NAIST-IS-DD1661213

Doctoral Dissertation

Lightweight Physical Layer Encryption Methods for IoT Sensor Transceivers

Hoang Dai Long

June 16, 2020

Graduate School of Information Science
Nara Institute of Science and Technology

A Doctoral Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Hoang Dai Long

Thesis Committee:

Professor Yasuhiko Nakashima	(Supervisor)
Professor Minoru Okada	(Co-supervisor)
Associate Professor Takashi Nakada	(Co-supervisor)
Assistant Professor Thi-Hong Tran	(Co-supervisor)
Assistant Professor Renyuan Zhang	(Co-supervisor)

Lightweight Physical Layer Encryption Methods for IoT Sensor Transceivers[*]

Hoang Dai Long

Abstract

Physical layer encryption (PLE) has been recently promoted as a new technique to enhance the security for the Internet of things (IoT) sensor transceivers. However, this technique appears with a big challenge that IoT sensor transceivers have limited power and computational resources to execute encryption tasks. Motivated by this issue, in this study, we propose two physical layer encryption methods. We aim to reduce the hardware complexity and preserve the performances of the system. On this basis, they can be applied for IoT sensor transceivers.

The first method is the sign bit encryption method, which encrypts only sign bit of the modulated symbols. To evaluate the performances of the proposed method, we simulate in Matlab following the IEEE 802.11ah standard. The simulation results show that our proposed method does not degrade the bit error rate (BER) and packet error rate (PER) performances of the system, while the conventional method degrades the performances about 3 dB. The proposed method uses only XOR-operation; thus, it is low complexity.

The second method is a low complexity joint encryption modulation (JEM) method for IoT sensor transceivers. Unlike the sign bit encryption, which only encrypts the signed bit, the JEM method encrypts all data. In this method, we merge the mapper and encryption in one block. Our proposed JEM method obtains the following advantages:1) Unlike the conventional method which reduces the BER and PER performances of the physical transceivers up to 3 dB, our proposed JEM method does preserve these performances; 2) The required hardware resources, ASIC area, and power of the proposed JEM method only increase

[*] Doctoral Dissertation, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD1661213, June 16, 2020.

slightly compared to the conventional mapper; 3) The required ASIC area and power of the proposed JEM method are approximately 40 times less than needed for the conventional encryption method.

Besides, to prove the feasibility of our method in practice, we design the proposed JEM method and decryption in hardware. We also do the hardware implementation of CORDIC based conventional encryption method to compare with our JEM method.

Keywords:

Physical Layer Security (PLS), Physical Layer Encryption (PLE), IoT Sensor Transceivers, High Order Modulation (HOM), Lightweight Stream Cipher, CORDIC Hardware

List of publications

Peer review journal papers (J)

1. Dai Long Hoang, Thi Hong Tran, and Yasuhiko Nakashima,“ A Low Complexity Joint Encryption-Modulation Method for IoT Sensor Transceivers”, MDPI Electronics, 9(4):663, April 2020. [Corresponds to Chapter 4, and 5]

Peer review international conference papers(C)

1. Dai Long Hoang, Thi Hong Tran, and Yasuhiko Nakashima, “Performance Evaluation of 802.11ah Physical Layer Phase Encryption for IoT Applications”, 2018 International Conference on Advanced Technologies for Communications (ATC’18), pp. 84-88, Ho Chi Minh City, Vietnam, October 2018. [Corresponds to Chapter 3]

2. Dai Long Hoang, Thi Hong Tran, and Yasuhiko Nakashima, “Hardware Implementation of CORDIC Based Physical Layer Phase Decryption for IEEE 802.11ah”, The 7th International Conference on Communications and Broadband Networking (ICCBN 2019), pp. 17-21, Nagoya, Japan, April 2019.[Corresponds to 5]

Workshops

1. Dai Long Hoang, Thi Hong Tran, and Yasuhiko Nakashima, “Security-Based Physical Layer Encryption of Wireless Communication 802.11ah for IoT Applications”, International Workshop on Frontiers in Computing Systems and Wireless Communications (FOSCOM), Nara, Japan, March 2018. (Oral Presentation)

2. Thi Hong Tran, Dai Long Hoang, and Yasuhiko Nakashima, “Hardware Design of Low-Cost Low Complexity Wireless Transceiver based on IEEE 802.11ah standard ”, FOSCOM, Nara, Japan, March 2018.

Contents

List of Figures

List of Tables

1 Introduction

This chapter first presents the overview problems of the security for IoT sensor transceivers. Then, the research contributions are included in chapter 1.2. Finally, the rest of this chapter shows the dissertation layout.

1.1 Overview

Nowadays, the concept of the Internet of Things (IoT) becomes very popular and plays a crucial role in numerous applications such as smart city, smart grid, and smart healthcare [1]. It has been demonstrated that IoT applications rely on wireless connections, which is vulnerable to eavesdropping attacks and active jammings [2]. Particularly, in IoT sensor networks with application scenarios such as health monitoring, monitoring sensors for the military, the collected data are private or confidential. Therefore, it is challenging to improve the security of these connections to protect users and data from attackers.

In wireless communication systems, the framework of security is provided in different protocol layers and can be classified into two main approaches, such as information-theoretic security and computational security. The former one is executed by physical layer security (PLS) techniques at the physical (PHY) layer, while the latter is a cryptography approach that performs the encryption and decryption at different layers such as medium access control (MAC), and physical (PHY) [3].

PLS, a well-known security approach at the PHY layer, aims to achieve information-theoretic security by exploiting channel characteristics [4, 5]. PLS methods can secure the system by generating artificial noise [6, 7, 8], creating beamforming [9, 10, 11], or utilizing cooperative relay [12, 13]. Although PLS brings some advantages such as flexible security level configurations, QoS guarantee, and less overhead as compared to the cryptography approach [14], its hardware implementation is complex. It needs high complexity coding and/or perfect/imperfect channel state information of the receiver and eavesdroppers [3]. Therefore, PLS is not a suitable security solution for small size transceivers such as IoT sensors.

In the aspect of cryptography, most of the conventional wireless communication systems implement security encryption/decryption at the upper layers, such as the MAC layer, where attackers are able to use general-purpose microprocessor and memory to collect data for cracking the encryption key. In addition, encryption at upper layers might result in the fact that many parts of transmission data are not encrypted, e.g., the MAC header and PHY header. Such information is thus vulnerable to eavesdroppers. Researchers have recently gained attention to move the encryption to the physical layer for improving the security of the wireless connections [15, 16, 17, 18, 19]. For instance, encryption schemes at the PHY layer based on Polar codes and chaotic sequences were proposed for the Orthogonal Frequency-Division Multiplexing (OFDM) system in [20, 21]. Since Polar code is a high complex encoder/decoder algorithm, it is an optional selection for a big transceiver that needs a high data rate and high reliability. Furthermore, most wireless transceivers nowadays utilize binary convolution code (BCC) encoder and Viterbi decoder as a mandatory. Therefore, a low complexity encryption method which is suitable for BCC code is needed.

Nominated as a promising solution, physical layer encryption (PLE) is considered in this work. PLE was deployed in various OFDM systems, such as constellation scrambling in the frequency domain, modulation symbols rotation [22, 23, 24], noise enhanced constellation rotation, rateless code [25], sparse code multiple access [26], subcarriers encryption [27]. The position of encryption can be placed before or after the data modulation. In case the encryption is performed after modulation, i.e., also known as phase encryption, the system is considered to be higher security because modulation type such as BPSK, QPSK, 16-256 QAM is also encrypted. However, conventional works on this method still have problems such as having high complexity if high-order modulation (64-256 QAM) is used and degrading packet error rate (PER) performance of the system. For that reason, PLE works in [28], and [29] only support low complex modulation OQPSK.

In this study, we propose lightweight physical layer encryption methods for IoT sensor transceivers. We aim to reduce the hardware complexity and evaluate the performance of the proposed methods. On this basis, our proposed methods can be applied to the systems that require low complexity and low power, such as IoT sensor transceivers.

1.2 Research Contributions

The aim of this dissertation is to provide new physical layer encryption methods that are low complexity for IoT sensor transceivers. We provide not only the algorithms but also the simulation evaluations and hardware implementation. In summary, the main contributions of this dissertation are:

- We propose the sign bit encryption method for the physical layer of IEEE 802.11ah. We simulate the system model based on the IEEE 802.11ah standard in Matlab to verify the BER and PER performances. The simulation results show that our proposed method does not degrade the BER and PER performances. Moreover, we only use the XOR for encryption; thus, it is low complexity. However, in this method, we encrypt only the sign bit, so the remaining data are still vulnerable to eavesdroppers.
- We propose a new joint encryption- modulation (JEM) method for IoT sensor transceivers, which is low complexity compared with the conventional method. We provide a performance analysis of the proposed JEM method. The simulation results show that the proposed JEM does not degrade the BER and PER performances of the system, while the conventional encryption method degrades about 3 dB.
- We design the hardware of the proposed JEM and the decryption. For comparison, we also design the hardware of CORDIC based conventional encryption method. The designs are synthesized in an application-specific integrated circuit (ASIC) using VDEC 180 nm CMOS technology and FPGA synthesis for hardware complexity analysis.

1.3 Dissertation Layout

The dissertation is divided into six chapters which are organized as follows:

- In Chapter 1, we introduce the overview, contributions, and the layout of this research.

- In Chapter 2, we first give an overview of the wireless technologies for IoT and their security countermeasures. Then we explain about physical layer encryption technique, which is mainly focussed on this thesis. Next, the related work is presented.

- In Chapter 3, we present the sign bit encryption method for the physical layer of IoT sensor transceivers. We describe the operation of this method, and then we show the simulation evaluation using Matlab with IEEE 802.11ah standard system model.

- In Chapter 4, we present the proposed joint encryption- modulation method for the PHY layer of IoT sensor transceivers. We describe the algorithm of the JEM method. Then we show the performance evaluation of this method.

- In Chapter 5, we first provide the hardware design of the proposed encryption and decryption. Then we present the hardware implementation of the CORDIC based conventional decryption method to compare with our proposed method.

- In Chapter 6, the last chapter of this thesis, we first conclude and emphasize the main contributions in our work. Then we provide some ideas for future work.

2 Background

This chapter presents the background of the dissertation. Chapter 2.1 introduces wireless technologies for IoT and their security countermeasures. Then chapter 2.2 provides the principle of physical layer encryption. Finally, the related work is surveyed in chapter 2.3.

2.1 Wireless Technologies for IoT and its Security Countermeasures

IoT aims to connect everything via wireless communication. Since IoT devices are usually small, embedded, and limited power, they are communicated with each other via low power wireless communication technologies. There are several popular wireless technologies for IoT, such as IEEE 802.15.4 (Zigbee), Bluetooth low energy (BLE), IEEE 802.11 n/ac, IEEE 802.11 ah, and LoRaWan [30, 31]. Among these technologies, we choose the IEEE 802.11ah standard to built the simulation model for further evaluating the performances of our proposed encryption methods for IoT sensor transceivers.

IEEE 802.15.4 IEEE 802.15.4 was defined as a standard for the PHY layer and MAC layer in wireless personal area networks [32]. It operates at an unlicensed industrial, scientific, and medical (ISM) 2.4 GHz frequency. IEEE 802.15.4 is power-saving and provides a data rate of up to 250 Kbps, which can use in IoT applications with limited data exchange requirements. It has been widely applied in wireless sensor networks, especially in industrial applications.

BLE BLE is an emerging wireless technology developed by the Bluetooth Special Interest Group for short-range communication (up to 100 meters) [33]. BLE operates at 2.4 GHz frequency and supports a data rate of 1 Mpbs. Moreover, BLE consumes extremely low power and can operate for months on standard coin-cell batteries [2]. BLE is widely employed in mobile phones, laptops, automobiles, etc. It is expected to be used in billions of devices in the near future.

Table 1. Wireless technologies for the Internet of Things (IoT)

Technique	Frequency	Range	Data Rate	Security Countermeasure	Applications
802.15.4 (Zigbee)	2.4 GHz	10 to 100 m	250 kpbs	AES in MAC layer	WSN, industrial, environment, and healthcare monitoring
BLE	2.4 GHz	50 to 150 m	1 Mbps	AES in link layer	Wearable devices, smartphones
802.11 n/ac	2.4 or 5 Ghz	50 m	>100 Mbps	WPA in MAC layer	Smart home, entertainment
802.11ah	sub 1 Ghz	1 km	150 kbps	WPA in MAC layer	Smart city, smart grid, smart home, healthcare
LoRaWan	sub 1 Ghz	>15 km	0.3 kbps to 50 kbps	Encrypt at network, application layer	M2M, smart city, and industrial applications

IEEE 802.11 n/ac IEEE 802.11 families are the most commonly used wireless local area network (WLAN) standards, which include IEEE 802.11 a/b/g/n. They work at 2.4/5 GHz and provide a high data rate ($>$ 100 Mpbs for 802.11 n/ac) [34]. They are widely used almost in smartphones, laptops, tablets, etc. However, they may not be suitable for many lightweight IoT applications due to its power consumption and coverage range.

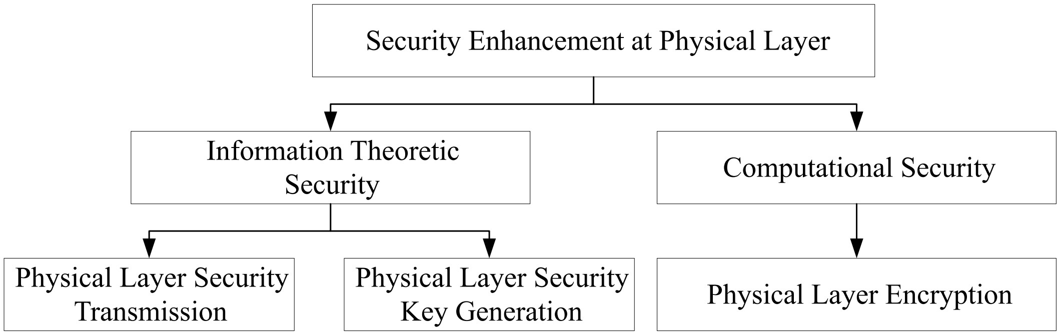
IEEE 802.11ah Recently, the Wi-Fi alliance announced to release the IEEE 802.11ah standard, which is a competitive candidate for developing IoT sensors' communication transceivers. It operates at sub 1 GHz frequency and provides a data rate from 150 kbps to 346 Mpbs. It supports long-range communication (up to 1 km), and a large number of devices (up to 8192 devices per access point) [35]. In addition, it employs the efficient mechanisms in the MAC layer to save energy; thus, it extends the sensors' battery life [36].

LoRaWan LoRaWan is also a sub 1 GHz wide area network (WAN) technology, which is released in June 2015 [37]. It supports low-power communications over long distances ($>$15km), millions of users, and low power consumption (up to ten years) [38]. Thus, it is suitable for IoT applications that require long-range coverage and low power consumption.

IEEE 802.15.4, Bluetooth, and IEEE 802.11 systems usually implement the security at the data link or MAC layer. For example, IEEE 802.15.4 and Bluetooth systems use an AES block cipher to protect the link layer. IEEE 802.11 systems employ the encryption at the MAC layer, which is called Wi-Fi protected access [39]. While LoRa executes the encryption at the network and application layer to provide the security. A summary comparison of the wireless technologies for the IoT is given in Table 1.

