ABSTRACT

Wireless Sensor Networks (WSNs) subsist on network of huge numbers of hubs that are distributed in unfriendly environments used for habitat monitoring and to observe changes in phenomena. These hubs are susceptible to faults such as energy exhaustion, hardware glitches, communication link errors, malicious attacks, which may lead to errors in message delivery. However, these systems are typically utilized to convey vital information in remote places. Thus, efficient error identification and correction is thereby required in order to maintain the accuracy of the received message. This paper aimed at designing high speed reverse converter for effective error control codes using Redundant Residue Number System in order to enhance the accuracy of messages delivered for real-time WSNs. In order to attain a greater error correction, a reverse converter which was based on Mixed Radix Conversion was designed for the moduli set \(\{2^{n+1}-1, 2^n, 2^n-1\}\). In addition to this moduli, an extra...
redundant moduli set\(\{2^{n+1}+3, 2^{2n} - 3\}\) is also designed for higher correctness. For correction and identification of corrupted values in the original data, maximum likelihood was used. The results obtained show that the suggested moduli set offers low energy utilization and consistency of the received message in real time WSNs.

**Keywords** - Wireless Sensor Network, Error Control, Reliability, Redundant Residue Number System, Moduli Set, Real-time

**Introduction**

Wireless Sensor Networks (WSNs) comprise of network of enormous number of randomly arrayed minute devices called nodes. These networks have inbuilt storage with computational and communication capabilities. They function in an unattended mode and collect required data from its coverage area [1, 2, 3].

WSNs are capable to sense data and then forward the collected data and process the data to generate suitable reactions depending on the nature of received data. It consists of two node. One is sensor nodes that sense the data and the other is sink node where information is transferred. The sensor nodes commonly are cheap. Their energy supply is restricted. There transmission range is also low. These transmission ranges are function for event detection or sensing data from the environmental. Conversely, the sink nodes are resource-richer nodes with abundant energy sources, higher communication and computation capability, and the ability to perform powerful reactions [4].

A sensor hub is a gadget which has three parts: first is a processing unit. It is utilized to information processing from the neighborhood, second part is a sensing subsystem. This system is used for obtaining information from its surroundings. Third part is a wireless communication transceiver. This part exchanges the detected data to the sink. These sensor hubs perform information detecting and preparing errands and furthermore communicate with each other [5]. In order to maintain the message reliability, it is obligatory for WSN to be capable of detecting the errors and take appropriate measures to correct them [2, 6,]. However, Residue Number System (RNS) has been utilized as a tool to reduce communication energy and increase integrity in WSNs [7]. A few more studies observed that for numerous reasons, the Redundant Residue Number Systems (RRNSs) are appropriate for utilization in real-time WSNs’ applications. They improve the real time operations and are fault tolerant. The capability to control error control is also improved. It also provides security and energy saving. [8, 9].

However, it has been established that energy, speed and reliability are the major challenges affecting many WSNs applications. This is because WSNs are furnished with limited energy and are used to collect vital information from the environment. Thus, WSNs need to be energy efficient, fast and reliable [2, 10]. A substantial efforts have been made by researches been with a focus on the development of techniques for high speed and reliability in WSNs. It was observed that none of the several researches has offered extensive reliability support for WSNs. Hence, this article presents an efficient energy utilization reverse converter with error detection and error correction that make use of RRNS to enhance the reliability of a received
message in real-time WSNs. The scheme employed the moduli set \{2^{n+1}-1, 2^n, 2^{n-1}\} to design the Mixed Radix Conversion (MRC)-based reverse converter and Chinese Remainder Theorem (CRT)-based error-control scheme with supplementary redundant moduli set \{2^{n+1}+3, 2^{2n} - 3\} so as to attain an enhanced error controllability. The results obtained indicate that the recommended scheme delivers low energy consumption and consistency of the received message in real-time WSNs.

1. Literature Review

Security and reliable information communications have been increasingly critical during the last decades [11].

It is vital to for a functioning of a sensor network to choose a perfect fault correcting mechanism, taking in consideration the vitality and execution [12].

