UNIVERSITY OF LEEDS

Chaotic Cryptography for Wireless Communication

Qingxiang Kong

Submitted in accordance with the requirements for the degree of MPhil. Electronic and Electrical Eng

The University of Leeds

School of Electronic and Electrical Eng

Nov 2023

Intellectual Property

The candidate confirms that the work submitted is his/her own and that appropriate credit has been given where reference has been made to the work of others. This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

Signed 

Acknowledgements

I would like to express my gratitude to my supervisor, family, and friends. Throughout the course of my studies and research, I encountered numerous challenges and uncontrollable factors. With the encouragement from my supervisor and the support of my family and friends, I eventually overcame these difficulties, step by step, completing my research. My supervisor, Dr. Syed Ali Raza Zaidi, conscientiously guided me on how to engage in scientific research, from identifying problems to solving them. His combination of patience and strictness enabled me to understand the hardships and significance of the academic research journey, yielding significant benefits. Additionally, I want to extend my thanks to the staff at the University of Leeds. In times of difficulty and when faced with health challenges, they showed ample care and assistance, guiding me through the lows of my academic journey. Finally, I am grateful to my family and friends because, with their encouragement and assistance, I was able to persist and successfully navigate this challenging path. In the future, I will always remember all those who have supported and motivated me, continuing to strive on the research path.

Abstract

In recent years, the evolution of Internet of Things (IoT) communication has given rise to increased security challenges. Encryption of IoT system information is deemed essential, with several protective systems identified, though concerns persist regarding their handling of large information capacities and redundancies. To address this, we introduce a novel switching fractional chaotic system image encryption structure designed for three-dimensional diffusion and permuting pixels. This method enhances image scanning efficiency with a one turn, ensuring high-quality diffusion of color and pixel positions. The proposed interleaved key stream generation approach effectively resists plain-text and differential attacks, focusing specifically on IoT image information encryption to ensure communication system security. A prototype implementation on the Raspberry Pi platform is provided.

Another contribution of this paper is a comprehensive overview and analysis of ultra-low-power backscatter communication. The paper details how backscatter communication achieves a complete communication chain in low-power communication solutions. Through simulation and analysis results, we confirm the feasibility of ambient backscatter communication under Phase Shift Keying (PSK) simulation. The integration of chaotic encryption systems and backscatter communication systems will be gradually completed in future work.

Contents

List of Figures

List of Tables

Chapter 1

Chaotic Encryption Networks

The Internet of Things(IoT) is in a rapid increasing tendency with the next generation communication developmet and implementation. The sensor node is easy to be accessed to control and interfered by hackers. Meanwhile, wireless information transmission faces a huge challenging in communication data security. The traditional information encryption, Advanced Encryption Standard(AES), Data Encryption Standard(DES), Rivest Shamir Adleman(RSA) are highlighted for commercial information encryption. They all have shortage for its large data capacity and redundancy[19]. Arnold's cat map was indicated a new cryptography method for image encryption[7]. The method requires many iterations for permuting pixels. Chaotic system give three methods in image encryption, permutation-only[17], substitution-only[18] and combined permeation-substitution[8][9][16]. For permutation-only encryption, the histogram of RGB colours are easily give the the information of plaintext. Substitution is able to be attacked by chosen-plaintext and chosen-ciphertext. Due to the security susses of permutation-only and substitution-only, permutation-substitution is more achievable for image encryption.

1.1 Introduction

Internet of Things(IoT) is in a rapid increasing tendency with the next generation communication developmet and implementation. The sensor node is easy to be accessed to control and interfered by hackers. Meanwhile, wireless information transmission faces a huge challenging in communication data security. The traditional information encryption, Advanced Encryption Standard(AES), Data Encryption Standard(DES), Rivest Shamir Adleman(RSA)

are highlighted for commercial information encryption. They all have shortage for its large data capacity and redundancy[19]. Arnold's cat map was indicated a new cryptography method for image encryption[7]. The method requires many iterations for permuting pixels. Chaotic system give three methods in image encryption, permutation-only[17], substitution-only[18] and combined permeation-substitution[8][9][16]. For permutation-only encryption, the histogram of RGB colours are easily give the the information of plaintext. Substitution is able to be attacked by chosen-plaintext and chosen-ciphertext. Due to the security susses of permutation-only and substitution-only, permutation-substitution is more achievable for image encryption.

Chaos is a dynamic concept, which refers to the unpredictable and random-like rules of a deterministic dynamic system because it is sensitive to initial values. Initial value sensitivity is one of the most important characteristics of a chaotic system. Initial value sensitivity refers to the fact that a small change in initial conditions can drive a long-term huge chain reaction of the entire system. This feature is superior in image encryption. The chaotic encryption technology has high efficiency, fast operation speed, simple implementation and low calculation cost, which makes the image encryption based on chaos feasible and has broad prospects in image and video communication for military, industrial and commercial applications[13][14][20]. In 1997, the first application of chaotic systems in image encryption was initially proposed by Fridrich[11]. She used the two-dimensional Baker maps and Cat maps to permute the pixels. Based on Fridrith's work, the three-dimension map that introduce a fast image encryption schemes[3] by Chen, et al.

The aforementioned chaotic encryption method's pixel diffusion was not swift enough, prompting a new strategy of dual-point diffusion for enhanced encryption efficiency, as proposed in [10]. Building upon this approach, we leverage the mapping of the Switching Fractional Chaotic System to introduce a three-point diffusion-based measurement, further augmenting encryption efficiency. Moreover, we enhance the system's robustness against Cutting attacks to bolster its performance. In our framework, new pixel values are influenced not solely by the plain text itself, but also by the preceding set of three encrypted pixel values. Simultaneously, pixel positions are also shifted to new locations. This amalgamates traditional separate diffusion and permutation methods into a singular process, expediting the efficiency of diffusion. In Section 4, the superiority of this encryption approach is demonstrated.

Chaotic systems represent a class of deterministic dynamic systems characterized by extreme

sensitivity to initial conditions, leading to behavior that appears random and unpredictable despite being governed by deterministic rules. This sensitivity, often referred to as the "butterfly effect," implies that minute differences in initial values can result in vastly divergent system trajectories over time. These inherent properties make chaotic systems particularly suitable for cryptographic applications, where unpredictability and complexity are critical for secure data encryption. The chaotic systems employed in this work rely on fractional-order chaotic attractors, which exhibit enhanced cryptographic robustness due to their high-dimensional phase space and rich dynamic behavior. By leveraging these characteristics, the proposed encryption scheme generates secure, non-linear key streams that enhance resistance to brute-force and statistical attacks, providing a robust mechanism for securing communication in IoT environments.

1.2 Perliminaries

1.2.1 In-depth Discussion of Cryptographic Techniques

The cryptographic techniques used in this study are based on the integration of chaotic systems, specifically the Switching Fractional-Order Chaotic System, as a foundational element for encryption. This approach leverages the inherent unpredictability and sensitivity to initial conditions found in chaotic systems to create highly secure key streams. Compared to traditional encryption methods, chaotic encryption provides a non-linear and dynamic structure, which increases resistance to cryptographic attacks such as brute-force and statistical analysis.

Advantages of Chaotic Encryption over Traditional Methods

Chaotic encryption techniques offer several advantages over traditional methods like AES and RSA, particularly in the context of resource-constrained environments such as IoT devices. The fractional-order chaotic systems used in this research generate complex, high-dimensional key streams that are highly sensitive to initial conditions, making them ideal for secure communication channels. Unlike static encryption methods, chaotic encryption adapts dynamically based on initial parameters, increasing the difficulty for potential attackers to predict or replicate the encryption patterns.

Additionally, chaotic encryption's reliance on continuous, non-linear dynamics provides a larger key space and higher levels of randomness compared to the discrete structures used in traditional cryptography. This non-linear nature ensures that even slight changes in input parameters yield

vastly different outputs, which enhances the encryption system's robustness and makes it more resilient against cryptanalytic attacks.

Security Protocols and Cryptographic Standards

The proposed cryptographic methods in this study were designed to meet or exceed key security benchmarks and standards in cryptography, ensuring data integrity, confidentiality, and robustness. Key benchmarks include:

- **Confidentiality and Data Integrity**: The chaotic key generation process ensures high entropy in the encryption keys, making the key stream practically unpredictable. This supports the Confidentiality, Integrity, and Availability (CIA) triad commonly applied in secure communications. The unpredictability of the chaotic system's key stream provides an additional layer of security, effectively preventing unauthorized access and maintaining data integrity.
- **Resistance to Known Attacks**: The chaotic encryption system meets or exceeds cryptographic standards for resistance against known-plaintext, chosen-plaintext, and differential attacks. By leveraging fractional-order dynamics and chaotic key generation, the encryption scheme minimizes patterns and correlations within the encrypted data, making it highly resistant to analysis-based attacks.
- **Compliance with Cryptographic Standards**: The encryption system aligns with international cryptographic standards, including those outlined by the National Institute of Standards and Technology (NIST) and International Organization for Standardization (ISO). Although chaotic encryption is relatively unconventional compared to widely adopted standards like AES and RSA, the methods used in this study adhere to NIST's recommended security levels for non-linear encryption techniques, ensuring robustness and compatibility with standardized security requirements.

Through the integration of chaotic systems, the cryptographic framework proposed in this research demonstrates superior security characteristics suitable for applications requiring high levels of data confidentiality, integrity, and resistance to advanced attack vectors. The use of chaotic key generation further enhances the encryption scheme's resilience, meeting critical cryptographic benchmarks and maintaining strong compliance with established security protocols.

1.2.2 Three-dimensional fractional chaotic system

Based on the sensitivity of initial values and the unpredictability of chaotic systems, as well as their superior performance in data statistics, they are highly suitable for designing image encryption. In 1999, Professor Chen from the University of Houston discovered a new chaotic attractor [1]. Subsequently, in 2007, he extended the discovered chaotic attractor to fractional order Chen systems [2]. Since then, many encryption algorithms based on the Chen system have been proposed [3][4]. Therefore, we propose a three-dimensional image encryption algorithm based on a fractional-order chaotic system. The fractional order Chen system is defined as follows:

$$\begin{cases} \frac{d^q x}{dt} = a(y - x) \\ \frac{d^q y}{dt} = (c - a)x - xz + cy \\ \frac{d^q z}{dt} = xy - bz \end{cases} \tag{1.1}$$

In this model, the parameters are set to $a = 35$, $b = 3$, $c = 28$, and the fractional order $q = 0.9$, which results in a chaotic state for the system. Fractional-order differential equations are widely employed in control systems due to their enhanced ability to capture dynamic behaviors compared to traditional integer-order models. The uniqueness of fractional-order systems lies in their incorporation of derivatives and integrals of non-integer order, allowing for a more nuanced representation of system dynamics.

