A Reliable & Energy-Efficient Scheme for Real Time Wireless Sensor Networks Applications

Ali Barati¹, Ali Movaghar², Samira Modiri³, Masoud Sabaei⁴

¹Ph.D Student, Department of Computer Engineering and Information Technology, Qazvin Branch, Islamic Azad University, Qazvin, Iran,
²Department of Computer Engineering, Sharif University of Technology, Tehran, Iran
³Department of Computer Engineering, Dezful University, Dezful, Iran
⁴Computer and IT Department, Amir-Kabir University of Technology, Tehran, Iran

ABSTRACT

Reliability and energy efficiency are the main issues in real time wireless sensor networks applications. In this paper, a new real-time, error control, low energy consumption scheme using redundant residue number system is proposed to enhance the wireless sensor network's lifetime and increase reliability in data delivery. New moduli set \(\{2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^{2n} - 1, 2^{2n+1} - 1, 2^{2n} + 1\}\) is proposed that \(\{2^{2n+1} - 1, 2^{2n} + 1\}\) are redundant modulus. Efficient reverse converter for the main modulus \(\{2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1\}\) is presented using mixed radix conversion. Using simulation of error control algorithm for the proposed moduli set using C++ programming language by setting \(n = 2\), error control capability for 145692 different error bits states is evaluated. Results show that the proposed scheme has considerable excellence in terms of reduction of end to end delay, error control and energy efficiency comparison with the other existing work.

KEYWORDS: Redundant Residue Number Systems, Reverse Converter, Mixed Radix Conversion, Wireless Sensor Networks

1. INTRODUCTION

Recent advancement in wireless communications, Micro Electro Mechanical System (MEMS), and also a tendency to use low cost, tiny, and autonomous high performance products have led to the emergence of wireless sensor networks (WSNs) [1]. The main task of wireless sensor networks is to collect the sensed data from environment and send them back to the sink node to be processed [2]. Because of the dynamic, lossy and low-power nature of sensor networks, important factors in all of these types are reducing energy consumption and reliability [3]. Moreover, some applications of wireless sensor networks need to real time operations such as forest fire detection. Thus end-to-end delay of operations in such as wireless networks is important too.

There are some works considered one or more critical issues in wireless sensor networks, such as delay, error control and energy efficiency. In [4, 5, 6, 7], energy and delay constraints were studied, without reliability consideration in data delivery. RAG [4] is a structure-free real time protocol for data aggregation that has advantages in terms of energy consumption, miss ratio, and aggregation gain. Energy-latency trade offs was explored in [5] using rate adaption techniques. A simple, distributed online protocol relay on the local information available at each sensor node proposed in this work. Authors in [6] presented an energy efficient algorithm for real time wireless sensor networks applications. Using this algorithm, a data aggregation tree under energy and latency constraints can construct. Dynamic time out for data aggregation in wireless sensor networks studied in [7]. In this work, a novel scheme proposed for time out control that can provide efficient trade off between energy consumption and latency in wireless sensor networks. As mentioned before, none of the aforesaid works has not considered the reliability problem in WSNs. In [8], error control for real time industrial wireless sensor networks considered, because in such as wireless networks, reliability is a very important issue. Different FEC codes were compared in terms of memory size requirement and processing time. Golay code, BCH code, and Reed Solomon code were evaluated and the results showed that Reed Solomon (15, 11) is the best. Energy consumption wasn't studied in this paper.

To improve reliability in data delivery, error control schemes such as Automatic Repeat reQuest (ARQ) and Forward Error Correction (FEC) are used. In ARQ-based schemes, the receiver must to detect lost packets and then request the sender to retransmit packets [9], thus it is a slow and energy consumer method. In FEC codes, sensor nodes encode the sensed data before sending them to the sink node, and at the receiver side, this encoded data are decoded, and if some of them were lost or damaged, receiver can reconstruct the original message, if enough number of encoded packets successfully received. In these schemes for transmit data from sensor to the sink, each aggregator must to decode the received data, aggregate them with new data and encrypt them again before sending to the sink. These operations cause to superabundant end-to-end delay and high
energy consumption, thus cause to decrease the network lifetime. Thus these schemes are not suitable for real time applications. Thereupon, it is necessary to find a new solution with low power consumption that simultaneously provides proper reliability in real time wireless sensor networks applications. Redundant residue number systems (RRNSs) are appropriate for use in real time wireless sensor networks applications, because of obtaining enhancement in 4 factors: 1) real time operations, 2) strong error control capability, 3) energy saving, and 4) security. In this systems, by considering some modulus, instead of sending number X, the remainders of X is transmitted, so less packets are send and power consumption is reduced. In the receiver side using a reverse converter, the received packets are decoded and the original message is recovered.

