An XOR Encoding for Wireless Body Area Networks*

Keigo YOKOTA, Akiko MAMADA and Hiroyoshi MORITA
The University of Electro-Communications
1-5-1 Chofugaoka, Chofu, Tokyo, JAPAN
yokota@appnet.is.uec.ac.jp, {amanada, morita}@is.uec.ac.jp

ABSTRACT
Devices using Wireless Body Area Networks (WBANs) have been gaining strong interest these days, especially in the medical area. Such devices must be constructed to reduce the power consumption, while the robustness against errors during transmission needs to be considered. In this paper, we propose some encoding methods which are low-power and strong against communication errors, by serially concatenating an XOR encoding and a BCH encoding.

Keywords
WBAN, XOR encoding, BCH encoding, concatenated encoding, power consumption

1. INTRODUCTION
Recently, medical devices (such as ECG's) have been developed rapidly in order to measure biological information of patients, even though they are outside of hospitals. Such medical devices tend to be unwired in order to reduce burdens for patients. A Wireless Body Area Network (WBAN) is a network which is mainly applied for these devices. More precisely, a WBAN is used to collect biological information via sensors over/implanted under patients' skins.

Devices used in WBANs must be small and light, so they must be constructed to reduce power consumption. At the same time, data transmitted from devices must have high robustness against errors to reduce the risk of medical errors. In this paper, we propose an encoding scheme that is low-power and strong against communication errors. More precisely, we first examine the encoding power consumptions for XOR encodings [3] and a BCH(63,51) encoding which is recommended by IEEE standard [2]. We then analyze performances of encoding methods against communication errors by serially concatenating an XOR encoding and a BCH(63,51) encoding. We conclude such these concatenations can produce, compared with only BCH encoding, good encodings in terms of robustness against communication errors, meanwhile they use low power.

We organize the rest of paper as follows. In Section 2, we introduce XOR encodings, which have high robustness against packet erasures. In Section 3, we estimate the power consumptions for various encodings (XOR encodings and a BCH encoding) by using the simulator in the Field-Programmable Gate Array (FPGA) designing tool. In Section 4, we propose concatenations of BCH and XOR encodings, and then simulate their robustnesses against errors and power consumption. We also examine the robustness by changing XOR encoding patterns. We terminate this paper with conclusion in Section 5.

2. PREVIOUS WORK
One of the methods that is expected to be low-power in WBANs is an encoding that uses only XOR operations [3], which we call an "*XOR encoding*". An XOR encoding is shown to be able to recover data when packet erasures occur during data transmission [1].

For example, consider the case that we transmit 2 packets (data) P_1 and P_2. In an encoding process, calculate $P_1 \oplus P_2$ by XORing P_1 and P_2. A transmitter sends P_1, P_2 and $P_1 \oplus P_2$ to a receiver. If some packet erasure occurs and the receiver receives only two of P_1, P_2 and $P_1 \oplus P_2$, then the receiver can obtain P_1 and P_2 by applying an XOR operation towards received 2 packets (if necessary). Because this encoding uses only XOR operations that are lightweight calculations, it is expected to use less power than other encodings.

3. ENCODING POWERS
We first examined the power consumptions for encodings we focus on throughout this paper. More precisely, we examined the power consumptions for XOR encodings and the BCH(63,51) encoding, and compared them each other. The BCH(63,51) encoding produces a 63-bit encoded packet consisting of a 51-bit original information packet followed by 12-bit parity check bits.

In the case of mounting an encoding in a device, the power consumption depends on the design of the encoding, so it is difficult to simulate the power consumption by using formula manipulation softwares. Therefore, we simulate power

*This work was supported by JSPS KAKENHI Grant Number 25870228.

BODYNETS 2013, September 30-October 02, Boston, United States

DOI 10.4108/icst.bodynets.2013.253570

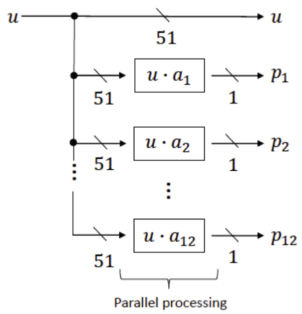

Figure 1: The BCH(63,51) encoding

consumptions by Xilinx XPower Analyzer in the FPGA designing tool. A target device is set to be Xilinx Vertex-5 XC5VLX50, and design 3 encodings to the device; the BCH(63,51) encoding, the 51-bit XOR encoding and the 63-bit XOR encoding; here the frequency of the operation is set to be 25.0[MHz].