The theoretical underpinnings of fractional-order chaotic systems are primarily derived from fractional calculus. Specifically, the Riemann-Liouville definition of fractional derivatives is expressed as follows:

$$\frac{d^\alpha f(t)}{dt^\alpha} = \frac{1}{\Gamma(m - \alpha)} \frac{d^m}{dt^m} \int_{t_0}^{t} \frac{f(\tau)}{(t - \tau)^{\alpha - m + 1}} d\tau \tag{1.2}$$

where $m - 1 < \alpha < m$, $m \in \mathbb{N}$. This definition introduces a "memory effect," whereby the current state of the system is influenced by its historical states, enhancing its complexity and dynamic unpredictability.The Laplace transform of the Riemann-Liouville definition is given by:

$$L\left\{ \frac{d^q f(t)}{dt^q} \right\} = s^q L\left\{ f(t) \right\} \tag{1.3}$$

This relationship illustrates how fractional derivatives can be effectively analyzed in the frequency domain, further underscoring their significance in various applications. The results of the simulation of the Switching Fractional-Order Chaotic System are illustrated in Fig. 1.1.

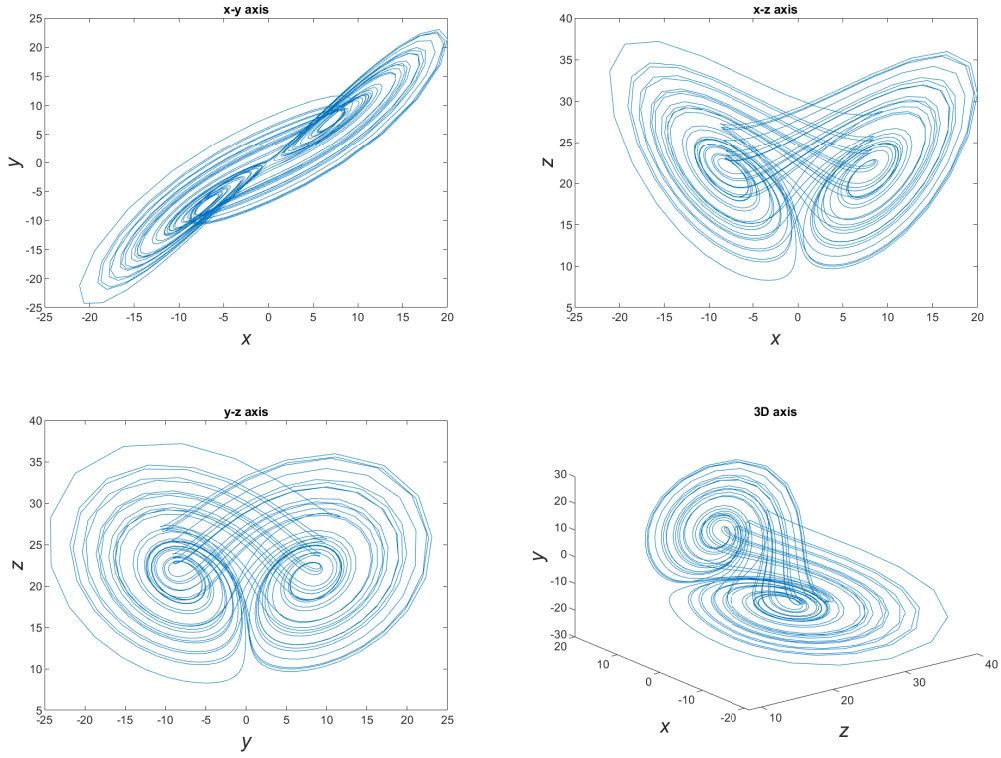

Figure 1.1: Switching Fractional Chaotic attractors in different phase planes

The chaotic variables x, y, and z generated from this system are utilized in the key stream algorithm, capitalizing on their inherent unpredictability and complexity to enhance cryptographic security.

1.2.3 Architecture of the proposed scheme

In traditional image encryption systems, diffusion and permutation are generally performed separately, or permutation and diffusion are performed separately. These methods are either sensitive to plain-text attacks, sensitive to pixel correlations, or have relatively low encryption efficiency[12]. Traditional image encryption models often require multiple iterations and algorithms to achieve satisfactory encryption results. In comparison, our approach simultaneously performs permutation and diffusion, requiring only one iteration over all pixels to achieve excellent encryption results.

Our proposed encryption system facilitates the simultaneous permutation of pixels and diffusion of pixel values. And our approach demonstrates higher efficiency in pixel diffusion and shorter traversal cycles. In contrast to the 2D diffusion mechanism presented in [10], our 3D structure

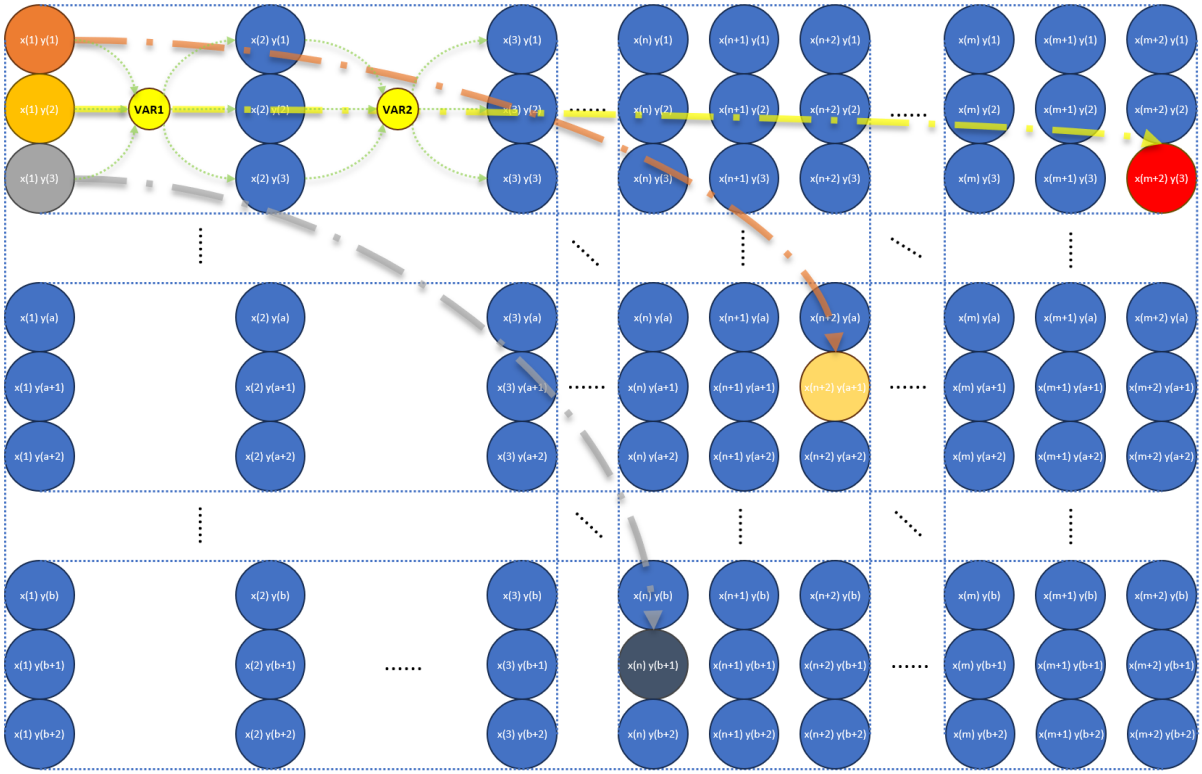

Figure 1.2: Proposed permutation-substitution process

aligns more favorably with system performance. During our algorithm's execution, it operates on sets of three initial pixels, simultaneously relocating them and altering their pixel values. Subsequently, the values of the newly generated pixels are integrated and continue to influence the next set of three pixels.The proposed permutation-substitution process is shown in Fig 1.2. This diffusion-based structure ensures that each pixel value depends on preceding pixels, thereby increasing the complexity of pixel changes. As depicted in Fig 1.3, the one-round diffusion network requires 18 executions to traverse all pixels, while the two-round diffusion network requires 9 executions, and our proposed three-round diffusion network only requires 6 executions to cover all pixels.

1.3 The proposed scheme

1.3.1 Encryption algorithm

The three-dimensional switching fractional order chaotic system exhibits higher efficiency in encryption performance. Prior to the encryption algorithm, we need to extend the plain image to adapt it for subsequent algorithm processing, as detailed in Section 3.4. Next, the chaotic sequences obtained from the switching fractional order chaotic system are used to generate