2.2 Physical Layer Encryption

In this section, we will describe the physical layer encryption at the PHY layer, which is the background for our proposed encryption methods. As shown in Figure 1, the security enhancement at the physical layer is divided into information-theoretic security and computational security. In this regard, computational secu-



Security Enhancement at Physical Layer
Information Theoretic Security
Computational Security
Physical Layer Security Transmission
Physical Layer Security Key Generation
Physical Layer Encryption

Figure 1. Classification of security enhancement techniques at the physical layer

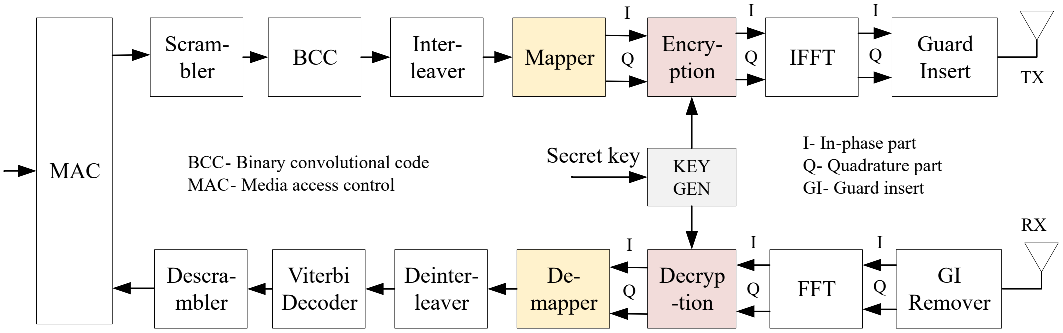


Figure 2. Physical wireless transceiver model

rity performs the encryption in different layers, including physical layer encryption (PLE).

The PLE methods can perform the encryption in different modulation stages, such as channel coding, mapping, inverse fast Fourier transform (IFFT) operation (for OFDM systems) [2]. In this work, we consider the encryption that processes after mapping.

Fig. 2 shows a block diagram of the PHY layer of a basic wireless communication system when implementing PLE. At the transmitter, data received from the MAC layer will be scrambled, encoded by Binary convolution code (BCC), interleaved, and mapped into the constellation of predefined modulation. The modulated data is then encrypted by the ciphering key generated from the “KEY GEN” block. “KEY GEN” can be any cryptography algorithm such as Grain, RC4. Because data is encrypted after modulation such as QPSK, 16-64QAM, it must encrypt both data streams, i.e., in-phase part (IP) and quadrature part (QP). The processing inside the encryption determines how the ciphering key makes a change to the modulated data. At the receiver side, the decryption is executed oppositely to recover the IP and the QP of the modulated symbols. In order to decrypt data successfully, the receiver must implement the same cryptography algorithm and keep the same secret key as the transmitter does.

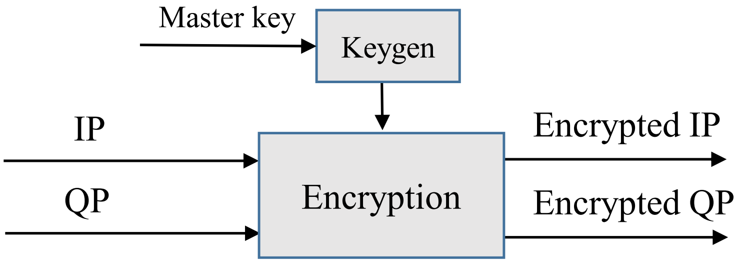


Figure 3. Physical layer encryption after the mapper

2.3 Related Work

Currently, PLE brings many advantages, such as the independence on the eavesdropper channel conditions, low computational cost, security enhancement at the signal level [40]. Therefore, it has attracted many researchers to find out new PLE methods for IoT sensor transceivers.

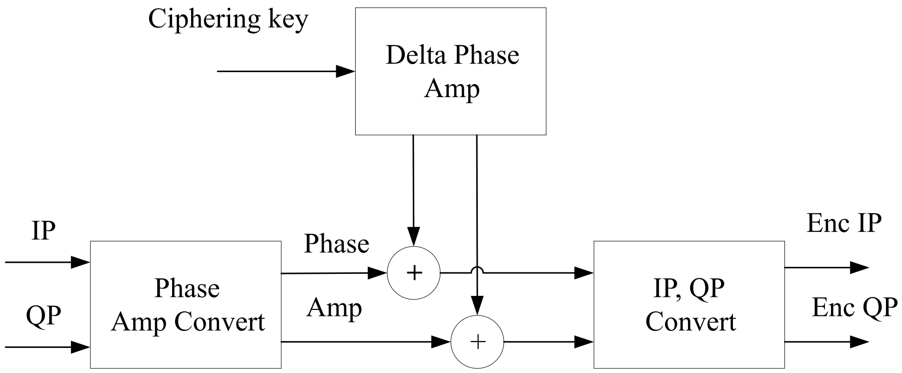


Figure 4. Conventional physical layer encryption

As surveying in the literature, many typical physical layer encryption works have been proposed. In the study [41], the authors prove that security at the physical layer results in the fact of the lowest impact on the network and offers low latency without introducing any overhead. Huo et al. [42] proposed a phase encryption method for general communication systems. This study showed that phase encryption at the PHY layer could resist traffic analysis attacks. In another approach [28], the authors performed physical layer encryption for IEEE 802.15.4 transceiver. They showed that security was enhanced by implementing encryption at the PHY layer instead of at the MAC layer as the original 802.15.4 standard does. However, in 802.15.4, there was only OQPSK modulation mode, which was encrypted by stream cipher RC4. Similarly, another secure 802.15.4 transceiver based on lightweight block cipher Simeck 32/64 was proposed in [29]. By exploiting the lightweight cipher, they reduced the hardware resource consumption. However, in this system, encryption was executed by XOR-operation between the binary data and ciphering keys before the mapper, so security was not high [42].

As shown in Fig. 4, another encryption method encrypts both the sign and amplitude information of the modulated data [22]. In this method, the IP and QP of the modulated data must be converted to amplitude and phase components. Based on the value of the ciphering key, the amplitude and phase of the data are accordingly changed. For further processing, the encrypted amplitude and phase then must be converted back to IP and QP parts. As a result, the encrypted data no longer exists in the modulation constellation. This affects the BER performance of the system. Since the complexity of the convert between phase/amplitude value and IP/QP value is affected by modulation types, this method is only applicable for simple modulations such as BPSK, QPSK. For high complexity modulations such as 64-QAM, 256-QAM, using this encryption method requires many computational resources and is not suitable for IoT sensors. For that reason, there is no implementation of this encryption method for high modulation such as 64-QAM, etc., available yet.

In another approach [43], the authors presented an asymmetrical PLE scheme based on elliptic curve cryptography for wireless communications. However, the drawback of asymmetric cryptosystems is a high computational overhead [44].

The mathematical models, design frameworks, and cryptographic primitives of PLE were established in [40]. However, the constellation modulation lays on the surface of the sphere, which is different from the conventional modulation.

To deal with these drawbacks, we propose low complexity physical layer encryption methods for IoT sensor transceivers. We aim to reduce the complexity and evaluate the performance of the encryption so that it can apply in the system that requires low complexity and low power, such as IoT sensors.

3 Sign Bit versus 8-MSB Bit Encryption Method

3.1 Introduction

In this chapter, we show our research results on physical layer encryption after the mapper at the PHY layer of the IEEE 802.11ah communication system, which is suitable for IoT sensor transceivers. We propose two encryption methods, which are the sign bit encryption and the eight most significant bit (8-MSB) encryption. We show that the sign bit encryption method satisfies the BER/PER requirement. It completely does not degrade the BER/PER performance while the conventional work does. With the purpose of low complexity, we use the popular stream cipher RC4 for generating the ciphering key for encryption, and XOR-operation for encryption.

This chapter is organized as follows. Chapter 3.1 presents the background of stream cipher RC4. Then, chapter 3.2 describes the proposed sign bit and 8-MSB bits. The simulation evaluation is provided to verify the performance of the proposed methods in chapter 3.3. Finally, chapter 3.4 gives the conclusion.

3.2 Stream Cipher RC4

In this section, we briefly explain stream cipher RC4, which is chosen to generate the ciphering key.

RC4 stream cipher was created by Ron Rivest from RSA Data Security in 1987. RC4 generates a pseudorandom bitstream, which is called as ciphering key. The ciphering key will then be used to encrypt plaintext data. To generate the ciphering key, RC4 uses a secret internal state that has two parts: 256-byte array memory S-box and three 8-bit index pointers i, j, and k. To generate a pseudorandom ciphering key, S-box values are permuted through two stages: Key Scheduling Algorithm (KSA) and Pseudorandom Generator Algorithm (PRGA). Fig. 5 illustrates the procedure of generating ciphering keystream from a provided master key. The KSA stage performs an initial permutation on S-box based on a secret master key, which is typically between 5 and 32 bytes. The PRGA uses the results of KSA, which has become a pseudorandom S-box to generate a pseudo ciphering key. Detail of RC4 processing can be found at [45].

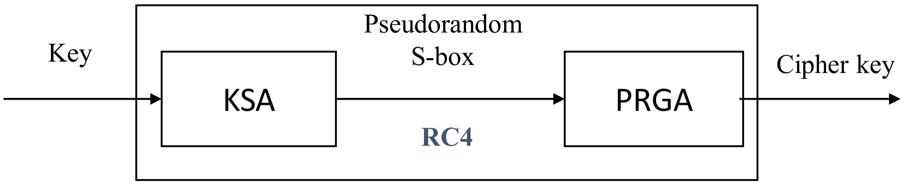


Figure 5. Procedure of cipher key generating of RC4

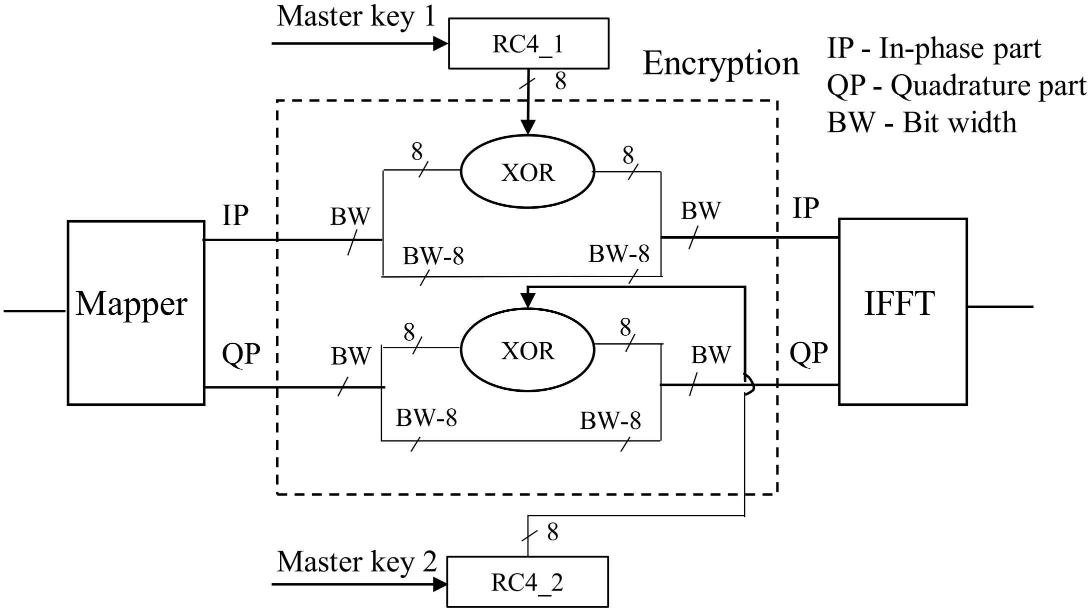


Figure 6. Method 1: 8-MSB encryption at the transmitter

3.3 The Proposed Sign Bit and 8-MSB Bits Encryption Methods

In 802.11ah, the variety of modulation types are used, including BPSK, QPSK, 16-QAM, 64-QAM, 256-QAM. To normalize the signal power to one, and to assure the peak to average power ratio (PAPR) of the transmitted signal, the modulated data is multiplied with the normalization factors, as be shown in Table. 2. Therefore, after the gain multiplication, the absolute value of IP and QP parts of modulated data in all cases of modulation (BPSK to 256-QAM) is always less than two. For the sake of hardware implementation, we can represent the *sign*, the *integer*, and the $fraction$ of IP/QP of modulated data by 1 bit, 1

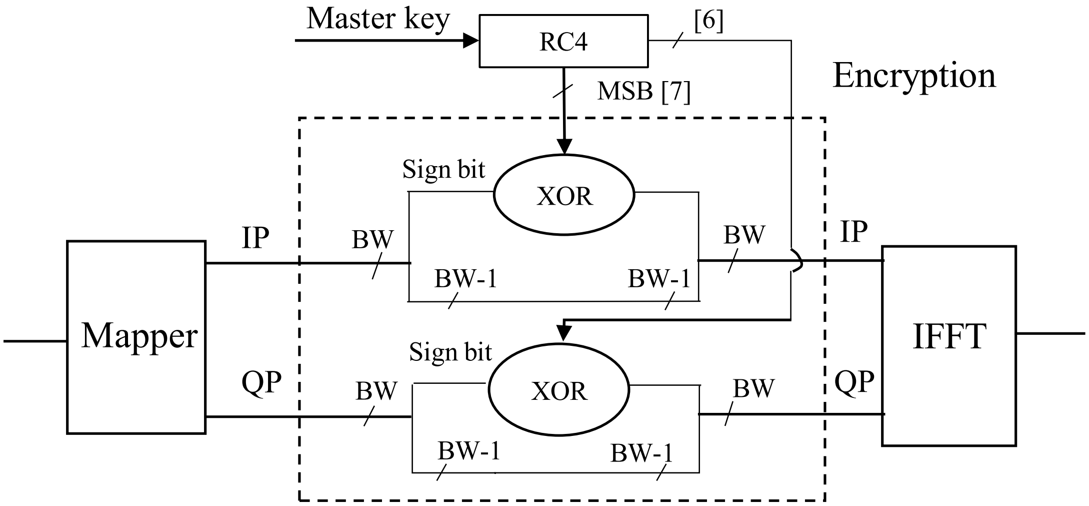


Figure 7. Method 2: Sign-bit encryption at the transmitter

Table 2. Amplitude of modulated symbols

Modulation type	Amplitude
BPSK	$0 < a < 1$
QPSK	$\frac{-1}{\sqrt{2}} < a < \frac{1}{\sqrt{2}}$
16-QAM	$\frac{-3}{\sqrt{10}} < a < \frac{3}{\sqrt{10}}$
64-QAM	$\frac{-7}{\sqrt{42}} < a < \frac{7}{\sqrt{42}}$
256-QAM	$\frac{-15}{\sqrt{170}} < a < \frac{15}{\sqrt{170}}$

bit, and N bits, respectively. Finally, we need a bit width (BW) $BW = N + 2$ bits to represent the value of each data IP and QP. According to our research experience, choosing $N \geq 16$ can result in an acceptable error between hardware and software results.

To generate the ciphering key for encryption, we chose the low complexity stream cipher RC4.

For the encryption, we propose a method that simply XOR the ciphering key with the IP/QP values of modulated data. However, one ciphering key generated by RC4 has eight bits, whereas each data of IP/QP has $BW = N + 2$ bits or

$BW \geq 18$ bits. The question is how to XOR 8 bits of the RC4 ciphering key with $BW \geq 18$ bits of IP/QP data. In our research, we focus on two cases. The first case is to encrypt only eight most significant bit (MSB) of each data IP/QP with 8-bits of ciphering key. The second one is to encrypt only the sign bit of IP/QP data with the MSB bit of ciphering key.

3.3.1 Method 1: 8-MSB Encryption

In this method, the encryption at the transmitter is operated, as shown in Fig. 6. In this method, we use two engines of RC4 to generate the ciphering key. We need two secret master keys for these two RC4 engines. One RC4 generates a ciphering key for encrypting IP data; the other generates the ciphering key for encrypting QP data. Only eight MSB of IP and QP data will be XOR with the ciphering key. The remaining bits are still kept as they are. After the encryption, the unencrypted bits are combined with encrypted bits before being sent to the IFFT processor.

