A Secure Phase-Encrypted IEEE 802.15.4 Transceiver Design

Ajay Kumar Nain, Jagadish Bandaru,
Mohammed Abdullah Zubair, and
Rajalakshmi Pachamuthu, *Member, IEEE*

Abstract—With the proliferation of Internet of Things (IoT), the IEEE 802.15.4 physical layer is becoming increasingly popular due to its low power consumption. However, secure data communication over the network is a challenging issue because vulnerabilities in the existing security primitives lead to several attacks. The mitigation of these attacks separately adds significant computing burden on the legitimate node. In this paper, we propose a secure IEEE 802.15.4 transceiver design that mitigates multiple attacks simultaneously by using a physical layer encryption approach that reduces the computations at the upper layers. In addition to providing confidentiality and integrity services, the proposed transceiver provides sufficient complexity to various attacks, such as cryptanalysis and traffic analysis attacks. It also significantly improves the lifetime of the node in the presence of a ghost attacker by preventing the legitimate node from processing the bogus messages and hence combats against energy depletion attacks. The simulation results show that a high symbol error rate at the adversary can be achieved using the proposed transceiver without affecting the throughput at the legitimate node. In this paper, we also analyze the hardware complexity by developing an FPGA and ASIC prototype of the proposed transceiver.

Index Terms—Attacks in wireless networks, energy depletion attack, hardware encryption, IEEE 802.15.4 transceiver, physical layer security, physical layer encryption, phase encryption, security in internet of things, traffic analysis attack

1 Introduction

Internet of Things plays an important role in recent wireless network applications, such as smart cities, e-health and so forth. IEEE 802.15.4 is one of the most popular protocols for these applications due to its low power consumption [1]. However, secure data transfer using devices based on this technology is a challenging aspect due to various constraints, such as limited processing power availability, limited storage memory and so forth [2]. Many techniques have been proposed in the literature to provide security services at different layers of the protocol stack [3]. These techniques are usually categorized into computational security and information-theoretic security. Computational security includes encryption at the upper layers as well as at the physical layer. Information-theoretic security, on the other hand, is provided by physical layer security (PLS) methods at the physical layer.

Most of the security methods are provided by upper layer encryption schemes. However, physical layer encryption (PLE) is also gaining attention since recent past. Upper layer encryption schemes are implemented at different layers having associated benefits and limitations. For instance, end-to-end encryption occurs in the application layer and is system independent. Although computational complexity and latency are not increased by end-to-end encryption, it does not secure the network from any threats, such as denial of service (DoS), energy depletion, traffic analysis, and network flooding attacks, among others. Secure sockets layer (SSL)/transport layer security (TLS) and Internet protocol security (IPSEC) provide security at the transport layer and network layer, respectively [4], [5]. These techniques provide the network with resistance against a few attacks but at the same time introduce overhead on the header field in the transport layer and network layer, respectively. This overhead causes a major issue for communication over low-power devices due to a small maximum transferable unit (MTU) size (128 bytes in 802.15.4). All the upper layer security primitives are implemented in software, and the strength of the security service depends on the complexity of the underlying encryption algorithm. This type of security can be broken by a competently stronger system in the case of constrained devices.

In the PLS approaches, the characteristics of the communication channel and underlying modulation schemes are utilized to provide security. Rather than being reliant on computational complexity like upper layer encryption, these techniques provide information-theoretic security that cannot be broken by computing power [6]. Multiple PLS techniques have been proposed in the literature, such as artificial noise-added security, security-oriented beam-forming and diversity-assisted security [7], [8]. However, there are many challenges, such as unavailable eavesdropper's channel state information (CSI), complex computation, excessive power consumption due to the requirement of co-operative relays or MIMO systems and so forth [9], that prevent the use of the PLS scheme in protecting commercial wireless systems.

The PLE methods modify the data modulation and are dependent on the modulation schemes unlike upper layer encryption. However, encryption can also be performed prior to the modulation process without changing the modulation schemes, but this method uses XOR encryption, in which an exclusive OR (XOR) operation is performed between the message bits and the key bits generated by a key-stream generator [10]. In terms of implementation, it is hardware efficient, but its security strength solely depends on the underlying algorithm used for key-stream generation and does not provide any additional strength. Conversely, PLE schemes can provide high decoding error in the ciphertext itself at the adversary and thus provide additional strength to the underlying encryption algorithm [11]. PLE approaches are also computationally secure, but unlike upper layer encryption approaches, they provide additional strength to the underlying algorithm because they hinder the adversary in receiving the ciphertext itself. Because PLE schemes perform encryption during the modulation process rather than performing it on the incoming data bits directly, these schemes are modulation specific and require special attention for each wireless technology.

In the case of 802.15.4, security services are provided through a medium access control (MAC) layer package that offers basic services such as confidentiality, integrity and so forth [12]. These services are achieved at the cost of computing energy that is far from ignorable [13], [14]. By investigating some potential flaws in these services, new attacks have been presented [15], [16], [17]. Different methods have been proposed to mitigate different attacks at the cost of additional computing power, but all the methods have not been concurrently studied and adopted by 802.15.4. However, for secret data transmission, some steganography methods have also been proposed for 802.15.4 to create a covert channel along with the main channel [18], but these methods suffer the drawbacks of a low data rate over the covert channel and depend on the primary data transmission.

In this work, we propose a PLE scheme for IEEE 802.15.4. To the best of our knowledge, we are the first to implement and analyze PLE for 802.15.4. In the following subsections, we discuss the existing PLE approaches with associated problems followed by the contributions of this work.

- *The authors are with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad, Telangana 502285, India.*
E-mail: {ee14resch11001, ee15resch02010, ee14mtech01003, raji}@iith.ac.in.