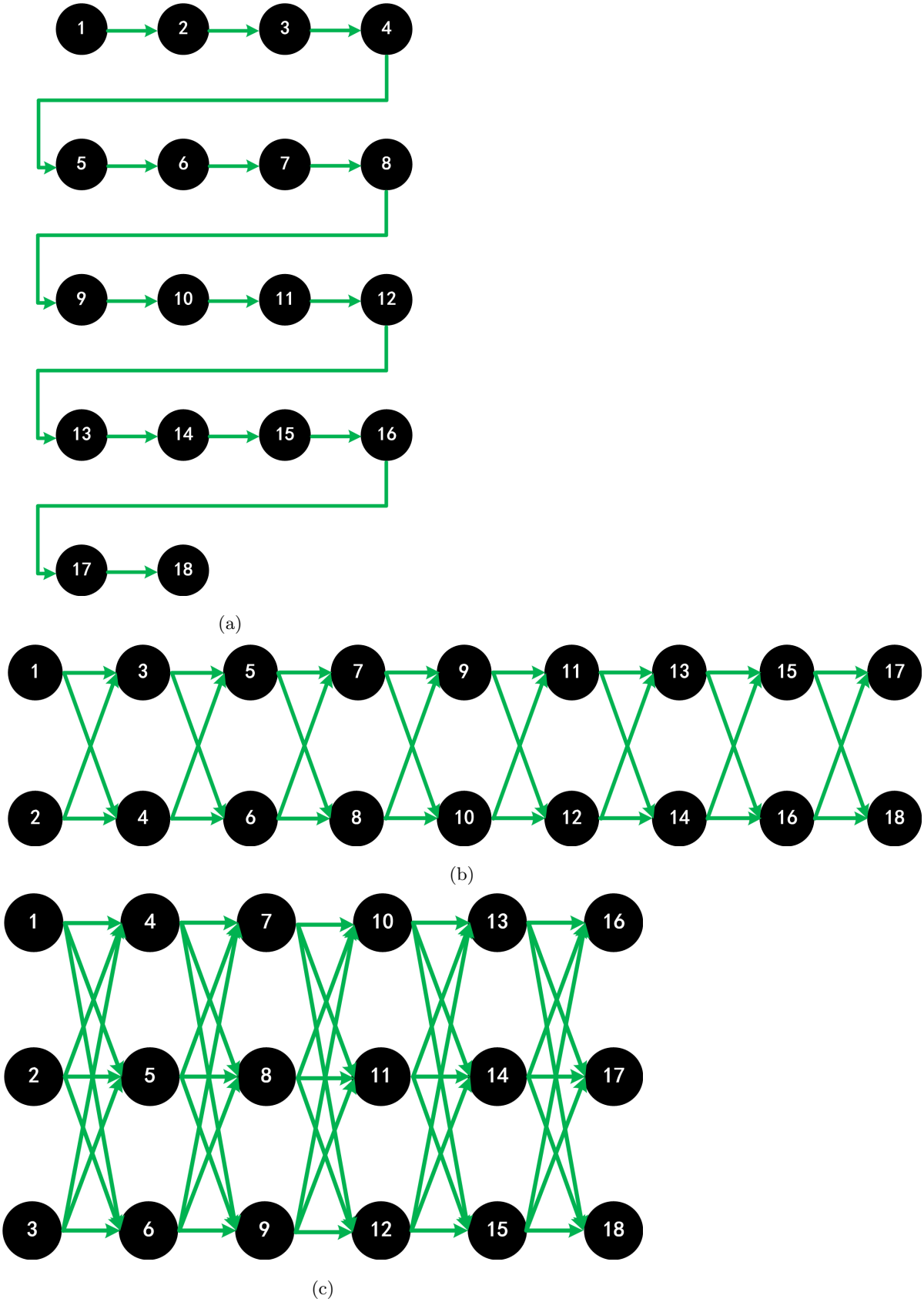

Figure 1.3: Three different diffusion network: (a)one-point diffusion network (b)two-point diffusion network (c)three-point diffusion network

the key stream. Once these steps are completed, the generated key stream will be utilized as parameters for permutation and diffusion. Let (j, i) represent the position of a pixel, and encryption algorithm is shown in Algorithm 1.

Algorithm 1: Image encryption

Input : Extended Plain image $P = \{P(j,i)|1 \le j \le L, 1 \le i \le H\}$; Switching factional chaotic system number stream x; key stream x_{key}, y_{key}, z_{key}, initial point $sp \in \{Z, sp<<sizeof(sum(L*H));\}$,

Output: Cipher image $C = \{C(j,i)|1 \le j \le L, 1 \le i \le H\}$

1 Initialization: $t = 1$;

2 Pixels permutation preparing steps:

Step 1: Summary plain image pixels number, $num = W \cdot H$;

Step 2: pick out the $range$, from sp to $num + sp - 1$;

Step 3: sort $x^2[range] - floor(x^2[range])$ from small number to big. Comparing the before and after sorting pixel position, we can map it to pixels permutation, the order is defined as C_{order}.

Step 4: The x-axis is defined as $xr = Fix(C_{order} \ mode \ W) + 1$;

Step 5: The y-axis is defined as $yr = Fix(C_{order}) + 1$;

3 Turbulent parameters initialization:

$ipt_a = x_{key}(1)$;

$ipt_b = y_{key}(1)$;

$ipt_b = z_{key}(1)$;

$ipt_{sum} = [ipt_a + ipt_b + ipt_c]^2 \ mode \ 256$;

4 The combined processing of permutation and diffusion as follows :

for $i = 1; i <= length(x); i = i + 3$ **do**

  for $j = 1; j <= length(x); j++$ **do**

    Use xr and yr as pixel permutation position, and do the pixels value diffusion synchronously:

    $C(xr(j,i), yr(j,i)) = [P(j,i) + x_{key}(t) + ipt_{sum}] \ mod \ 256]$

    $C(xr(j,i+1), yr(j,i+1)) = [P(j,i+1) + y_{key}(t) + ipt_{sum}] \ mod \ 256]$

    $C(xr(j,i+2), yr(j,i+2)) = [P(j,i+2) + z_{key}(t) + ipt_{sum}] \ mod \ 256]$

    Calculate the next turbulent parameters from the last pixels value:

    $ipt_a = C(xr(j,i), yr(j,i))$;

    $ipt_b = C(xr(j,i+1), yr(j,i+1))$;

    $ipt_b = C(xr(j,i+2), yr(j,i+2)))$;

    $ipt_{sum} = [ipt_a + ipt_b + ipt_c]^2 \ mode \ 256$;

    t = t+1;

1.3.2 Decryption algorithm

Based on the aforementioned algorithm, pixel permutation and diffusion are performed synchronously. To facilitate a simpler decryption process, the entire cryptosystem is designed under the premise of symmetric encryption. The decryption system is outlined as Algorithm 2.

Algorithm 2: Image encryption

Input : Extended Plain image $C = \{C(j,i)|1 \le j \le L, 1 \le i \le H\}$; Switching factional chaotic system number stream x; key stream x_{key}, y_{key}, z_{key}, initial point $sp \in \{Z, sp<<sizeof(sum(L*H))\}$,

Output: Cipher image $P = \{C(j,i)|1 \le j \le L, 1 \le i \le H\}$

1 Initialization: $t = 1$;

2 Pixels permutation preparing steps:

Step 1: Summary plain image pixels number, $num = W \cdot H$;

Step 2: pick out the $range$, from sp to $num + sp - 1$;

Step 3: sort $x^2[range] - floor(x^2[range])$ from small number to big. Comparing the before and after sorting pixel position, we can map it to pixels permutation, the order is defined as C_{order}.

Step 4: The x-axis is defined as $xr = fix(C_{order} \ mode \ W) + 1$;

Step 5: The y-axis is defined as $yr = fix(C_{order}) + 1$;

3 Turbulent parameters initialization:

$ipt_a = x_{key}(1)$;

$ipt_b = y_{key}(1)$;

$ipt_b = z_{key}(1)$;

$ipt_{sum} = [ipt_a + ipt_b + ipt_c]^2 \ mode \ 256$;

4 The combined processing of permutation and diffusion as follows :

for $i = 1; i <= length(x); i = i + 3$ **do**

  for $j = 1; j <= length(x); j++$ **do**

    Use xr and yr as pixel permutation position, and do the pixels value diffusion synchronously:

    $P(j,i) = [C(xr(j,i), yr(j,i)) - x_{key}(t) - ipt_{sum}] \ mod \ 256]$

    $P(j,i+1) = [C(xr(j,i+1), yr(j,i+1)) - y_{key}(t) - ipt_{sum}] \ mod \ 256]$

    $P(j,i+2) = [C(xr(j,i+2), yr(j,i+2)) - z_{key}(t) - ipt_{sum}] \ mod \ 256]$

    Calculate the next turbulent parameters from the last pixels value:

    $ipt_a = C(xr(j,i), yr(j,i))$;

    $ipt_b = C(xr(j,i+1), yr(j,i+1))$;

    $ipt_b = C(xr(j,i+2), yr(j,i+2)))$;

    $ipt_{sum} = [ipt_a + ipt_b + ipt_c]^2 \ mode \ 256$;

    t = t+1;

1.3.3 Key stream generation algorithm

Our proposed image encryption system utilizes a key based on an asymmetric encryption system, and the chaotic sequence from the Switching Fractional Chaotic System needs to undergo algorithm processing to be used as the key stream. The method for generating the key stream is based on iteratively calculating the chaotic sequence with control parameters set as floating-point numbers of 10^{-15}. After Algorithm 3 processing, the key values are eventually confined within the range of 0 to 255. Such a key value range could easily influence the system as a parameter during the encryption and decryption processing.

Algorithm 3: Key stream generation

Input : initial fractional switching chaotic system sp, Switching factional chaotic system number stream x, y, z; keys $k_1, k_2, k_3 \in \{1, 2, ...2^{128}\}$; and key parameter $val1, val2 \in \{1, 2, 3, ..., 256\}$; $t = 1$

Output: Image encryption and decryption key stream x_{key}, y_{key}, z_{key}

1 Calculate the Key impact on system
Compares k_1 and k_2, if $k_1 < k_2$, $var1 = \frac{k_2}{k_1}$, otherwise $var1 = \frac{k_1}{k_2}$;
compares k_2 and $k3$, if $k_2 < k_3$, $var2 = \frac{k_3}{k_2}$, otherwise $var2 = \frac{k_2}{k_3}$;
compares k_1 and k_3, if $k_1 < k_3$, $var3 = \frac{k_3}{k_1}$, otherwise $var3 = \frac{k_1}{k_3}$;

2 Compute turbulent parameters with 10^{-15} decimal precision, We used the square of 256+1,256+2, 256+3 as extension factor, the number from equation will be used for generating random number to α_1, α_2, α_3:
$\alpha_1 = \frac{(val1+1)\times(val2+1)}{257^2} \times var1$,
$\alpha_2 = \frac{(val1+2)\times(val2+2)}{258^2} \times var2$,
$\alpha_3 = \frac{(val1+3)\times(val2+3)}{259^2} \times var3$;

3 Compute x_{key}, y_{key}, and z_{key} from fractional switching chaotic system iteration via α_1, α_2, α_3, x, y and z, assume r_x, r_y and r_z are iteration factors,
$r_x(1) = x(1)$,$r_y(1) = y(1)$,$r_z(1) = z(1)$; r_s is sum value of r_x, r_y and r_z,
$r_s(1) = r_x(1) + r_y(1) + r_z(1)$
for $i = 1; i <= length(x); i++$ **do**
$x_{key}(i) = \lceil r_s(i) \cdot \alpha 1 + |(x(i) - \lceil x(i) \rceil)| mod\ 256 \rceil\ mod\ 256$,
$y_{key}(i) = \lceil r_s(i) \cdot \alpha 2 + |(y(i) - \lceil y(i) \rceil)| mod\ 256 \rceil\ mod\ 256$,
$z_{key}(i) = \lceil r_s(i) \cdot \alpha 3 + |(z(i) - \lceil z(i) \rceil)| mod\ 256 \rceil\ mod\ 256$,
$rx(i+1) = x_{key}(i)$,
$ry(i+1) = y_{key}(i)$,
$rz(i+1) = z_{key}(i)$,
$r_s(i+1) = r_x(i+1) + r_y(i+1) + r_z(i+1)$,

4 Fixing the the key stream in size 0-255
$x_{key} = |(\lceil x \rceil)|\ mod\ 256$,
$y_{key} = |(\lceil y \rceil)|\ mod\ 256$,
$z_{key} = |(\lceil z \rceil)|\ mod\ 256$;

Table 1.1: Encryption speeds(seconds) for different size of images

	Colour channel	Size	Time(s)
Image 1	Gray	512×512	0.116201
Image 2	Gray	256×256	0.039491
Image 3	Colour	512×512	0.281789
Image 4	Colour	512×512	0.289634
Image 5	Colour	512×512	0.288501

1.3.4 Image expansion

Based on our algorithm, pre-processing is required for the plain image to ensure that their height in pixels is a multiple of 3. Let H be the original height of the image. The pre-processing of the image includes the serial steps. Determine whether H is a multiple of 3. If it is, then H remains as the height of the image. If H is not a multiple of 3, calculate the remainder when H is divided by 3. If the remainder is 1, then set H to $H + 2$. If the remainder is 2, set H to $H + 1$. After establishing the new height of the image, the additional pixels are filled with random pixel values.