It is necessary to find a novel solution with little vitality utilization and at the same time, provide proper reliability in real-time WSNs’ applications. Hamamreh et al (2018) presented a new energy-aware algorithm. This algorithm is known as Minimum Residual Hop Capacity (MRHC) [3]. This algorithm is incorporated into the frequently adopted protocol commonly identified as Low Energy Adaptive Cluster Hierarchical (LEACH) within the broadcast procedure in the clusters. This reduces energy utilization in the transmission process, extends network durability and improves the amount of data delivered to the base station. Usually, nodes communicate within cluster through multi-hop communications and data is sent via other nodes in the cluster till it reaches the CH. This result in few duplicates of a similar packet could get to the CH, hence there is need for data aggregation which usually result in communication suspension between the CH and the sink [3]. In addition, other studies also reported that the simplicity in the design of sensor hardware and limited utilization of power necessitate prevention of deployment of high complexity codes along with energy efficiency and error control [6, 13, 7]. Moreover, WSNs are naturally unprotected to numerous sources of unpredictability such as errors generated by hardware noise, transmission errors and errors in hubs, hence, these require the need for reliability measures [6, 3]. Similarly, when a message is corrupted, not just the errors ought to be recognised, but they also need to be corrected so that the message can return to its original and useful form [14, 15].

An upgraded moduli-set \{2^{n} - 1, 2^{n+1} + 1, 2^{2n} - 1\} reverse converter was proposed by Habibi and Salehnamadi 2016 which was based on CRT algorithm. The recommended moduli set improveD complexity of circuit, energy consumption but there is need to reduce hardware requirements, improve the speed and is applicable to WSNs [16]. Additionally, Bankas and Gbolagade (2014) recommend a noveleffective reverse converter for the 4-moduli set \{2^{n}, 2^{n+1}, 2^{n} - 1, 2^{2n+1} - 1\} which was built on an improved CRT and MRC. The area cost of reverse converter is \(tF4 + tMUX\). The converter is found to be faster, to the detriment of slightly more hardware assets, which surpass the state of the art reverse converter. Barati et al, 2014 present a new scheme with improved vitality and rapid control of error that relies on the RRNS in real-time WSNs. The theoretical outcome implemented by simulation outperforms renowned error control methods for WSNs with respect to vitality productivity, error controllability and reduction of end-to-end delay though with a little high hardware requirement [14]. Therefore, it is
indispensable to provide an efficient, rapid operation and reliable error correction scheme for real-time WSN while still preserving the little vitality of the network.

2. Residue Number System

Residue Number System (RNS) is characterized as a non-weighted numeral system that was introduced by Garner, in 1959. It delivers a high speed, carry-free, parallel secure arithmetic operation which is fault-tolerant [14, 17]. In RNS, instead of represented by the original number itself, the remainders of the original number with reference to moduli set is commonly used for representation. Hence, the number will be fragmented into some smaller independent numbers. Thus, operations can be accomplished on them separately. This makes the computations much simpler and swift [16, 18]. However, RNS are very useful in detection and correction of errors because no other digit is corrupted by error in one digit. Furthermore, RNS are founded on the congruence relation, which is described in terms of relatively prime moduli set \( \{m_1, m_2, \ldots, m_n\} \) that is the greatest common divisor \( \gcd(m_i, m_j) = 1 \) for \( i \neq j \). Therefore, a weighted number \( X \) can be given as follows:

\[
X = (x_1, x_2, \ldots, x_n), \text{ where:}
\]

\[
x_i = x \mod m_i = \left\lfloor x \right\rfloor p_i, \quad 0 \leq x_i < p_i
\]

This depiction is distinctive for any integer \( X \) which is in the range \([0, M)\) when \( M = (m_1 m_2 \ldots m_n) \) is referred to as dynamic range of the moduli set \( \{m_1, m_2, \ldots, m_n\} \). Conversely, RRNS is a kind of RNS which detects and corrects the errors, due to adding redundant modules to the original modules. RRNS is indicated by moduli set \( \{m_1, m_2, m_3, \ldots, m_n, m_{n+1}, m_{n+2}, \ldots, m_{n+r}\} \) and \( m_i > m_{i-1} \) for \( i = 2, 3, \ldots, n+r \). Given that all the moduli set are pairwise prime, the Dynamic Range (DR) of the system is given by

\[
[0, M = \prod_{i=1}^{n+r} m_i)
\]