Manuscript received 22 Oct. 2016; revised 17 Jan. 2017; accepted 10 Feb. 2017. Date of publication 21 Feb. 2017; date of current version 17 July 2017.
Recommended for acceptance by W. Liu.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TC.2017.2672752

1.1 Related Works

The motivation for providing security at the physical layer lies in the fact that it has the lowest impact on the network and offers low latency without introducing any overhead [19]. Various PLE schemes have been proposed in the literature, although most of these schemes are for securing orthogonal frequency division multiplexing (OFDM) systems. Some techniques are implemented by scrambling the constellation symbols [20], [21], whereas in [22] and [23], PLE is achieved by adding a small amount of random noise to each constellation symbol. A few methods [11], [24] have proposed the encryption of training symbols along with data symbols to combat traffic analysis attacks, which have not been mitigated by other methods. The aforementioned approaches provide security for OFDM systems, but these cannot be applied to IEEE 802.15.4 due to their unsuitability with its devices.

Recently, Huo et al. proposed a method called phase encryption for combating against traffic analysis attacks [25]. They compared the phase encryption with XOR encryption and generalized the phase encryption for various modulation schemes, such as BPSK, QPSK, and QAM, among others [26]. However, its implementation along with the key-stream generation has not been well studied and implemented for IEEE 802.15.4. Moreover, security analyses against various attacks, such as energy depletion and traffic analysis attacks, have not been performed for 802.15.4 with PLE schemes. This work focuses on the phase encryption for 802.15.4 with an extensive analysis of the different attacks.

1.2 Contributions of this Paper

The contributions of this paper include the following:

1) We propose a secure IEEE 802.15.4 transceiver with the PLE approach using physical layer phase encryption, which reduces the computation at the upper layers aiming toward energy savings.
2) Analysis of the proposed system in terms of security services and capability of combating against attacks such as brute force search, cryptanalysis, traffic analysis and resource depletion attacks.
3) Performance comparison of the proposed transceiver with standard transceivers reported in the literature by considering security strength, power consumption and symbol error rate as the key performance metrics.
4) Comparison of the message reception at the legitimate and adversary receivers.
5) Implementation of the proposed system in ASIC using UMC 0.18 μ m CMOS technology and FPGA prototyping for hardware complexity analysis.

1.3 Organization of the Paper

The remainder of this paper is organized as follows. Section 2 describes the necessary background related to the work, followed by the requirements for the system. The proposed system design is described in Section 3, followed by the implementation of the system in Section 4. The performance of the system is analyzed through various performance metrics in Section 5. Section 6 concludes the paper with future directions for research.

2 Background and System Requirements

In this section, we describe the phase encryption that is used in the proposed system for security and the system requirements for implementing phase encryption.

2.1 Phase Encryption

In phase encryption, the phase of the modulated symbol is varied according to a key stream whose size depends on the underlying modulation scheme. In IEEE 802.15.4, each modulated symbol contains 2 bits of the message and is of the form (I, Q) where I and Q takes the value from the set $\{1, -1\}$. Therefore, the key stream's I and Q components take values from the binary set $\{1, -1\}$, and the ciphertext is generated by multiplying the respective components of the key stream and modulated symbols.

If k_i, d_i, and c_i are the i th sample of the key stream, modulated symbol and ciphertext, respectively, then the ciphertext generation can be explained by

$$c_i = a_i \times Re\{d_i\} + jb_i \times Im\{d_i\}, \tag{1}$$

where $a_i + jb_i = k_i$, and a_i, b_i are the in-phase and quadrature-phase components of the key stream, respectively.

2.2 System Requirements

The specification of the IEEE 802.15.4 PHY layer supports a data rate of up to 250 Kbps. The modulated symbol rate for this data rate is 1 Mbps as bit-to-symbol encodes k = 4 bits into 2^k = 16 symbols and each symbol is mapped to a 32-bit-long chip sequence. To perform encryption after the modulation, there is a need for a key stream at the required rate of 1 Mbps. The stream cipher is used to generate the key stream, and the reasons for choosing a stream cipher rather than a block cipher are as follows:

- Block ciphers have a complex architecture that requires a large chip area and high power consumption. These may not be suitable for constrained devices at the mentioned data rate.
- In block ciphers, there is no one-to-one relationship between individual bits in plaintext and ciphertext as in stream ciphers. Even a single bit error in the ciphertext introduced in the channel due to noise will change the plaintext dramatically [27]. This reflects the unsuitability of block ciphers when error-correcting codes are applied prior to encryption.

We used the RC4 stream cipher for encryption and key-stream generation purposes due to its simplicity and suitability for low-power devices [28]. However, some biases have been found in RC4 that make it insecure [29], but for the proposed system, RC4 provides sufficient complexity to perform the cryptanalysis as the adversary faces additional difficulty in frame synchronization and receives the ciphertext with a high error rate.

3 System Design

The block diagram of the proposed transceiver is presented in Fig. 1. Based on functionality, it can be categorized into three main units: transmitter, receiver and key-stream generator. First, we discuss the key-stream generator, which is common for transmitter and receiver operation. Later, we briefly describe the transmitter and receiver units.

3.1 Key-Stream Generator

To generate the key stream, we use the RC4 stream cipher with hardware implementation using loop unrolling [28]. We have modified the hardware implementation proposed by S. Gupta et al. to make it more suitable for phase encryption [28]. The reason for choosing hardware rather than software is due to the unsuitability of software implementation in the transceiver. The hardware implementation of RC4 with loop unrolling provides the fastest results with minimum latency. In addition, through the loop unrolling design, we obtain two bytes of the key stream on every alternate clock, where one can be used for the real part of the key stream and the other for the imaginary part. Thus, both parts of the key streams can be generated at the same time without any lag between the two, which is the requirement of the system.

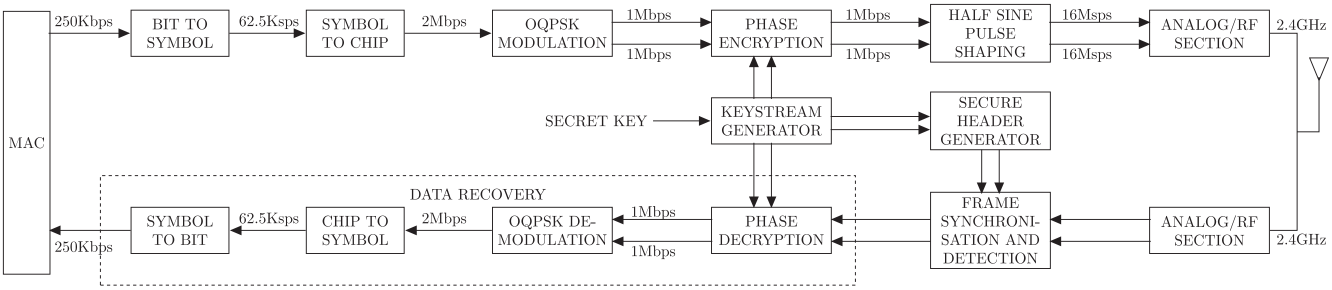


Fig. 1. Block diagram of proposed transceiver.