We designed the 51-bit and the 63-bit XOR encodings so that they produce the outputs from the inputs in 1 cycle. We also designed the BCH(63,51) encoding so that it produces the output from the input in 1 cycle as follows so as to make a comparison easier. Let G be the systematic generator matrix of the BCH(63,51), so G can be represented as $G = (I|A)$ with the 51×51 unit matrix I and a 51×12 matrix A. Let $\mathbf{u}$ be a 1×51 row vector representing a 51-bit original packet to be encoded, and $\mathbf{c}$ be a 1×63 row vector representing the 63-bit encoded packet given by $\mathbf{c} = \mathbf{u}G$. However, since G is systematic, it is enough to calculate the 12 parity check bits $p_1, p_2, \cdots, p_{12}$ to obtain $\mathbf{c}$, where these parity check bits are calculated by $\mathbf{u}A$. Therefore, each parity check bit p_i ($i = 1, 2, \ldots, 12$) can be independently calculated as $p_i = \mathbf{u} \cdot \mathbf{a}_i$ (the inner product of $\mathbf{u}$ and $\mathbf{a}_i$), where $\mathbf{a}_i$ is the i-th column in the matrix A. Thus, it is possible to produce the output from the input in 1 cycle by calculating these 12 parity check bits $p_1, p_2, \cdots, p_{12}$ in parallel (as shown in Figure 1).

The simulation results are shown in Table 1, which are given by taking the averages of the power consumptions per cycle by simulating 100,000 times. Recall that the power consumption by cycle means the power consumption required for one encoding. Also, the power P of each module for an encoding can be calculated as

$$P = k \times C \times V^2,$$

where k is the number of switching rate per cycle, C is the capacitance and V is the supply voltage. Hence, the total power is given by summing up all powers of modules.

Table 1 shows that both the 51-bit and the 63-bit XOR encodings require less power (about 0.7[mW/cycle] lower) compared with the BCH(63,51) encoding. Hence, we can conclude that the power consumption per bit for XOR encodings is much lower than that for the BCH encoding.

Table 1: The power consumption for each encoding

	BCH(63,51)	51-bit XOR	63-bit XOR
Power[mW/cycle]	0.74	0.04	0.05

4. DECODING PROBABILITIES

4.1 Concatenations of encodings

An XOR encoding has high robustness against packet erasures, while it cannot correct other types of errors, such as bit flip errors (BFEs). In fact, a decoding with XOR operations can propagate BFEs. For example, consider 2 packets $P_1 = (1, 0, 0, 1)$ and $P_2 = (0, 0, 1, 1)$. If P_1 and P_2 have BFEs in the first position and the second position, respectively (sp $\hat{P}_1 = (\underline{0}, 0, 0, 1)$ and $\hat{P}_2 = (0, \underline{1}, 1, 1)$), then the resulting packet $\hat{P}_1 \oplus \hat{P}_2$ under the XOR operation has two BFEs at the first and second positions ($\hat{P}_1 \oplus \hat{P}_2 = (\underline{0}, \underline{1}, 1, 0)$).

Therefore, we consider to concatenate XOR and BCH encodings in order to have robustness against not only packet erasures but also BFEs with low power consumption. To examine the efficiencies of such concatenations, we simulated the success probability of data transmission through a noisy channel. We call the probability the "*decoding probability*". For simplicity, we did the simulations under the assumptions that (1) a packet erasure and a BFE in the noisy channel occur independently with fixed probabilities, where q and p represent the probability of a packet erasure (packet erasure rate, PER) and the probability of a BFE (bit error rate, BER), respectively; and (2) prepare 9 51-bit packets $P_1, P_2, \ldots, P_9$, and they are sent to a receiver with redundancy 2 after encoding; that is, the total number of packets to be sent to a receiver is 18. The simulations were implemented by MAGMA [4].

We first examined the performance of decoding probabilities when only the BCH(63,51) encoding is used (Type 1) in a following way.

Simulation for Type 1

(1) Prepare 9 51-bit packets $P_1, P_2, \ldots, P_9$ randomly, and then, encode them into 63-bit packets $P_1^{'}, P_2^{'}, \ldots, P_9^{'}$ under the BCH(63,51) encoding.

(2) For each packet $P_i^{'} (= P_{(i,1)}^{'})$, $i = 1, 2, \ldots, 9$, make a copy $P_{(i,2)}^{'}$ of $P_i^{'}$; so there exist 18 packets $P'_{(1,1)}, P'_{(1,2)}, \ldots, P'_{(9,1)}$, $P'_{(9,2)}$ at this moment.

