for $P=\{3, 4, 5\}$, can be studied using simulation. The input is assumed to be an integer-valued ramp spanning the range [0, 60), the scale factor k is again chosen to be $k=M/2^5=$

60/32=15/8, and the values of m_i ' and m_i^{-1} are computed as before. For the case where no fractional precision is retained (i.e., f=0 or n'=5), the computed error variance measured at points σ_i^2 of Figure 1, due to the ε -CRT scaling, are found to be:

$$\sigma_1^2 = 0.0740741; \quad \sigma_2^2 = 0; \quad \sigma_3^2 = 0.32$$

The output error, measured at point σ_s^2 of Figure 1 is also found to be the sum of the individual errors, or $\sigma_s^2 =$ 0.394074 (~ -0.7 bits), suggesting that the individual channel errors can be considered to be statistically independent. The system's simulated input-output behavior is graphically interpreted in Figure 2, which compares a ramp input to an ε -CRT's output with full fractional precision (f=5), and zero fractional bit precision (f=0). The full fractional precision (f=5) ε -CRT conversion is seen to be zero-free. Errors are present in the zero fractional bit precision (f=0) cases, but it is well within the predicted statistical error bound.

These factors suggest that the ε -CRT enabled RAC can be made a practical reality, as the next example illustrates. Suppose that a RAC is required that produces an analog output with 12-bit precision fom a 24-bit RNS system defined by the moduli set P={253,255,256}. Choosing k = $M/2^{12} \sim 4032$ will require three lookup tables configured as 2^8x12 , along with a 12-bit binary adder and 12-bit DAC. Such a system can be easily configured using standard commercial off-the-shelf parts or embedded directly into that RAC design.

6. SYSTEM DESIGN ISSUES

In the study of the three-moduli system, defined be $P=\{3, 4, 5\}$, it was observed that the third channel (i.e., mod(5) channel) had the highest error variance (i.e., $\sigma_1^2 = 0.0740741$, $\sigma_2^2 = 0$, $\sigma_3^2 = 0.32$). It can be argued that this is due to that path having the highest error power gain. Interpreting this observation in the context of the standard uniform random error model to represent a quantization process [8], the error variance is modeled to be $\sigma^2 = Q^2/12$, where Q is called the *quantization step-size*. The quantization error variance for the mod(3) path is then predicted to be $\sigma_{3Q}^2 = Q^2/12 = 3^2/12$, for the mod(4) channel, $\sigma_{4Q}^2 = Q^2/12 = 4^2/12$, and for the mod(5) channel, $\sigma_{5Q}^2 = Q^2/12 = 5^2/12$. From the ε -CRT parameters, note that:

 $m_1=20; m_1=round(20/(15/8)=round(10.66) \rightarrow 11$ $m_1=15; m_2=round(15/(15/8)=round(8) \rightarrow 8$ $m_1=12; m_3=round(12/(15/8)=round(6.4) \rightarrow 6$

Therefore, the roundoff error in the mod(3) channel is 0.33, zero for the mod(4) path, and 0.4 for the mod(5) path. The

error power gain for the mod(3) channel is given by $G_3=0.33^2$, $G_4=0.0^2$ for the mod(4) path, and $G_5=0.4^2$ for the mod(5) path. The error variance, defined at points σ_i^2 , in Figure 1, can therefore be modeled as $\sigma_i^2 = G_i * \sigma_{Q_i}^2$ which translates to predicted values of $\sigma_1^2=0.333$, $\sigma_2^2=0.0$, and $\sigma_3^2 = 0.8$. It can be noted that the predicted values are in good agreement with the measured values of 0.32, 0.0, and 0.74. Based on this mathematical model, one can pose the question, what change in architecture shown in Figure 1, would produce the most dramatic improvement RAC statistical performance? The answer is motivated by the observation that the second channel has an error power gain of zero and third channel has a computed error power gain four times larger that of first channel. This suggests that the largest improvement should come from improving the statistical performance of Channel 3. Once the moduli are chosen (fixing the values of σ_0^2), the only Channel 3 parameter that is under user control is the power gain G_5 . Adding just 1-bit of additional fractional precision to this channel would change the rounded value of m_3 ' from 6 to 6.5. This, in turn, reduces G_5 from 0.4^2 to 0.1^2 , for a 16-fold error power gain reduction. That is, the predicted value of $\sigma_3^2 = G_3 * \sigma_{50}^2$ is reduced from 0.8 to 0.02. This conjecture can be tested using simulation. The computed error variance, for Channel 3 (only) having 1 additional bit of fractional precision, was (see Figure 1):

$$\sigma_1^2 = 0.0740741; \quad \sigma_2^2 = 0; \quad \sigma_3^2 = 0.02$$

The output error measured at point σ_s^2 of Figure 1 is the sum of the individual errors or $\sigma_s^2 = 0.0940741$ (~ -1.7 bits) which is in good agreement with the predicted outcome.

7. CONCLUSIONS

A practical residue number system to analog converter (RAC) architecture is presented. Its intended use is to connect systems having a high RNS content to the analog signal domain. The presented architecture is based on the use of the ε -CRT algorithm to convert RNS number to integers using only a few LUT cells and a mod(2ⁿ) adder. Using existing technology, a RAC can be designed having essentially the speed and precision of a modern DAC with only a modest increase in complexity. Statistical error prediction formulas were presented and validated using computer simulation. A technique was also presented that allows for targeted precision improvement that requires only a modest investment of digital hardware.

8. REFERENCES

[1] M., Soderstrand, et. al., *Residue Number System Arithmetic: Modern Applications in Digital Signal Processing*, IEEE Press, 1986 [2] H., Safiri, et. al., "Design of FPGA Implementation of Systolic FIR Filters Using Fermat Number AU," Asilomar Conference, Monterey, 1997.

[3] U., Meyer-Bäse, et. al., "Fast Implementation of Orthogonal Wavelet Filter Banks Using Field Programmable Gate Arrays," IEEE ICASSP Conference, Phoenix, May, 1999.

[4] U., Meyer-Base, et. al., "Implementation of a Communication Channelizer Using FPGAs and RNS Arithmetic," Journal of VLSI Signal Processing, May/June. 2001
[5] J., Ramirez, "High Performance RNS Wavelet Processors Using Custom IC Technologies," Journal of VLSI Signal

Processing, Special Issue on Signal, Image, and Video Technology, #33, 2003.

[6] J., Mellott, et. al., "ASAP A 2D DFT VLSI Processor and Architecture." Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Atlanta, GA: 1996.

[7] M., Griffin, et. al., "New Scaling Algorithms For The Chinese Remainder Theorem." Proceedings of 1989 IEEE ICASSP Conference. 1989.

[8] F., Taylor, F. and J., Mellott, *Hands-On Digital Signal Processing*, McGraw Hill, 1998.



Figure 1: RAC architecture (shown for 3-moduli case)



Figure 2: Performance of an ε -CRT enabled RAC. (top) input, (middle) ε -CRT with f=5, (bottom) ε -CRT with f=0.