In the RC4 algorithm, a secret key s_k is used to scramble the permutation of an array S and to generate an arbitrary number of pseudo-random key-stream bytes. There are two components in RC4, namely, key scheduling algorithm (KSA) and pseudo-random generator algorithm (PRGA). The KSA performs an initial permutation on S on the basis of an array K, where K is the repeated version of s_k of length 256. The PRGA uses this pseudo-random permutation to generate the key stream. The complete description of the RC4 algorithm and its loop unrolling architecture can be found in [28].

In the loop unrolling architecture, a consecutive pair of cycles is combined in a particular fashion such that the functionality of two consecutive loops is completed in a single loop. According to the authors in [28], the operations of the two consecutive cycles of the RC4 algorithm are performed simultaneously, and its functionality can be reduced to circuits as mentioned in Table 1. *Circuit1* and *Circuit2* are used to increment i_1, i_2 and j_1, j_2, respectively; *Circuit3* is for swapping; and *Circuit4* will calculate the output Z_1 and Z_2. These circuits can be found in [28], except for *Circuit2*, which has been modified by us to make it reusable and less complex. Because both KSA and PRGA need not run at the same time, the same circuitry should be utilized for both. The modified *circuit 2* is shown in Fig. 2.

The complete architecture of the key-stream generator is depicted in Fig. 3. The clock selector circuit is for selecting the clock from the two clocks CLK_{system} and $CLK_{derived}$. Here, CLK_{system} is 16 MHz, and $CLK_{derived}$ is the required clock rate such that the output bit rate of the key stream matches the modulated data rate. This is to ensure that KSA completes in the minimum possible time, while during PRGA, the output of the key stream is 1 Mbps. Two control signals KSA_{en} and $PRGA_{en}$ are used for controlling the execution timing of the two algorithms because KSA has to complete its operations prior to the start of PRGA. CLK_{system} and $CLK_{derived}$ should be given at the trailing edge of KSA_{en} and $PRGA_{en}$, respectively, while no clock should be given to the KSA and PRGA in the absence of a valid incoming frame to reduce the power consumption.

The task of the clock scheduler block is to generate clocks for all the circuits given in Table 1 because these circuits require different clocks to run at different time intervals. *Circuit1*, *Circuit2* and *Circuit4* run on the trailing edge of odd cycles of ϕ, whereas *Circuit3* runs on the trailing edge of even cycles of ϕ. Here, ϕ is the clock to the clock scheduler. *Circuit1, Circuit2* and *Circuit3* should be ON for both KSA and PRGA, whereas *Circuit4* and the serializer are only needed for PRGA. The behavior model of the clock selector and clock scheduler is described in Algorithm 1. The purpose of the serializer is to generate serial bit streams driven by the outputs Z_1 and Z_2 of *Circuit4* during PRGA execution. In this way, the key stream for the i^{th} modulated symbol is $[KS_i, KS_q]$.

TABLE 1
Loop Unrolling of RC4

Task	First Loop	Second Loop	Corres. Circuit
Increment of i	$i_1 \leftarrow i_0 + 1$	$i_2 \leftarrow i_0 + 2$	*Circuit1*
Increment of j	$j_1 \leftarrow j_0 + S_0[i_1]$	$j_2 \leftarrow j_0 + S_0[i_1] + S_1[i_2]$	*Circuit2*
Swap	$S_0[i_1] \leftrightarrow S_0[j_1]$	$S_1[i_2] \leftrightarrow S_1[j_2]$	*Circuit3*
Output	$Z_1 \leftarrow S_1[S_0[i_1] + S_0[j_1]]$	$Z_1 \leftarrow S_2[S_1[i_2] + S_1[j_2]]$	*Circuit4*

Algorithm 1. Clock Scheduling Algorithm

```
procedure CSA ($CLK_{system}, CLK_{derived}, KSA_{en}, PRGA_{en}$)
    Initialize $KSA_{done} \leftarrow 0$
    if $PRGA_{en}$ then
        $\phi \leftarrow CLK_{derived}$
        $CLK_1 \leftarrow \frac{\phi}{8}$ and $CLK_3 \leftarrow \sim CLK_1$
    else if $KSA_{en}$ then
        $\phi \leftarrow CLK_{system}$
        if $KSA_{done} = 0$ then
            $CLK_1 \leftarrow \frac{\phi}{2}$ and $CLK_3 \leftarrow \sim CLK_1$
        else
            $CLK_1 \leftarrow 0$ and $CLK_3 \leftarrow 0$
        end if
    else
        $\phi \leftarrow 0$
    end if
end procedure
```

3.2 Transmitter

In addition to the standard 802.15.4 transmitter [30], the proposed transmitter has key generation and phase encryption modules. The data coming from the MAC layer are first mapped to symbols, and then each symbol is mapped to corresponding chip sequences. The modulated data along with the key stream are fed to the phase encryption block, which rotates the phase accordingly. When the transceiver receives the request to send the data from the upper layer, it sends KSA_{en} signal to the key-stream generator block.

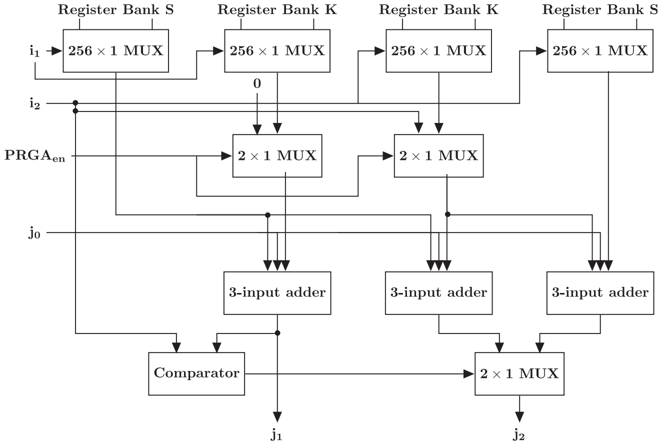


Fig. 2. Circuit 2 to compute j_1 and j_2.

