RNS Encoding Based Folding ADC

C. H. Vun and A. B. Premkumar
School of Computer Engineering
Nanyang Technological University
Singapore 639798
Email: {aschvun, asannamalai}@ntu.edu.sg

***Abstract*—This paper presents a novel encoding scheme based on the residue number systems for folding ADC to enable highly efficient hardware implementation of high speed, high resolution ADC. The input analog signal is folded by multiple groups of zero-crossing based folding circuits of different folding factors corresponding to the moduli used for their implementation. Each group of folding circuits in turn contains multiple parallel zero-crossing based folding circuits of same folding factor, but with their outputs phase shifted with respect to one another such that their collective output bits pattern form a residue of the modulus of the group. Multiple groups of folding circuits with different moduli that are relatively prime to each other are then combined in parallel. When taken together, their digital outputs form residue digits of a relatively prime moduli set that allows unique representation of the input signal magnitude over a dynamic range that is equal to the product of the moduli used in the different groups in the ADC.**

I. INTRODUCTION

The most common solid-state circuit based high speed Analog to Digital Converter (ADC) in use today is the flash ADC where multiple parallel comparators, equal to the number of quantization levels to resolute, are used to convert an analog input signal to the corresponding digital output. A flash converter of n-bit resolution that provides 2^n dynamic range will have 2^n-1 quantization levels (also known as the least significant bit, LSB) and require a total of 2^n-1 parallel comparators. For instance, an 8-bit flash ADC will hence use 2^8-1=255 comparators. As the number of parallel comparators needed increases exponentially with higher resolution, the skew time among parallel paths used by these comparators becomes a difficult issue to manage in such converters. Furthermore, the overall power dissipation and chip area required also increase tremendously with the number of parallel paths. These factors, among others, impose a practical limit to the resolution that can be achieved in these types of high speed Flash ADCs.

To reduce the number of parallel paths while retaining the high speed of the flash ADC, one alternative that has been proposed is the folding ADC [1]. The folding ADC is similar to a conventional two-step ADC that consists of two parts: a coarse quantizer to output the MSBs (most significant bits) data, and a fine quantizer to digitize the residue signal and output the remaining LSBs (least significant bits). In the case of the folding ADC, the residue signal is obtained directly through a folding circuit, unlike the two-step ADC that obtains it through the output of the coarse quantizer. As such, the folding ADC can operate at the full speed of a flash ADC without the need to wait for the coarse quantizer to first complete its operation. Furthermore, the number of parallel paths in a folding ADC is significantly lower when compared to the flash ADC, and minimum number of parallel paths occurs when its MSBs and LSBs use the same number of bits. For example, an 8-bit folding ADC using 4-bit each for MSBs and LSBs will only require $2\times(2^n\text{-}1)$=30 parallel comparators, compared to the 255 comparators required in the case of flash ADC.

In this paper we propose an encoding scheme that has the advantage of using a smaller number of parallel paths when compared with the folding ADC of similar resolution. In addition, this new encoding scheme uses folding circuits that are similar to those used in folding ADC, and hence retains similar conversion speed characteristics. This is possible because encoding of the signal is based on principles of modular arithmetic used in residue number systems (RNS) where the range of residues required is generally much smaller than conventional number systems. For example, an 8-bit ADC can be adequately implemented using 5+7+8=20 comparators based on [5,7,8] moduli set. The difference becomes even more pronounced when this is extended to higher resolution ADCs. For a 10-bit ADC, the flash ADC will need 1023 comparators, the optimum folding ADC will need $2\times(2^5\text{-}1)$=62 comparators, while the proposed RNS based scheme uses only 5+7+8+9=29 comparators based on the [5,7,8,9] moduli set, but having an better resolution of slightly higher than 11-bits (exact value is 2520).

Furthermore, the modular arithmetic approach also allows the provision to include built-in bit error detection and error correction capability through the use of redundant modulus/moduli. Hence a judicious choice of a moduli set will allow implementation of an ADC of minimum hardware complexity or of better reliability based on system requirements.

It is to be emphasized that the proposed scheme makes use of standard zero-crossing based folding circuits (typically together with interpolation in practical case) that can be implemented using proven and well established circuit techniques. As a corollary, the standard flash ADC is effectively an extreme case of the proposed scheme that is based on a single modulus. Hence, a 10-bit flash ADC effectively uses a single modulus of [1023] and thus requires 1023 parallel comparators. This obviously is not efficient due to choosing a single modulus in the moduli set, as compared to choosing a set of relatively prime moduli that will result in the use of much fewer comparators for a similar resolution.

