An Energy Efficient Wireless Sensor Network by using Ternary with Silent Symbol Coding Technique
Ashutosh Ranjan¹, Beauty Sharma², Yogesh Barade³, Prof. L.K. Wadhwa⁴
Department of Electronics and Telecommunications
Dr. D Y Patil Institute of Technology Pimpri, Pune, India

Abstract:
Today Wireless Sensor Networks (WSNs) are used in many industries. Wireless sensor networks typically require low cost devices and low power operations. WSN consist of a large number of sensor nodes. Maximizing node or network lifetime is primary objective in WSN because it is very difficult to charge or replace exhausted batteries. Thus energy efficient communication is very important part in WSN to reduce the device recharging cycle period and hence provide connectivity for longer duration. Hence we are using a new energy efficient communication coding technique for wireless sensor networks that is based on the ternary number system encoding of data. An efficient algorithm for conversion from binary to ternary and vice versa is used that does not involve any division or multiplication but only addition. Ternary with Silent Symbol (TSS) coding technique is similar in concepts to the RBNSiZeComm protocol proposed in [1]–[3]. However, in contrast to RBNSiZeComm, TSS simultaneously saves energy at both the transmitter and receiver due to shortening of the transmission duration.

Keywords: Energy-efficient communication, wireless net-works, ternary encoding, silent communication, sensor networks.

I. INTRODUCTION
Wireless sensor networks (WSNs) are attracting increasing research attention, due to their wide spectrum of applications, including military purposes for monitoring, tracking and surveillance of borders, intelligent transportation systems for monitoring traffic density and road conditions, and environmental applications to monitor, for example, atmospheric pollution, water quality, agriculture, etc. AWSN is composed of a number of sensor nodes (SN) transmitting wirelessly the information they capture. An SN is generally composed of a power unit, processing unit, sensing unit, and communication unit. Power consumption is the main limiting factor of an SN. In fact, SNs are in general required to operate autonomously and independently for a large period of time in areas where power infrastructure may not be available. Thus, battery-powered SNs should be able to operate with very low power consumption. Some SNs have batteries rechargeable by solar power, thus ensuring longer autonomous operation. The processing unit is responsible to collect and process signals captured from sensors before transmitting them to the network. The sensing unit is a device that produces a measurable response to a change in a physical condition like temperature or pressure.

The wireless communication unit is responsible for transferring the sensor measurements to the exterior world, e.g., to be stored on a server, where they can be distributed on the internet or accessed by specialized personnel. The wireless communication unit can also ensure a mechanism for ad-hoc communication between SNs forming a WSN. In fact, in some scenarios, it might be more energy efficient to transmit a message via multihop communications over short distances instead of a single hop long distance transmission to the base station (BS). In practice, most existing transmission schemes not only utilize non-zero voltage levels for both 0 and 1 so as to distinguish between a silent and a busy channel, they also keep both the transmitter and the receiver switched on for the entire duration of the transmission of a data frame. Communication strategies that require energy expenditure for transmitting both 0 and 1 bit values are known as energy based transmission (EbT) schemes. In other words, if the energy required per bit transmitted is $e_b$, the total energy consumed to transmit an n-bit data would be $n \cdot e_b$. Most current research efforts on reducing energy consumption have focussed on the MAC layer design [11], [12], [18], [20], optimizing data transmissions by reducing collisions and retransmissions [13], [14], [21] and through intelligent selection of paths or special architectures for sending data [10], [12], [15]–[17], [20]. In all such schemes, the underlying communication strategy of sending a string of binary bits is energy based transmission.

In contrast to EbT based communication schemes, a new communication strategy called Communication through Silence (CtS) was proposed in [4] that involves the use of silent periods as opposed to energy based transmissions. CtS, however, suffers from the disadvantage of being exponential in communication time. An alternative strategy, called Variable-Base Tacit Communication (VarBaTaC) was proposed in [5] that uses a variable radix-based information coding coupled with CtS for communication. However, for an n-bit binary string, the duration of transmission is in general significantly longer than n. Neither [4] nor [5] talk about the amount of energy saved by CtS and VarBaTaC for noisy channels and considering real-life device characteristics.