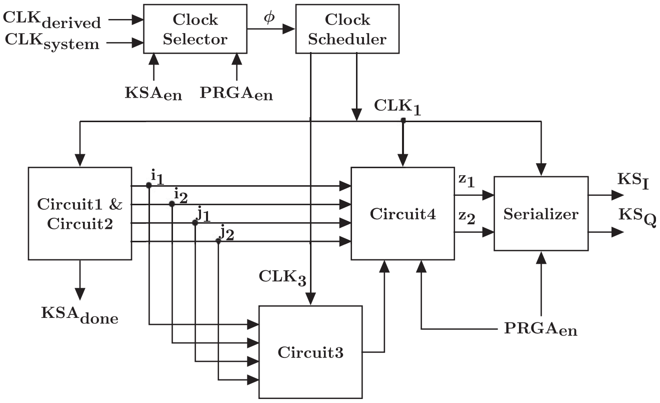


Fig. 3. Block diagram of key-stream generator.

After completion of the KSA, it will set $PRGA_{en}$ to high and start the transmission of the data. During KSA, the key-stream generator operates with a clock frequency of 16 MHz, and KSA requires 256 clock cycles for completion. So, the latency caused by encryption during transmission is 16 microseconds.

3.3 Receiver

In Fig. 1, the receiver unit has two main blocks: one is frame synchronization and detection, and the other is the data recovery block. The data recovery block is activated only if a valid frame is detected by the former block. The functionality of both blocks is explained below:

3.3.1 Frame Synchronization and Detection

After detection of valid energy in the channel, this block starts its operation. There are two main tasks for this block: to perform the correlation of the incoming samples with a known header and to detect a peak in the correlation output for finding the location of the header. For frame synchronization, the cross-correlation method is used [31], [32]. The receiver should have prior knowledge of the corresponding header, which is similarly encrypted after modulation as it is in the transmitter. This is generated by the secure header generator block.

After completion of the correlation, KSA_{en} needs to be made high such that KSA can complete its cycles before the detection of a peak. Upon the detection of the peak, if its magnitude is found to be above a predefined threshold, the $PRGA_{en}$ is set to high to run the PRGA. At the same time, the data recovery block is also activated. In this way, the latency introduced in the receiver is due to frame synchronization and detection only. However, if the peak is found below the threshold, KSA_{en} and $PRGA_{en}$ are set to zero such that the key generation block is ready for the next frame.

3.3.2 Data Recovery

In the data recovery section, the phase of each sample is first shifted back to the original phase in the phase decryption block according to the key stream. The decrypted data are demodulated, de-spread and converted to bits in their respective blocks, and then data bits are sent to the MAC layer.

4 System Implementation

We have developed the Verilog-RTL for the proposed transceiver. The complete design is simulated, synthesized, and validated using Xilinx Kintex-7 FPGA KC705 and Xilinx Chipscope Pro Analyzer. The latency of transmitting a frame is 16 microseconds, whereas the latency of receiving a frame is 128 microseconds (this latency includes the frame synchronization and detection latency). The result obtained is summarized in Table 2.

TABLE 2
FPGA Implementation Results

Designs	Power in mw			Hardware Usage	
	Static	Dynamic	Total	Slices	LUTs
RC4	163.43	74.89	238.32	2,476	7,064
Proposed Transceiver	220.06	141.67	361.73	6,507	15,954

The proposed system has also been implemented in an application-specific integrated circuit (ASIC) using the UMC 0.18 μm CMOS technology. The clock frequency used in the implementation is 16 MHZ. The gate-level synthesis is conducted using the Synopsys Design Compiler. The synthesis result of the proposed transceiver provides a total number of gate counts of 1,32,046, whereas the count for the standard transceiver is 1,04,477. This results in a 26 percent higher number of gate counts, which is a reasonable amount considering the security benefits. The resource overhead is primarily due to the RC4 stream cipher, which consumes 0.681 mW of power, while the proposed transceiver consumes 3.931 mW of power. The phase encryption/decryption block is quite simple in terms of resources because it only consists of two multipliers.

5 Performance Analysis

In the first two subsections, the performance of the proposed transceiver is compared with the standard receiver in terms of frame synchronization and detection analysis and de-spreading analysis, respectively. The following two subsections discuss the effect of the proposed PLE on the symbol error rate (SER) and power consumption, respectively. Finally, the security strength of the proposed transceiver is analyzed in terms of provided services and mitigated attacks.