The remainder of the paper is organized as follows. Section II explains the principle of the proposed technique. Section III describes the circuit arrangement needed to implement this technique, and Section IV discusses its system operation. Section V presents the circuit simulation results of an illustrative example used for the implementation of the proposed scheme. Section VI provides the concluding remarks on the new technique proposed in this paper.

II. Residue Number Systems and ADC

RNS is a number system in which an integer X within the range [0, P) can be uniquely defined by a set of remainders (known as residues) obtained when X is divided by a set of pair wise prime positive integers (called the moduli), with P equal to the product of the moduli. Using a $[m_1,m_2,\ldots,m_M]$ moduli set, an integer $X < P = \prod_{i=1}^{M} m_i$ can hence be uniquely represented by its residues, written as $X \cong \langle x_1,x_2,\ldots,x_M\rangle$.

The main feature of the RNS is that an integer within a large dynamic range can be uniquely represented by a set of residue digits that are of much smaller magnitudes, corresponding to the size of the moduli set used in the representation. For instance, using the [5,7,8] moduli set, the integer X = 178 can be represented by ⟨3,3,2⟩ residue set and the integer Y = 254 by ⟨4,2,6⟩ residue set.

If one were to translate this feature into a hardware implementation with resolution equivalent to an 8-bit flash ADC that normally requires 255 parallel comparators, the RNS based ADC could potentially be implemented with just 4+6+7=17 parallel comparators based on the [5,7,8] moduli set. This would result in tremendous reduction in the ADC circuit complexity if means can be found to implement the RNS conversion directly in hardware circuitry. (It turns out that the RNS based ADC using the proposed scheme would require 5+7+8=20 parallel comparators, which is still a tremendous saving compared to the flash ADC).

Another useful property of RNS is its bit error detection and correction capability when it is used with redundant moduli. This implies that an RNS based ADC can also implement self-contained error detection and error correction features, albeit with the use of extra parallel circuitry to provide the redundant moduli. This leads to a rather profound implication that an error in the conversion process can be detected and corrected by the ADC itself, which further improves the reliability and accuracy of such a converter.

Yet another unique property of RNS is that the number of moduli used can also be reduced, the consequence being having lower resolution and smaller dynamic range. For instance, starting with [5,7,8,9] moduli set, it is possible to remove the modulus 9 thus reducing the original moduli set to become a [5,7,8] moduli set with reduced dynamic range of 5×7×8=280. This feature can be of advantage when implemented in hardware since the circuit corresponding to modulus 9 can be disabled under appropriate conditions so as to reduce power consumption while the device continues to operate at lower resolution.

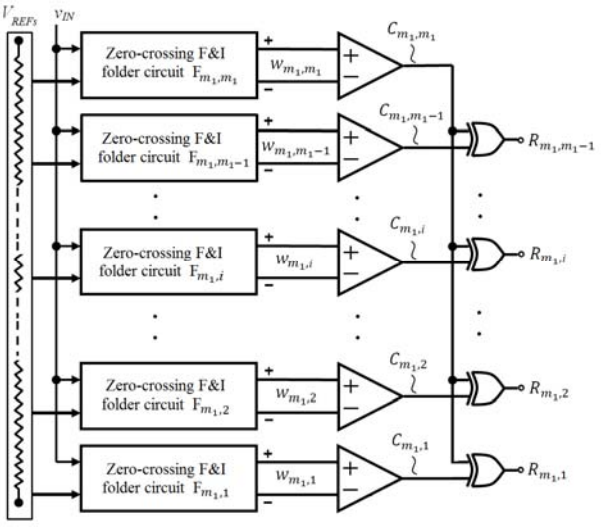


Figure 1. Connection of F&I circuit for RNS output.

III. Circuit Implemenatation

Consider the pattern of the residue digits produced by an integer number corresponding to a modulus. As the integer value increases, its residue digit will repetitively be reset to 0 as it reaches the value of the corresponding modulus, as well as at multiples of the modulus. For example, the residue digit of an integer with modulus m=5 will have the output pattern of the form {0,1,2,3,4,0,1,2,3,4,0,1,2,…} as the integer is increased linearly. Hence the requirement of an RNS based ADC is that its digital output should have the same pattern as the residue digit which resets itself repetitively when the input reaches the level of the modulus, as well as multiple levels of the modulus selected for the implementation.