In this paper, a new high speed, reliable and low energy consumer using redundant residue number system is proposed and for create it, a new three moduli set \( \{2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1\} \) is presented and an efficient reverse converter in terms of area utilization and speed of operations is designed. Moreover by adding two redundant modulus \( \{2^{3n+1} - 1, 2^{4n} + 1\} \) to the new three main moduli set, a new five-moduli set is created that enables error control capability in the proposed scheme. Finally, error detection and correction capability of the proposed scheme will be evaluated.

The rest of this paper is organized as follows: In section 2 the preferences of the proposed scheme comparison with the other existent works are presented. An efficient reverse converter for the new proposed 3-main moduli set \( \{2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1\} \) is designed in section 3. In section 4, error control performance of the proposed moduli set, \( \{2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1, 2^{3n+1} - 1, 2^{4n} + 1\} \) is evaluated using simulation of redundant residue number system algorithm's functionality for the proposed moduli set using C++ language programming. Finally, the paper is concluded in section 5.

2. PREFERENCES OF THE PROPOSED SCHEME

Residue number system is an unconventional, high speed, and fault tolerance number system. By using redundant residue number systems in wireless sensor networks, below excellences will be acquired:

1. **High speed operations:** In residue number systems, the calculations are performed paralleled, that cause to increase the speed of the operations. Moreover, in often schemes that using data aggregation, in each aggregator node must accomplish 3 phases: decryption the received data, aggregation, and encryption the aggregated data again. But using the proposed scheme, in each aggregator node, no need to decrypt the received data, and encrypt them again after aggregation operation. Only aggregation process is accomplished, and no need to consume energy and time for the other 2 phases. If the numbers of nodes be very large, these decreases will be very salient.

2. **Energy efficiency:** Hardware requirement and complexity for data aggregation is reduced and small arithmetic units are used using the proposed scheme, because the remainders of the sensed data are sent by sensors, instead of data itself. Therefore, energy consumption is very low and speed of operations increased.

3. **Error detection and correction capability:** by adding some redundant modulus to the residue number system, redundant residue number system is created that has error control capability. Moreover in such as systems, if error occurs in one modulo, it is not propagates between the other residues.

4. **Security:** Because the remainders of sensed data are sent instead of data itself, the received packet is encrypted and for decrypt the message, need to have the moduli set. Thus the moduli set act as secret key. In the proposed scheme, the sink node using the modulus, decrypt the received packets and recover the original message. If an adversary can acquire the transmitted message, while he doesn't know the moduli set, he can not decrypt this message.

5. **Confidentiality:** As mentioned above, in often proposed scheme for data aggregation in wireless sensor networks, in each aggregator, the received message will be decrypt, and encrypt again after accomplishment of aggregation operations. These two phases cause to the encrypted data lie in danger, and maybe an adversary can achieve the transmitted message easily. But using the proposed scheme, the confidentiality of data will be preserved.

3. DESIGN AND IMPLEMENTATION OF THE PROPOSED REVERSE CONVERTER

A residue number system is defined in terms of relatively prime moduli set \( \{P_1, P_2, ..., P_n\} \) that is gcd\((P_i, P_j) = 1 \) for \( i \neq j \). A weighted number \( X \) can be represented as \( X = (x_1, x_2 ... x_n) \) where

\[
x_i = X \mod P_i = |X|_{P_i}, \quad 0 \leq x_i < P_i
\]

Such a representation is unique for any integer \( X \) in the range \( [0,M) \) where \( M = P_1 P_2 ... P_n \) is the dynamic range of the moduli set \( \{P_1 P_2 ... P_n\} \). [10]. The residue to binary conversion can be performed using the MRC as follows:

\[
X = V_n \prod_{i=1}^{n} P_i + \cdots + V_2 P_2 P_1 + V_1 P_1 + V_1
\]

The coefficients \( V_i P \) can be obtained from residues by:
In the general case, we have:

\[ V_1 = x_1 \quad (3) \]

\[ V_2 = \left| (x_2 - x_1) |P_1^{-1}|_{p_2} \right| \quad (4) \]

\[ V_3 = \left| ( (x_3 - x_1) |P_1^{-1}|_{p_3} - V_2 ) |P_2^{-1}|_{p_3} \right| \quad (5) \]

In the general case, we have:

\[ V_n = \left| ( (x_n - V_1) |P_1^{-1}|_{p_n} - V_2 ) |P_2^{-1}|_{p_n} - \ldots - V_{n-1} ) |P_{n-1}|_{p_n} \right| \quad (6) \]

Where \( |P_i^{-1}|_p \) is multiplicative inverse of \( P_i \) modulo \( P_j \). The modular multiplicative inverse of a moduli \( m \) can be found using the extended Euclidean algorithm.