At the receiver side, the decryption is performed oppositely. To decrypt successfully, two master keys of the receiver must be the same as those of the transmitter.

3.3.2 Method 2: Sign-bit Encryption

In this method, we only encrypt the sign bit of IP and QP data. We also use one RC4 engine to encrypt both IP and QP data. The MSB bit of ciphering key is used to encrypt the sign bit of IP, and the second significant bit of ciphering key is used to encrypt the sign bit of QP. The remaining bits of IP and QP data are kept as they are.

After the encryption, the unencrypted bits are combined with encrypted bits before being sent to the IFFT processor.

3.4 Evaluation Results

To check how the implementation of the proposed methods affect the BER and PER performance of the system, we have run the simulation in Matlab. Our simulation model follows the IEEE 802.11ah standard. Figure. 8 shows the block

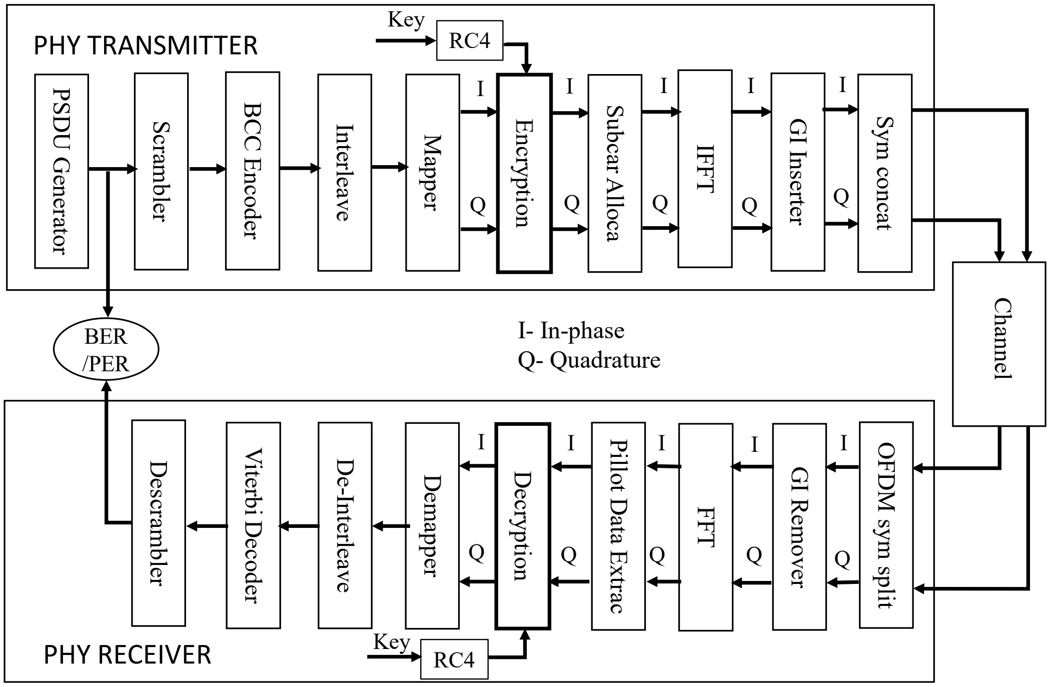


Figure 8. Simulation model based on IEEE 802.11ah standard

diagram of our simulator. Table. 3 presents the parameters using during the simulation. We suppose that the channel is corrupted either by the additive white Gaussian noise (AWGN) or fading channel. The transfer data is random, with 100 bytes per packet.

3.4.1 Simulation Model

The simulation model is described as follows. At the transmitter side, the source of sending random bitstreams is generated by 'PSDU Generator'. These bitstreams are scrambled by the 'Scrambler' block to keep away from a long sequence of zero or one bits. They are encoded by binary convolutional code (BCC) at 'BCC Encoder'. Then the 'Interleave' block permutes the encoded bits. Next, they are mapped into a constellation at the 'Mapper' block. In this simulation model, two types of modulation 16-QAM and 256-QAM are created. The modulated symbols are encrypted at 'Encryption' block. The output data of 'En-

Table 3. Simulation parameters

Simulation parameters	Value
Simulator	IEEE 802.11ah
Number of iterations	5000
Number of spacial streams in TXxRX	1x1
Channel type	AWGN, Fading
Channel estimation	Ideal
Modulation types	16-QAM, 256-QAM
Code rate	3/4
Transfer data type	Random

cryption', which passes 'IFFT' block is orthogonal as a result of the invert fast Fourier transform. This block also changes the frequency-domain of data into time-domain. The orthogonal frequency division multiplexing (OFDM) symbol is inserted by a guard interval at the 'GI Inserter' block for avoiding the interference with data of adjacent symbols. Finally, these bitstreams are transmitted to the receiver via the additive white Gaussian noise (AWGN) channel.

At the receiver side, many functional blocks are built for performing the reverse operations. At first, the guard intervals are discarded from receiving bitstreams at the 'GI Remover' block. The next operation is converting data from the time domain to the frequency domain by performing fast Fourier transfer at the 'FFT' block. Before the data subcarriers are delivered to the 'Demapper' block, they are decrypted at the 'Decryption' block. In the case of hard decision, 'Demapper' evaluates the input values of 'Mapper'. In the case of soft decision, 'Demapper' calculates the LLR values of input data of 'Mapper'. 'De-Interleave' converts the bit order into the original position. 'Interleave' and 'De-Interleave' are implemented to reduce the effect of the burst error. The received data is decoded at 'Viterbi Decoder'. Finally, the data is descrambled at the 'Descrambler' block to recover the transmitting information. This data is used to compare with the data at the input of 'Scrambler' to evaluate the BER and PER performance of the simulation model. The readers refer to [46] and [47] for more detail about processing inside each block.

3.4.2 Simulation Results

We evaluate the BER and PER of the system in five cases: with sign-bit encryption; with 8-MSB encryption; without encryption; with using Ref. [22], and with an unexpected user who does not obtain the correct key in the receiver side.

Figure 9 and 10 show the BER performance of the system for 16-QAM and 256-QAM modulation in AWGN channel and fading channel. Figure 11 and 12 indicate the PER performance of the system for 16-QAM and 256-QAM modulation in AWGN channel and fading channel. From these figures, we obtain the following results:

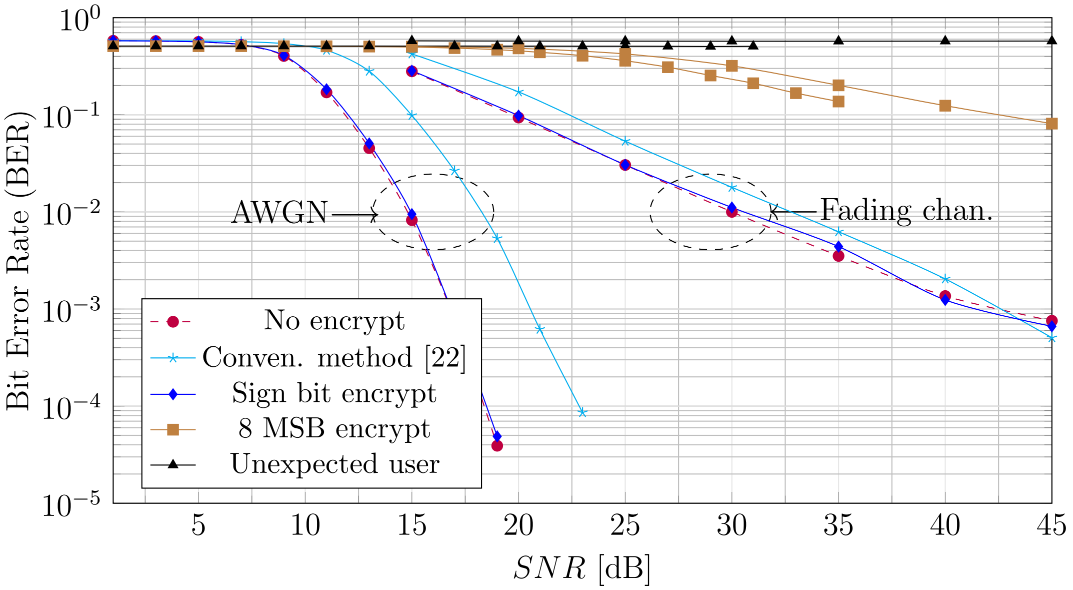


Figure 9. BER performance of 802.11ah for 16-QAM with the sign bit encryption, AWGN and fading channel

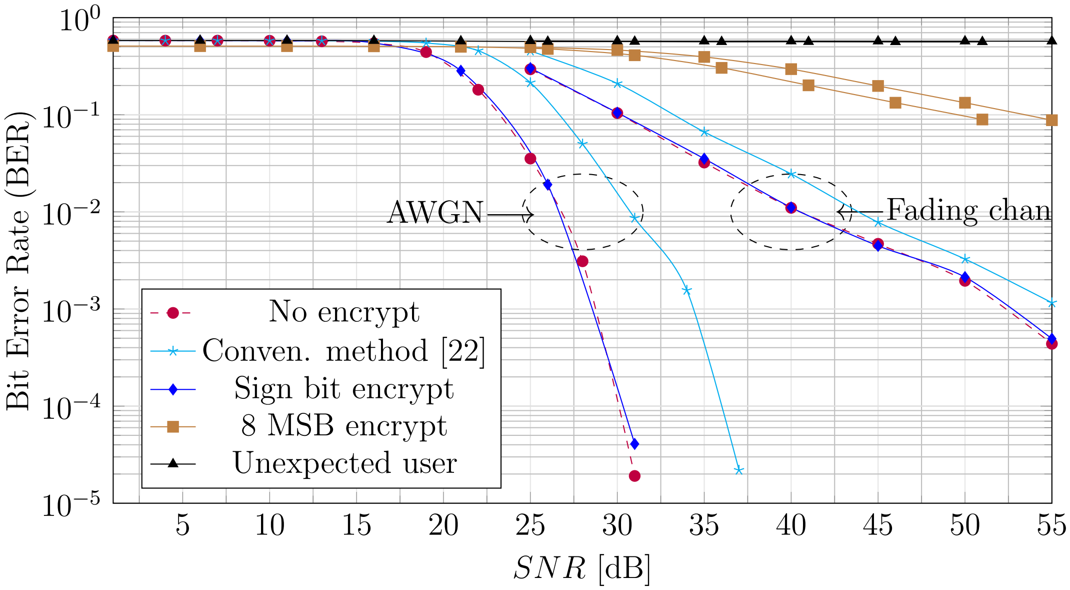


Figure 10. BER performance of 802.11ah for 256-QAM with the sign bit encryption, AWGN and fading channel

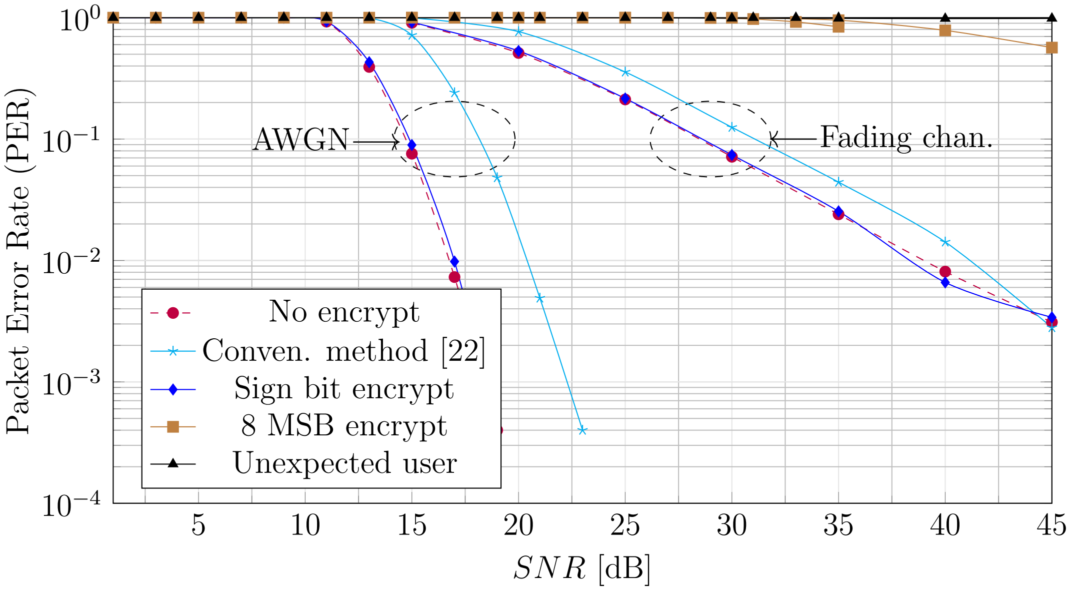


Figure 11. PER performance of 802.11ah for 16-QAM with the sign bit encryption, AWGN and fading channel

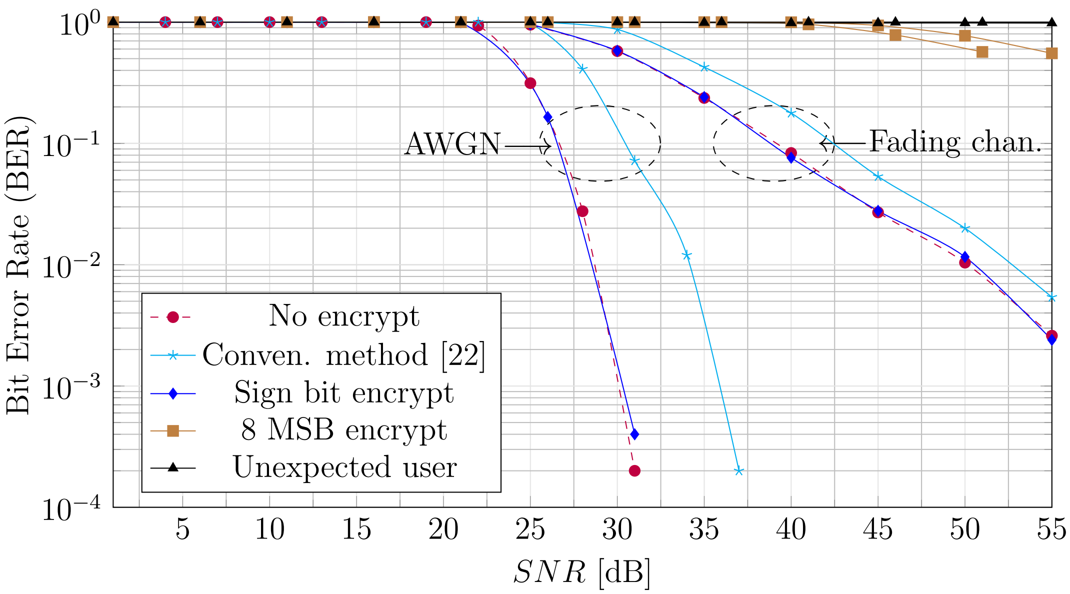


Figure 12. PER performances of 802.11ah for 256-QAM with the sign bit encryption, AWGN and fading channel

Firstly, in both case of 16-QAM and 256-QAM, the implementation of our proposed encryption method 2, i.e., Sign-bit encryption, does not degrade the BER and PER performance of the system in both kinds of channel. Whereas, the conventional work Ref. [22] degrades both BER and PER performance by about 3 dB in case of deploying AWGN channel, and about 2 dB in case of fading channel .

Secondly, in both cases of 16-QAM and 256-QAM, if an unexpected user does not have the correct key that the transmitter has used to encrypt the data, he/she is entirely not able to recover the transmitted data. The decryption is unsuccessful.