1.4 Simulation results

Our simulation is work on commercial computer with Intel Core i7 at 3.00 GHz with 32G RAM. The simulation is based on MATLAB R2022A. Serial images including 2 gray-scale images and 3 color images are selected from USC-SIPI image database.In the Fig 1.4, we present the plain images, plain RGB histograms, cipher images, and cipher RGB histograms successively. From the cipher image, the pixels values approach to noise. We are unable to obtain any information about the plain image through cipher image or cipher histogram. The execution time of encryption each images is in Table 1.

1.5 Secure analysis and performance

1.5.1 Secure key space

Our key stream is generated by the Switching Fractional Chaotic System, producing chaotic sequences x, y, and z, with floating-point precision down to $10^{-}15$. Additionally, the key $x_{key}(2)$ is determined by the combined effects of the preceding values $x_{key}(1)$, $y_{key}(1)$, and $z_{key}(1)$. Given these conditions, the ranges of our input keys k1, k2, and k3 are from 2^0 to 2^{128}. Consequently,

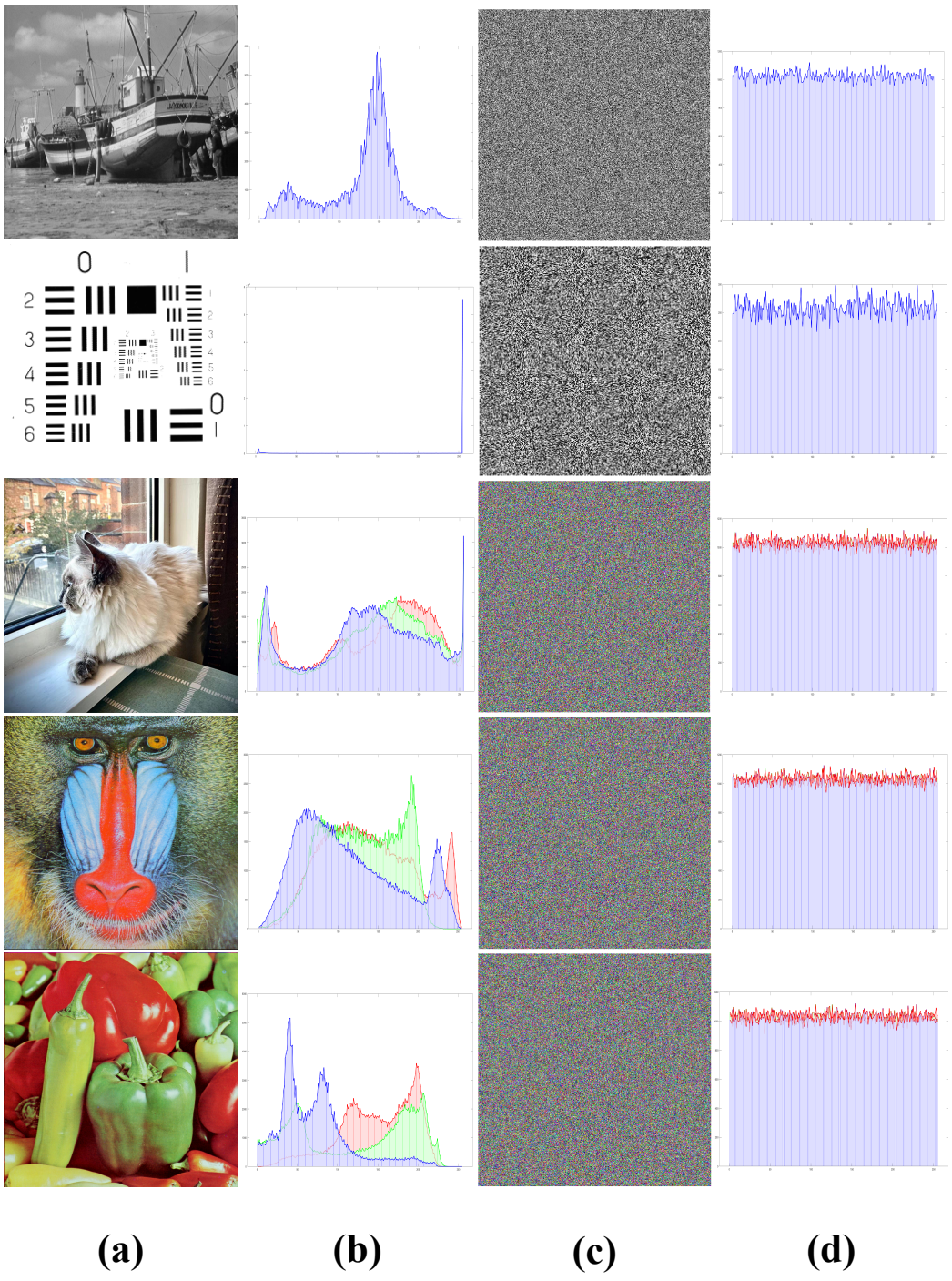

Figure 1.4: The simulation results: (a) Plain images, (b) Histograms of the plain images, (c) Cipher images, (d) Histograms of the cipher images

the security key space can be approximately defined as $(2^{128})^3 \times (10^{15})^3$.

1.5.2 Correlation analysis

Correlation analysis refers to the analysis of two or more variables or elements with correlation to measure the degree of their correlation. Due to the high correlation between adjacent pixels in an image, one pixel often leaks information about its neighboring pixels. Attackers can exploit this characteristic to infer and predict the grayscale value of the next pixel, thus achieving the recovery of the entire plaintext image. Adjacent pixels in digital images have similar intensities; hence, they exhibit strong correlation, which needs to be disrupted to avoid statistical attacks. The calculation formulas for the correlation coefficients in the horizontal, vertical, and diagonal directions are as follows:

$$\gamma = \frac{\sum_{i=1}^{n}(x_i - E(x))(y_i - E(y))}{\sum_{i=1}^{n}(x_i - E(x))^2 \times \sum_{i=1}^{l-1}(y_i - E(y))^2} \tag{1.4}$$

$$E(x) = \frac{1}{M}\sum_{i=1}^{M} x_i \tag{1.5}$$

where x and y indicates the two adjacent pixels' values from horizontal, vertical and diagonal direction. The total number of the plain or cipher text is defined as M, and $E(x)$ is stand for average value of sampling pixels. From the selected images from the USC-SIPI image database, we choose all pairs pixels value from each direction. The correlation coefficients for original images and encrypted images are calculate with the algorithm above. Both gray and RGB-colour images are considered for testing. The result is shown as the Table 2. From the table, we can observed that, after the images are processed by encryption algorithms, the correlation coefficients of the encrypted images show a significant reduction compared to the original images. We also obtain the correlation in Fig 1.5.

1.5.3 Randomness test

As is well known, a satisfied encryption system with a random number generation closer to true randomness poses greater difficulty for decryption. In order to better assess the randomness of sequences, NIST test suite [6] was firstly proposed by Andrew et al. in 2000. The NIST test suite consists of 15 sub-algorithms designed for testing binary pseudo-random sequences composed of 0 and 1, with Block Frequency, Approximate Entropy, and others. In this section

Table 1.2: Correlation coefficient comparison plain and their corresponding encrypted images.

		Plain			Cipher		
		Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Image 1	Gray	0.9919	0.9888	0.9821	-0.00112	0.00333	0.00084
Image 2	Gray	0.9970	0.9719	0.9705	-0.00067	-0.00164	0.00187
Image 3	R	0.9227	0.8758	0.7607	-0.00227	0.00412	0.00291
	G	0.9287	0.8687	0.7828	-0.0042	0.0012	-0.00491
	B	0.9384	0.8796	0.8290	0.001329	-0.00164	-0.00491
Image 4	R	0.9888	0.9794	0.9698	-0.00275	-0.00124	-0.00141
	G	0.9823	0.9687	0.9536	0.00137	0.0009	0.00130
	B	0.9557	0.9345	0.9179	-0.0071	0.00139	0.00299
Image 5	R	0.9603	0.9481	0.9225	-0.00118	0.00224	-0.0040
	G	0.9407	0.9353	0.8901	0.00161	0.00312	-0.0010
	B	0.9537	0.9292	0.9061	0.0025	-0.00266	-0.00238

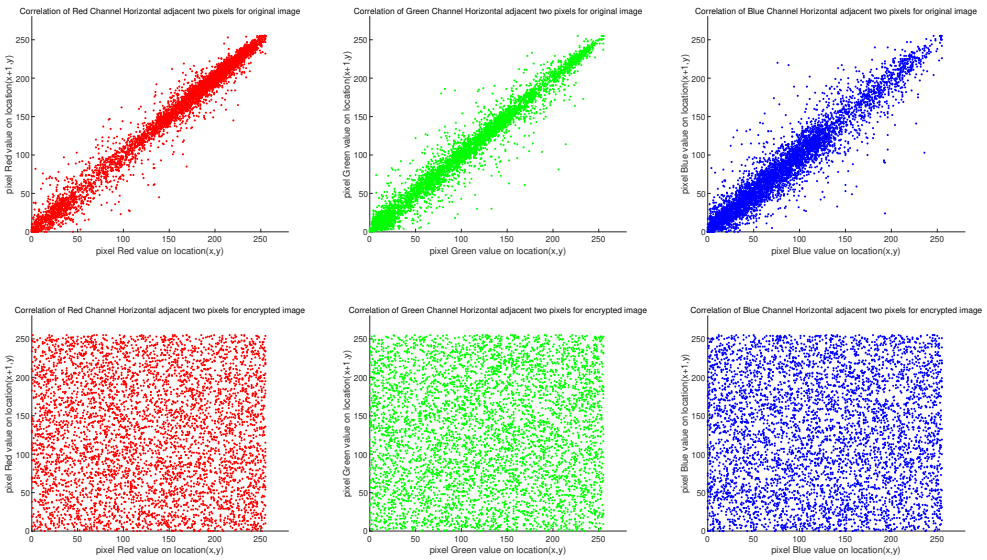

Figure 1.5: Correlation histogram analysis of Lena.