According to the above equations, the proposed reverse converter for the new 3-moduli set can be designed as follows: Consider the three-moduli set with three corresponding residues \( (x_1, x_2, x_3) \). For design of a residue to binary converter, firstly need to prove that the modulus of proposed moduli set are in fact pair wise relatively prime for the validity of the RNS. Next, should to find the multiplicative inverses, and then the values of the multiplicative inverses and moduli set must be substitute in the conversion algorithm formulas. Then, the resulted equations should be simplified using arithmetic properties. Finally, simplified equations would be realized using hardware components such as full adders and logic gates. Based on Euclid’s Theorem, we have:

\[ \text{gcd} (a,b) = \text{gcd} (b, a \mod b) \quad , \quad a > b \quad (7) \]

Hence,

\[ \text{gcd}(2^{2n+1} + 2^n - 1, 2^{2n+1}) = \text{gcd}(2^{2n+1}, 2^n - 1) = \text{gcd}(2^n - 1, 2) = 1 \quad (8) \]

\[ \text{gcd}(2^{2n+1} + 2^n - 1, 2^n - 1) = \text{gcd}(2^n - 1, 2) = 1 \quad (9) \]

\[ \text{gcd}(2^{2n+1}, 2^n - 1) = \text{gcd}(2^n - 1, 2) = 1 \quad (10) \]

Since the greatest common divisors are 1s in \((8-10)\), thus the numbers \(< 2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1 > \) are relatively prime together. In what follows, using three propositions \((1-3)\), the closed form expressions for the multiplicative inverses under the MRC algorithm are derived that form the basis of our algorithm for the proposed reverse converter.

**Proposition 1:** The multiplicative inverse of \((2^{2n+1} + 2^n - 1) \) modulo \((2^{2n+1}) \) is \( k_1 = -1 \).

**Proof:**

\[ |-2^{2n+1} - 2^n + 1|_{2^{2n+1}} = 1 \quad (11) \]

**Proposition 2:** The multiplicative inverse of \((2^{2n+1} + 2^n - 1) \) modulo \((2^n - 1) \) is \( k_2 = 2^{n-1} \).

**Proof:**

\[ |2^{n-1} \times (2^{2n+1} + 2^n - 1)|_{2^{n-1}} = |2^{2n} - 2^{2n-1} - 2^{n-1}|_{2^{n-1}} = 1 \quad (12) \]

**Proposition 3:** The multiplicative inverse of \((2^{2n+1}) \) modulo \((2^n - 1) \) is \( k_3 = 2^{n-1} \).

**Proof:**

\[ |2^{n-1} \times 2^{2n+1}|_{2^{n-1}} = |2^{2n}|_{2^{n-1}} = 1 \quad (13) \]

Therefore, let the values \(< k_1 = -1, k_2 = 2^{n-1}, k_3 = 2^{n-1}, P_1 = 2^{2n+1} + 2^n - 1, P_2 = 2^{2n+1}, P_3 = 2^n - 1 > \) in \((2-5)\) and we have:

\[ X = x_1 + P_1 (V_2 + V_3 P_2) = x_1 + (2^{2n+1} + 2^n - 1)(V_2 + (2^{2n+1})V_3) \]

\[ V_1 = x_1 \quad (14) \]

\[ V_2 = \left| (x_2 - x_1) |P_1^{-1}|_{p_2} \right| = \left| (x_1 - x_2) \right|_{2^{2n+1}} \quad (15) \]

\[ V_3 = \left| ( (x_3 - x_1) |P_1^{-1}|_{p_3} - V_2 ) |P_2^{-1}|_{p_3} \right| = \left| 2^{2n-2} (x_3 - x_1) - (2^n - 1 \times V_2) \right|_{2^{n-1}} \quad (16) \]

According to the following two properties, \((17)\) can be simplified to decrease the hardware complexity.