(3) Each packet is independently erased with probability q. For all packets $P_{(i,j)}^{'}$ which are not erased, provoke BFEs with probability p (this step represents the noisy channel).

(4) Apply decoding for all non-erased packets $P_{(i,j)}^{'}$. If at least one of $P_{(i,1)}^{'}$ and $P'_{(i,2)}$ can be decoded as P_i, we say that P_i is successfully transmitted. If all P_is are successfully transmitted, we call the case "*Decode success*".

We next examined a case of concatenated encoding with the BCH(63,51) encoding followed by the 63-bit XOR encoding (Type 2) in a following way.

Simulation for Type 2

(1) Prepare $P_i^{'}$ from P_i ($i = 1, 2, \ldots, 9$) as denoted in the simulation for Type 1.

(2) Generate 18 packets from $P_i^{'}$s using the XOR encoding scheme denoted in Table 2, which is proven to have strong robustness against packet erasures [1].

(3) For these 18 packets generated in (2), we write $P_m^{'} \oplus P_n^{'}$ as $P'_{(m,n)}$ (when $m \neq n$), and $P_m^{'}$ as $P'_{(m,m)}$; for example, $P_{(2,5)}^{'} = P_2^{'} \oplus P_5^{'}$ and $P_{(1,1)}^{'} = P_1^{'}$.

(4) For a boolean variable $b_{(m,n)}$ with $1 \leq m, n \leq 9$, set

$b_{(m,n)} = true$ if there exists a corresponding packet $P'_{(m,n)}$ (including the case of $m = n$), and $b_{(m,n)} = false$ otherwise.
(5) Set $b_{(m,n)} = false$ if $P'_{(m,n)}$ is erased in the transmission. For all packets $P'_{(m,n)}$ such that $b_{(m,n)} = true$, provoke BFEs with probability p.
(6) Apply XOR decoding by following Algorithm 1. Observe that each P'_i is uniquely decoded by using the minimum number of XOR operations amongst all possible decoding ways (if it can really be decoded).
(7) All packets decoded in (6) are further decoded by applying the parity check matrix of BCH(63,51). If all P_is are successfully retrieved, we call the case *"Decode success"*.

Table 2: XOR encoding scheme

P'_1	$P'_2 \oplus P'_5$	$P'_3 \oplus P'_4$	$P'_4 \oplus P'_5$	$P'_5 \oplus P'_6$	$P'_6 \oplus P'_1$
P'_4	$P'_5 \oplus P'_8$	$P'_6 \oplus P'_7$	$P'_7 \oplus P'_8$	$P'_8 \oplus P'_9$	$P'_9 \oplus P'_4$
P'_7	$P'_8 \oplus P'_2$	$P'_9 \oplus P'_1$	$P'_1 \oplus P'_2$	$P'_2 \oplus P'_3$	$P'_3 \oplus P'_7$

Algorithm 1 XOR decoding scheme

Input : $P'_{(m,n)}, b_{(m,n)} (m, n = 1, 2, \cdots, 9)$
Output : $P'_{(m,n)}, b_{(m,n)}$

```
c ⇐ 0
while c = 0 do
  c ⇐ 1
  for m = 1 to 9 do
    if b_(m,m) is true then
      for n = 1 to 9 do
        if (b_(m,n) is true) and (b_(n,n) is false) then
          P'_(n,n) ⇐ P'_(m,m) ⊕ P'_(m,n)
          b_(n,n) ⇐ true
          c ⇐ 0
        end if
      end for
    end if
  end for
end while
```

We further examined a case of concatenated encoding with the 51-bit XOR encoding followed by the BCH(63,51) encoding (Type 3) in the same manner as Type 2. More precisely, in Type 3, P'_i in Table 2 is changed to P_i, and $P'_{(m,n)}$ in Algorithm 1 is changed to $P_{(m,n)}$, where $P_{(m,n)} = P_m \oplus P_n$ if $m \neq n$ and $P_{(m,n)} = P_m$ if $m = n$.

The decoding probability for each type is given by

$$\frac{\text{The number of decode success}}{\text{The number of trials}}, \tag{1}$$

where the number of trials is set to be $100,000$, and the PER p and the BER q are ranged from 0.001 to 0.006 and from 0 to 0.5, respectively. These result are shown in Figures 2,3,$\cdots$,7.