Table 1.3: NIST test results for the proposed encryption scheme with Lena image

Statistical test [Proposed]	$P\ value_T$	Proportion	Result
Frequency	0.116519	238/240	SUCCESS
Block frequency	0.242986	237/240	SUCCESS
Runs	0.551026	239/240	SUCCESS
Longest runs of ones	0.253551	238/240	SUCCESS
Rank	0.542566	239/240	SUCCESS
Spectral DFT	0.772760	237/240	SUCCESS
Non-overlapping templates	0.551026	238/240	SUCCESS
Overlapping templates	0.970538	239/240	SUCCESS
Maurers universal	0.500934	239/240	SUCCESS
Linear complexity	0.407091	238/240	SUCCESS
Serial0	0.881915	237/249	SUCCESS
Approximate entropy	0.654467	237/240	SUCCESS
Cumulative sums	0.213309	238/240	SUCCESS
Random excursions	0.542566	239/240	SUCCESS
Random excursions variant	0.748229	239/240	SUCCESS

of the design, we first selected a 512×512 Lena image as the plain image and generated 20 different ciphers using random keys. Next, we converted the individual color channels of the images into 8-bit binary arrays (unit8 pixels) and merged all arrays into a long sequence. This merged sequence was then stored in a txt document. In the NIST tests, the length of a single binary sample was $512 \times 512 \times 8 \times 3 = 6,291,456$ bits. There were 240 samples taken in total. The final test results are shown in the Table 3.

1.5.4 Different analysis

One method for deciphering encrypted image data is differential crypt-analysis, which is defined as the attackers making subtle modifications to the original plaintext digital image data. They use the proposed encryption algorithm to encrypt both the altered digital image and the original plaintext digital image separately. By comparing the two encrypted ciphertext images, they try to identify the relationship and patterns between the original plaintext digital image data and the encrypted ciphertext digital image data. With this knowledge, they attempt to decipher the ciphertext image. Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) are defined as

$$NPCR = \frac{\sum_{i,j} D(i,j)}{M \times N} \times 100 \tag{1.6}$$

Table 1.4: The NPCR and UACI results

	NPCR				UACI			
	R	G	B	Gray	R	G	B	Gray
Image 1				99.6090				33.4693
Image 2				99.5926				33.5022
Image 3	99.5968	99.6231	99.6140		33.4154	33.4587	33.4624	
Image 4	99.5895	99.6140	99.6109		33.4128	33.4587	33.4528	
Image 5	99.6166	99.6094	99.6151		33.4928	33.4534	33.5885	

$$D(i,j) = \begin{cases} 0, & \text{if } C_1(i,j) = C_2(i,j); \\ 1, & \text{if } C_1(i,j) \neq C_2(i,j); \end{cases} \tag{1.7}$$

$$UACI = \frac{\sum_{i,j} |C_1(i,j) - C_2(i,j)|}{255 \times M \times N} \times 100 \tag{1.8}$$

where M*N represents the total number of pixels in the cipher image. C1 and C2 represent the cipher images encrypted using different keys. Our test subjects consist of gray-scale images and RGB color images with sizes of 256×256 and 512×512, respectively, obtained from the USC-SIPI image database. The expected values of NPCR (Number of Pixel Change Rate) are 99.61% and UACI (Unified Average Changing Intensity) is 33.46% [9]. The results are displayed in Table 4.

1.5.5 Occlusion noise analysis

During the information transmission process, there is a certain probability of encountering interference from Occlusion noise. Based on this scenario, our encryption scheme exhibits a certain resistance to the system performance affected by Occlusion noise. As shown in Fig 1.6(b), we have simulated the possibility of encountering blockage attacks by adding some black blocks to replace the original pixels. Fig 1.6(e) demonstrates that even under such circumstances, the decrypted image can still be recognized.

1.5.6 Gaussian noise robustness test

To test the robustness of our system under Gaussian noise, in Fig 1.6(c), we randomly selected 10,000 pixels on the cipher and replaced their original values with random pixel values to simulate the impact of Gaussian noise on the transmission channel. As shown in Fig 1.6(f), even under the simulated noise interference, the decrypted cipher can still clearly recognize the original image information.

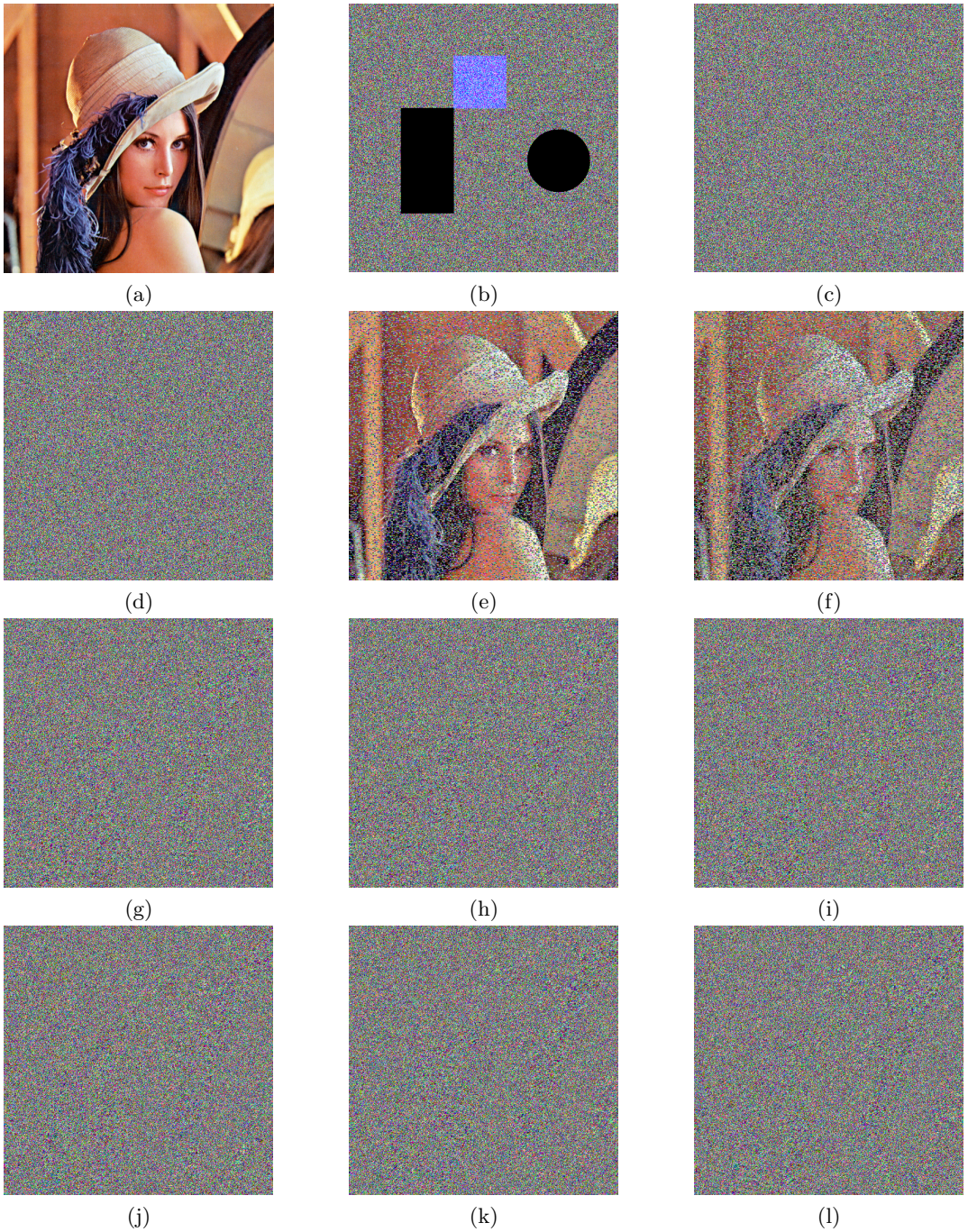

Figure 1.6: Noise and robustness test: (a) plain-text; (b) cipher under cut attack; (c) cipher under Gaussian noise; (d) cipher; (e) image deciphered from cut attack; (f) cipher from Gaussian noise. Key sensitivity analysis: (g) image deciphered with wrong x(1) = 2.156617425511379; (h) image deciphered with wrong y(1) = 0.959376638739491; (i) image deciphered with wrong z(1) = 4.795281393542876; (j) image deciphered with wrong k1 = 70; (k) image deciphered with wrong k2 = 100; (l) image deciphered with wrong k3 = 303.

1.5.7 Key sensitivity

Key sensitivity is refer to when a slight change in the initial key during the encryption or decryption processing would lead to a significant change impacted by the generated key after being subjected to a key sequence generator or iterative function. As a result, the processing of image encryption or decryption undergoes a substantial transformation. If the pixel values of the image are set as control parameters and used as the initial key during the encryption or decryption process, the algorithm not only exhibits key sensitivity but also demonstrates resistance against known plain-text attacks. Is our scheme, decrypted image is sensitivity with both key and input chaotic sequence. The decryption process of the image is influenced by both the encryption key and the input chaotic sequence. The original encryption image's keys and chaotic sequences are given as k1 = 69, k2 = 99, k3 = 302, and x(1) = 2.156617425511378, y(1) = 0.959376638739490, z(1) = 4.795281393542875. Next, we introduce minimal changes separately to the key and chaotic sequence. Fig 1.6(a) depicts the selected Lena image, Fig 1.6(d) shows the encrypted cipher, Fig 1.6(g) displays the decrypted result with an incorrect key k1 = 70, Fig 1.6(h) shows the decrypted result with an incorrect key k2 = 100, Fig 1.6(i) displays the decrypted result with an incorrect key k3 = 303, Fig 1.6(j) shows the decrypted result with an incorrect initial value x1 = 2.156617425511379, Fig 1.6(k) displays the decrypted result with an incorrect initial value y1 = 0.959376638739491, and Fig 1.6(m) shows the decrypted result with an incorrect initial value x1 = 4.795281393542876. From the above results, any slight change can lead to the unrecognizable result.It is evident that our system exhibits good key sensitivity.

1.6 Experimental Results and Implementation

Our system design is simple and convenient for implementation in Internet of Things (IoT) devices. To test the practicality of the system in a commercial wireless network, we conducted transmission and encryption-decryption experiments on a Raspberry Pi 3B via WiFi and UDP protocol connecting with each other. As shown in Fig 1.7, we successfully implemented the encryption algorithm in IoT devices using Python. Our next step is to integrate real-time monitoring into our encryption system.