A. Our Contribution
One interesting open issue related to the RBNSiZeComm protocol proposed in [2], [3] was whether a re-coding of binary data to higher radix such as ternary and then using the silent symbol strategy from [2], [3] would yield comparable savings in energy. In this work, we propose a new communication
scheme based on recoding data from binary to the ternary radix and the silent symbol strategy, with the aim of generating energy savings simultaneously at the transmitter and the receiver. An efficient algorithm for conversion from binary to ternary and vice versa is used that does not involve any division or multiplication but only addition.

A comprehensive analysis of the energy-efficiency of a communication strategy needs to address the MAC protocols along with adequate physical layer awareness. However, we do not address the issues related to MAC protocols in this work and make an effort to improve the energy efficiency with physical-layer centric approach. We study the performance of the proposed ternary with silent symbol (TSS) protocol in the context of low power wireless networks for noisy channels with a new transceiver design. In order to understand the energy-efficiency aspect of the proposed scheme (TSS combined with ASK and PSK/FSK), we therefore examine first the candidate source coding scheme, i.e., TSS.

Thereafter, we estimate the energy spent in the baseband electronic circuits (device consumption) followed by the evaluation of the energy consumed during RF transmission for a given BER specification with appropriate analytical models. While there are numerous coded modulation techniques in the literature [22], applications requiring low power and low cost radio devices (e.g., sensor networks) usually employ radios with only simple modulation techniques such as ASK, OOK and FSK [10].

Our proposed transceiver design uses a hybrid modulation scheme utilizing FSK and ASK to keep both the cost and complexity low. With a non-coherent de-tection based receiver and assuming equal likelihood of all possible binary strings of a given length, there is a 20% savings in energy on an average at the transmitter compared to binary FSK, for additive white gaussian noise (AWGN) channels. In addition, there is a savings of about 36.9% in battery energy at the receiver resulting from a reduction in the length of the transmitted data (and hence transmission duration) as a result of recoding to ternary radix from binary data.

Thus, while the energy savings generated by TSS at the transmitter are inferior to that of RBNSiZeComm (which is around 41% on an average for noisy channels [11]), TSS scores over RBNSiZeComm in saving energy at the receiver too. These results hence demonstrate that TSS may be useful for low power wireless networks, particularly for multi-hop communications.

II. BASIC IDEA AND ALGORITHM DESCRIPTIONS

Let a given binary message \( B \) be represented by an \( n \)-bit binary string \( b_{n-1}b_{n-2} \ldots b_0 \). Let \( T \) be its equivalent \( m \)-digit ternary representation given by \( T = t_{m-1}t_{m-2} \ldots t_0 \). We assume that \( n \) is even, otherwise we pad the message with a 0 bit at the leftmost (msb) position.

For conversion from binary to ternary, we successively scan every two bits of the binary message starting from its msb position and then convert the leftmost part of the binary message (starting from its msb to the currently scanned bit position) to its equivalent ternary representation. To do this, we replace the currently scanned two bits (bit-pair) by either a single ternary digit as shown in below table.

<table>
<thead>
<tr>
<th>Binary bit pair (radix-2)</th>
<th>Ternary bits (radix-3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(00)2</td>
<td>(0)3</td>
</tr>
<tr>
<td>(01) 2</td>
<td>(1)3</td>
</tr>
<tr>
<td>(10) 2</td>
<td>(2)3</td>
</tr>
<tr>
<td>(11) 2</td>
<td>(10)3</td>
</tr>
</tbody>
</table>

And then add it to four times the so far obtained ternary number \( T \) from all the previous bit positions (on the left of this bit-pair). Note that this addition will be in ternary number system, and the multiplying factor four is to adjust the weight of \( T \) with respect to the currently scanned bit-pair. Also, since this addition will be in ternary number system, the weight of four can be assigned to \( T \) by adding the two ternary numbers \( T \) 0 and 07.