Fig. 1 shows the connection of a group of m_1 zero-crossing based folding and interpolation (F&I) circuits that can produce the repetitive pattern required of the residue digits corresponding to a modulus m_1. An analog input signal v_{IN} is fed in parallel to the m_1 F&I circuits, where it is compared against a set of reference voltages by the individual folding circuit chosen to produce the folding waveforms with zero-crossings occurring at suitable code transition voltage levels. With F&I circuit, additional folding waveforms are also generated by interpolation within the folding circuits.

The reference voltages for the F&I circuits are typically generated from a reference resistor ladder network. For the RNS based ADC, these reference voltages are chosen in such a way that the F&I circuits' output folding waveforms $w_{m_1,i}$, i = 1,..., m_1, are folded and phase shifted with respect to each other by the appropriate amounts, based on the modulus value and resolution chosen for the group of circuits.

These m_1 folding circuits' output waveforms are analog in nature and typically produced in differential form in practice. They are next applied to m_1 comparators as shown in Fig. 1 to convert them into m_1 single-ended digital bits

$C_{m_1,i}$, i = 1,..., m_1, which are suitable for further digital logic manipulation. The manipulation consists of applying the exclusive OR (XOR) operation to the signals $C_{m_1,1}$ to C_{m_1,m_1-1} bits with the C_{m_1,m_1} bit. This produces the binary bits $R_{m_1,1}$ to R_{m_1,m_1-1} that corresponds to the (m_1-1) bits that form the residue digit for the m_1 modulus chosen for the group of F&I circuits.

While multiple groups of different modulus values are required to produce a complete residue set, only one modulus group is described here as similar circuit arrangements and operation principles can be equally well applied to the other modulus groups of different values. In addition, the conversion operations are performed in parallel for the multiple groups in order to obtain the complete residue set corresponding to the instantaneous v_{IN} value applied to the circuits.

IV. System Operation Principle

This section describes the detailed operation of the circuit shown in Fig. 1. In order to provide a clear understanding of the operating principle, a modulus m_1=5 is used for illustrative purposes in the following discussion.

With reference to the circuit arrangement in Fig. 1 and the waveforms shown in Fig. 2 (which corresponds to the specific case of modulus m_1=5), the number 1 F&I circuit, $F_{m_1,1}$ in a modulus group always has its first zero crossing occurring at 1ΔV in its output waveform $w_{m_1,1}$, where ΔV is the least significant bit size of the ADC device, typically in volts and represents the resolution of the ADC. Its subsequent zero crossings then regularly occur at m_1ΔV intervals apart. The output waveform $w_{m_1,2}$ of the number 2 F&I circuit $F_{m_1,2}$ in a modulus group always has its first zero crossing occurring at 2ΔV, with its subsequent zero crossings similarly occurring regularly at m_1ΔV apart, but phase shifted by 1ΔV when compared to the number 1 waveform. Similar arrangements are required for all subsequent F&I circuits in the modulus group. In essence, all the m_1 F&I circuits within the group generate the same zero-crossing folding waveform shape at a folding interval of m_1ΔV, but are phase shifted with respect to each other by 1ΔV.

Total number of zero crossings generated by each of the F&I circuit will depend on the dynamic range intended for the ADC device. For example, an 8-bit ADC will have the number of zero crossings to be either $(2^8\text{-}1)/m_1$ or $(2^8)/m_1$, depending on the position of the individual F&I circuit and its output waveform relative phase shift within the modulus group. As a group, sufficient zero crossings are required to be generated in order to represent the total number of LSBs required by the ADC device.

Fig. 2 shows the relative phase shift of the differential waveforms $w_{5,1}$ to $w_{5,5}$ for a modulus-5 F&I circuits as the input voltage v_{IN} is linearly increased. Its number 1 waveform $w_{5,1}$ has its first zero crossing occurring at position 1ΔV, with subsequent zero crossings occurring at positions

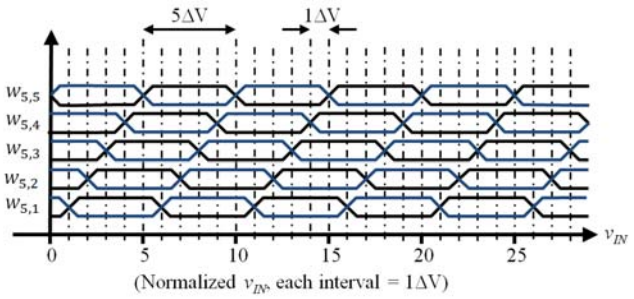


Figure 2. Output waveforms of F&I circuits for modulus = 5.