Finally, the traditional CRT is characterized per given formula as: for a moduli set \( \{m_1, m_2, m_3, \ldots, m_k\} \) having the dynamic range \( M = \prod_{i=1}^{k} m_i \), the residue number \( (x_1, x_2, x_3, \ldots, x_k) \) can be altered to the decimal number \( X \), using:

\[
X = \left\lfloor \sum_{i=1}^{k} M_i \left| M_i^{-1} x_i \right|_{m_i} \right\rfloor
\]

Given that when \( M = \prod_{i=1}^{k} m_i \), \( M_i = \frac{M}{m_i} \), and \( M_i^{-1} \) is representing the multiplicative inverse of \( M_i \) in reference to \( m_i \).

3. Methodology
In order to realize the proposed scheme, the moduli set \( \{2^{n+1}-1, 2^n, 2^{n-1}\} \) is applied to design MRC-based reverse converter and CRT-based error control scheme utilizing supplementary redundant moduli set \( \{2^{n+1}+3, 2^{2n} - 3\} \). As shown in Figure 1, a node will send a residue of the sensed message to the cluster head (CH), which will forward the gathered residues in its cluster to the sink node which will convert the received residues to binary equivalent. Then the sink will check for errors, locate, correct the errors, and then recover the sink from error to have error free message.

3.1 Design of the High Speed Reverse Converter

For the purpose of designing reverse converter that will convert the received residues to its original message, it is required that the moduli set \( \{2^{n+1}-1, 2^n, 2^{n-1}\} \) be somewhat prime numbers. This can be done using Euclidean algorithm:

\[
gcd (a, b) = gcd (b, \frac{|a|}{b}), \quad a > b
\]

Where the \( \gcd \) represents greatest common divisor, consequently:

\[
gcd (2^{n+1}-1, 2^n) = gcd (2^n, \frac{2^{n+1}-1}{2^n}) = gcd (2^n, 1) = 1
\]

\[
gcd (2^{n+1}-1, 2^{n-1}) = gcd (2^n-1, \frac{2^{n+1}-1}{2^n-1}) = gcd (2^n-1, 2) = 1
\]

Figure 1. Proposed Error Detection and Correction Process
gcd \(2^n, 2^n - 1\) = gcd \(2^n - 1, 2^n\) = gcd \(2^n - 1, 1\) = 1

Thus, all the greatest common divisors (gcd) are equal to 1, this indicates that the moduli set \\{2^n+1-1, 2^n, 2^n-1\} are somewhat prime. Then, let us calculate the multiplicative inverse based on MRC algorithm and then in the conversion algorithm, substitute those values with the modulus set.

First, the multiplicative inverse of \(2^n+1-1\) with respect to \(2^n\) is \(K_1\).

\[
\left[ m_1^{-1} \right]_{m_2} = \left[ (2^n+1-1)^{-1} \right]_{2^n} = \left[ (-1 \times 2^n+1-1) \right]_{2^n} = 1
\]

Therefore, \(K_1 = -1\)

Also, multiplicative inverse of \(2^n+1-1\) analogous to \(2^n-1\) is \(K_2 = 1\)

\[
\left[ m_2^{-1} \right]_{m_4} = \left[ (2^n+1-1)^{-1} \right]_{2^n-1} = \left[ (1 \times 2^n+1-1) \right]_{2^n-1} = 1
\]

Therefore, \(K_2 = 1\)

And lastly, the multiplicative inverse of \(2^n\) analogous to \(2^n-1\) is \(K_3\).

\[
\left[ m_3^{-1} \right]_{m_6} = \left[ (2^n)^{-1} \right]_{2^n-1} = \left[ (1, 2^n) \right]_{2^n-1} = 1
\]