**Property 1:** The residue of a negative residue number \((-v)\) in modulo \((2^n - 1)\) is the one’s complement of \(v\), where \(0 \leq v < (2^n-1) \).
Property 2: The multiplication of a residue number \( v \) by \( 2^p \) in modulo \( (2^n - 1) \) is carried out by \( P \) bit circular left shift, where \( P \) is a natural number [11].

For designing an efficient reverse converter, firstly we must to design \( V_2 \) based on (16), then designing \( V_3 \) is performed based on (17), because for designing \( V_3 \), we need to have the value \( V_2 \). Finally, \( X \) is designed based on (14). For (15), no need to design anything, and input \( x_1 \) is connected straightly to the port \( V_1 \). Using FAs, carry save adders (CSAs) with end around carry (EAC) and carry propagation adder (CPA) with EAC, the proposed reverse converter is implemented.

For designing \( V_2 \), simplify (16) as follows:

\[
V_2 = (x_2 - x_1)[x_1^{-1}]_{p_2} = \left| x_1 - x_2 \right|_{2^{2n+1}} = x_1 x_{2n+1} + (-x_2) x_{2n+1} = V_{21} + V_{22}
\]  

(18)

\[
V_{21} = \left| x_1 \right|_{2^{2n+1}} = \frac{x_{1.2n} x_{1.2n-1} \ldots x_{1.0}}{(2n+1)\text{bits}}
\]

(19)

\[
V_{22} = \left| -x_2 \right|_{2^{2n+1}} = \frac{x_{2.2n} x_{2.2n-1} \ldots x_{2.0}}{(2n+1)\text{bits}}
\]

(20)

For implementation of \( V_2 \) based on (19, 20), we need to a \((2n+1)\text{-bits CPA with EAC}, because we have two \((n+1)\text{-bits} \) inputs, one input in (19) and another in (20). Figure 1 shows the realization of \( V_2 \). Note that there are \((2n+1)\) one complements in (20) that are prepared using \((2n+1)\) \text{NOT gates} by \text{OPU} 1.

Now, for realize \( V_3 \) based on (17), we have:

\[
V_3 = \left| ((x_3 - x_4)[x_2^{-1}]_{p_3} - V_2) \right|_{p_3} = \left| 2^{n-2} x_3 - x_4 - 2^n x_V \right|_{2^n-1} = V_{31} + V_{32} + V_{33}
\]

(21)

\[
V_{31} = \left| 2^{n-2} x_3 \right|_{2^n-1} = \frac{x_{3.1} x_{3.0} x_{3.2} \ldots x_{2.0}}{(2n-2)\text{bits}}
\]

(22)

\[
V_{32} = \left| -2^{n-2} x_4 \right|_{2^n-1} = \frac{x_{1.1} x_{1.0} x_{1.2} \ldots x_{1.2}}{(2n-2)\text{bits}}
\]

(23)

\[
V_{33} = \left| (-2^{n-1} x_V) \right|_{2^n-1} = \frac{1 \ldots 1 \ldots 1}{(2n-2)\text{bits}}
\]

(24)

Note that there are \((n-2)\) and \((n-1)\) consecutive 1s, respectively. Thus \((n-2)\) and \((n-1)\) FAs can be substituted by \((n-2)\) and \((n-1)\) \text{pairs of OR/XNOR gates} in CSA 2 and CSA 5 with EAC, respectively. Moreover, for implementation of
(23) and (24), we need to $(2n + 2)$ and $(2n + 1)$ NOT gates, respectively, that are prepared using \textit{OPU 1 and OPU 2}, respectively.

Finally, to find $X$ based on (14), we have:

$$X = x_1 + P_1 (V_2 + V_3 P_2) = x_1 + \left(2^{2n+1} + 2^n - 1\right) \frac{V_2 + (2^n+1)V_3}{c} = x_1 + \left(2^{2n+1} + 2^n - 1\right)c$$

(25)

$$c = V_2 + (2^n+1)V_3$$

(26)

$$C = \text{Concatenation} (V_2, V_3)$$

(27)

$$X = x_1 + \left(2^{2n+1} + 2^n - 1\right)c$$

(28)

As shown in (27), realizing of $C$ no need to hardware requirement, and is based on a simple concatenation. For implementation of (28), we need to two $(5n + 2)$ – bits \textit{CSAs with EAC} and a $(5n + 2)$ – bits \textit{Regular CPA}. Implementation of $X$ is shown in Figure 3.