Thirdly, the implementation of our proposed encryption method 1, i.e., 8-MSB encryption, degrades the BER and PER performance significantly. The reason is that we encrypt not only the sign but also the significant bits that represent the integer and fraction parts of data. These encrypted data then be affected by noise and interference of the channel. The effect of the noisy channel makes the receiver is unable to recover the transmitted data even though it has the correct master key.

From the first and the second results, we conclude that our proposed encryption method 2 (Sign-bit encryption) is suitable for 802.11ah standard and that it is applicable for IoT sensors. In summary, it is low complexity (only use XOR operation), high performance (not degrade BER/PER performance). From the third result, we conclude that if using XOR operation for phase encryption, we should only encrypt the sign bit. Encrypting the bits that represent integer and fraction value of data will make the encrypted data becomes sensitive to the noisy channel. As a result, the receiver is not able to decrypt the data successfully.

3.5 Conclusion

In this chapter, we have presented two encryption methods for the physical layer of IEEE 802.11ah. We have built the simulation model and evaluated the BER and PER performance of these methods for 16-QAM and 256-QAM modulation in AWGN noisy channel and in fading channel. These simulation results have shown that our proposed encryption method 2 (Sign-bit encryption) is suitable for 802.11ah standard and that it is applicable for IoT sensors because of its low complexity, high performance. Implementing this method does not degrade the BER and PER performance of the system while the conventional work degrades the performances. We also show that if using XOR operation for phase encryption, we should only encrypt the sign bit. Encrypting the other bits of data will make data become sensitive to noise and is not able to decrypt successfully.

Despite the low complexity of the proposed sign bit encryption method, the remaining bits of modulated data are not encrypted. Therefore, in the next chapter, we will present another encryption method that is not only low complexity but also encrypts the entire modulated data.

4 Joint Encryption-Modulation (JEM) Method

This chapter presents a low complexity physical layer encryption method for IoT sensor transceivers, which is called joint encryption-modulation (JEM) method. This chapter is organized as follows. Chapter 4.1 provides an introduction. Then Chapter 4.2 describes the algorithm of the proposed JEM method. The simulation evaluation is provided in Chapter 4.3. Finally, the conclusion is given in Chapter 4.4.

4.1 Introduction

Most encryption methods nowadays perform XOR operation to data and ciphering key because of its low complexity. Unfortunately, if merely performing XOR operation into the modulated data, the results will no longer locate in the constellation map. For example, in the case of 16-QAM, as shown in Fig. 13, the modulated data is mapped into the IP part and QP part in which the values are $-3, -1, 1, 3$. If we apply XOR operation into these values, the results may jump out of the range $-3, -1, 1, 3$. It means that the shape of the constellation will be changed. Once the transmitted data is not located in the constellation point, the receiver will be more difficult to recover the transmitted data. Consequently, the error rate increases, and the performance of communication reduces.

For the sake of simplicity, we prefer to use XOR operation for encrypting data. At the same time, the encryption should not degrade the performance of the system. A new idea should be proposed to guarantee that the result of XOR-operation between the modulated data and the ciphering key still remains in the constellation shape. Our initial idea is to convert the values of constellation points into continuous positive integers such as $0, 1, 2, 3$ (16 QAM). These data will be encrypted by XOR with the ciphering key. The encrypted data will certainly still in the range $0, 1, 2, 3$ (16 QAM). Finally, the encrypted data is converted back to values of constellation points, i.e., $-3, -1, 1, 3$ (16-QAM).

In this proposed method, we aim to reduce the hardware complexity and evaluate the performance of the proposed JEM. On this basis, our proposed JEM method can be applied to the systems that require low complexity and low power, such as IoT sensor transceivers. To evaluate the performances of the proposed

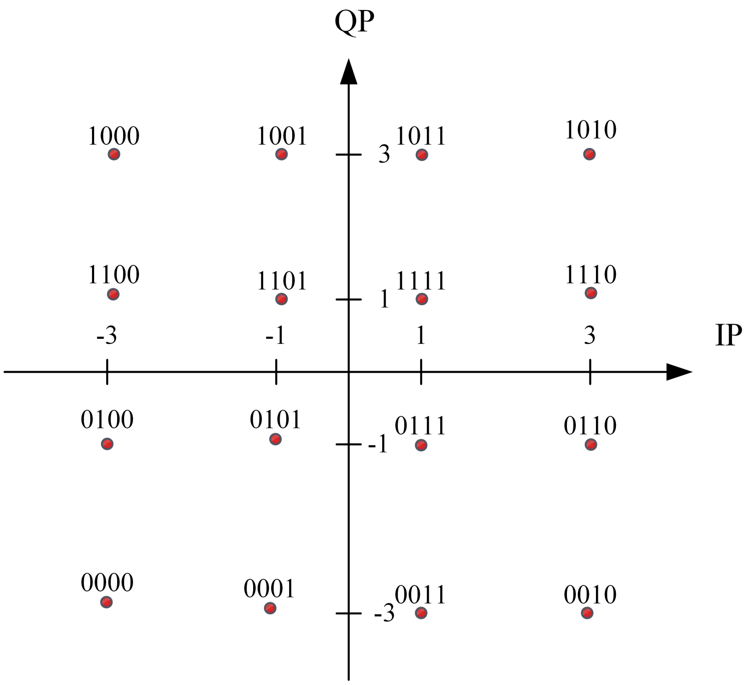


Figure 13. Constelation for 16-QAM modulation

method, we simulate in Matlab following IEEE 802.11ah standard system model.

In summary, the main contributions of this chapter are:

- We propose a new low-complexity JEM method for IoT sensor transceivers.
- We provide a performance analysis of the proposed JEM method. The simulation results show that unlike the conventional method, which reduces the bit error rate (BER) and packet error rate (PER) performances of the physical transceivers up to 3 dB, our proposed method does preserve these performances.

4.2 Algorithm of the JEM Method

Base on the mentioned-above basic idea, we propose the join-encryption-modulation (JEM) method, as shown in Fig. 14. The JEM merges the operation of mapper and encryption into one component. The encryption is divided into two stages, named as PRE-MAP, and POST-MAP. The XOR operation is placed in between two stages to encrypt data. The detailed principle of the proposed JEM method is described as follows.

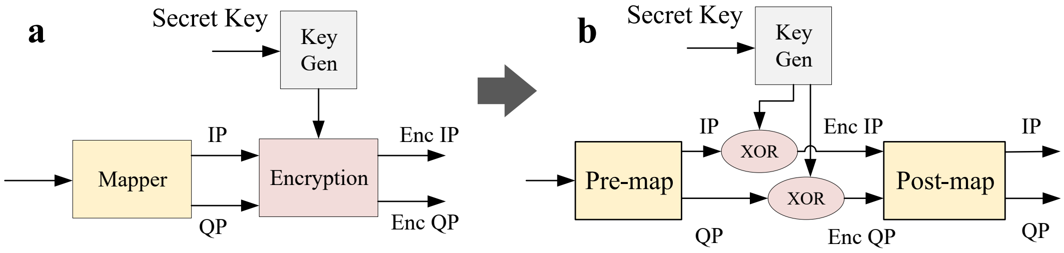


Figure 14. Conventional PLE (a) versus the proposed JEM (b)

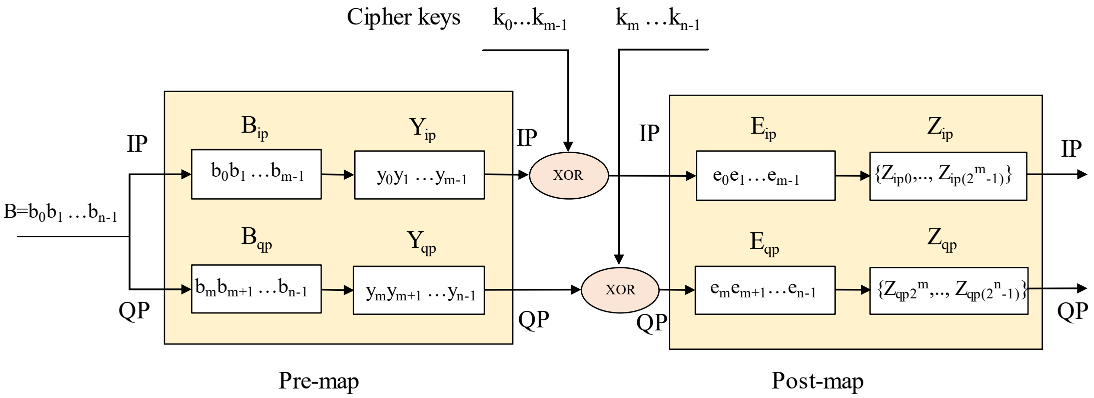


Figure 15. The detailed operation of the proposed JEM method

The input binary data, $B = b_0b_1...b_{n-1}$, are divided into the IP part, $B_{ip} = b_0b_1...b_{m-1}$, and the QP part, $B_{qp} = b_mb_{m+1}...b_{n-1}$, where $n = 2m$ is the number of bits per modulated symbol. According to the modulation type m is the number of bits per IP or QP part, defined as in Equation 1. In this study, we design and evaluate the proposed JEM for the most used modulation types up to 256-QAM of the wireless communication.

$$m = \begin{cases} 1 & \text{if} \quad \text{BPSK, QPSK} \\ 2 & \text{if} \quad \text{16-QAM} \\ 3 & \text{if} \quad \text{64-QAM} \\ 4 & \text{if} \quad \text{256 QAM} \end{cases} \tag{1}$$

Since the IP and QP part are similar to each other, we only describle the encryption process of the IP part. Note that, for BPSK, the input data only consists of IP part. The operation of the proposed JEM for the various modulation types in wireless communication is executed in the following three steps, illustrated in Fig. 15.

- Step 1, PRE-MAP: The input data of IP part, $B_{ip} = b_0b_1...b_{m-1}$, is first mapped into the continuous pre-map values $Y_{ip} = y_0y_1...y_{m-1}$ by the following mapping rule.

  $$B_{ip} \mapsto Y_{ip} \tag{2}$$

  Where the continuous pre-map values $y_0y_1...y_{m-1}$ are calculated based on the input values $b_0b_1...b_{m-1}$ and are formulated by the following equation.

  $$\begin{cases} y_0 = b_0 \\ y_1 = b_0 \oplus b_1 \\ ... \\ y_{m-1} = b_0 \oplus b_1 \oplus ... \oplus b_{m-1} \end{cases} \tag{3}$$

- Step 2, ENCRYPTION: The pre-map data Y_{ip} is hence encrypted by a XOR-operation with the cipher keys $K_{ip} = k_0k_1...k_{m-1}$ to obtain encrypted data $E_{ip} = e_0e_1...e_{m-1}$.

  $$E_{ip} = Y_{ip} \oplus K_{ip} \tag{4}$$

- Step 3, POST-MAP: The encrypted data E_{ip} are then mapped to the post-map values Z_{ip} of the conventional modulation by the mapping rule in Equation 5. Since the IP part consists of m bits, there are 2^m possible combinations of them. We number these combinations according to an ascending value of combination as $0, 1, ..., 2^m - 1$.

  $$\begin{cases} E_{ip0} \mapsto Z_{ip0} \\ E_{ip1} \mapsto Z_{ip1} \\ ... \\ E_{ip(2^m-1)} \mapsto Z_{ip(2^m-1)} \end{cases} \tag{5}$$

Where Z_{ip0},...,$Z_{ip(2^m-1)}$ are calculated as follows.

$$\begin{cases} Z_{ip0} = -(2^m-1) + 2 \times 0 \\ Z_{ip1} = -(2^m-1) + 2 \times 1 \\ \quad ... \\ Z_{ip(2^m-1)} = -(2^m-1) + 2 \times l \end{cases} \tag{6}$$

Where $l = 0, 1, ..., 2^m - 1$.

Table 4. The proposed JEM for 16-QAM modulation

Pre-Map In b_0b_1/b_2b_3	Pre-Map Out y_0y_1/y_2y_3	Cipher Key k_0k_1/k_2k_3	Encryption Out E_{ip}/E_{qp}	Post Map Out Z_{ip}/Z_{qp}
00	00	00	00	-3
01	01	00	01	-1
11	10	00	10	1
10	11	00	11	3
00	00	01	01	-1
01	01	01	00	-3
11	10	01	11	3
10	11	01	10	1
00	00	10	10	1
01	01	10	11	3
11	10	10	00	-3
10	11	10	01	-1
00	00	11	11	3
01	01	11	00	1
11	10	11	01	-1
10	11	11	00	-3

For a clear explanation of the proposed JEM, we describe in detail the operation for 16-QAM modulation in the following.

16-QAM Modulation For 16-QAM, we have the number of bits per IQ data $m = 2$. Table 4 illustrates the detailed operation of the proposed JEM. First the input data IP $B_{ip} = b_0b_1$ are mapped into Y_{ip}= y_0y_1 using a pre-map. Then the pre-map outputs are encrypted by a XOR-operation with the cipher key. The cipher keys $K_{ip} = k_0k_1$ are selected from {00, 01, 10, 11}, so the encrypted data E_{ip} is permuted. In the post-map, the encrypted data E_{ip} are again mapped into conventional mapper values $Z_{ip} = (-3) \vee (-1) \vee (1) \vee (3)$. However, these post-map values express different input information data. Consequently, illegitimate receivers without the correct key cannot decrypt the correct data.

At the receiver, the decryption performs as the reversed operation of the proposed JEM. It is assumed that the legitimate receiver obtains the same secret key as the transmitter, so the same ciphering keys are generated. In this decryption method, we apply the hard-decision decoding method. The input values of IP and QP parts first compare with the threshold values of the conventional modulation to determine the decoded output as 1 or 0. Then, by using cipher keys, they are decrypted.

4.3 Performance Evaluation

Our simulation model follows the IEEE 802.11ah wireless communication network, which is suitable for IoT applications. Table 5 shows some important parameters used in the simulation. We simulated the system for high order modulations such as 16-QAM, 64-QAM, and 256-QAM. It is assumed that the channel was corrupted either by the additive white Gaussian noise (AWGN) or by the fading channel. The transfer data is random, with 100 bytes per packet. The simulation was run 10,000 times.

Figure 16, 17, 18 and Figure 19, 20, 21 show the BER and PER performance of the system corrupted by AWGN channel and fading channel for four cases: with the proposed encryption; with no encryption; with using conventional encryption method; and with an unexpected user who does not obtain the correct key in the receiver side (an eavesdropper). From these figures, we can make the following observations.

Firstly, the proposed JEM has nearly the same BER and PER performances as the unencrypted system for all modulation types (16-QAM, 64-QAM, and 256-

Table 5. Simulation parameters based on IEEE 802.11ah

Simulation parameters	Value
Simulator	IEEE 802.11ah
Number of iterations	10^4
Number of spacial streams in TXxRX	1x1
Channel type	AWGN, Fading
Channel estimation	Ideal
Modulation types	16,64,256-QAM
Code rate	3/4
Transfer data type	Random

QAM) in both types of channels AWGN channel and fading channel. This means that the proposed JEM does not degrade the BER or PER performance. The reason is that the proposed JEM does not change the constellation of modulated symbols. Consequently, it does not affect the error correction performance of the system.

Secondly, these figures also show that the conventional method in [22] degrades the BER and PER performances about 3 dB in the AWGN channel and 2 dB in the fading channel as compared with the proposed JEM at the observation point of $BER = 10^{-3}$ and $PER = 10^{-3}$. The main reason for BER and PER performance degradation of [22] is that the encrypted data of [22] (which is transmitted to the receiver) do not locate on the constellation points anymore. The transmitted data is thus more sensitive to noise and interference during transmission. In addition, the receiver has more difficulty in recovering the transmitted data, which originally does not locate on the constellation. Error detection rate increases, and BER/PER performance degrades.