Therefore, \(K_3 = 1\)

MRC is generally given by:

\[
X = a_1 + a_2 m_1 + a_3 m_2 m_3 
\]

(1)

where

\[
a_1 = x_1 
\]

(2)

\[
a_2 = \left[ (x_2 - a_1) m_1^{-1} \right]_{m_2} \left[ m_2^{-1} \right]_{m_4}
\]

(3)

And

\[
a_3 = \left[ (x_2 - a_1) m_1^{-1} \right]_{m_2} \left[ m_2^{-1} \right]_{m_4} - a_2 
\]

(4)

Next is to substitute values for \(a_2\) in equation (3) and simplify to decrease the hardware requirements:

\[
a_2 = \left[ (x_2 - a_1) m_1^{-1} \right]_{m_2} \left[ m_2^{-1} \right]_{m_4} = -x_2 \left[ x_2 + a_1 \right]_{2^n} 
\]

\[
= -x_2 \left[ x_2 + 1 \right]_{2^n} = a_{21} + a_{22}
\]

Hence, the binary representation of the residues is:
\[
a_{21} = \overline{x_2}_{2^{n-1}} = \overline{x_{2,n-1}} \overline{x_{2,n-2}} \ldots \overline{x_{2,1}} \overline{x_{2,0}} \tag{5}
\]

\[
a_{22} = x_1_{2^{n-1}} = x_{1,n-1} x_{1,n-2} \ldots x_{1,1} x_{1,0} \tag{6}
\]

To implement \(a_3\) by simplifying equation (4) as follows:

\[
a_3 = \left[ (x_3 - a_1) - a_2 \right]_{\text{bit}} = \left[ (x_3 - x_i) - a_2 \right]_{2^{n-1}}
\]

\[
= \overline{x_3}_{2^{n-1}} + \overline{x_i}_{2^{n-1}} - a_2_{2^{n-1}} = a_{31} + a_{32} + a_{33},
\]

where the binary representation of the residues is:

\[
a_{31} = \overline{x_1}_{2^{n-1}} = x_{3,n-1} x_{3,n-2} \ldots x_{3,1} x_{3,0} \tag{7}
\]

\[
a_{32} = \overline{x_1}_{2^{n-1}} = \overline{x_{3,n-1}} \overline{x_{3,n-2}} \ldots \overline{x_{3,1}} \overline{x_{3,0}} \tag{8}
\]

\[
a_{33} = \overline{a_2}_{2^{n-1}} = a_{2,n-1} a_{2,n-2} \ldots a_{2,1} a_{2,0} \tag{9}
\]

To obtain \(X\) based on the equation (1),

\[
X = a_1 + a_2 m_1 + a_3 m_2 = x_i + m_i (a_2 + a_3 m_2)
\]

\[
= x_i + (2^{n+1} - 1)(a_2 + (2^n)a_3) = x_i + (2^{n+1} - 1)C \text{ where } C = a_2 + (2^n)a_3
\]

\[
\begin{aligned}
C &= a_{3,n-1} \ldots a_{3,0} a_{2,n-1} a_{2,n-2} \ldots a_{2,1} a_{2,0} \\
&\quad + \overline{1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots a_{2,n-1} \ldots a_{2,0}} \tag{10}
\end{aligned}
\]

\[
X = x_i + (2^{n+1} - 1)C = \text{Concatenation of } (x_i, C)
\]

\[
\begin{aligned}
X &= C_{3n-1} \ldots C_0 C_{n-1} \ldots C_0 C_{n+1} \ldots C_0 + \\
&\quad \overline{1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots 1 \ldots 1} \times x_{1,n+1} \ldots x_{1,0} \tag{11}
\end{aligned}
\]

Therefore, the implementation of \(X\) based on equation (10) and (11) is given in Figure 2:
In order to implement $X$, consider equations 10 and 11, it needs $(2n)$ bits and $(3n+2)$ bits Regular Carry Propagate Adder (CPA). There is also need for Operand Preparation Unit-2 (OPU-2) and OPU (3) for preparing the 1's complements respectively. However, $(n)$ bits and $(n+1)$ bits are 1 in equations 10 and 11 respectively. Therefore $(n)$ and $(n+1)$ Full Adders (FAs) can be substituted with $(n)$ and $(n+1)$ pairs of OR/XNOR gates respectively.