![Figure 2: Implementation of $V_3$ based on (22 – 24)](image)

![Figure 3: Implementation of $X$ based on (27, 28)](image)

Note that in the third row in (28), we have $(2n + 1)$ and $(n)$ consecutive 1s and 0s, respectively. Thus in \textit{CSA 6}, $(2n + 1)$ and $(n)$ \textit{FAs} can be substituted by $(2n + 1)$ and $(n)$ pairs of \textit{OR/XNOR} and
AND/XOR gates, respectively. Moreover, in (28), we need to \((3n+1)\) NOT gates that are prepared by OPU 3. Addition to these, in the forth row of (28), \((5n+1)\) FAs can be substituted by \((5n+1)\) pairs of AND/XOR gates. Area and delay specifications of each requirement part for implementation of the proposed reverse converter are shown in Table 1. Using Table 1, total hardware requirement (hardware complexity) and delay of retrieving the original data can be calculated. These values are shown in (29) and (30).

Then, specification comparison of the proposed reverse converter for new moduli set \([2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1]\) with the reverse converters with the same or less dynamic range is shown in Table 2.

Finally, performance evaluation of the proposed scheme is shown in the next section.

Table 1: Hardware requirements for implementation of the proposed reverse converter based on the new three moduli set \([2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1]\)

<table>
<thead>
<tr>
<th>Parts</th>
<th>FA</th>
<th>NOT</th>
<th>AND/XOR</th>
<th>OR/XNOR</th>
<th>Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPU 1</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>(t_{\text{NOT}})</td>
</tr>
<tr>
<td>CPA 1</td>
<td>((2n + 1))</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>((4n + 2)t_{FA})</td>
</tr>
<tr>
<td>CSA 1</td>
<td>(n)</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>(t_{FA})</td>
</tr>
<tr>
<td>CSA 2</td>
<td>2</td>
<td>–</td>
<td>((n - 2))</td>
<td>–</td>
<td>(t_{FA})</td>
</tr>
<tr>
<td>OPU 2</td>
<td>((2n + 1))</td>
<td>–</td>
<td>–</td>
<td>(t_{\text{NOT}})</td>
<td></td>
</tr>
<tr>
<td>CSA 3</td>
<td>(n)</td>
<td>–</td>
<td>–</td>
<td>(t_{FA})</td>
<td></td>
</tr>
<tr>
<td>CSA 4</td>
<td>(n)</td>
<td>–</td>
<td>–</td>
<td>(t_{FA})</td>
<td></td>
</tr>
<tr>
<td>CSA 5</td>
<td>1</td>
<td>–</td>
<td>((n - 1))</td>
<td>(t_{FA})</td>
<td></td>
</tr>
<tr>
<td>CPA 2</td>
<td>((3n + 1))</td>
<td>–</td>
<td>–</td>
<td>((2n)t_{FA})</td>
<td></td>
</tr>
<tr>
<td>CSA 6</td>
<td>((2n + 1))</td>
<td>(n)</td>
<td>((2n + 1))</td>
<td>(t_{FA})</td>
<td></td>
</tr>
<tr>
<td>CSA 7</td>
<td>1</td>
<td>–</td>
<td>((5n + 1))</td>
<td>(t_{FA})</td>
<td></td>
</tr>
<tr>
<td>(R-CPA)</td>
<td>((5n + 2))</td>
<td>–</td>
<td>–</td>
<td>((5n + 2)t_{FA})</td>
<td></td>
</tr>
</tbody>
</table>

\[
\text{Total area} = (13n + 6)A_{FA} + (9n + 5)A_{\text{NOT}} + (6n + 1)A_{\text{AND}} + (6n + 1)A_{\text{XOR}} + (4n - 2)A_{\text{XNOR}}
\]

\[
\text{Total delay} = (11n + 11)t_{FA} + 3t_{\text{NOT}}
\]

Table 2: Area and Delay comparison between the proposed reverse converter and related works