Finally, the figures show that an unexpected user, who does not have the correct key has approximately 50% BER, and 100% PER. Note that a bit has value either 0 or 1; the 50% of BER is a random bit error rate detection in case the receiver has no information about the transmitter. It does not mean that the receiver of unexpected users can detect the transmitted data. PER

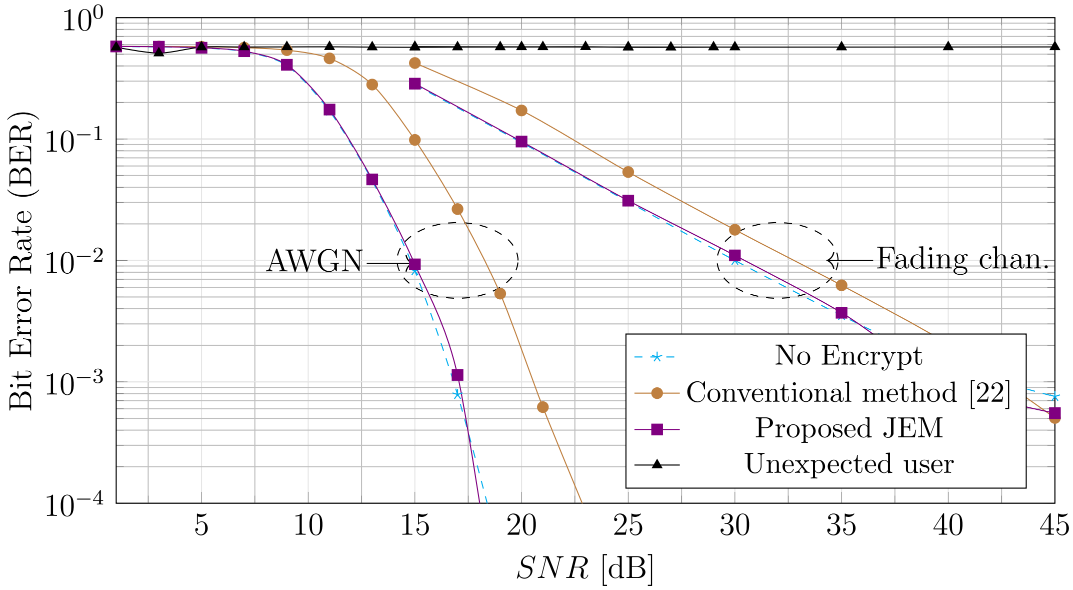


Figure 16. BER performance of 802.11ah for 16-QAM with the JEM method, AWGN and fading channel

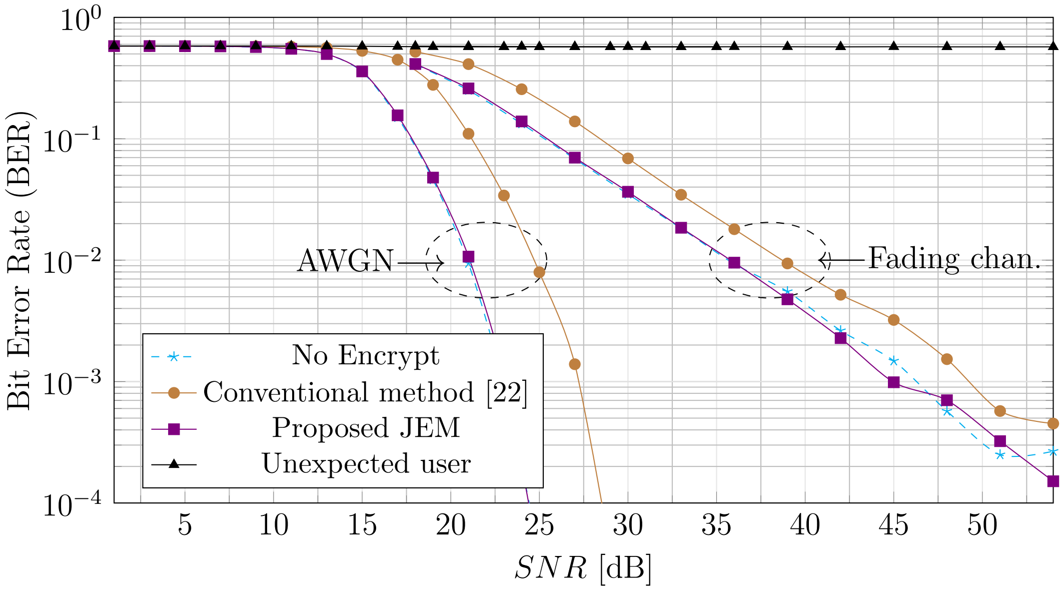


Figure 17. BER performance of 802.11ah for 64-QAM with the JEM method, AWGN and fading channel

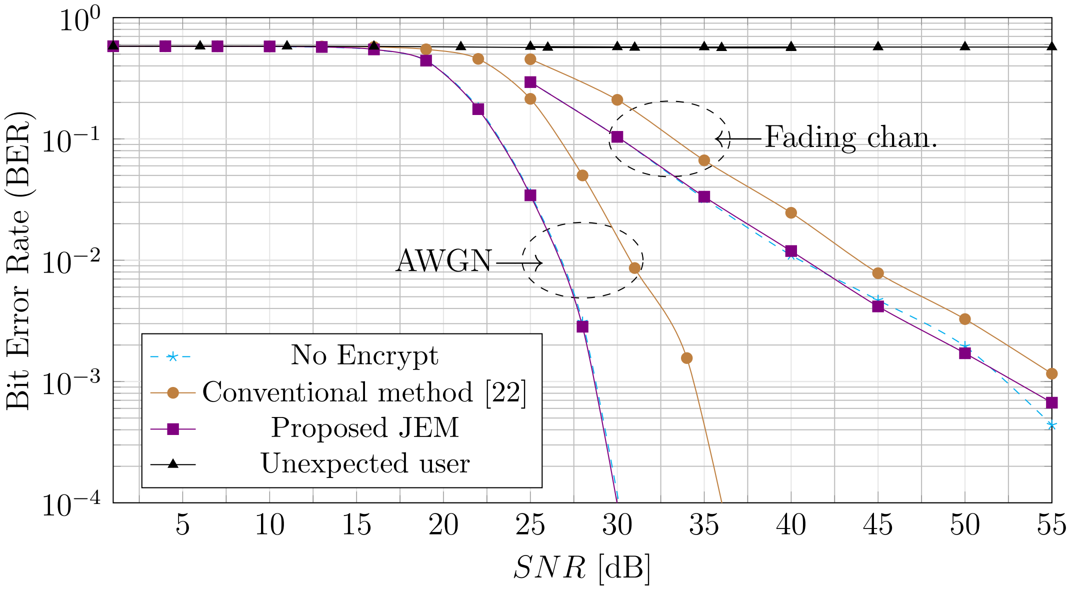


Figure 18. BER performance of 802.11ah for 256-QAM with the JEM method, AWGN and fading channel

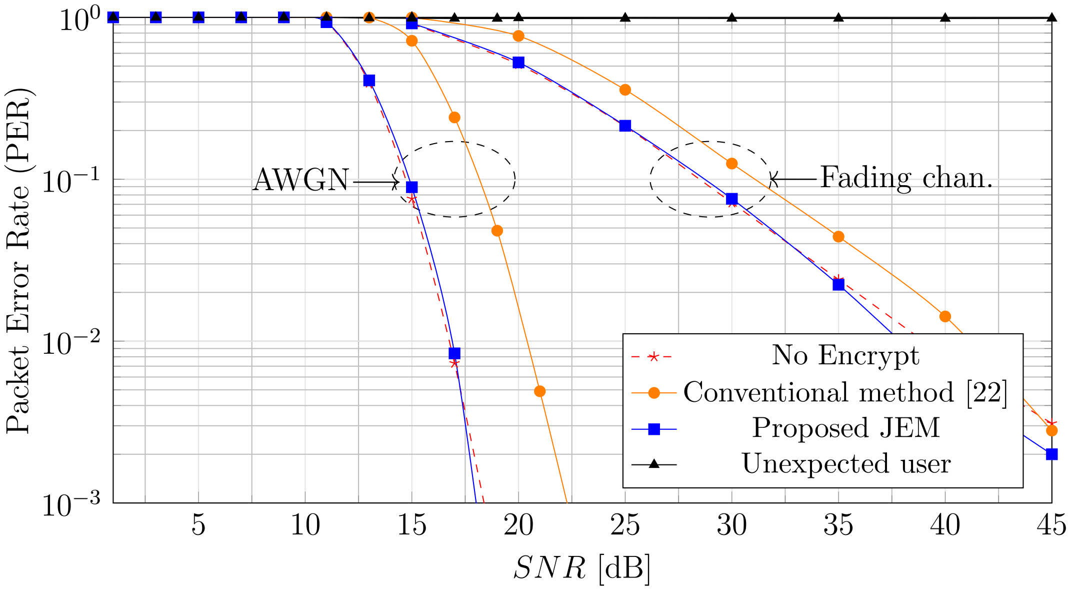


Figure 19. PER performance of 802.11ah for 16-QAM with the JEM method, AWGN and fading channel

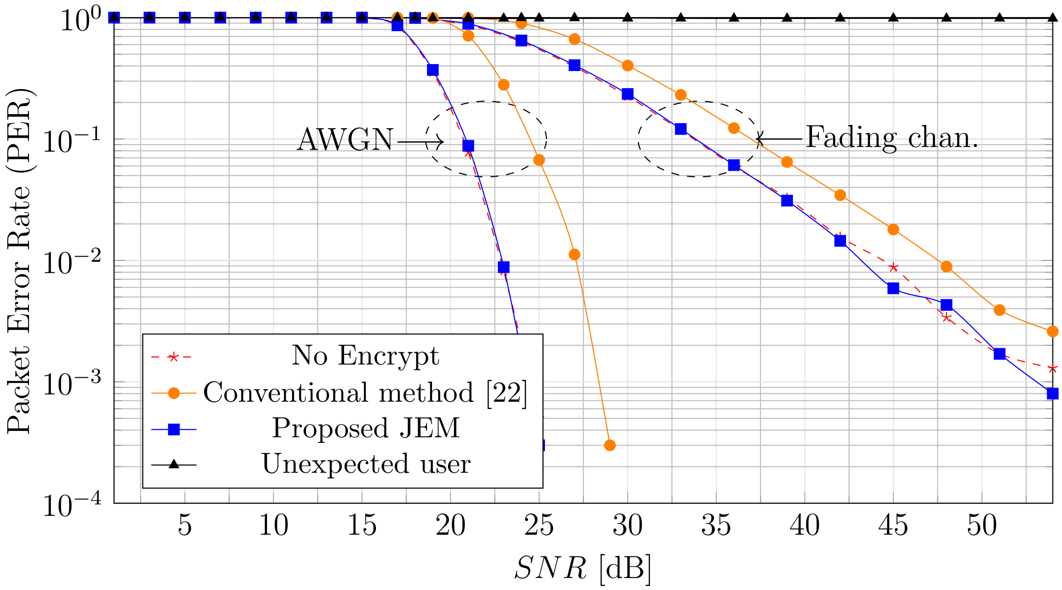


Figure 20. PER performance of 802.11ah for 64-QAM with the JEM method, AWGN and fading channel

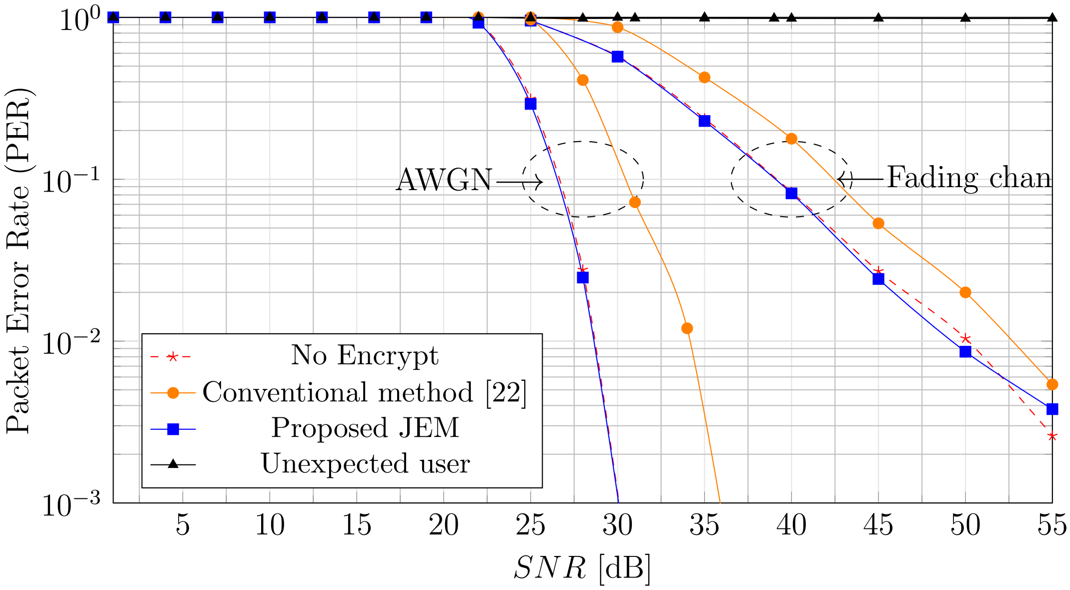


Figure 21. PER performance of 802.11ah for 256-QAM with the JEM method, AWGN and fading channel

= 100% shows that the unexpected user completely not be able to recover the transmitted data. It means the proposed JEM method successfully protects data from an unexpected user who does not own the correct private key.

4.4 Conclusion

In this chapter, we have presented a new low complexity physical layer encryption method for IoT sensor transceivers able to support up to 256-QAM. We built a simulation model to evaluate the BER and PER performances of the proposed JEM method to compare with existing methods. The simulation results show that our proposed JEM method does not degrade the BER and PER performances of the system compared with unencrypted transmissions, while the conventional method degrades the performance by 3dB in the AWGN channel and 2 dB in the fading channel.

5 Hardware Design

5.1 Introduction

In this chapter, we show the hardware design of the proposed JEM method. Then, we implement the hardware of the conventional encryption method based on the Coordinate Rotation Digital Computer (CORDIC) to compare with our method.

This chapter is organized as follows. Chapter 5.1 presents the hardware design of the JEM method. Then, chapter 5.2 provides the hardware implementation of the CORDIC based conventional decryption method. Next, the implementation results are provided in chapter 5.3. The conclusion is given in chapter 5.4.

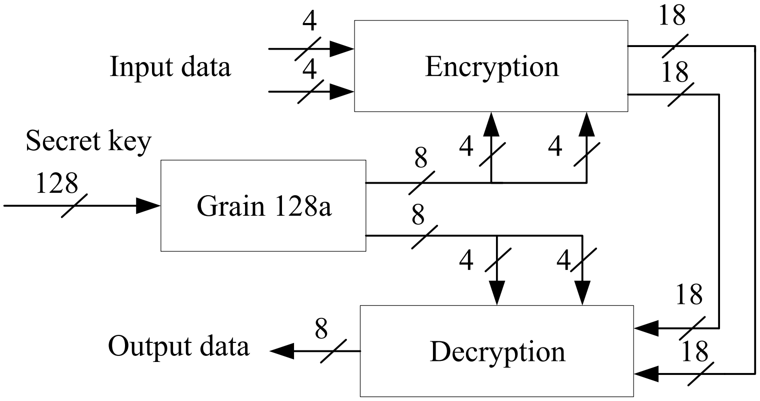


Figure 22. Overview of the hardware architecture

5.2 JEM's Hardware Design

5.2.1 Overview of the Architecture

After completing simulation and evaluating BER and PER performances of the proposed JEM, we designed the hardware to analyze the hardware complexity and to prove that our proposed JEM can be practically implemented. The hardware of the proposed JEM supports a variety of modulation types, BPSK, QPSK, 16-QAM, 64-QAM, and 256-QAM. Besides, we also designed the hardware of the decryption to prove that it is feasible to decrypt the data using our proposed encryption method.