3.2 RRNS Error Control Scheme

When the base station received the collected residues from the CH, it reconstructs the original information using the information moduli as well as their residues. The remainders of the calculated number is divided by redundant moduli. The remainders collected for those redundant moduli are unequal, an error is discovered, and which will initiate a process for error correction. Then, three residues out of five can be utilized by the base station to restructure the original data. Thus, RRNS codes which has ann redundant modulus can identify $n$ errors and correct $n-1$ given that the condition $m_1 < m_2 < ... < m_k < m_{k+1}$ is met. However, for the purposes of error detection and correction, the complete moduli set will now contain $\{2^{n+1}-1, 2^n, 2^n-1, 2^{n+1}+3, 2^{2n}-3\}$ where $\{2^{n+1}-1, 2^n, 2^n-1\}$ are information moduli while $\{2^{n+1}+3, 2^{2n}-3\}$ are
redundant moduli. The actual number can be represented by just the residues obtained using the main moduli. The legitimate range is defined by the product of the information moduli whereas the illegitimate range is characterized by product of the entire moduli set.

However, error can be detected if \( a_j \neq 0 \) with \( j \in [k+1, n] \). Assuming a single error corrupts the \( i^{th} \) residue digit which appeared in the residue of the received integer \( X \), this is signified as an adding of a number \( e_i \) to the residue when \( 0 < e_i < m_i \), that is:

\[
X' \rightarrow [r_1, r_2, r_3 ... (r_j + e_j) ... , r_{n-1}, r_n]
\]

Where altered number is represented by \( X' \). Similarly, \( X' \) can be described as the sum of the altered number \( X \) and an integer \( E \) created by the residue \( e_i \),

\[
X' \rightarrow X + [0, 0...e_j,...,0,0].
\]

\[
[0,0...e_j,...,0,0] \equiv E(\text{mod } m_j).
\]

Therefore,

\[
E = \sum_{j=k+1}^{n} a_j \prod_{i=1}^{j-1} m_i \text{ such that}
\]

\[
X' = X + \sum_{j=k+1}^{n} a_j \prod_{i=1}^{j-1} m_i
\]

In view of the fact that the redundant modulus \( m_{k+1} \) is greater as compared to every information moduli, and \( E \) is greater than the legitimate range \( M \). The only case decoding the integer is \( a_j = 0 \) which shows up in the legitimate range, where \( j \in [k+1, n] \). Accordingly, \( a_j \neq 0 \) specifies the residue digit has been corrupted, else no error occurs.

For illustration purpose, let us consider \( n = 2 \) for moduli set \( \{2^{n+1}-1, 2^n, 2^{n-1}, 2^{n+1}+3, 2^{2n} - 3\} \). The moduli set will now be \( \{7, 4, 3, 11, 13\} \) where \( \{7, 4, 3\} \) is the information moduli set and \( \{11, 13\} \) is the redundant moduli set. The legitimate Range called Dynamic Range (DR) is 84 and the illegitimate range is 12012.

For example, if the gathered message \( X = 52 \) is to be conveyed from CH to the sink using the moduli set above. The number would then be represented as the residue set \( (3, 0, 1, 8, 0) \) where \( (3, 0, 1) \) is the non-redundant moduli set and \( (8, 0) \) is the redundant moduli set. This residue representation is all needed to manage transmitted data in WSN communication. Accordingly, in order to check if the transmitted data was received correctly by the recipient node in real time WSN, a reverse conversion of the residue is done using CRT to reconstruct the original number from the residue set, using:

\[
X = \left[ \sum_{j=1}^{k} M_j \right] \left[ M_i^{-1} X_j \right]_{m_i} \frac{1}{M_i}
\]
where

\[ M_i = \frac{M}{m_i} = \prod_{i=1}^{N} \frac{m_i}{m_i}. \]