<table>
<thead>
<tr>
<th>Moduli set</th>
<th>DR</th>
<th>Area ((A_{FA}))</th>
<th>(t_{FA})</th>
</tr>
</thead>
<tbody>
<tr>
<td>([2^{2n+1} - 1, 2^n + 1, 2^{2n+1} - 1])</td>
<td>5n + 1</td>
<td>8n + 2</td>
<td>12n + 5</td>
</tr>
<tr>
<td>([2^{2n+1} - 1, 2^n + 1, 2^{2n+1} - 1, 2^{2n+1} - 1])</td>
<td>5n</td>
<td>((5n^2 + 43n)/6 + 16n - 1)</td>
<td>18n + 7</td>
</tr>
<tr>
<td>([2^{2n+1} - 2^n - 1, 2^n + 1, 2^{2n+1} - 1])</td>
<td>5n - 1</td>
<td>10n + 5</td>
<td>13n + 1</td>
</tr>
<tr>
<td>([2^{2n+1} - 1, 2^n + 1, 2^{2n+1} + 1])</td>
<td>5n + 1</td>
<td>12.5n + 6</td>
<td>12n + 6</td>
</tr>
<tr>
<td>([2^{2n+1} - 2^n + 1, 2^n + 1, 2^{2n+1} - 1, 2^{2n+1} - 1])</td>
<td>5n - 1</td>
<td>10n + 5</td>
<td>12n + 1</td>
</tr>
<tr>
<td>([2^{2n+1} - 2^n - 1, 2^n + 1, 2^{2n+1} - 1])</td>
<td>4n + 1</td>
<td>9n + 5</td>
<td>11.5n + 6</td>
</tr>
<tr>
<td>([2^{2n+1} - 2^n - 1, 2^n + 1, 2^{2n+1} + 1])</td>
<td>4n + 1</td>
<td>(n^2 + 12n + 12)</td>
<td>16n + 22</td>
</tr>
<tr>
<td>([2^{2n+1} - 2^n + 1, 2^n + 1, 2^{2n+1} - 1])</td>
<td>4n + 1</td>
<td>9n + 10</td>
<td>11n + 14</td>
</tr>
<tr>
<td>([2^{2n+1} - 2^n - 1, 2^n + 1, 2^{2n+1} - 1])</td>
<td>4n + 1</td>
<td>(n^2/2 + 11n + 4)</td>
<td>11n + 8</td>
</tr>
<tr>
<td>([2^{2n+1} - 2^n - 1, 2^n + 1, 2^{2n+1} + 3])</td>
<td>4n</td>
<td>2.5n^2 + 25.5n + 12</td>
<td>18n + 23</td>
</tr>
<tr>
<td>([2^{2n+1} - 2^n - 1, 2^n + 1, 2^{2n+1} + 3])</td>
<td>4n</td>
<td>2.5n^2 + 37.5n + 28</td>
<td>12n + 15</td>
</tr>
<tr>
<td>([2^{2n+1} - 2^n + 1, 2^n + 1, 2^{2n+1} + 3])</td>
<td>4n</td>
<td>20n + 17</td>
<td>13n + 22</td>
</tr>
<tr>
<td>([2^{2n+1} - 2^n - 1, 2^n + 1, 2^{2n+1} + 3])</td>
<td>4n</td>
<td>23n + 11</td>
<td>16n + 14</td>
</tr>
</tbody>
</table>

4. PERFORMANCE EVALUATION

As mentioned before, in section 2, the proposed scheme is reliable, real time and energy efficient. In this section, the performance of the proposed moduli set \([2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1, 2^{2n+1} - 1, 2^{4n} + 1]\) is evaluated in terms of error detection and error correction capability. Firstly we must to explain that why these two redundant modulus were selected for the proposed main modulus. Note that the redundant modulus must be greater than the main modulus, and all of the modulus including the main and redundant modulus must be primed together. In the proposed moduli set, \([2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1, 2^{3n+1} - 1, 2^{4n} + 1]\) it is obvious that the redundant moduli are bigger than the main modulus. For indicate that the modulus in the proposed
moduli set are pair wise prime, only need to show each redundant modulo is prime respect to each main modulo, and the redundant modulus are prime together. Because before, in the previous section, we demonstrate that the main modulus are prime together. Based on Euclid’s theorem (equation (7)), we have:

\[
gcd(2^{3n+1} - 1, 2^{2n+1} + 2^n - 1) = 1
\]
\[
gcd(2^{3n+1} - 1, 2^{2n+1}) = \gcd(2^{3n+1} - 1) = 1
\]
\[
gcd(2^{3n+1} - 1, 2^n - 1) = \gcd(2^n - 1, 1) = 1
\]
\[
gcd(2^{2n+1} + 2^n + 2^n - 1) = 1
\]
\[
gcd(2^{2n+1} + 2^n, 1) = \gcd(2^{2n+1}, 1) = 1
\]
\[
gcd(2^n + 1, 2^{3n} - 1) = \gcd(2^n - 1, 2) = 1
\]
\[
gcd(2^n + 1, 2^{3n} + 1) = \gcd(2^n - 1, 2^n + 1) = \gcd(2^n - 1, 1, -2) = 1
\]

It is obvious from (31 – 37) that the redundant modulus are prime together and are prime against the main modulus, thus these redundant moduli are suitable for these three main modulus.