5.1 Frame Synchronization and Detection Analysis

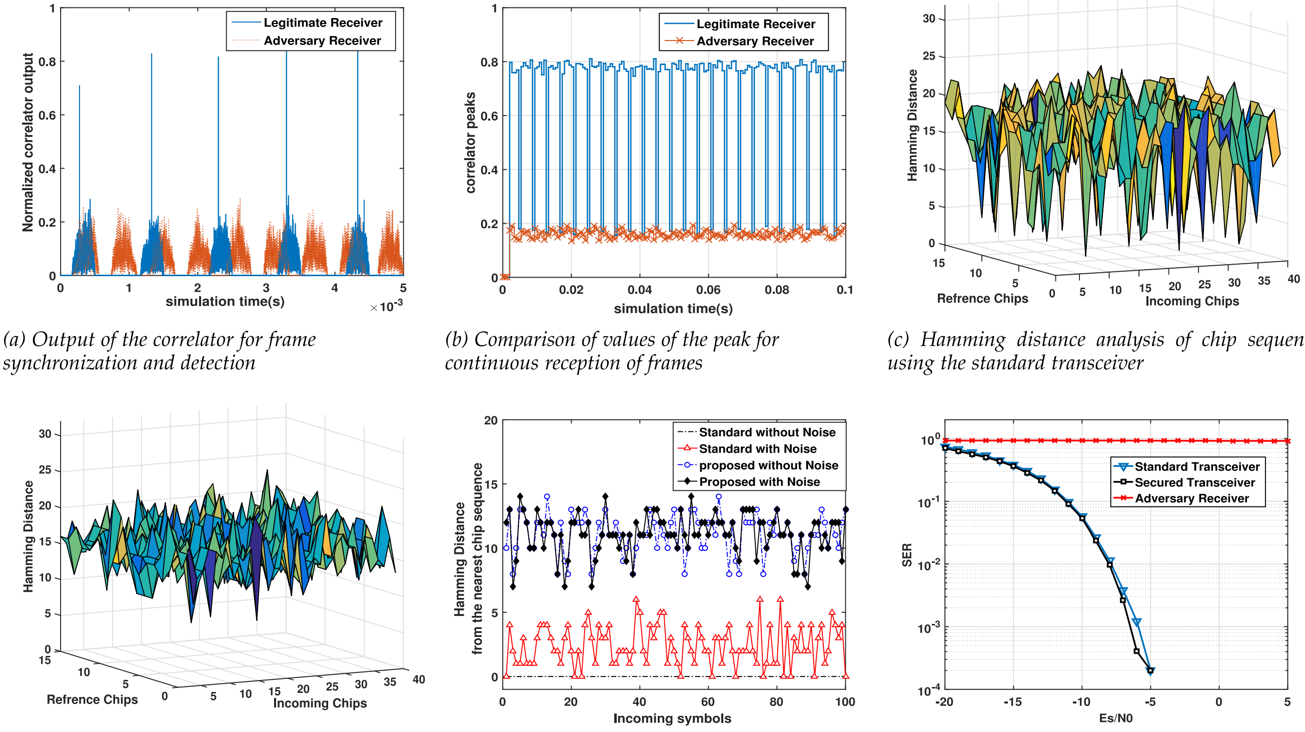
Using the proposed system, the adversary faces difficulty in the detection and synchronization of the frames. Fig. 4a shows the normalized output of the correlator for 5 frames at an SNR of 3 dB. We can observe that the legitimate receiver has very subtle peaks, whereas the adversarial receiver could not obtain such a peak. For better clarity, the normalized calculated peak for the continuous reception of frames is plotted in Fig. 4b. After detecting the energy in the channel, it starts correlating and computes the peak of the correlation for that frame. This value is the same for the time until the frame reception is completed. Then, it is again calculated and updated. Smaller values at the repetitive intervals indicate that there is no valid frame in the channel during that period of time.

From both Figs. 4a and 4b, it can be inferred that the average normalized amplitude of the peak is found to be 0.8 and the adversary output is averaged at 0.2. For a successful reception of data, the threshold can be set anywhere between 0.4 to 0.6. Such a threshold can help in the detection of valid frames at the legitimate receiver with a guaranteed amount of accuracy. The correlator then extracts the payload from the frame, and the receiver starts processing the extracted data.

However, we can observe from the figures that the adversary is unable to obtain a subtle peak above the threshold and hence fails to detect the valid frames. Moreover, even if the adversary attempts to decode the message by having a low threshold, it is difficult to perform the cryptanalysis as there is no guarantee that it has received the valid ciphertext exactly.

5.2 De-Spreading Analysis

In the 802.15.4 standard, a set of sixteen pseudo-noise (PN) sequences is used to spread a 4-bit symbol into 32-bit chip sequences.



(a) Output of the correlator for frame synchronization and detection

(b) Comparison of values of the peak for continuous reception of frames

(c) Hamming distance analysis of chip sequences using the standard transceiver

(d) Hamming distance analysis of chip sequences at the adversary using the proposed transceiver

(e) Performance of the de-spreading in the presence of noise

(f) Comparison of SER

Fig. 4. Results of the proposed transceiver compared with the standard transceiver.

These PN sequences are quasi-orthogonal, and the Hamming distance between any two chip sequences lies between 12 and 20. In this way, there is a capability to tolerate six chip errors without a symbol error. With the use of the proposed method, this property is destroyed, resulting in more error at the adversary even in the absence of noise.

In the de-spreading block, the incoming 32-bit chips are correlated with the known chips, and the chip sequence that provides the maximum correlation is selected for symbol mapping. For the standard receiver, the incoming chip has a very small Hamming distance from one sequence while having large Hamming distances from the others. However, using the proposed method, the Hamming distance at the adversary is almost the same for all the known sequences, resulting in more ambiguity in the result. This can be observed from Figs. 4c and 4d, where 40 incoming sequences are correlated with the known sequences.

To demonstrate the ambiguity in taking a decision during the despreading at the receiver, we have observed one hundred incoming symbols of the frame for both of the transceivers and analyzed the Hamming distance of each symbol from the nearest symbol. As shown in Fig. 4e, the effect of noise is visible for the standard receiver as the minimum Hamming distance from any chip sequence is increased. However, using the proposed method, the adversary has faced almost the same ambiguity in the absence of noise.

5.3 SER Analysis

The performance of the proposed system is compared with the standard transceiver in a noisy environment in terms of SER. As shown in Fig. 4f, the proposed transceiver does not degrade the SER performance at the legitimate receiver. However, the SER at the adversary receiver is very high for all densities of noise.

5.4 Power Analysis

If the proposed transceiver is used, we can avoid the MAC layer encryption approaches, which consume considerable computing power when encrypting a frame and can be found in [17]. The proposed transceiver reduces this considerable computing burden at the cost of a small amount of additional communication energy. We discuss an analytical model for the lifetime of the node for this purpose in two cases. In the first case, we compare its energy consumption with that of the standard transceiver in the presence of an attacker in the channel. In the second case, its energy consumption is analyzed in two different scenarios depending upon whether crafted or genuine packets are present in the channel.