Moreover, in case one of the residues is corrupted by an electrical fault or noise, the proposed scheme has the capacity to identify the error and correct it. For illustration purpose, assuming the residue subset of \( \{3, 0, 1\} \) has been corrupted into the values \( \{3, 3, 1\} \) (i.e. \( x_2 \) has been altered from the right value of 0 to the incorrect value of 3). The conversion process then gives:

\[ X = \left[ 12 \cdot 3 + 21 \cdot 3 + 28 \cdot 3 \right]_{84} \]

\[ X = [15]_{84}, \text{ therefore } X = 31 \]

In order to correct the error, the maximum likelihood principle is utilized and obtained the original data. To achieve this, extra combinations of information and redundant residues are created. Each combination can then be changed utilizing the CRT. In this case, the value within the legitimate range that appears the most often will be the correct value of the original data as indicated in Table 1 in blue background.

Consider the previous instance where residue \( x_2 \) for the decimal number 52 of \( X = (3, 0, 1, 8, 0) \) was corrupted into the incorrect value from 0 to 3, now with the residue digits of \( X = (3, 3, 1, 8, 0) \) corresponding to it. With CRT's approach, the integer \( X \) in the dynamic range \([0, 84)\) can be recovered by raising any of the three moduli and their equivalent residue numbers given that no errors happened in the received RNS representation. Then, reflect all likely cases and aim at recovering the number \( X \) represented by \( (3, 3, 1, 8, 0) \) from moduli set \( (7, 4, 11, 13) \).

The additional residue mixture can be formed from likely combinations of three residue digits out of five residue digits as shown in Table 1:

<table>
<thead>
<tr>
<th>Residues</th>
<th>Residues Values</th>
<th>Moduli Set</th>
<th>( X_{ijk} )</th>
<th>( X )</th>
<th>DR</th>
</tr>
</thead>
<tbody>
<tr>
<td>{x_1, x_2, x_3}</td>
<td>(3, 3, 1)</td>
<td>{7, 4, 3}</td>
<td>( X_{123} )</td>
<td>31</td>
<td>84</td>
</tr>
<tr>
<td>{x_1, x_2, x_4}</td>
<td>(3, 3, 8)</td>
<td>{7, 4, 11}</td>
<td>( X_{124} )</td>
<td>283</td>
<td>308</td>
</tr>
<tr>
<td>{x_1, x_2, x_5}</td>
<td>(3, 3, 0)</td>
<td>{7, 4, 13}</td>
<td>( X_{125} )</td>
<td>143</td>
<td>364</td>
</tr>
<tr>
<td>{x_1, x_3, x_4}</td>
<td>(3, 1, 8)</td>
<td>{7, 3, 11}</td>
<td>( X_{134} )</td>
<td>52</td>
<td>231</td>
</tr>
<tr>
<td>{x_1, x_3, x_5}</td>
<td>(3, 1, 0)</td>
<td>{7, 3, 13}</td>
<td>( X_{135} )</td>
<td>52</td>
<td>273</td>
</tr>
<tr>
<td>${x_1, x_4, x_5}$</td>
<td>(3, 8, 0)</td>
<td>${7, 11, 13}$</td>
<td>$X_{145}$</td>
<td>52</td>
<td>1001</td>
</tr>
<tr>
<td>${x_2, x_3, x_4}$</td>
<td>(3, 1, 8)</td>
<td>${4, 3, 11}$</td>
<td>$X_{234}$</td>
<td>19</td>
<td>132</td>
</tr>
<tr>
<td>${x_2, x_3, x_5}$</td>
<td>(3, 1, 0)</td>
<td>${4, 3, 13}$</td>
<td>$X_{235}$</td>
<td>91</td>
<td>156</td>
</tr>
<tr>
<td>${x_2, x_4, x_5}$</td>
<td>(3, 8, 0)</td>
<td>${4, 11, 13}$</td>
<td>$X_{245}$</td>
<td>195</td>
<td>572</td>
</tr>
<tr>
<td>${x_3, x_4, x_5}$</td>
<td>(1, 8, 0)</td>
<td>${3, 11, 13}$</td>
<td>$X_{345}$</td>
<td>52</td>
<td>429</td>
</tr>
</tbody>
</table>
4. Performance Evaluation and Discussion of Results

The reverse converter was compared with the existing advance reverse converters. The results of theoretical analysis depicts that the suggested error correction scheme is efficient with regard to area and delay when compare with Barati et al, 2014 [14] and better in terms of delay as compared to Habibi & Salehnamadi, 2016 [16] as shown in Table no. 2.