6ΔV, 11ΔV, 17ΔV etc., extending to the full scale input range required of the ADC. For its number 2 waveform $w_{5,2}$, the zero crossing occurs at positions 2ΔV, 7ΔV, 12ΔV etc. Similarly, waveform $w_{5,3}$ starts with the zero crossing occurring at 3ΔV, waveform $w_{5,4}$ starts with the zero crossing occurring at 4ΔV, and waveform $w_{5,5}$ starts with the zero crossing occurring at 5ΔV, with all their subsequent zero crossings occur at 5ΔV apart, spanning the full range of v_{IN}.

Fig. 3 shows the digital signals $C_{5,1}$ to $C_{5,5}$ generated from the $w_{5,1}$ to $w_{5,5}$ zero crossing folding waveforms by the five comparators. The digital output signals of the comparators collectively display a circular code bit pattern, repeating at an interval of two times the modulus value (equivalent to 10ΔV for m_1=5).

By applying the XOR operation to the lower 4 bits, $C_{5,1}$ to $C_{5,4}$ with the uppermost bit $C_{5,5}$ based on the connection shown in Fig. 1, the resultant digital signals $R_{5,1}$ to $R_{5,4}$ consist of a repetitive bit patterns in the thermometer code format as shown in Fig. 4. These thermometer code encoded residue's (TCR) bits are equivalent to the decimal values changing in the repetitive pattern {0,1,2,3,4,0,1,2,3,4,0…} as the input voltage is linearly increased. This set of values hence corresponds to the residue digits of a modulus 5 as used in the example here.

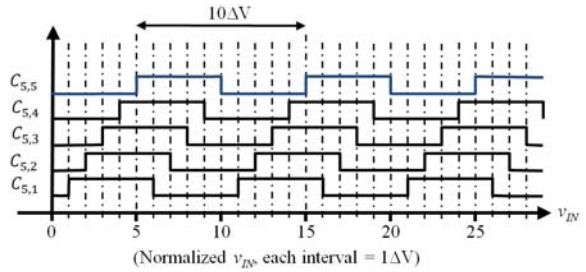


Figure 3. Comparators output digital signals for modulus = 5.

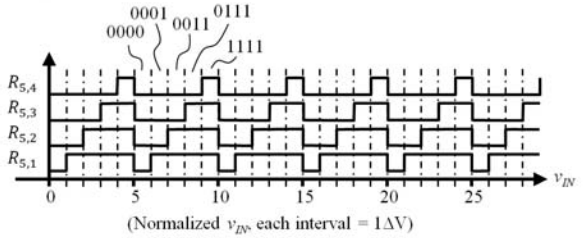


Figure 4. RNS encoded digital signals for modulus = 5.

Additional residue digits for other moduli values can be similarly generated based on the same approach to produce a complete residue set corresponding to the chosen moduli set. Their combined output forms a RNS based code which can then be applied to a decoder circuit to identify the input v_{IN}. The decoding circuit could be a logic based device capable of interpreting the combined RNS codes to derive the magnitude of v_{IN}, or a Read Only Memory (ROM) device containing the truth table relating the RNS codes to the magnitude of v_{IN}. In the case of ROM based decoder, an alternative arrangement [2] for the XOR logic gates can be used to produce a one-hot (i.e. 1-out-of-n) encoded residue (OHR) format from the comparator output's circular code, which would be preferred as it is simpler to be used for the purpose of decoding look-up table. Another possibility is to directly perform the signal processing algorithms using the output data in the OHR format or TCR format based on modular arithmetic operations, which has certain advantages as explained in [3].

V. Circuit Simulation Results

Operation of a modulus-5 RNS based F&I ADC circuit is simulated using LTspice SPICE simulator. A 2-stage cascaded zero-crossing F&I based architecture [1][4] is used to construct the RNS based ADC to verify the correctness of the theoretical operations described earlier.

Fig. 5 shows the block diagram of the 2-stage F&I ADC used in the simulation, with the ADC's LSB value chosen to be ΔV=50mV. The first stage consists of six parallel folding circuits (folders) with folding degree of 5 and folding period of 15ΔV, phase shifted from each other by 5ΔV. Four additional folding waveforms are then generated through interpolation between each pair of the folding circuits, filling in between the two folding circuits' output waveforms at 1ΔV interval. These waveforms are then applied to the second stage circuits that consist of five ×3 folding circuits. This ×3 folding increases the folding rate by 3 times such that the first stage's 15ΔV folding period is shortened to 5ΔV, corresponding to the modulus 5 used for the RNS ADC. The second stage's five output waveforms are then applied to five comparators to convert them into the single-ended circular code digital signals. These are then encoded by the four XOR logic gates to produce the residue digit in TCR encoded bits format corresponding to the modulus 5 of the group.