5.4.1 Lifetime Comparison with Standard Transceiver

We consider two nodes for comparison purposes: the first node N^P employs the proposed PLE method, whereas the second node N^U has the standard transceiver with advanced encryption standard (AES) as the upper layer encryption scheme. We consider a counter with the CBC-MAC mode of operation using AES with a 128-bit key (AES-CCM-128), which is the most secure approach at the MAC layer for 802.15.4. We assume that both nodes have the same central processing unit (CPU). We consider that the transceiver works in a duty cycle mode, which is a valid assumption and also a standard assumption made by multiple studies [17]. Let the duty cycle of the node be $\frac{\tau}{T}$, where τ is the active period and T is the length of the cycle. We assume that the attacker is sending a crafted packet at a fixed rate and that n_p is the number of crafted packets received by the receiver in the active period. Let P^a_{cpu}, P^i_{cpu}, P_{rx} and P_{srx} denote the power consumption by the active CPU, idle CPU, standard transceiver and proposed transceiver, respectively. The CPU remains idle during the reception of messages by the transceiver, whereas it is active during the encryption/decryption of upper layer security methods. Let T_{srx} and T_{rx} denote the times to receive a bogus message by the proposed and standard transceivers, respectively. Let T_a be the total time to receive, process and decrypt a packet; for the proposed transceiver, $T_a = T_{srx}$ as the packet is decrypted in the transceiver only. However, for the standard transceiver, it is

$T_a = T_{rx} + T_{dec}$, where T_{dec} is time taken by the CPU to decrypt the packet.

If we neglect the energy consumption during sleep, i.e., when the node is not involved in communication, the total energy consumed per cycle E_c can be considered as the sum of the communication cost E_{comm} and computational cost E_{comp}, where E_{comm} and E_{comp} are the energy consumed per cycle by its transceiver and CPU, respectively. E_{comm} for nodes N^P and N^U are given below

$$E^P_{comm} = \tau P_{srx} \tag{2}$$

$$E^U_{comm} = \begin{cases} n_p T_a P_{rx} & \text{if } n_p T_a \geq \tau, \\ \tau P_{rx} & \text{otherwise.} \end{cases} \tag{3}$$

The E^U_{comm} depends on the number of frames detected in the active period, whereas E^P_{comm} depends only on the duty cycle. This is because the proposed transceiver decrypts the data during the demodulation itself, and hence, extra computation and latency due to decryption in the CPU can be ignored. In this way, the CPU can also be considered idle with the proposed transceiver in the active period, i.e., the CPU in node N^p is always idle. The computation costs for nodes N^P and N^U are given below

$$E^P_{comp} = \tau P^i_{cpu} \tag{4}$$

$$E^U_{comp} = n_p(T_{dec} P^a_{cpu} + T_{rx} P^i_{cpu}). \tag{5}$$

Because the lifetime of the node is inversely proportional to the energy consumed by the node per cycle, the ratio of lifetime of nodes N^P and N^U can be expressed as

$$\begin{aligned} \frac{L_P}{L_U} &= \frac{E^U_{comp} + E^U_{comm}}{E^P_{comp} + E^P_{comm}} = \frac{n_p(T_{dec} P^a_{cpu} + T_{rx} P^i_{cpu} + T_a P_{rx})}{\tau P_{srx} + \tau P^i_{cpu}} \\ &= \frac{n_p T_a}{\tau} \frac{\left(\frac{T_{dec}}{T_a}(P^a_{cpu} + P_{rx}) + \frac{T_{rx}}{T_a}(P^i_{cpu} + P_{rx})\right)}{P_{srx} + P^i_{cpu}} \\ &= \frac{n_p T_a}{\tau}\left(\frac{P^i_{cpu} + P_{rx}}{P^i_{cpu} + P_{srx}} + \frac{T_{dec}}{T_a}\frac{P^a_{cpu} - P^i_{cpu}}{P^i_{cpu} + P_{srx}}\right). \end{aligned} \tag{6}$$

As the power consumption is primarily dominated by the RF section [33], we consider that the power consumption by the proposed secure transceiver is approximately equal to the standard transceiver since the only change is in base-band resources whose power consumption is relatively minimal. So, $P_{srx} \approx P_{rx}$. We can use $T_{dec} \gg T_{rx}$ and $P^a_{cpu} > P^i_{cpu}$, which is a valid assumption and can be referenced in [17]. In the case of a ghost attack, when $n_p T_a$ is considerably larger than τ, it can be inferred from Equation (6) that the lifetime of the node with the proposed transceiver is significantly larger than that of the node with the standard transceiver.

5.4.2 Lifetime Comparison with Valid and Bogus Packets

In this case, we considered the dynamic power of the proposed transceiver for detailed analysis. Let P_{rf} be the power consumed by the transceiver when it waits for energy in the channel. After the successful detection of energy, it starts synchronization for duration T_{fs} and lets it consume P_{fs} energy during that interval. If it detects a valid frame, the data recovery block is activated. Let T_{dr} and P_{dr} be the time duration and power consumed in the reception of the valid frame, respectively.

For the analytical model, the power consumed by the node can be categorized into 3 scenarios: no energy in the channel, only genuine packets in the channel and only bogus packets in the channel. The energy consumed by the CPU is the same for all the cases. If the energy consumed by the transceiver in all three cases are E^o_{comm}, E^g_{comm} and E^b_{comm}, respectively, then the following equations describe these energy consumptions

$$\begin{aligned} E^o_{comm} &= \tau P_{rf} \\ E^g_{comm} &= n_p T_{fs} P_{fs} + n_p T_{dr} P_{dr} + (\tau - n_p T_{fs} - n_p T_{dr}) P_{rf} \\ E^b_{comm} &= n_p T_{fs} P_{fs} + (\tau - n_p T_{fs}) P_{rf}. \end{aligned}$$

If the lifetimes of the node in the three cases are L_o, L_g and L_b, respectively, then the reduction in lifetime due to the reception of genuine packets is

$$\begin{aligned} \frac{L_o}{L_g} &= \frac{\tau P^i_{cpu} + E^g_{comm}}{\tau P^i_{cpu} + E^o_{comm}} \\ &= 1 + \frac{n_p T_{fs}}{\tau}\frac{P_{fs} - P_{rf}}{P^i_{cpu} + P_{rf}} + \frac{n_p T_{dr}}{\tau}\frac{P_{dr} - P_{rf}}{P^i_{cpu} + P_{rf}}. \end{aligned} \tag{7}$$

Similarly, the reduction in lifetime when only bogus packets are present in the channel is

$$\frac{L_o}{L_b} = \frac{\tau P^i_{cpu} + E^g_{comm}}{\tau P^i_{cpu} + E^b_{comm}} = 1 + \frac{n_p T_{fs}}{\tau}\frac{P_{fs} - P_{rf}}{P^i_{cpu} + P_{rf}}. \tag{8}$$

From Equations (7) and (8), we can observe that if the number of packets per unit time is the same for both cases, then the reduction in lifetime due to legitimate packets is more than that due to bogus packets. In other words, the proposed receiver does not waste considerable resources on the processing of the bogus messages.