Thus, the equivalent ternary representation of the part of the binary message from its msb to the currently scanned bit-pair will then be obtained as \( 70 \oplus T \oplus x \), where \( \oplus \) represents a ternary addition operator, and \( x \) is the ternary number equal to 0, 1, or 2 (when the scanned bi-pair is 00, 01 or 10, respectively), or equal to \((10)_{3}\) (when the scanned bi-pair is 11).

**Example 1: Binary to Ternary Conversion**

Consider the binary number 101011. First, we get \( T = 2 \) from the leftmost two bits 10. Then, for the next two bits, the equivalent ternary digit is 2. So we add (in ternary) the three numbers \((20)_{3}\), \((02)_{3}\), and \((2)_{3}\), to get the changed value of \( T = (24)_{3} \) (which is equal to 10 in decimal corresponding to the binary digits 1010 scanned so far) but we are not using the digit-4 in ternary format that’s why we convert it into its equivalent ternary bits, and it is \( T=(101)_{3}\).

Then for the next two bits i.e. \((11)_{2}=(10)_{3}\), we get the equivalent ternary representation as \((10)_{3}\), which is added to \((1010)_{3}(=T'0)\) and \((0101)_{3}(=07)\) in ternary to get the changed value of \( T = (1121)_{3} = (43)_{10} \). In a similar manner, we can reconvert the received \( m \)-digit ternary message to its equivalent binary form by using algorithm Ternary2Binary, where again we scan the given ternary number from its most significant (leftmost) digit position in a digit by digit manner, and convert the part of the so far scanned ternary digits to its equivalent binary representation. In the algorithm, we denote a ternary digit by \( t \) and the converted binary number by \( B \).

As in the previous Binary2Ternary algorithm, we need to adjust the weight of the binary representation of the previously scanned ternary digits, by attaching a weight of three to it, with respect to the currently scanned ternary digit, and this is achieved by adding \( B0, 0B \) and the equivalent binary representation of the currently scanned ternary digit (note that, this time it is a binary addition).

**Example 1: Ternary to Binary conversion**

Let the given ternary number \( T = 1121 \). The leftmost digit is 1 whose binary representation is 01. So we initially form the binary number as \( B = 01 \). The next ternary digit is 1. We add the three binary numbers \( 010(=B0), 001(=0B) \) and 1 to get the changed value of \( B = 100 \). The next scanned ternary digit is 2. So we add 1000 to 0100 to get the changed value of \( B = 1110 \). Finally, for the rightmost ternary digit 1, we add 11100, 01110 and 01 to get 101011 as the equivalent binary representation \( B \) of the given ternary number \( T \).
Algorithm for Binary2Ternary Conversion

```plaintext
procedure BINARY2TERNARY(IN bit vector B, integer n, OUT ternary number T)
/* Initialization */
if \( b_{n-1}b_{n-2} = 00 \) then
    \( T \leftarrow 0 \);
end if
if \( b_{n-1}b_{n-2} = 01 \) then
    \( T \leftarrow 1 \);
end if
if \( b_{n-1}b_{n-2} = 10 \) then
    \( T \leftarrow 2 \);
end if
if \( b_{n-1}b_{n-2} = 11 \) then
    \( T \leftarrow 10 \);
end if
for \( i = n - 3 \) down to 1 step -2 do
    if \( bb_{i-1} = 00 \) then
        \( T \leftarrow T \oplus 07 \) /*this is a ternary addition*/
    end if
    if \( bb_{i-1} = 01 \) then
        \( T \leftarrow T \oplus 07 \oplus 1 \) /*ternary addition*/
    end if
    if \( bb_{i-1} = 10 \) then
        \( T \leftarrow T \oplus 07 \oplus 2 \) /*ternary addition*/
    end if
    if \( bb_{i-1} = 11 \) then
        \( T \leftarrow T \oplus 07 \oplus 10 \) /*ternary addition*/
    end if
end for
end procedure
```

Steps for Transmission of Data:

**Step 1:** Recode the binary data to ternary using the Binary2Ternary protocol.

**Step 2:** Transmit the ternary data bit by bit, obtained by the recoding process from step 1 using the following rules:

1. **Step 2.1:** If the bit to be transmitted is a 0, then switch off the transmitter for the bit period.