Figure 22 presents the overview of the hardware architecture. It consists of three blocks, encryption, decryption, and stream cipher Grain 128a blocks. The input data is encrypted in the encryption block using cipher keys generated by the Grain 128a block. Similarly, the decryption block decrypts the encrypted data produced by the encryption block. Besides this, we also designed the hardware of the conventional encryption and decryption method in [22], and conventional mapper and de-mapper to compare with our encryption and decryption. Based on the results of [48] and our experience in hardware implementation, we chose 18 bits as the fixed point for data representation, which does not affect the performance of the system and benefits the hardware implementation.

To achieve low complexity, in addition to utilizing the proposed JEM, the lightweight keystream generator also plays an important role.

5.2.2 Lightweight Keystream Generator

To perform encryption and decryption, a cipher is required to generate keystreams. There are two types of ciphers called block cipher and stream cipher. In this design, we chose a stream cipher rather than a block cipher due to following reasons.

- Block ciphers require a large chip and high power consumption due to their complex architecture[28]. These disadvantages make block ciphers unsuitable for IoT sensor transceivers.

- In block ciphers, an error in one symbol can cause the corruption of an entire block. On the other hand, in stream ciphers, an error in encrypting one symbol does not affect subsequent symbols.

There are many types of stream ciphers, such as RC4, Grain 128a, A5, Mickey, Trivium. Although RC4 is one of the most widely used stream cipher, compared with other stream ciphers, it consumes a large hardware resource[29]. Therefore, we chose stream cipher Grain 128a because it is lightweight, making it suitable for IoT sensors. We developed the hardware for both RC4 and Grain 128a stream cipher, but since Grain 128a had lower power and hardware resource consumption, it is more fitting for our proposed JEM. The details of Grain 128a can be found in [49]. Further, we designed the Grain 128a hardware to generate 8-bit keystreams per clock cycle to encrypt and decrypt the IP and QP parts.

As clearly shown in Table 6, the slices look-up tables (LUTs) of Grain 128a is less than half of RC4's slice LUTs. In addition, the static power and area of Grain 128a is also about half of the designed RC4 version. When compared with the RC4 in [28], the power consumption of our Grain 128a hardware implementation is less than 1%.

Table 6. Comparison of Grain 128a with RC4 at the same clock frequency of 16 MHz

Implemented design	Clocks	Slice registers	Slice LUTs	Static Power(mW)	Area (μm^2)
GRAIN 128a	256	2195	2715	1.53	188175
Designed RC4	256	2213	7954	3.25	344461
RC4[28]	256	2478	7064	163.43	-

5.2.3 JEM's Encryption and Decryption

As indicated in Fig. 23, the hardware of JEM consists of a pre-map block, post-map block, and XOR-encryption. The pre-map block converts input data into the proposed output pre-map values. Then, in the post-map block, the output pre-map values are mapped to the conventional mapper values. We design the hardware of the pre-map, and post-map blocks that support multiple modulation types.

It is clearly shown in Fig. 24 that the pre-map and post-map only consist of multiplexers. Therefore, they consume significantly low hardware resources. Moreover, the hardware design of the proposed JEM employs only the XOR-operation to encrypt the IP and QP parts with the cipher keys; it is thus low complexity.

As illustrated in Fig. 25, the hardware for decryption consists of a Gain block, joint de-mapper-decryption, and a mux block. In this regard, the gain block is responsible for recovering any decrease in the amplitude of the input data by multiplying with a corresponding gain constant of each modulation type, as illustrated in Fig. 26. The joint de-mapper-decryption supports different modulation types. In this block, we use several binary fix-point comparators to compare

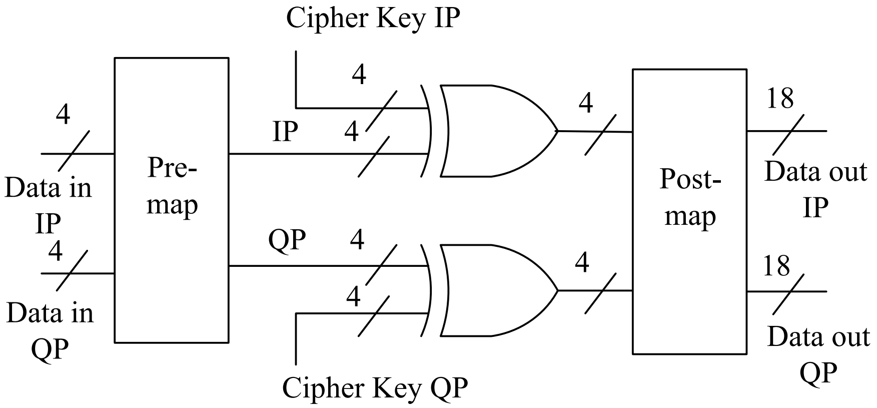


Figure 23. Hardware architecture of the JEM encryption

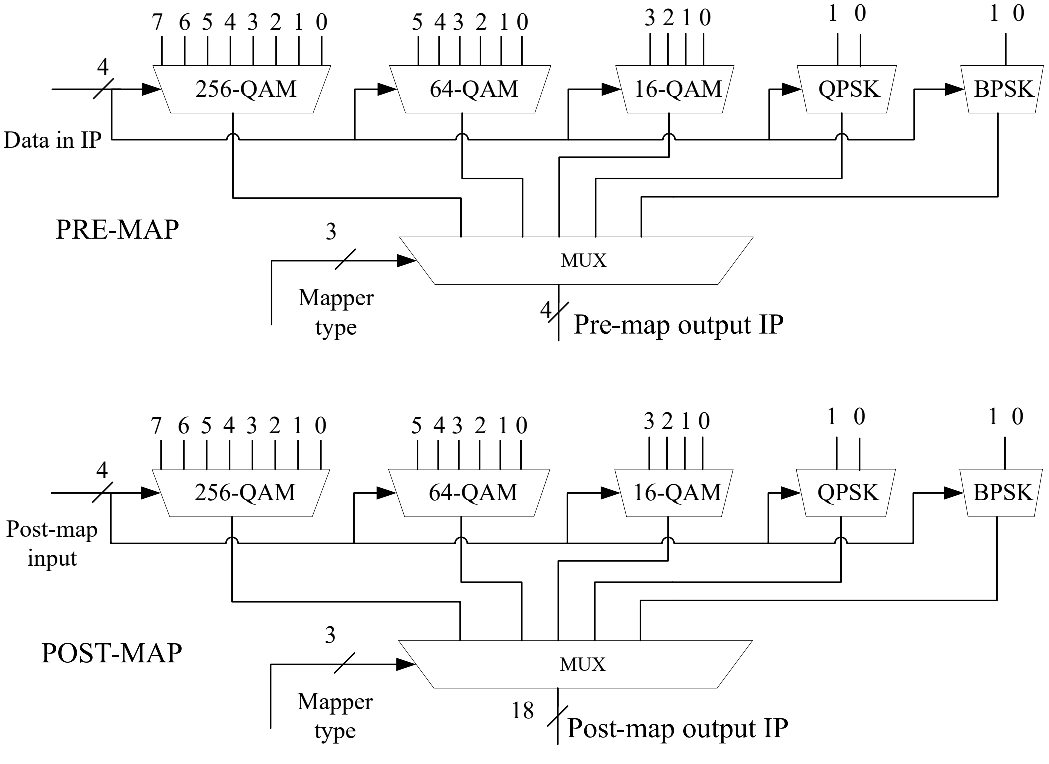


Figure 24. The pre-map and post-map architecture

the input data with the threshold values to determine the output values. Then, depending on the value of the cipher key, the decrypted output data is decided.

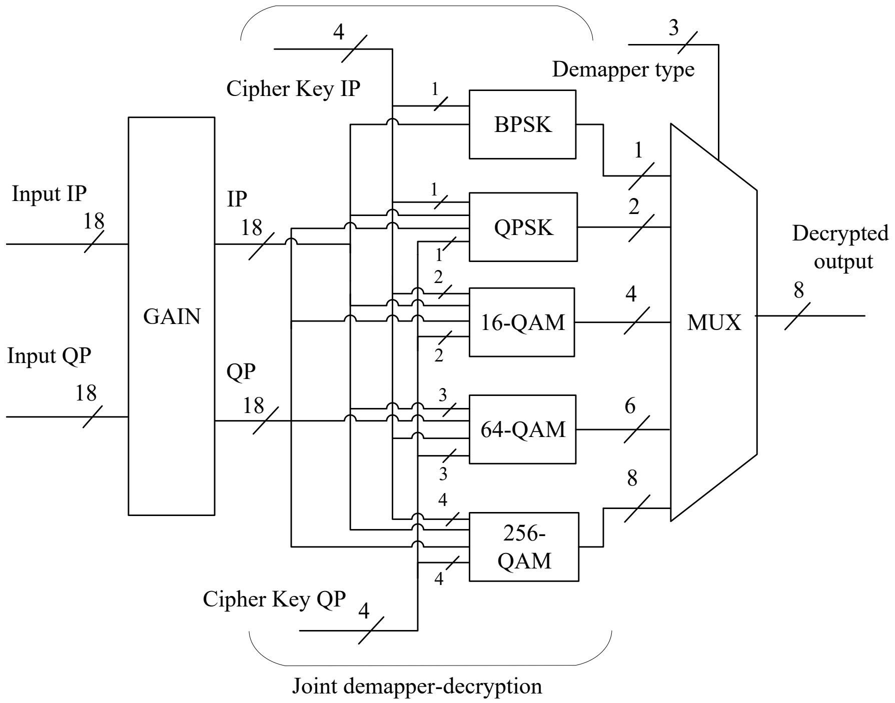


Figure 25. Hardware architecture of the JEM decryption

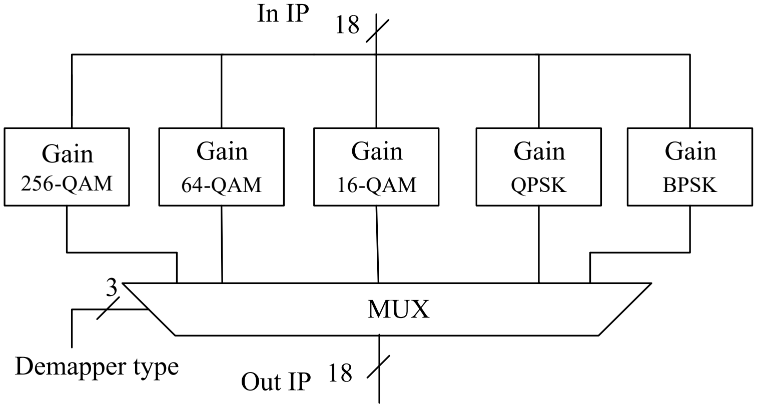


Figure 26. Gain block architecture.

5.3 CORDIC Based Conventional Phase Decryption Circuit

In this section, we present the hardware implementation of the PHY layer phase decryption for high complex modulation types of 802.11ah. We use the decryption algorithm, which is presented in [22]. This method requires to calculate the amplitudes, phases, and trigonometric functions of the modulated data. Therefore, we apply the Coordinate Rotation Digital Computer (CORDIC) algorithm, which can result in reducing the complexity of the hardware. This work is published in [24].

5.3.1 Phase Decryption Algorithm

In this section, we briefly explain phase encryption and decryption that exploit ciphering keys to encrypt and decrypt data at the PHY layer.

The phase encryption is a type of encryption at the PHY layer, in which the modulated symbol data is adjusted by changing phases and amplitudes. The phase decryption is the reverse process in the receiver. The detailed explaining of these operations could be found in [50].

The encryption and decryption block inside the PHY transmitter and receiver of 802.11ah are arranged as shown in Fig. 27. The applied decryption method in [22] could be explained as follows. At the receiver side of the 802.11ah, the decryption block receives output data of FFT block, which consists of in-phase parts (IP) and quadrature parts (QP). From these data, the amplitudes and phases are calculated as in Eq. 7 and Eq. 8:

$$Amplitude = \sqrt{IP^2 + QP^2} \tag{7}$$

$$Phase = \arctan \frac{QP}{IP} \tag{8}$$

Whereas the adjustment of amplitude (Δ_amp) and the adjustment of phase (Δ_phase) are computed according to the value of cipher key. The calculation is shown in Eq. 9 and Eq. 10.

$$\Delta\_amp = \frac{K_a \times M_a}{(2^n - 1)} \tag{9}$$

$$\Delta\_phase = \frac{K_p \times M_p}{(2^n - 1)} \tag{10}$$

Where K_a is the value of cipher key for amplitude, M_a is the maximum value of amplitude, n is the number of bits of cipher key, M_p is the maximum value of phase, K_p is the value of cipher key for phase.

Then the decrypted amplitude ($Decrypted\_Amp$) and decrypted phase ($Decrypted\_Phase$) will be calculated as in Eq. 11 and Eq. 12:

$$Decrypted\_Amp = Amplitude - \Delta\_amp \tag{11}$$

$$Decrypted\_Phase = Phase - \Delta\_phase \tag{12}$$

Next these decrypted amplitude and phase have to convert again to decrypted in-phase ($Decrypted\_IP$) and decrypted quadrature ($Decrypted\_QP$) as shown in Eq. 13 and Eq. 14:

$$Decrypted\_IP = Decrypted\_Amp \times \cos(Decrypted\_Phase) \tag{13}$$

$$Decrypted\_QP = Decrypted\_Amp \times \sin(Decrypte\_Phase) \tag{14}$$

At the end of the decryption operation, the decrypted IP and decrypted QP then pass to the Demapper.

5.3.2 CORDIC Based Calculation Flow

As explained above, the calculation of phases, amplitudes requires multiplication, division, and square root operations. It leads to the high complexity of the hardware circuit of precise calculation. In our work, we exploit the CORDIC algorithm to compute these values approximately.

The CORDIC, also known as Volder's algorithm, is a computing technique to calculate a variety of functions, include trigonometric functions [51].

CORDIC Based Amplitude and Phase Calculation The input data of the decryption have complex values, and we assume that they are presented as $D_i = X + j \times Y$. According to the CORDIC algorithm, the amplitude of the input data can be computed if it is rotated to have a phase of zero; then Y value would be zero, so the amplitude would be given entirely by the new X value. However, in the rotation process, D_i has been multiplied by a CORDIC gain K=1.6467. As a result, the actual value of amplitude has to divide by constant K [52]. The desired phase is the sum of phases after the addition or subtraction angle from the angle lookup table. The details of the algorithm are illustrated in Algorithm 1. The angle lookup table values are indicated in Table. 7.