The proposed moduli set \{2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1, 2^{2n+1} - 1, 2^{4n} + 1\} can correct up to \(2n + 2\) adjacent error bits, while the error bits sat in the first modulo of the received remainders. For study the error correction capability, we consider a special case of the proposed moduli set by setting \(n = 2\). As result, the proposed moduli set is equal to \{35, 32, 3, 127, 257\} that \{35, 32\} are the main modulus and \{127, 257\} are redundant modulus. Error detection capability in this special case is up to 6 bits, because by setting \(n = 2\) in \(2n + 2\), the number 6 is acquired. For study the error detection capability, we study 18 different sent numbers that have error bits in more than one moduli. Thus error correction is impossible. These numbers are in \(X \in [0, 3360]\) that is equal to dynamic range of the moduli set \{35, 32, 3\}, because \(35 \times 32 \times 3 = 3360\). The selected numbers are \{0, 200, 400, 600, 800, 1'000, 1'200, 1'400, 1'600, 1'800, 2'000, 2'200, 2'400, 2'600, 2'800, 3'000, 3'200, 3'359\}. Overall, 150’000 error bits are considered. The results are shown in Figure 4.

![Figure 4: Percent of error detection capability of the proposed scheme by setting \(n = 2\) for 18 different sent data by considering 150’000 different error bits states](image)

It is obvious from Figure 1 that the percent of error detection is always more than 71\%. The best percent of error detection capability is 74.51\% and in the worst case, it is 71.57\%, as shown in the Figure 1, and the average percent of the error detection for 150’000 under consideration error bits states is equal to 73.28\%.

5. CONCLUSION

This paper presents a new high speed, reliable and energy efficient scheme for real time wireless sensor networks applications. For create the proposed scheme, a new moduli set \{2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1, 2^{3n+1} - 1, 2^{4n} + 1\} presents that the modulus \{2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^n - 1\} are the main modulus and the modulus \{2^{3n+1} - 1, 2^{4n} + 1\} are redundant modulus. Moreover, an efficient reverse converter based on mixed radix conversion designs for the main proposed moduluss that has supremacy in terms of hardware requirement and speed of operations comparison with the other reverse converters for the moduli sets with the same or less
dynamic ranges. Addition to these, error control capability of the proposed moduli set for the proposed scheme evaluates using simulation of error detection and correction algorithm by C++ programming language by setting \( n = 2 \) in the proposed moduli set \( \{2^{2n+1} + 2^n - 1, 2^{2n+1}, 2^{n+1} - 1, 2^{3n+1} - 1, 2^{4n} + 1\} \) and for 18 different sensed sent data under consideration of 150'000 error bit states, error detection capability studies. The results show that the proposed scheme can detect error bits with the probability of 73.28% on average. Moreover, in the special case \((n=2)\), the error detection and correction algorithm can correct up to 6 error bits. As totality results, the proposed scheme has excellent preference in terms of real time operations, reliability in data delivery and energy efficiency comparison with the other schemes in wireless sensor networks.

REFERENCES

12. Molahosseini, A. S., K. Navi, C. Dadkhah, O. Kavehei, S. Timarchi, 2010. Efficient reverse converter designs for the new 4-moduli sets \{2^n-1, 2^n, 2^{n+1}, 2^{2n+1}-1\} and \{2^n-1, 2^n+1, 2^{2n}, 2^{2n+1}+1\} based on new CRTs. IEEE Transactions on Circuits and Systems-I: Regular papers, 57 (4): 1-13.
17. Mohan, P.V.A. and A.B. Premkumar, 2007. RNS to binary converters for two four moduli set \{2^n-1, 2^n, 2^{n+1}, 2^{2n+1}-1\} and \{2^n-1, 2^n, 2^{n+1}, 2^{2n+1}+1\}. IEEE Transactions on Circuits and Systems- I: Regular papers, 24 (6): 1245-1254.