2. **Step 2.2:** Otherwise, switch on the transmitter and transmit the bit (1 or 2) in the time slot.

The receiver side steps would be exactly opposite to the steps of transmission of data.

Algorithm for Ternary2Binary

```plaintext
procedure TERNARY2BINARY(IN ternary number T, integer m, OUT binary number B)
/* Initialization */
if \( t_{m-1} = 0 \) then
    \( B \leftarrow 00 \);
end if
if \( t_{m-1} = 1 \) then
    \( B \leftarrow 01 \);
end if
if \( t_{m-1} = 2 \) then
    \( B \leftarrow 10 \);
end if
for \( i = m - 2 \) to 0 step -1 do
    if \( t_i = 0 \) then
        \( B \leftarrow B \oplus 0.2; /* this is a binary addition*/
    end if
    if \( t_i = 1 \) then
        \( B \leftarrow B \oplus 0.2 + 1; /* binary addition*/
    end if
    if \( t_i = 2 \) then
        \( B \leftarrow B \oplus 0.2 + 10; /* binary addition*/
    end if
end for
end procedure
```

Steps for Reception of Data:

**Step 1:** Receive the ternary encoded data symbol by symbol as follows:

1. **Step 1.1:** If a signal is received in the given time slot (symbol period), then the received symbol is either a 1 or a 2, depending on the predetermined received signal to symbol
mapping.

Step 1.2: Otherwise, if the channel is quiet, then the received symbol is a 0.

Step 2: Convert the received ternary message to binary using the Ternary2Binary protocol.

III. RESULT:

Table 1. Comparison of Peak Transmitter Powers

<table>
<thead>
<tr>
<th>BER Values</th>
<th>Corresponding Eth Value (volts)</th>
<th>Required SNR in dB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TSS Protocol</td>
<td>FSK Binary</td>
</tr>
<tr>
<td>6.747 × 10^-4</td>
<td>0.55</td>
<td>14.00</td>
</tr>
<tr>
<td>1.573 × 10^-3</td>
<td>0.53</td>
<td>16.23</td>
</tr>
<tr>
<td>4.4763 × 10^-4</td>
<td>0.53</td>
<td>16.5</td>
</tr>
<tr>
<td>4.6849 × 10^-4</td>
<td>0.54</td>
<td>15.5</td>
</tr>
</tbody>
</table>

IV. CONCLUSION:

We have presented in this paper a new low energy communication scheme that can generate energy savings simultaneously at the transmitter and the receiver, unlike the RBNSiZeComm protocol in [1]–[3]. Using low cost and low complexity implementation scheme based on a hybrid modulation utilizing FSK and ASK for the TSS protocol proposed we show that for AWGN noisy channels, there is an average savings of 20% in battery energy at the transmitter for equal likelihood of all possible binary strings of a given length. Simultaneously, there is a savings of 36.9% energy at the receiver. An efficient algorithm involving only addition (and no multiplication or division) for conversion from binary to ternary and vice versa is used in order to keep the energy consumed for the radix conversion process low at both the transmitter and the receiver. Coupled with the low cost and low complexity of transceiver, these savings clearly demonstrate the usefulness of TSS for low power wireless sensor networks, particularly for multi-hop communications.

V. REFERENCES:

[17]. C. Intanagonwvat, R. Govindan, D. Estrin, J. S.