Fig. 6 shows the single-ended version of selected representative output folding waveforms of the first stage folders, including the four interpolated folding waveforms derived from the two outer folding waveforms shown. Fig. 7

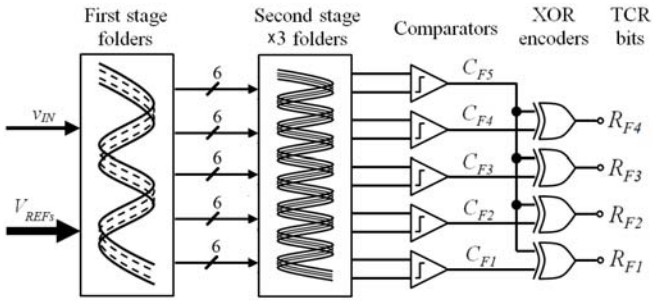


Figure 5. 2-stage cascaded F&I based RNS ADC.

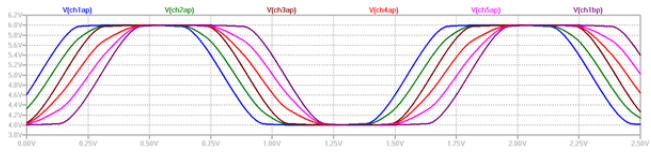


Figure 6. First stage folding and interpolated waveforms.

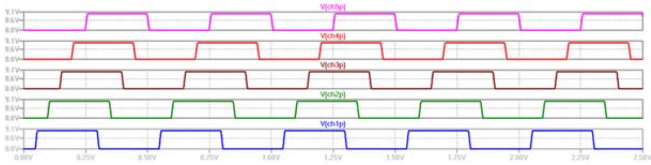
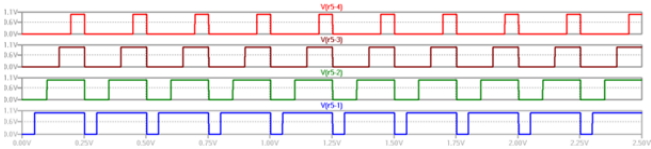
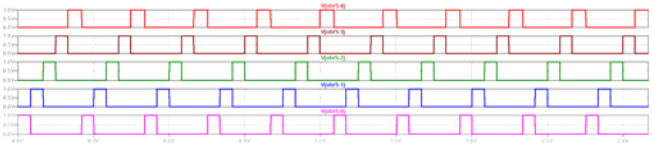


Figure7. Second stgae ×3 cascaded folding waveform.



(a) TCR format



(b) OHR format

Figure 8. Digital output signals from XOR based encoders.

shows the outputs of the second stage folders with ×3 folding to produce the waveforms with the appropriate folding period and phase shift patterns described earlier, corresponds to pattern of Fig. 3. Fig. 8(a) shows the TCR digital signals output by the XOR encoding circuit which matches the required TCR patterns shown in Fig. 4. For completeness, Fig. 8(b) shows the OHR bit patterns output by the same ADC through a slight change in the XOR encoder arrangements [2].

VI. Conclusion

This paper presents a novel RNS based encoding scheme to minimize the hardware complexity required to implement high resolution high speed folding ADCs. This is achieved by applying RNS principles to the hardware embodiment of the folding circuits, and can be realized using well-established folding ADC circuit implementation techniques. Using a conventional 2-stage cascaded F&I circuits architecture, the simulated results obtained validate the correctness and confirm the practical feasibility of the proposed scheme.

References

[1] M. J. Choe, B. S. Song and K. Bacrania, "An 8-b 100-MSample/s CMOS pipelined folding ADC", IEEE Journal of Solid-State Circuits, vol. 36 no. 2, pp. 184-194, Feb 2001.

[2] C. H. Vun and A. B. Premkumar, "Residue number system based encoding circuits for analog to digital conversion", provisional patent application filed with USPTO, 30 June 2011.

[3] C. H. Vun and A. B. Premkumar, "Method and apparatus for thermometer code based modular arithmetic operations", provisional patent application filed with USPTO, 18 Sept 2011.

[4] P. M. Figueiredo and J. C. Vital, "High-speed ADC architectures", Chapter 1, Offset Reduction Techniquess in High-Speed Analog-to Digital Converters, Springer Science+Business Media B.V. 2009.