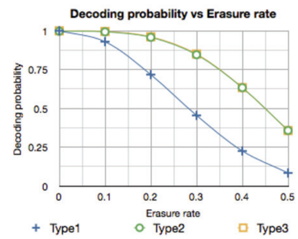

Figure 2: $p = 0.001$

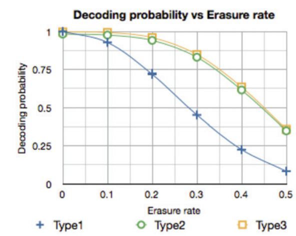

Figure 3: $p = 0.002$

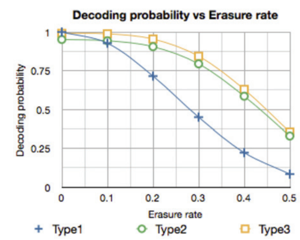

Figure 4: $p = 0.003$

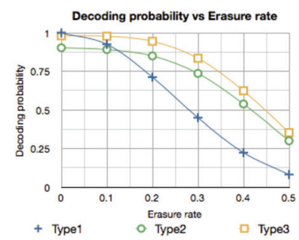

Figure 5: $p = 0.004$

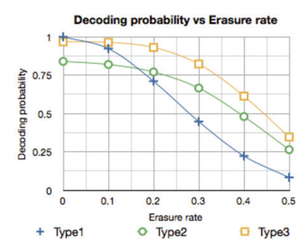

Figure 6: $p = 0.005$

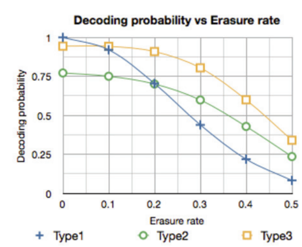

Figure 7: $p = 0.006$

These simulations show that
(a) the decoding probabilities of Type 2 and Type 3 are higher than that of Type 1 as q gets higher;
(b) the decoding probability of Type 2 is smaller than that of Type 3 as p gets higher; and
(c) in the case of p gets higher and q gets smaller, Type 1 provides the highest decoding probability. This is because many packets affected by BFEs are sent to the receiver and BFEs are propagated during the XOR decoding.

To summarize, the best type for WBANs should be determined properly based on p and q. For example, in the case of $p = 0.005$, the best type is Type 1 if $q < 0.05$, and Type 3 if $q \geq 0.05$ (as shown in Figure 6).

We also estimated the power consumption [mW] of each type. Type 1 needs 9 times of the BCH(63,51) encoding. Type 2 needs 9 times of the BCH(63,51) encoding followed by 15 times of the 63-bit XOR encoding. Type 3 needs 15 times of the 51-bit XOR encoding followed by 18 times of the BCH(63,51) encoding. Therefore, from the results in Table 1, we can estimate the power consumption for each type and the results are shown in Table 3.

Table 3: The power estimation of each type

	BCH(63,51)	51-bit XOR	63-bit XOR	Total Power
Type 1	6.66	-	-	6.66
Type 2	6.66	-	0.75	7.41
Type 3	13.32	0.60	-	13.92

The results show that the power consumption of Type 3 is almost twice the power consumption of Type 1. Using Type 3 brings in high decoding probability in the case when packet erasures happen with high probability. On the other hand, Type 3 requires more power since it does more BCH encodings compared with Type 1 and Type 2. However, in the case of $p = 0.001$, Type 2 also performs well in terms of robustness against errors; that is, the decoding probability of Type 2 is almost equal to that of Type 3 (as shown in Figure 2). Furthermore, the power consumption of Type 2 is reasonably low compared with that of Type 3. Hence, Type 2 can be a suitable encoding type when p is small.

4.2 XOR Encode Pattern Verification

We examined performances of Type 2 and Type 3 encoding methods from the perspective of the power consumptions and the decoding probabilities, using the encoding scheme in Table 2. However, it might be possible to obtain better performances by using different XOR encoding schemes. In this subsection, we further analyze Type 2 and Type 3 by implementing four distinct XOR encoding schemes. Throughout this subsection, we call the XOR encoding schemes in Table 2, Table 4, Table 5 and Table 6 "pattern 1", "pattern 2", " pattern 3" and "pattern 4", respectively.