5.5 Security Analysis

5.5.1 Brute Force Search Attack

In IEEE 802.15.4, the physical layer preamble consists of 32 zeros. It encodes $k = 4$ bits into $2^k = 16$ symbols, and each symbol is 32 bits long. In this way, the preamble consists of 8 symbols or 256 chips. After modulation, it will convert to 128 complex samples. During phase modulation, the phase of each sample is rotated with one of the four phases of QPSK according to the key. Thus, there are 4^{128} possible combinations of the key for one preamble, which can be considered sufficiently secure in today's standard [34]. With such large number of combinations, it is quite difficult for the adversary to detect the valid frame and perform brute force search attack.

5.5.2 Confidentiality

Even if the adversary is successful in obtaining the valid frame, confidentiality is provided by the phase encryption as correct data cannot be recovered without a correct key stream. Although we have used the RC4 stream cipher to generate the key stream and several biases have been found in RC4 making it insecure [29], finding the key by utilizing these vulnerabilities would require a large number of encrypted texts. In our case, it is difficult to perform the cryptanalysis as there is no guarantee that the adversary correctly receives the cipher text.

5.5.3 Integrity

Because all the upper layer headers are encrypted, including their check-sums, the proposed system can identify whether the source address or the data have changed in the medium, thus providing the integrity.

5.5.4 Authentication, Availability and Data Freshness

We assumed that these services are provided by the upper layers. For availability, the node maintains an access control list (ACL) to prevent unauthorized nodes from participating in the network. For data freshness, a 32-bit counter is used at the MAC layer such that the adversary can conduct a replay attack only after 2^{32} frames, which is considered cryptographically secure in practice. In addition, the proposed system provides more strength to these services by encrypting the security header rather than sending it as a plain text as in the existing security primitives.

5.5.5 Traffic Analysis Attack

Using the proposed system, the adversary cannot detect the timing of the data transmission exactly because it is difficult for the adversary to detect and synchronize the frames. Hence, the proposed system provides sufficient resistivity against traffic analysis attacks.

5.5.6 Energy Depletion Attacks

Ghost-in-ZigBee is a resource depletion attack that leverages the underlying vulnerabilities of existing security suites. This attack not only depletes the energy of nodes at a faster rate but also facilitates a variety of threats, such as denial of service and replay attacks [17]. In these attacks, bogus messages are transmitted by the attacker, having intention to deplete the energy of the legitimate nodes. In the proposed method, the bogus message will be dropped as soon as possible because it will not be detected as a valid frame. Hence, energy will not be wasted on receiving and processing these bogus messages. We have determined that the node with the proposed transceiver does not waste considerable energy on bogus messages and provides higher lifetime to the network.

6 Conclusion and Future Directions

We have proposed a secure IEEE 802.15.4 transceiver architecture along with FPGA and ASIC implementation. We have analyzed the performance through comparison with the standard transceiver. The proposed system provides a high error rate at the adversary without affecting SER performance at the legitimate receiver. In addition to confidentiality, it also offers protection from cryptanalysis and traffic analysis attacks by increasing the complexity for the attacker. It provides a significant improvement in the lifetime of the node in the presence of energy depletion attacks. The proposed physical layer encryption technique introduces a minimal latency of approximately 16 microseconds and has the lowest impact on the network. Future work can be to extend the proposed approach for a multi-radio secure transceiver that has ZigBee, WiFi and Bluetooth on board.

References

[1] A. Al-Fuqaha , M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of things: A survey on enabling technologies, protocols, and applications,” *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, Oct.–Dec. 2015.

[2] J. A. Stankovic, “Research directions for the internet of things,” *IEEE Internet Things J.*, vol. 1, no. 1, pp. 3–9, Feb. 2014.

[3] Y. Zou, J. Zhu, X. Wang, and L. Hanzo, “A survey on wireless security: Technical challenges, recent advances and future trends,” in *Proc. IEEE*, vol. 104, no. 9, pp. 1727–1765, Sept. 2016.

[4] N. Doraswamy and D. Harkins, “IPSec architecture,” in *IPSec: The New Security Standard for The Internet, Intranets, and Virtual Private Networks*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice Hall, 2003, ch. 4, pp. 59–80.

[5] S. Raza, D. Trabalza, and T. Voigt, “6LoWPAN compressed DTLS for CoAP,” in *Proc. IEEE 8th Int. Conf. Distrib. Comput. Sensor Syst.*, 2012, pp. 287–289.

[6] A. Mukherjee, S. A. A. Fakoorian, J. Huang, and A. L. Swindlehurst, “Principles of physical layer security in multiuser wireless networks: A survey,” *IEEE Comm. Surveys Tuts.*, vol. 16, no. 3, pp. 1550–1573, Jul.–Sep. 2014.

[7] D. Goeckel, S. Vasudevan, D. Towsley, S. Adams, Z. Ding, and K. Leung, “Artificial noise generation from cooperative relays for everlasting secrecy in two-hop wireless networks,” *IEEE J. Selected Areas Commun.*, vol. 29, no. 10, pp. 2067–2076, Oct. 2011.

[8] A. Mukherjee and A. Swindlehurst, “Robust beamforming for security in MIMO wiretap channels with imperfect CSI,” *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 351–361, Jan. 2011.

[9] W. Trappe, “The challenges facing physical layer security,” *IEEE Commun. Mag.*, vol. 53, no. 6, pp. 16–20, Jun. 2015.