<table>
<thead>
<tr>
<th>Reverse Converter</th>
<th>Moduli Set</th>
<th>Area ($A_{FA}$)</th>
<th>Delay ($t_{FA}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>[14]</td>
<td>${2^{2n+1}, 2^{2n+1}-1, 2^n-1}$</td>
<td>7n+3</td>
<td>9n+8</td>
</tr>
<tr>
<td>[16]</td>
<td>${2^n-1, 2^n+1, 2^{2n}-1}$</td>
<td>5n+1</td>
<td>10n + 6</td>
</tr>
<tr>
<td>[19]</td>
<td>${2n, 2n + 1, 2n- 1, 2n+1-1}$</td>
<td>12n+2</td>
<td>9n+6</td>
</tr>
<tr>
<td>Proposed</td>
<td>${2^{n+1}-1, 2^n, 2^n-1}$</td>
<td>6n +2</td>
<td>9n +2</td>
</tr>
</tbody>
</table>

Table 2: Comparison of Area and Delay

Additionally, the proposed error-control scheme is effective in terms of speed and energy utilization in comparison with the study of Roshanzadeh and Saqaeeyan [6].

The suggested converter has an area Full Adders of (6n +2)$A_{FA}$ with low cost hardware requirements and power usage and also a delay Full Adder of (9n +2)$t_{FA}$ that provides relatively low conversion delay and high performance for the WSNs in terms of energy usage and improved reliability when compared with other reverse converters. Similarly, conversion using the CRT that yields different values of X was displayed in table 1. $X_{ijk}$ signifies the yielded result by applying moduli $m_i$, $m_j$ and $m_k$ and their analogous residue digits’ $x_i$, $x_j$ and $x_k$. From the table 1, $X_{124}$, $X_{125}$, $X_{235}$ and $X_{245}$ all represents the illegitimate numbers, because the numbers are out of the legitimate range [0, 84). Of the six cases which were remaining, four cases ($X_{134}$, $X_{135}$, $X_{145}$, $X_{345}$) all the results are the same and equal to 52. For this case, the value within the legitimate range that appears the most often will be the correct value of the original data. As 52 appears the greatest number of times it is known to be the accurate original data. Furthermore, these results were obtained from three moduli excluding $x_2$. Consequently, it might be concluded that the accurate result is 52 or there might be an error in $x_2$, which can be amended by calculating:

$$x_2 = 52 \pmod{4} = 0.$$  

5. Conclusion

An effective error control scheme is recommended for real time WSNs having $\{2^{n+1}-1, 2^n, 2^n-1\}$ as a a moduli set and with $\{2^{n+1}+3, 2^n - 3\}$ as for error detection and correction. MRC-based reverse converter was designed in order to convert the received residues to its original message. The results obtained outperform existing advance reverse converter in terms of speed, hardware requirements and energy consumption. Maximum likelihood which involves the value within the legitimate
range that appears the frequently was used to correct the corrupted values to the original data. The proposed moduli set outperforms existing state of the art error-control schemes in real time WSNs.

Conflict of Interest
The authors declare no conflict of interest

References


8. Abdul-Mumin S, Gbolagade KA. An Improved Redundant Residue Number System Based Error Detection and Correction Scheme for the Moduli Set \{2 2n+ 1, 2 n+ 1+ 1, 2 n+ 1–1, 2 n+ 1, 2 n\}. Advances in Wireless Communications and Networks. 2016 Dec 5;2(1):11.