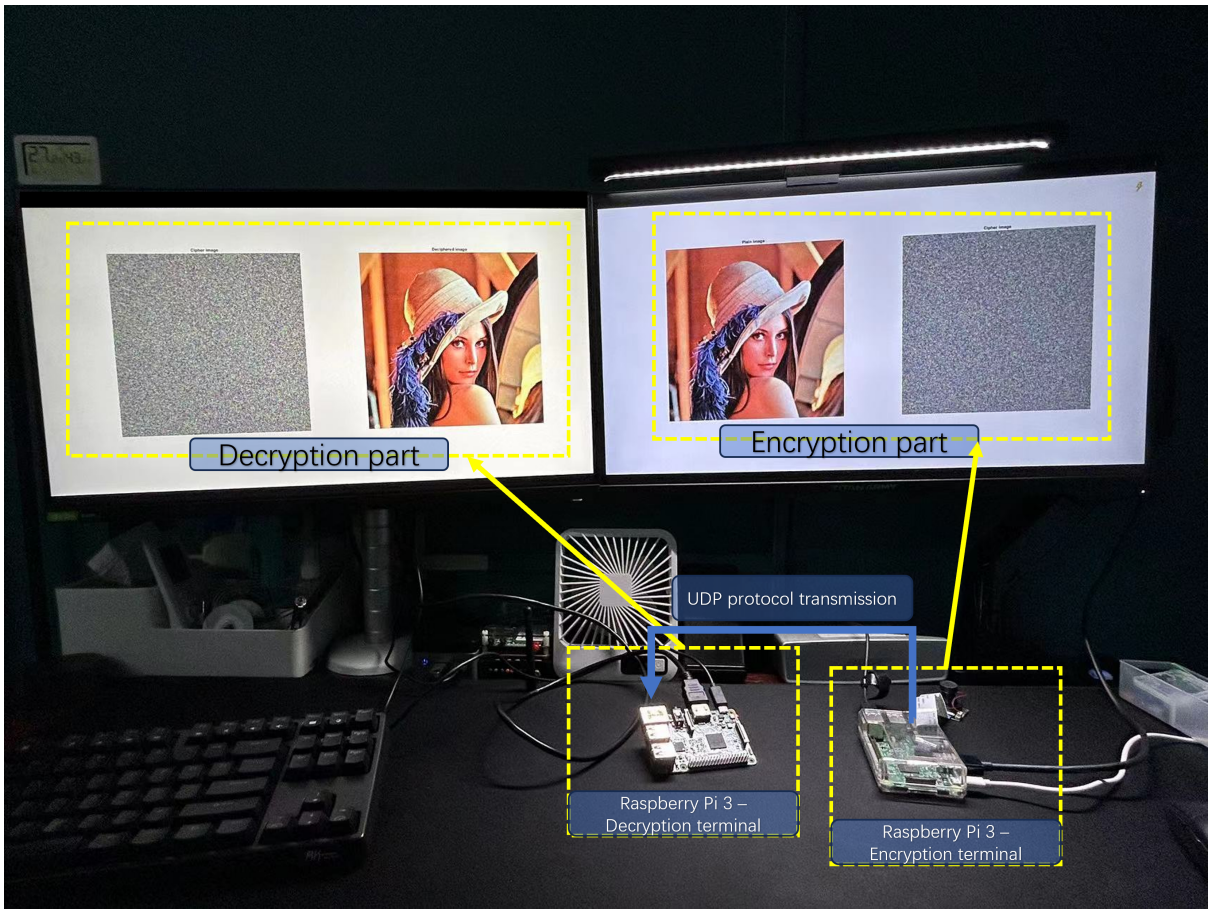

Figure 1.7: Implementation of Raspberry Pi3 with sending image via UDP protocol

1.7 Conclusion

In this chapter, we present an efficient algorithm for image encryption using the switching fractional order chaotic system. Firstly, we introduce three-point diffusion based on the existing two-point diffusion technique, which enhances the speed of pixel diffusion. Secondly, by employing the proposed algorithm, permutation and diffusion processes can be carried out synchronously, eliminating the need for multiple image scans and thus increasing the efficiency of the encryption system. Thirdly, on this basis, we enhance the robustness of the cipher. In conclusion, the proposed algorithm demonstrates high efficiency and improved robustness, making it of significant commercial value in the domain of commercial image encryption.

Moreover, this chapter substantiates the feasibility of chaos cryptography in the field of communication, providing instructive experience and technical accumulation for subsequent communication encryption in backscatter communication.To elaborate further, our switching fractional chaotic system image encryption system has validated its feasibility in communication. Its resistance to interference in information can be further integrated into wireless communication

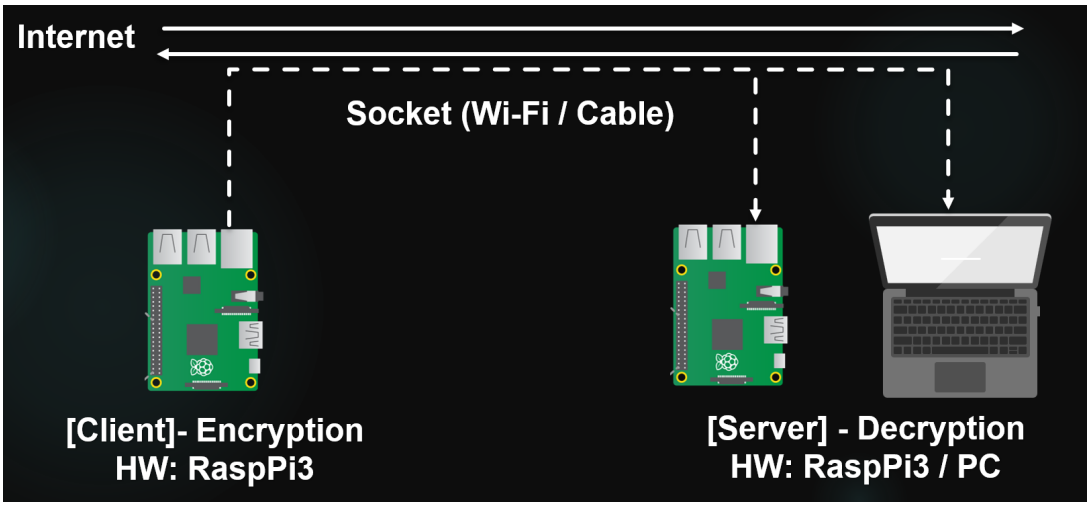

Figure 1.8: Client encrypts the image and sent out the cipher image under Socket protocol via WiFi or Cable. Servers receives the cipher image and decipher it back to plain image.

systems to enhance the stability of encoding channels. In light of these research findings, the algorithm presented in Chapter 1 can be explored for potential reuse in ambient backscatter systems, aiming to improve the stability of communication systems and enhance data security.

Chapter 2

Backscatter Communication

In recent years, we experienced a rapidly revolution of Internet of Things (IoT), otherwise, wireless sensor (WSN) network is one of them. IoT global market is increasing rapidly, which is rising from $157 billion in 2016 up to $457 billion in 2020 advance.[47] It observed that increasing demand requires bring inevitable technique problems, which caused by sensors, base station, and network. One for the first, we need to figure out the possibility of wireless communication information exchange is achievable and effective in principle. Some new schemes are indicated to challenge current problems. Backscatter, a new wireless communication technique, defined a low power, battery-free wireless communication system. Comparing to traditional wireless communication, backscatter communication has unique that harvesting energy from RF resource from environment firstly, then converter it to DC power. Backscatter can solve the power supply problem of large account of unit wireless microsystems devices. In the other hand, the solution can extend battery life to wireless devices, even replaces the role of batteries inset the devices.

Backscatter communication represents a cutting-edge wireless communication method that enables ultra-low-power data transmission by reflecting existing ambient radio frequency (RF) signals, rather than generating its own. This technique leverages ambient RF sources—such as WiFi, TV towers, and cellular signals—to power the communication process, thereby eliminating the need for dedicated power supplies and batteries. In this system, a backscatter tag modulates the reflection of incident RF signals to encode and transmit data, enabling robust, energy-efficient communication in Internet of Things (IoT) and wireless sensor networks (WSNs).

Due to its reliance on ambient energy, backscatter communication significantly extends the operational lifespan of IoT devices, facilitating sustainable, battery-free deployments. This communication mode is particularly advantageous in scenarios where device access is limited, as it reduces the dependency on battery maintenance and replacement. Consequently, backscatter communication is emerging as a critical technology for enabling scalable IoT solutions with minimal energy requirements.

2.1 Comprehensive Methodology

2.1.1 Backscatter Communication Methodology

The backscatter communication system is an integral component of the overall functionality of the proposed framework. The testing methodology for this system includes various experimental setups and data collection protocols.

- **Hardware Configurations**: The primary hardware used in the experiments includes a HackRF One, serving as the ambient RF source; an RF reflection module (tag) for reflecting signals; and another HackRF One as the receiver. This setup is capable of capturing signals within the ISM band, making it suitable for many IoT devices. Additionally, a high-efficiency antenna was employed to optimize signal reception and energy harvesting.
- **Ambient RF Source Specifications**: The ambient RF source in the experiments was generated by a HackRF One device, transmitting a sine wave signal at a frequency of 868 MHz. Detailed descriptions of the RF source's signal strength, frequency, and modulation type were provided in the simulation experiments to ensure reproducibility of the test conditions.
- **Data Collection Protocols**: During the experiments, real-time data collection was conducted to analyze the signals under different signal-to-noise ratio (SNR) conditions and error rates. The collected data was then systematically analyzed to assess the effectiveness of backscatter communication across varying conditions.

The performance evaluation of the backscatter system relies on comprehensive data analysis, with the following methods used to quantify and validate performance metrics:

- **Statistical Analysis Tools and Methods**: Various statistical tools were utilized, in-

cluding correlation analysis, regression analysis, and hypothesis testing, to thoroughly assess the system's performance metrics, such as data transmission reliability and energy harvesting efficiency.

- **Parameter Sensitivity Analysis**: Sensitivity analysis was conducted to evaluate the impact of changes in key parameters (e.g., ambient RF power density and device spacing) on system performance. This analysis aimed to identify critical thresholds, optimizing the design to meet different application requirements. For instance, at lower RF power densities, the data transmission rate decreased significantly, highlighting the importance of sufficient ambient energy sources.
- **Performance Metrics Analysis**: Core performance metrics, including energy efficiency, data transmission reliability, and latency, were analyzed. The results showed a positive correlation between higher ambient RF power levels and increased data transmission rates and energy harvesting efficiency. To ensure the reliability of the analysis, statistical significance testing was conducted to verify the statistical relevance of performance differences across different experimental conditions.