[10] A. Zqute and J. Barros, “Physical-layer encryption with stream ciphers,” in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2008, pp. 106–110.

[11] J. Zhang, A. Marshall, R. Woods, and T. Q. Duong, “Design of an OFDM physical layer encryption scheme,” in *Proc. IEEE Trans. Veh. Technol.*, vol. PP, no. 99, p.1, doi: 10.1109/TVT.2016.2571264.

[12] N. Sastry and D. Wagner, “Security considerations for IEEE 802.15.4 networks,” in *Proc. 3rd ACM Workshop Wireless Secur.*, 2004, pp. 32–42.

[13] Y. Xiao, H. Chen, B. Sun, R. Wang, and S. Sethi, “MAC security and security overhead analysis in the IEEE 802.15.4 wireless sensor networks,” *J. Wireless Commun. Netw.*, vol. 2006, pp. 1–12, May 2006, doi: 10.1155/WCN/2006/93830.

[14] M. Doomun, K. Soyjaudah, and D. Bundhoo, “Energy consumption and computational analysis of Rijndael-AES,” in *Proc. IEEE/IFIP Int. Conf. Central Asia Internet*, Sep. 2007, pp. 1–6.

[15] A. Back, U. Möller, and A. Stiglic, “Traffic analysis attacks and trade-offs in anonymity providing systems,” *Int. Workshop Inform. Hiding*, Springer Berlin Heidelberg, 2001, pp. 245–257, April 2001, doi: 10.1007/3-540-45496-9_18.

[16] E. Y. Vasserman and N. Hopper, “Vampire attacks: Draining life from wireless ad hoc sensor networks,” *IEEE Trans. Mobile Comput.*, vol. 12, no. 2, pp. 318–332, Feb. 2013.

[17] X. Cao, D. M. Shila, Y. Cheng, Z. Yang, Y. Zhou, and J. Chen, “Ghost-in-ZigBee: Energy depletion attack on ZigBee-Based wireless networks,” *IEEE Internet Things J.*, vol. 3, no. 5, pp. 816–829, Oct. 2016.

[18] A. K. Nain and P. Rajalakshmi, “A reliable covert channel over IEEE 802.15.4 using steganography,” in *Proc. IEEE World Forum Internet Things*, Dec. 2016, pp. 711–716.

[19] Y. S. Shiu, S. Y. Chang, H. C. Wu, S. C. H. Huang, and H. H. Chen, “Physical layer security in wireless networks: A tutorial,” *IEEE Wireless Commun.*, vol. 18, no. 2, pp. 66–74, Apr. 2011.

[20] H. Li, X. Wang, and W. Hou, “Secure transmission in OFDM systems by using time domain scrambling,” in *Proc. IEEE 77th Veh. Technol. Conf.*, 2013, pp. 1–5.

[21] L. Zhang, X. Xin, B. Liu, and Y. Wang, “Secure OFDM-PON based on chaos scrambling,” *IEEE Photon. Technol. Lett.*, vol. 23, no. 14, pp. 998–1000, Jul. 2011.

[22] R. Ma, L. Dai, Z. Wang, and J. Wang, “Secure communication in TDS OFDM system using constellation rotation and noise insertion,” *IEEE Trans. Consum. Electron.*, vol. 56, no. 3, pp. 1328–1332, Aug. 2010.

[23] D. Reilly and G. Kanter, “Noise-enhanced encryption for physical layer security in an OFDM radio,” in *Proc. IEEE Radio Wireless Symp.*, Jan. 2009, pp. 344–347.

[24] D. Dzung, “Data encryption on the physical layer of a data transmission system,” U.S. Patent 7752 430B2, Jul. 6, 2010.

[25] F. Huo and G. Gong, “Physical layer phase encryption for combating the traffic analysis attack,” in *Proc. IEEE Int. Symp. Electromagn. Compat.*, Aug. 2014, pp. 604–608.

[26] F. Huo and G. Gong, “XOR encryption versus phase encryption, an in-depth analysis,” *IEEE Trans. Electromagn. Compat.*, vol. 57, no. 4, pp. 903–911, Jan. 2015.

[27] R. E. Smith, “Encrypting volumes,” in *Elementary Information Security*, 2nd ed. Burlington, MA, USA: Johns & Barlett Learning, 2016, ch. 9, sec. 2, pp. 379–387.

[28] S. S. Gupta, A. Chattopadhyay, K. Sinha, S. Maitra, and B. P. Sinha, “High-performance hardware implementation for RC4 stream cipher,” *IEEE Trans. Comput.*, vol. 62, no. 4, pp. 730–743, Apr. 2013.

[29] P. Sepehrdad, S. Vaudenay, and M. Vuagnoux, “Discovery and exploitation of new biases in RC4,” in *Proc. 18th Int. Workshop Selected Areas Cryptography*, 2011, pp. 74–79, doi: 10.1007/978-3-642-19574-7_5.

[30] N. Salman, I. Rasool, and A. H. Kemp, “Overview of the IEEE 802.15.4 standards family for low rate wireless personal area networks,” in *Proc. 7th Int. Symp. Wireless Commun. Syst.*, 2010, pp. 701–705

[31] K.-H. Lin, W.-H. Chiu, and J.-D. Tseng, “Low-complexity architecture of carrier frequency offset estimation and compensation for body area network systems,” *Comput. Math. Appl.*, vol. 64. no. 5, pp. 1400–1408, Sep. 2012.

[32] K.-H. Chen and H.-P. Ma, “A low power ZigBee baseband processor,” in *Proc. Int. SoC Des. Conf.*, 2008, vol. 1, pp. I-40–I-43.

[33] W. Kluge, et al., “A fully integrated 2.4-GHz IEEE 802.15.4-Compliant transceiver for ZigBee^{TM}applications,” *IEEE J. Solid-State Circuits*, vol. 41, no. 12, pp. 2767–2775, Dec. 2006.

[34] W. Stalling, “Classical encryption techniques,” in *Cryptography and Network Security: Principle and Practice*, 5th ed. Englewood Cliffs, NJ, USA: Prentice Hall, 2011, ch. 2, pp. 35–38.