Table 4: Pattern 2

P_1'	$P_2' \oplus P_4'$	$P_3' \oplus P_5'$	$P_4' \oplus P_5'$	$P_5' \oplus P_6'$	$P_6' \oplus P_1'$
P_4'	$P_5' \oplus P_8'$	$P_6' \oplus P_7'$	$P_7' \oplus P_8'$	$P_8' \oplus P_9'$	$P_9' \oplus P_4'$
P_7'	$P_8' \oplus P_2'$	$P_9' \oplus P_1'$	$P_1' \oplus P_2'$	$P_2' \oplus P_3'$	$P_3' \oplus P_7'$

Table 5: Pattern 3

P_1'	P_2'	P_3'	$P_4' \oplus P_5'$	$P_5' \oplus P_6'$	$P_6' \oplus P_4'$
P_4'	P_5'	P_6'	$P_7' \oplus P_8'$	$P_8' \oplus P_9'$	$P_9' \oplus P_7'$
P_7'	P_8'	P_9'	$P_1' \oplus P_2'$	$P_2' \oplus P_3'$	$P_3' \oplus P_1'$

Table 6: Pattern 4

P_1'	P_2'	P_3'	$P_4' \oplus P_5'$	$P_5' \oplus P_6'$	$P_6' \oplus P_7'$
P_4'	P_5'	P_6'	$P_7' \oplus P_8'$	$P_8' \oplus P_9'$	$P_9' \oplus P_1'$
P_7'	P_8'	P_9'	$P_1' \oplus P_2'$	$P_2' \oplus P_3'$	$P_3' \oplus P_4'$

We simulated the decoding probabilities of Type 2 and Type 3 using patterns 1, 2, 3, and 4, where the BER p is fixed to be $p = 0.001$, and the PER q ranges from 0 to 0.5. We again used MAGMA for simulation, and followed the encoding/decoding procedures and the calculation of the decoding probabilities as described in Subsection 4.1. The results are shown in Figures 8 and 9, both of which include the decoding probability of Type 1 as a target for comparison.

These results show that the decoding probability of each pattern is higher than that of Type 1. Furthermore, we can also observe that the performances of patterns 1 and 2 are better than those of patterns 3 and 4 as q gets higher (the maximum difference between Pattern 1 and Pattern 3 in decoding probability is about 13%), which is expected from the results in [1] saying that the number of unencoded packets should be 3 for 4 in this situation (*i.e.*, a situation of the redundancy $r = 2$) to obtain high robustness. At the same time, the power consumption for each pattern is almost the same (the maximum difference is 0.3[mW] in the power

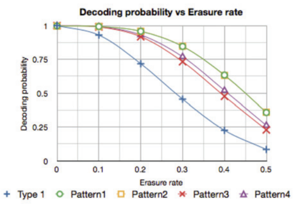

Figure 8: The valuation of patterns for Type 2

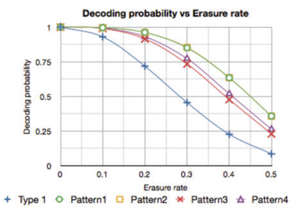

Figure 9: The valuation of patterns for Type 3

consumption), so Patterns 1 and 2 can be good candidates for XOR encoding schemes. We again want to emphasize that there is no big gap between Type 2 and Type 3 when $p = 0.001$.

5. CONCLUSIONS

In this paper, we have first checked that the power consumptions for XOR encodings are much lower than that of the BCH(63,51) encoding, using Xilinx Xpower Analyzer in the FPGA designing tool. Moreover, we found that a concatenation of XOR and BCH encodings (Type 2 or Type 3) provides a better decoding probability compared with that of only BCH encoding (Type 1). In particular, Type 2 gives us a good performance when the BFR p is small, meanwhile it requires a low power consumption which is almost equal to that of Type 1. We also implemented various pattens of XOR encoding schemes for Type 2 and Type 3, and observed that Pattens 1 and 2 (which are based on results in [1]) are good candidates for XOR encoding schemes.

As a future work, we aim to find a better XOR encoding scheme than Patterns 1 and 2 (if it really exists). Moreover, we plan to estimate a battery lifetime in an actual device using WBANs, together with other power consumptions required for the device (such as power for modulation or transmission). Furthermore, we also want to examine the decoding probabilities under a real environment using devices mounted with proposed encodings.

6. REFERENCES

[1] E. Byrne, A. Manada, S. Marinkovic, E. M. Popovici, " A graph theoretical approach for network coding in wireless body area networks, " Proceeding of the IEEE International Symposium on Information Theory(ISIT2011), pp. 405-409

[2] LAN/MAN Standards Committee, "IEEE Standard for Local and metropolitan area networks Part15.6: Wireless Body Area Networks," http://standards.ieee.org/getieee802/download/802.15.6-2012.pdf

[3] S. Marinkovic and E. Popovici, " Network Coding for Efficient Error Recovery in Wireless Sensor Networks for Medical Applications, " 2009 First Int. Conf. on Emerging Network Intelligence, pp. 15-20, 2009

[4] The Computational Algebra Group, School of Mathematics and Statistics, University of Sydney, "MAGMA Calculator," http://magma.maths.usyd.edu.au/calc/