2.2 Backscatter Communication Background

Radio Frequency (RF) energy harvesting technology converts electromagnetic waves into electrical energy via an RF receiving antenna, through a matching network and rectifier circuit in the RF front end. The first idea of Energy Harvesting (EH) came out from Nikola Tesla [41], who created the first wireless power transmission machine ever. In1948, Stockman defines technique of backscatter [42], which the first-time data transmission through backscatter was determined. Traditional backscatter, also known as conventional backscatter, the design as reflected signals receiver and RF resource settled in same device. The traditional backscatter device sets RF resource transmitter and receiver in the same device. We can notice the disadvantages of traditional backscatter, which consists of self-interference, limited the usage and coverage area. The new emerging technology, ambient backscatter device separates the RF resource and receiver, and fix the shortage of traditional backscatter.

Ambient backscatter communication (AmBC) system, a novel technology, via harvesting RF resource around devices in ambient reflect processing signals to receiver to realize communica-

tion between transmitters and receivers. Backscatter communication systems involve antenna impedance matching and standing wave reflection processes. The backscatter communication system needs to convert the signals collected by the sensors to regulate the RF switch, achieving incident signal reflection or absorption, and utilizes the amplitude difference of the received signals at the air interface. The advantages of AmBCS is not only extend the coverage but also provide more feasibility to reduce self-interference. It does not require especial transmitter. Moreover, RF resource is large range selected from environment broadcasted from TV towers, FM towers, Wi-Fi basement, etc.. RF resource is stable and little influenced by environment, other than solar, wind and hydraulic energy. AmBC system does not need occupy extra frequency spectrum, instead of reflecting signals and saves precious frequency band resources. Summarily, AmBC system talents showing itself, which is battery-free, low-power consumption, and simple structure wireless communication technique.

2.3 Motivation and related work

When the first ambient backscatter is indicated in [47], 2013, technique of ambient backscatter has been quickly becoming a critical novel solution for IoT systems without battery in future. It is observed that there are so many multiple solutions for battery-free wireless communication systems, which is also known as wireless transmission powered system. As an emerging technology, backscatter has its own advantages, e.g., low power computation, no occupation of frequency band, and simple logic of circuit. It a new area in wireless communication and IoT. There is so much work related to backscatter need to be worked out, e.g., communication system design, circuit design, communication modulation, and security of PHY layer, etc. What is the best performance in design? How is the most achievable circuit for backscatter transmitter tag? What kinds of problem can be solved with backscatter? All these questions will be the future work in my research. The current work is system model design of AmBC system, which contains signals detection and Bit Error Rate (BER) performance analysis with different modulations. After that, we plan to implement an ambient backscatter communication system with SDR analysis. For the more, extend the basic single link to multiply accessible transmitters in AmBC system. Finally, my future work is relative to AmBC system in dynamic environment with signal analysis and system modulated.

2.4 Backscatter Communication Overview

2.4.1 Energy Harvesting Scheme

Energy harvesting is a sustainable technology that captures and converts ambient energy sources—such as radio frequency (RF) waves, solar power, or vibrational energy—into usable electrical power, allowing electronic devices to operate independently of traditional battery supplies. In RF energy harvesting, electromagnetic waves are received by an antenna and converted to direct current (DC) power through a matching network and rectifier circuit, providing a reliable source of energy for low-power applications.

In backscatter communication systems, RF energy harvesting is instrumental for enabling continuous, battery-free operation of IoT devices. By utilizing ambient RF sources, such as WiFi signals or cellular transmissions, the devices can sustainably power their communication processes. This integration reduces dependency on battery replacements, making it particularly valuable for long-term IoT deployments in remote or difficult-to-access locations. As a result, energy harvesting serves not only to meet the power demands of backscatter systems but also to support environmentally friendly and scalable IoT infrastructure.

In [46], RF Power harvesting technique began from middle of last century. The experiment is set up a microwave wireless powered system for a model helicopter. The wireless power receiver consisted of an array dipole terminated in a bridge-rectifier array of point-contact silicon diodes. The power of the helicopter was received from microwave generator via focusing antenna. The experiment provided favourable evidence for wireless powered systems realization. The three well-known EH scheme for wireless powered communications network (WPCN) consist of Wireless Power Transfer (WPT), Wireless-powered Communication Network (WPSN), and Simultaneous Wireless Information and Power Transfer (SWIPT).

WPT: Wireless Power Transfer scheme allows the power energy to transmit without information. This scheme supports basic wireless sensor network (WSN) via wireless power transmission, for example, powered sensors, low-power electronics in home (smart house), etc.

WPCN: Wireless-powered Communication Network scheme enable transfer energy to wireless device and receive information back from device. This scheme can transfer information between end terminal and wireless devices.It would be widely using scheme in IoT systems during low-power systems in future.

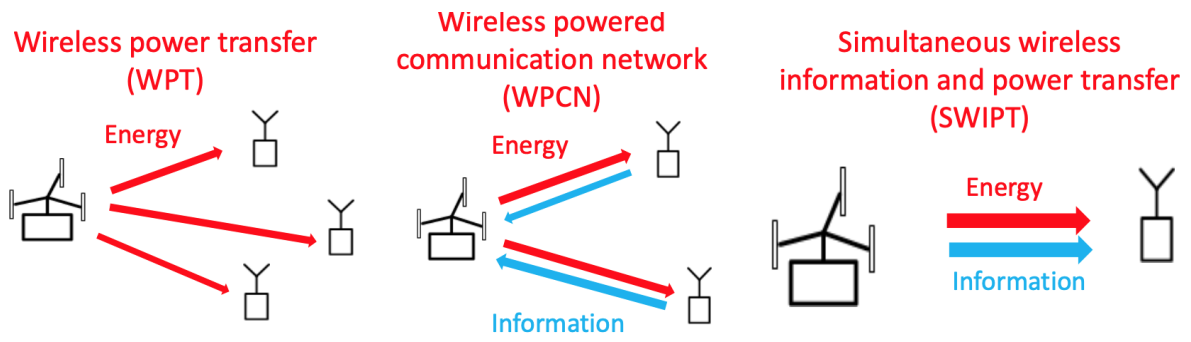

Figure 2.1: (1) WPT system model; (2) WPCN system model; (3) SWIPT system model

SWIPT: Simultaneous Wireless Information and Power Transfer scheme allow transmitter spreads energy and information at same time to wireless devices. Thereby, the performance of this scheme makes high efficiency, considering implementation in future.

2.5 Clarification of Energy Harvesting Mechanisms

The energy harvesting mechanisms utilized in this study are essential for enabling sustainable operation in resource-constrained IoT devices. Specifically, RF energy harvesting is leveraged to convert ambient RF signals into usable electrical energy, supporting both device power and data transmission.

2.5.1 RF Energy Harvesting Mechanisms and Efficiency

RF energy harvesting involves capturing electromagnetic energy from ambient RF sources, such as WiFi routers, cellular base stations, and broadcast towers, and converting it into DC power. The energy conversion process typically includes three main components: the antenna, matching network, and rectifier circuit.

- **Antenna**: The antenna captures RF energy from the environment, with its design and frequency range significantly influencing the overall efficiency of the energy harvesting process. The system's efficiency is highly dependent on the ambient RF signal strength and the antenna's ability to capture a broad spectrum of frequencies, ranging from low-frequency signals like AM radio to higher frequencies such as WiFi at 2.4 GHz.
- **Matching Network**: The matching network maximizes power transfer from the antenna to the rectifier by ensuring impedance matching, which reduces energy losses and improves overall efficiency. This is particularly critical in low-power RF environments, where even

small losses can impact the device's ability to operate effectively.

- **Rectifier Circuit**: The rectifier converts the AC RF signal captured by the antenna into DC power, which can then be stored or used to power the device. Typical rectifiers for RF energy harvesting employ Schottky diodes or CMOS technology, both of which provide high efficiency in low-power environments. The rectification efficiency varies depending on the strength of the input RF signal and can reach rates of 40-60% under ideal conditions with strong RF sources.

In varying environmental conditions, such as urban vs. rural areas, the RF energy harvesting efficiency fluctuates based on the availability and strength of ambient RF sources. For instance, in urban environments with high densities of RF sources, the system can achieve higher energy conversion rates, while in rural areas, efficiency may be lower due to limited RF energy availability.

2.5.2 Integration of Energy Harvesting with Data Transmission

The integration of RF energy harvesting with data transmission capabilities is critical for achieving a fully self-sustained IoT device. In this thesis, an innovative approach is proposed to optimize the simultaneous energy harvesting and data transmission processes, ensuring that both operations can occur without compromising performance.

- **Energy Storage and Power Management**: Harvested energy is first stored in a capacitor or a rechargeable battery, which ensures a continuous power supply for data transmission even when ambient RF sources fluctuate. The power management system regulates the distribution of harvested energy between powering the device and storing excess energy for later use.

- **Duty Cycling and Power Allocation**: To maximize energy efficiency, a duty-cycling mechanism is implemented, allowing the device to alternate between active transmission and low-power idle states. During idle periods, more energy is allocated to storage, which provides additional power for the next transmission cycle. This approach minimizes energy consumption during data transmission without sacrificing communication performance.

- **Dynamic Power Adjustment for Data Transmission**: The system dynamically adjusts the power allocated to data transmission based on the available harvested energy.