Algorithm 1 CORDIC Based Amplitude and Phase Calculation

```
Input: In-phase parts (X), Quadrature parts (Y), Angle $Z_0$=0, Angle Look-up Table, K=1.6467
Output: Amplitude and Phase
$X_0$ = $abs(X)$
if $sign(X) \neq sign(Y)$ then
    $Y_0 = -abs(Y)$
else
    $Y_0 = abs(Y)$
end if
for $i \leftarrow 0$ to $N$ do
    if $Y_i >= 0$ then
        $a = 1$
    else
        $a = -1$
    end if
    $X_{i+1} = X_i + a \times (Y_i >> i)$
    $Y_{i+1} = Y_i - a \times (X_i >> i)$
    $Z_{i+1} = Z_i + a \times Angle[i]$
end for
Results
Phase $= Z$
Amplitude $= X/K$
```

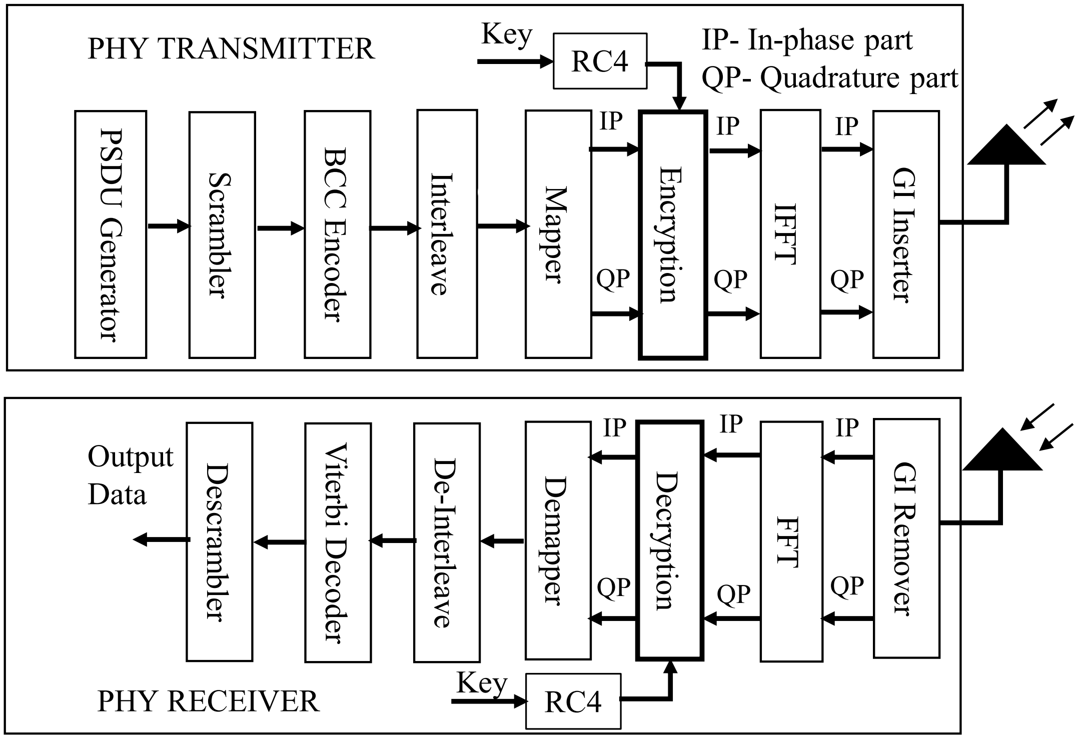


Figure 27. Encryption and decryption inside the PHY transmitter and receiver of 802.11 ah

CORDIC Based Sine and Cosine Calculation of an Angle The main calculation of sine and cosine of an angle based on CORDIC is described in Algorithm 2. The starting values of the rotation are X=1/K and Y=0 at angle zero. If the current angle is less than the desired angle, then it is added to the CORDIC angle from the angle lookup table; otherwise, the current angle is subtracted from the CORDIC angle. This process is continuing until the desired angle is found. The cosine and sine values are the value of X and Y, respectively, at the desired angle.

5.3.3 Hardware Architecture

Overview of Hardware Architecture The overview of the hardware architecture of the decryption is presented in Fig. 28. From IP and QP input data, amplitudes and phases are computed based on the CORDIC algorithm at the

Algorithm 2 CORDIC Based Sine and Cosine Calculation of an Angle

```
Input: Angle(Z), Angle Look-up Table, K=1.6467, X =1/K, Y=0;
Output: Cosine and Sine
    for $i \leftarrow 0$ to $N$ do
        if $Z > 0$ then
            $a = 1$
        else
            $a = -1$
        end if
        $X = X - a \times (Y >> i)$
        $Y = Y + a \times (X >> i)$
        $Z = Z - a \times Angle[i]$
    end for
    Results
    Cosine $= X$
    Sine $= Y$
```

block “AMP PHASE CAL”.“RC4” block generates the cipher keys from the input secret master key by using an RC4 stream cipher. The “DELTA AMP PHASE CAL” block calculates the adjustment of phases and amplitudes by multiplying the cipher keys with corresponding constants according to the modulation types. The constants are chosen on the base of the Eq. 9 and Eq. 10. As a result of calculating the maximum amplitude and the maximum value of cipher key according to each modulation type, the constants for phase and amplitude adjustment are in Table. 8. In the block “DECRYPTED AMP PHASE CAL”, the decrypted amplitudes and phases are computed. From these phases, the cosine and sine values are calculated by using the CORDIC algorithm at the block “ COSINE SINE CAL”. At block “DECRYPTED IP AND QP”, the decrypted in-phase and quadrature parts are calculated according to the Eq. 13 and Eq. 14 respectively.

Hardware Description of CORDIC Based Amplitude and Phase Calculation The CORDIC hardware uses the same iterative operations with only shifters, adders, and subtracters. The hardware of the CORDIC based amplitude

Table 7. Angle lookup table

i	Angle in Degree	Angle in Radian
0	45°	0.7854
1	26.565°	0.4636
2	14.036°	0.2450
3	7.125°	0.1244
4	3.576°	0.0624
5	1.7876°	0.0312
6	0.8938°	0.0156
7	0.4469°	0.0078
8	0.2234°	0.0039
9	0.1117°	0.0019
10	0.0558°	0.0009
11	0.0279°	0.0004

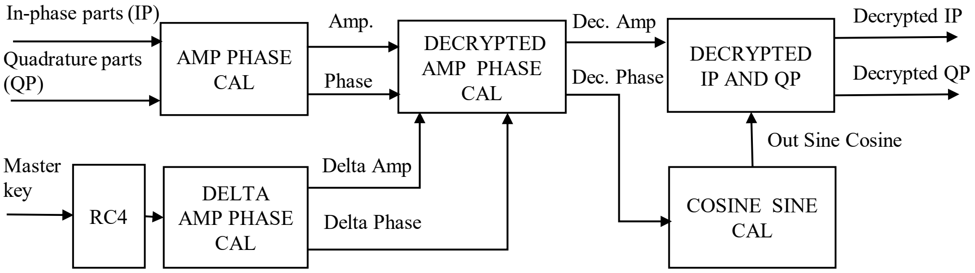


Figure 28. Overview of the decryption hardware architecture

and phase calculation is shown in Fig. 29. It requires a lookup table to store values of the angle table in Tab. 7. For each iteration, the hardware deploys three registers for in-phase part (X), quadrature part (Y), angle (Z), and two shifters for shifting X and Y to the adder/subtractor units. For this calculation, the vectoring mode of the CORDIC is used, and therefore the di-factor (-1, 1) in Algorithm 1 depends on the sign of input Y [52].

In CORDIC, only angles in the range from $-90°$ to $+90°$ can be rotated [52]. Therefore, before starting the iteration, we have to convert the sign of X

Table 8. Constants for phase and amplitude adjustment calculation

Modulation type	For Amplitude	For Phase
16-QAM	$\frac{\sqrt{3^2+3^2}}{2^2-1} = \frac{\sqrt{18}}{3}$	$\frac{2*\pi}{2^2-1} = \frac{2*\pi}{3}$
64-QAM	$\frac{\sqrt{7^2+7^2}}{2^3-1} = \frac{\sqrt{98}}{7}$	$\frac{2*\pi}{2^3-1} = \frac{2*\pi}{7}$
256-QAM	$\frac{\sqrt{15^2+15^2}}{2^4-1} = \frac{\sqrt{450}}{15}$	$\frac{2*\pi}{2^4-1} = \frac{2*\pi}{15}$

to positive, and the sign of input Y must be normalized. If the sign of X and Y differs from each other, the sign of Y needs to convert to negative. Otherwise, the sign of Y is positive.

Based on simulation on Matlab, we choose the number of rotations is twelve. It is enough to guarantee the maximum error rates of calculated amplitudes and phases based on CORDIC compared with using Matlab functions are 0.0037 % and 0.1607 %, respectively.

Hardware Description of CORDIC Based Cosine and Sine Calculation

The hardware of this calculation also performs twelve iterations for approximating the values of cosine and sine. In this calculation, it differs from phase and amplitude calculation; the rotation mode of the CORDIC algorithm is applied and, therefore, the di-factor (-1 and 1) in Algorithm 2 depends on the sign of angle input Z [52]. The angle lookup table is also necessary.

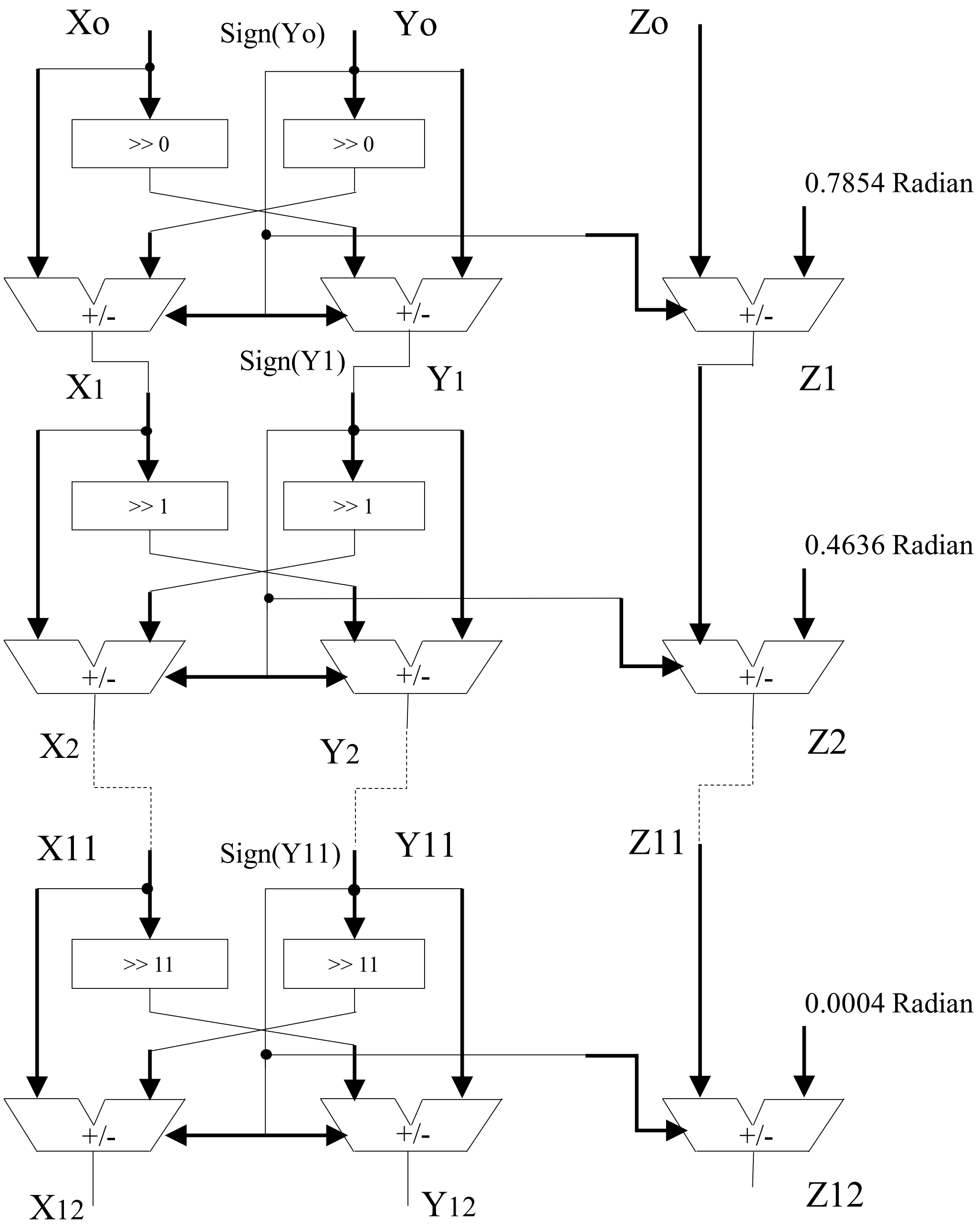


Figure 29. Hardware architecture of CORDIC based phase and amplitude calculation

5.4 Evaluation Results

The hardware was designed by Verilog hardware description language and was verified by using the ModelSim simulation tool. FPGA synthesis was performed with Intel Quartus II using a Cyclone IVE EP4CE115F29C7 device. The ASIC synthesis was executed at 16 Mhz frequency by Synopsys' Design Compiler using VDEC's Rohm 180 nm technology library.

In our encryption and decryption, we merged the encryption with the mapper and the decryption with de-mapper. Therefore, we compare our encryption and decryption with the combined conventional method with mapper and de-mapper, which are called conventional encryption and conventional decryption. According to the FPGA and ASIC syntheses, our encryption and decryption have the following advantages.

Table 9. Hardware resources comparison of the JEM with the conventional method at the same frequency of 16 Mhz

Implemented design	Encryption (Proposed) (+Mapper)	Conventional Mapper (Only)	Conventional Encryption [22] (+Mapper)
Slice registers	51	34	1326
Slice LUTs	229	139	3516

Table 10. Hardware resources comparison of the JEM decryption with the conventional method at the same frequency of 16 Mhz

Implemented design	Decryption (Proposed) (+Demapper)	Conventional Demapper (Only)	Conventional Decryption[22] (+Demapper)
Slice registers	90	54	1352
Slice LUTs	713	629	4016

Low FPGA Resource From the FPGA syntheses result given in Table 9, we can give the following observations. First, the slices LUTs of the encryption are

Table 11. ASIC results comparison of the JEM and the conventional method at the same frequency of 16 Mhz

Implemented design	Encryption (Proposed) (+Mapper)	Conventional Mapper (Only)	Conventional Encryption[22] (+Mapper)
Area(μm^2)	9,902	6,899	436,942
Static Power(mW)	0.09	0.08	3.62

Table 12. ASIC results comparison of the JEM decryption and the conventional method at the same frequency of 16 Mhz

Implemented design	Decryption (Proposed) (+Demapper)	Conventional Only Demapper (Only)	Conventional Deccryption[22] (+Demapper)
Area(μm^2)	124,372	116,628	548,767
Static Power(mW)	1.63	1.55	5.23

just about 6.5% (229 versus 3516) of that of the conventional work. The slice registers of the encryption increase about 50% compared with the conventional mapper (51 versus 34), but about 26 times less than the conventional encryption (51 versus 1326).

As shown in Table 10, the slices LUTs of the decryption are about 17.8% (713 versus 4016) of conventional work and approximately equal to those of the conventional de-mapper (713 versus 629). The slice registers of the decryption are only 6.5% (90 versus 1352) of the conventional decryption.

Small Area As shown in the Table 11, the ASIC area of the proposed encryption is 44 times smaller than the area of the conventional method (9902 versus 436942), although it is about 40 % larger than the area of the conventional mapper (9902 versus 6899). Similarly, as indicated in Table 12, the ASIC area of the decryption is approximately equal to the area of the conventional de-mapper (124372 versus 116628) and about 22 % (124372 versus 548767) of the area of the conventional decryption.

Low Power Consumption According to Table 11, the encryption consumes 40 times less static power than the conventional method (0.09 versus 3.62). In addition, the encryption also does not consume more power than conventional mapper. At the same time, as indicated in Table 12, the static power consumption of the decryption almost does not increase compared to the conventional demapper and is about 30% (1.63 versus 5.32) of the conventional decryption.

5.5 Conclusion

In summary, we can conclude that the proposed encryption, when compared with the conventional mapper, does not significantly increase the FPGA resource usage, ASIC area, or static power. Moreover, as compared with the conventional method, the proposed encryption method has much lower hardware resources, ASIC area, and static power usage. Similarly, compared with the conventional method, the decryption method also greatly reduces hardware resource consumption, ASIC area usage, and static power usage.

6 Conclusion

Physical layer encryption is a promising solution to enhance the security of IoT sensor transceivers. However, it is challenging to realize in practice due to the limitation on the power and computational resources of IoT sensor transceivers. In this dissertation, we have provided two lightweight physical layer encryption methods for IoT sensor transceivers. To evaluate the BER and PER performances of these methods, we have simulated in Matlab using IEEE 802.11 ah standard system model. Furthermore, to analyze the hardware complexity of the proposed methods, we have designed the hardware of the proposed JEM method. We also have implemented the hardware of the conventional method based on CORDIC architecture to compare with our proposed JEM method. These contributions are presented in chapters 3, 4, and 5.

The first lightweight encryption method is presented in chapter 3. In this method, we encrypt only the sign bit of the modulated data. The simulation results have shown that the sign bit encryption method does not degrade the BER and PER performance of the system. We also have evaluated the performances of the 8 MSB encryption method. Since the performances of this method are significantly low, we can not apply this method for IoT sensor transceivers.

The second lightweight encryption method is described in chapter 4. This method is named the joint encryption modulation method. Unlike the sign bit encryption method encrypts only the signed bit, the JEM method encrypts all data. The simulation results showed that our proposed encryption method does not degrade the BER and PER performances of the system compared with unencrypted transmissions, while the conventional method degrades the performance by 3dB.

Based on the hardware implementation in chapter 5, we have verified the practicality and analyzed the complexity of the proposed JEM method. According to the FPGA syntheses results, our proposed JEM method does not significantly increase the FPGA resources compared with the conventional mapper and much lower than the conventional encryption method. The ASIC results showed that the required area for our proposed JEM is about the same as needed for the conventional mapper and about 44 times less than needed for the conventional encryption method. Similarly, the static power usage of the proposed PLE is about 40 times less than needed the conventional method. For the decryption, the ASIC area, and the static power usage are less than the conventional method by about four times, and three times, respectively. For generating cipher keys, we chose and designed the hardware of the stream cipher Grain 128a, which is lightweight, making it suitable for IoT sensor transceivers.

In summary, we can conclude that our proposed JEM substantially reduced the power consumption and hardware complexity compared with the conventional method.

Future work Since it is still challenging to realize in practice the physical layer security in general and particularly physical layer encryption in IoT sensors transceivers. Our future work will focus on the implementation of the physical layer encryption solutions to the real system. We will attempt to design a full wireless digital transceiver for IoT sensors using our proposed encryption and decryption method. Then we will implement in the FPGA to verify the functional operation of the system.

References

[1] J. A. Stankovic. Research directions for the internet of things. *IEEE Internet of Things Journal*, 1(1):3–9, Feb 2014.

[2] Junqing Zhang, Trung Q. Duong, Roger Woods, and Alan Marshall. Securing wireless communications of the internet of things from the physical layer, an overview. *Entropy*, 19(8), 2017.

[3] Y. Zou, J. Zhu, X. Wang, and L. Hanzo. A survey on wireless security: Technical challenges, recent advances, and future trends. *Proceedings of the IEEE*, 104(9):1727–1765, Sep. 2016.

[4] A. Mukherjee, S. A. A. Fakoorian, J. Huang, and A. L. Swindlehurst. Principles of physical layer security in multiuser wireless networks: A survey. *IEEE Communications Surveys Tutorials*, 16(3):1550–1573, Third 2014.

[5] Matthieu Bloch and João Barros. *Physical-Layer Security: From Information Theory to Security Engineering.* Cambridge University Press, 2011.

[6] S. Goel and R. Negi. Guaranteeing secrecy using artificial noise. *IEEE Transactions on Wireless Communications*, 7(6):2180–2189, June 2008.

[7] X. Zhou and M. R. McKay. Secure transmission with artificial noise over fading channels: Achievable rate and optimal power allocation. *IEEE Transactions on Vehicular Technology*, 59(8):3831–3842, Oct 2010.

[8] W. Liao, T. Chang, W. Ma, and C. Chi. Qos-based transmit beamforming in the presence of eavesdroppers: An optimized artificial-noise-aided approach. *IEEE Transactions on Signal Processing*, 59(3):1202–1216, March 2011.

[9] A. Mukherjee and A. L. Swindlehurst. Robust beamforming for security in mimo wiretap channels with imperfect csi. *IEEE Transactions on Signal Processing*, 59(1):351–361, Jan 2011.

[10] A. Khisti and G. W. Wornell. Secure transmission with multiple antennas i: The misome wiretap channel. *IEEE Transactions on Information Theory*, 56(7):3088–3104, July 2010.

[11] N. Romero-Zurita, D. Mclernon, and M. Ghogho. Physical layer security by robust masked beamforming and protected zone optimisation. *IET Communications*, 8(8):1248–1257, May 2014.

[12] L. Dong, Z. Han, A. P. Petropulu, and H. V. Poor. Improving wireless physical layer security via cooperating relays. *IEEE Transactions on Signal Processing*, 58(3):1875–1888, March 2010.

[13] L. J. Rodriguez, N. H. Tran, T. Q. Duong, T. Le-Ngoc, M. Elkashlan, and S. Shetty. Physical layer security in wireless cooperative relay networks: state of the art and beyond. *IEEE Communications Magazine*, 53(12):32–39, Dec 2015.

[14] Li Sun and Qinghe Du. A review of physical layer security techniques for internet of things: Challenges and solutions. *Entropy*, 20:730, 09 2018.

[15] M. A. Khan, M. Asim, V. Jeoti, and R. S. Manzoor. On secure ofdm system: Chaos based constellation scrambling. In *2007 International Conference on Intelligent and Advanced Systems*, pages 484–488, Nov 2007.

[16] L. Zhang, X. Xin, B. Liu, and Y. Wang. Secure ofdm-pon based on chaos scrambling. *IEEE Photonics Technology Letters*, 23(14):998–1000, July 2011.

[17] H. Li, X. Wang, and W. Hou. Secure transmission in ofdm systems by using time domain scrambling. In *2013 IEEE 77th Vehicular Technology Conference (VTC Spring)*, pages 1–5, June 2013.

[18] D. Reilly and G. S. Kanter. Noise-enhanced encryption for physical layer security in an ofdm radio. In *2009 IEEE Radio and Wireless Symposium*, pages 344–347, Jan 2009.

[19] R. Ma, L. Dai, Z. Wang, and J. Wang. Secure communication in tds-ofdm system using constellation rotation and noise insertion. *IEEE Transactions on Consumer Electronics*, 56(3):1328–1332, Aug 2010.

[20] X. Lu, J. Lei, W. Li, K. Lai, and Z. Pan. Physical layer encryption algorithm based on polar codes and chaotic sequences. *IEEE Access*, 7:4380–4390, 2019.

[21] X. Lu, Y. Shi, W. Li, J. Lei, and Z. Pan. A joint physical layer encryption and papr reduction scheme based on polar codes and chaotic sequences in ofdm system. *IEEE Access*, 7:73036–73045, 2019.

[22] M. Kloos. Method and apparatus for encryption of over-the-air communications in a wireless communication system, april 2010. US Patent US7693284B2.

[23] D. Dzung. Data encryption on the physical layer of a data transmission system, Jul 2010.

[24] Dai Long Hoang, Thi Hong Tran, and Yasuhiko Nakashima. Hardware implementation of cordic based physical layer phase decryption for ieee 802.11ah. In *Proceedings of the 7th International Conference on Communications and Broadband Networking*, ICCBN 2019, pages 17–21, 2019.

[25] Y. Huang, W. Li, and J. Lei. Concatenated physical layer encryption scheme based on rateless codes. *IET Communications*, 12(12):1491–1497, 2018.

[26] K. Lai, J. Lei, L. Wen, G. Chen, W. Li, and P. Xiao. Secure transmission with randomized constellation rotation for downlink sparse code multiple access system. *IEEE Access*, 6:5049–5063, 2018.

[27] J. Zhang, A. Marshall, R. Woods, and T. Q. Duong. Design of an ofdm physical layer encryption scheme. *IEEE Transactions on Vehicular Technology*, 66(3):2114–2127, March 2017.

[28] A. K. Nain, J. Bandaru, M. A. Zubair, and R. Pachamuthu. A secure phase-encrypted ieee 802.15.4 transceiver design. *IEEE Transactions on Computers*, 66(8):1421–1427, Aug 2017.

[29] Prachin Bhoyar, S.B. Dhok, and R.B. Deshmukh. Hardware implementation of secure and lightweight simeck32/64 cipher for ieee 802.15.4 transceiver. *AEU - International Journal of Electronics and Communications*, 90:147 – 154, 2018.

[30] Mahmoud Elkhodr, Seyed A. Shahrestani, and Hon Cheung. Emerging wireless technologies in the internet of things: a comparative study. *ArXiv*, abs/1611.00861, 2016.

[31] Alem Čolaković and Mesud Hadzialic. Internet of things (iot): A review of enabling technologies, challenges, and open research issues. *Computer Networks*, 144, 07 2018.

[32] V. Gazis, M. Görtz, M. Huber, A. Leonardi, K. Mathioudakis, A. Wiesmaier, F. Zeiger, and E. Vasilomanolakis. A survey of technologies for the internet of things. In *2015 International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 1090–1095, Aug 2015.

[33] Carles Gomez, Joaquim Oller, and Josep Paradells. Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology. *Sensors*, 12(9):11734–11753, 2012.

[34] Eldad Perahia and Robert Stacey. *Next Generation Wireless LANs: 802.11N and 802.11Ac*. Cambridge University Press, New York, NY, USA, 2nd edition, 2013.

[35] Victor Baños-Gonzalez, M. Shahwaiz Afaqui, Elena Lopez-Aguilera, and Eduard Garcia-Villegas. Ieee 802.11ah: A technology to face the iot challenge. *Sensors*, 16(11), 2016.

[36] Y. Zhao, O. N. C. Yilmaz, and A. Larmo. Optimizing m2m energy efficiency in ieee 802.11ah. In *2015 IEEE Globecom Workshops (GC Wkshps)*, pages 1–6, Dec 2015.

[37] LoRa Alliance . Available Online. `https://lora-alliance.org`. Accessed: 2019-09-20.

[38] U. Raza, P. Kulkarni, and M. Sooriyabandara. Low power wide area networks: An overview. *IEEE Communications Surveys Tutorials*, 19(2):855–873, Secondquarter 2017.

[39] Ieee standard for information technology-telecommunications and information exchange between systems-local and metropolitan area networks-specific

requirements-part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications: Amendment 6: Medium access control (mac) security enhancements. *IEEE Std 802.11i-2004*, pages 1–190, July 2004.

[40] W. Li, D. Mclernon, J. Lei, M. Ghogho, S. A. R. Zaidi, and H. Hui. Cryptographic primitives and design frameworks of physical layer encryption for wireless communications. *IEEE Access*, 7:63660–63673, 2019.

[41] Y. Shiu, S. Y. Chang, H. Wu, S. C. . Huang, and H. Chen. Physical layer security in wireless networks: a tutorial. *IEEE Wireless Communications*, 18(2):66–74, April 2011.

[42] F. Huo and G. Gong. Xor encryption versus phase encryption, an in-depth analysis. *IEEE Transactions on Electromagnetic Compatibility*, 57(4):903–911, Aug 2015.

[43] W. Li, D. Mclernon, K. Wong, S. Wang, J. Lei, and S. A. R. Zaidi. Asymmetric physical layer encryption for wireless communications. *IEEE Access*, 7:46959–46967, 2019.

[44] Z. Zhang, M. C. Y. Cho, C. Wang, C. Hsu, C. Chen, and S. Shieh. Iot security: Ongoing challenges and research opportunities. In *2014 IEEE 7th International Conference on Service-Oriented Computing and Applications*, pages 230–234, Nov 2014.

[45] Thi Hong TRAN, Jr. Leonardo LANANTE, Yuhei NAGAO, and Hiroshi OCHI. Hardware design of multi gbps rc4 stream cipher. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E96.A(11):2120–2127, 2013.

[46] T. H. Tran, H. Kato, S. Takamaeda-Yamazaki, and Y. Nakashima. Performance evaluation of 802.11ah viterbi decoder for iot applications. In *2015 International Conference on Advanced Technologies for Communications (ATC)*, pages 320–325, Oct 2015.

[47] Ieee standard for information technology–telecommunications and information exchange between systems - local and metropolitan area networks–specific requirements - part 11: Wireless lan medium access control (mac)

and physical layer (phy) specifications amendment 2: Sub 1 ghz license exempt operation. *IEEE Std 802.11ah-2016 (Amendment to IEEE Std 802.11-2016, as amended by IEEE Std 802.11ai-2016)*, pages 1–594, May 2017.

[48] R. Ferdian, Y. Hou, and M. Okada. A low-complexity hardware implementation of compressed sensing-based channel estimation for isdb-t system. *IEEE Transactions on Broadcasting*, 63(1):92–102, March 2017.

[49] Martin Ågren, Martin Hell, Thomas Johansson, and Willi Meier. Grain-128a: a new version of grain-128 with optional authentication. *IJWMC*, 5:48–59, 2011.

[50] D. L. Hoang, T. Hong Tran, and Y. Nakashima. Performance evaluation of 802.11ah physical layer phase encryption for iot applications. In *2018 International Conference on Advanced Technologies for Communications (ATC)*, pages 84–88, Oct 2018.

[51] J. E. Volder. The cordic trigonometric computing technique. *IRE Transactions on Electronic Computers*, EC-8(3):330–334, Sep. 1959.

[52] P. K. Meher, J. Valls, T. Juang, K. Sridharan, and K. Maharatna. 50 years of cordic: Algorithms, architectures, and applications. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 56(9):1893–1907, Sep. 2009.

Acknowledgements

First of all, I would like to express my sincere thanks to Professor Yasuhiko Nakashima for giving me a great opportunity to study and research in the high-level laboratory, guiding me in research, providing valuable comments, and encouraging me during my research.

Then, I want to thank Professor Minoru Okada for delivering me helpful comments and advice about my research as a co-supervisor. He raised a lot of essential questions that help me to improve my study a lot.

Next, I would like to give thanks to Affiliate Professor Mutsumi Kimura, Associate Professor Takashi Nakada, and Assistant Professor Renyan Zhang for their guidance, supports, and comments for growing my research.

Especially, I would like to give many thanks to Assistant Professor Tran Thi Hong. As a close co-supervisor, she has supported me from my beginning study at Computing Architecture Lab until graduation. She taught me to do the research, provided vital comments, and guided me a lot on writing scientific papers.

Furthermore, I spread my thanks to the Ministry of Education and Training (MOET) of Vietnam, Vietnamese International Education Department (VIED) for the scholarship grant, and financial support during my doctoral course in Nara Institute of Science and Technology (NAIST). I also would like to acknowledge NAIST for giving me a three-year exemption fee for my doctoral course.

I want to acknowledge all the past and present members of the Computing Architecture Lab for their help and unforgettable time. Thanks to the Vietnamese friends in NAIST for all spirit support and enjoyable time together.

Besides, I want to express my acknowledgments to all staff members in the Division of Information of Science and in NAIST, especially members in the International Student Affairs Section, for all their supports to me. I also want to give thanks to the volunteer enthusiastic Japanese teachers who taught me the Japanese language at NAIST.

Last but not least, my sincere gratitude to my family, especially my beloved wife Phuong Nhi and my two lovely daughters Phuong Linh and Thuy An, for their encouragement, love, and unconditional support during my doctoral course far from home.