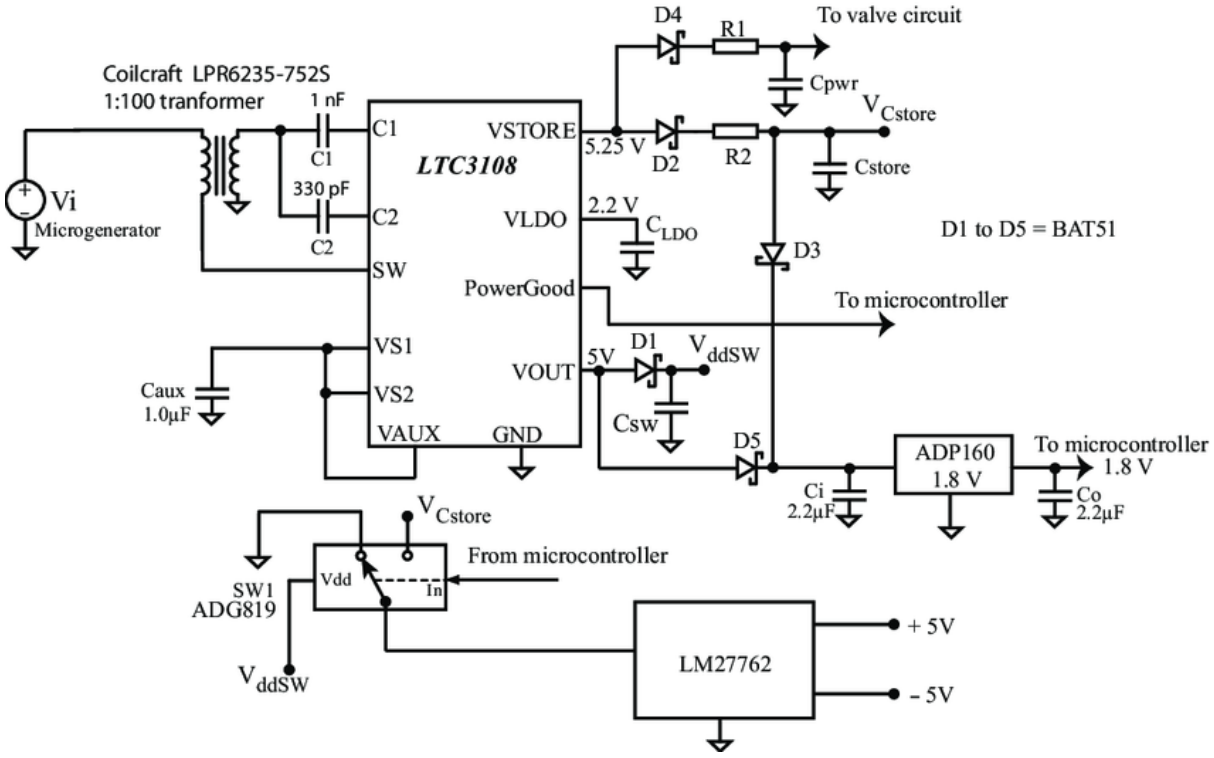

Figure 2.2: Energy Harvesting Circuit Diagram

When ambient RF energy levels are high, more power is allocated to increase transmission strength and data rate, whereas under lower energy conditions, transmission power is reduced to conserve energy. This dynamic adjustment ensures a balance between maintaining reliable data communication and conserving energy resources.

These innovative solutions allow for a seamless integration of RF energy harvesting with data transmission, making it possible for the device to function independently without the need for external power sources. By optimizing both energy conversion and power management, the proposed framework enhances the reliability and longevity of IoT devices operating in diverse environments.

Energy Harvesting Circuit Diagram: This diagram shows the RF-to-DC conversion process, including components such as the antenna, matching network, and rectifier. Each element is labeled to demonstrate how RF energy is collected from ambient sources and converted into usable DC power for device operation. This clarity is crucial for understanding the efficiency of energy harvesting under varying RF conditions.

2.5.3 Classification of Backscatter

The classification of backscatter communication systems consists of three different types, Monostatic Backscatter Communication Systems (MBCS), Bistatic Backscatter Communication Systems (BBCS), and Ambient Backscatter Communication Systems (AmBCS). The difference of these backscatter communication systems based on inner architectures.

Monostatic Backscatter Communication Systems (MBCS)

MBCS contains two major parts: reader, and backscatter transmitter. Reader itself both have the function of RF resource transmitter and receiver. The design of Backscatter transmitter is to be activate by the received RF resource signals from reader. Backscatter components also modulate and reflect the RF signal back to receiver, which places in same device in reader. Therefore, MBCS scheme has some disadvantages. Due to RF resource and receiver place in same device, the reflected signal influences deeply of path loss and interference. It also limited the distance of two components. With the distance increasing, the MBCS affects severely. MBCS architectures limits the using range, which is suitable in short-range application.

Bistatic Backscatter Communication Systems (BBCS)

Comparing MBCS, BBCS separates RF resource (carrier emitter) and backscatter receiver (reader). BBCS scheme reduces the double round trip path loss. Though reasonable layout of the carrier emitter, the performance of the BBCS increases remarkably. The implementation of this reasonable layout contains setting multiply carrier emitters around the backscatter transmitters; meanwhile backscatter receiver in location range is able to receive the backscatter signals from each backscatter transmitter. BBCS architectures is enable the multiple backscatter communication achieved. Moreover, BBCS reduces the cost from the simple structure of backscatter components.

Ambient Backscatter Communication Systems (AmBCS)

AmBCS share some similar points with BBCS, they both separates the RF resource and backscatter receiver. The difference is that, AmBCS use the ambient RF resource (TV tower, cellular base station, Wi-Fi signals source) around the backscatter transmitter as power instead of special carrier emitter. Intuitively, AmBCS reduces lots of cost from setting the individual carrier emitter. Meanwhile, via reusing the ambient RF resource reduces the waste extra fre-

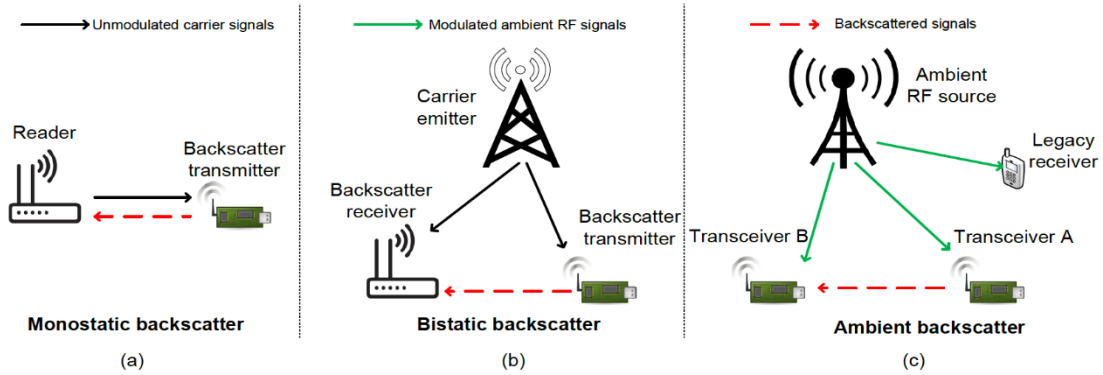

Figure 2.3: (a) Monostatic backscatter (b) Bistatic backscatter (3) Ambient backscatter

quency spectrum. It improves the frequency spectrum efficiency. However, power source and communication of AmBCS rely on ambient RF resource; AmBCS also have some disadvantages from BBCS. BBCS has stable and continuous with solid-state RF resource from carrier emitter. AmBCS RF resources are dynamic that its suffered interference and performance are unpredictable. These unstable factors will lead to difficulties of complex design and deployment of AmBCS.

2.5.4 Principle of Backscatter Communication

Therefore, different backscatter schemes have different configurations, they are still in the basic principle, via antenna converter, the signals from RF resource to power to send backscatter signals to receiver and enable realise wireless communication systems. The bit stream of backscatter modulated on antenna impedance. The following equation represents the reflection of antenna coefficient Γ_i,

$$\Gamma_i = \frac{Z_i - Z_a^*}{Z_i + Z_a} \tag{2.0}$$

In formula, Γ_i is Antenna coefficient. Z_a and Z_a^* are impedance of antenna and its complex conjugate operator. The scheme of backscatter is load modulation, via switching the impedance of antenna to change the antenna state. In the equation, i represents different switch state of antenna; i can be two, four and eight states. Generally, in backscatter communication modulation, i usually is two, which simple the basic structure. However, Z_1 and Z_2 is used as two different states, absorbing and reflecting. In Z_1, absorbing state, RF signals absorbed represent '0' in bit stream. In contrast, Z_2 reflecting state, represents '1' in bit stream. In Fig. 2.3., shows the backscatter tag logic circuit.

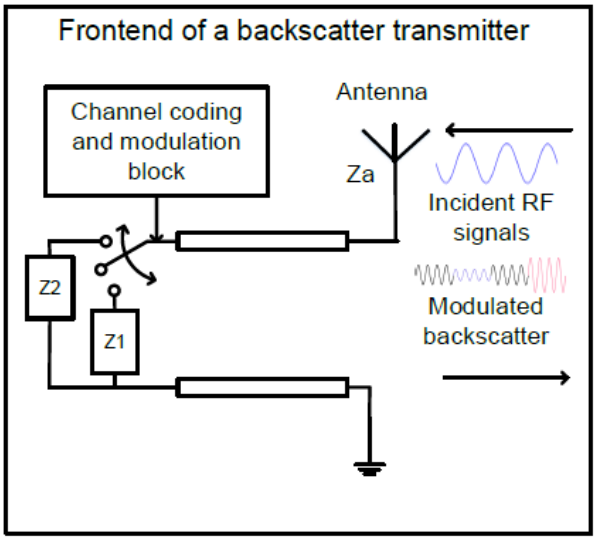

Figure 2.4: Frontend of a backscatter transmitter[29]

Backscatter receiver ordinary decode modulation method is analog-to-digital converter (ADC). The backscatter receiver receives both RF resource signals and backscattered signals. The equation $y[n]$ below represents backscatter signals processing:

$$\mathbf{y}\left[\mathbf{n}\right] = \mathbf{x}\left[\mathbf{n}\right] + \propto \mathbf{B}\left[\mathbf{n}\right]\mathbf{x}\left[\mathbf{n}\right] + \mathbf{w}\left[\mathbf{n}\right] \tag{2.1}$$

where, $y[n]$ represents the received signals of backscatter receiver, α is the complex attenuation of the backscattered signals, $x[n]$ are samples from RF resource which received by backscatter receiver. $B[n]$ is backscatter tag signals which only represent the state of backscatter tag ('0' or '1'), $w[n]$ is addition white Gaussian noise (AWGN). Though transferring the signal information to power, backscatter tag reflect signals is able to detect. The average received signals power expressed as follows:

$$\mathbf{P_{recevied}} = \frac{1}{\mathbf{N}}\sum_{\mathbf{i=1}}^{\mathbf{N}}\left|\mathbf{y}\left[\mathbf{n}\right]\right|^2 = \frac{1}{\mathbf{N}}\sum_{\mathbf{i=1}}^{\mathbf{N}}\left|\mathbf{x}\left[\mathbf{n}\right] + \propto \mathbf{B}\left[\mathbf{n}\right]\mathbf{x}\left[\mathbf{n}\right] + \mathbf{w}\left[\mathbf{n}\right]\right|^2 \tag{2.3}$$

In this calculation, N is the received samples of backscatter receiver. $B[n]$ is able to assume as consist, for it only represents '0' or '1', two states. AWNG $w[n]$ is not correlated with the signal information. The formula is simplified to following:

$$\mathbf{P_{recevied}} = \frac{1}{\mathbf{N}}\sum_{\mathbf{i=1}}^{\mathbf{N}}\left|\mathbf{y}\left[\mathbf{n}\right]\right|^2 = \frac{\left|1 + \propto \mathbf{B}\right|^2}{\mathbf{N}}\sum_{\mathbf{i=1}}^{\mathbf{N}}\left|\mathbf{x}\left[\mathbf{n}\right]\right|^2 + \frac{1}{\mathbf{N}}\sum_{\mathbf{i=1}}^{\mathbf{N}}\left|\mathbf{w}\left[\mathbf{n}\right]\right|^2 \tag{2.4}$$

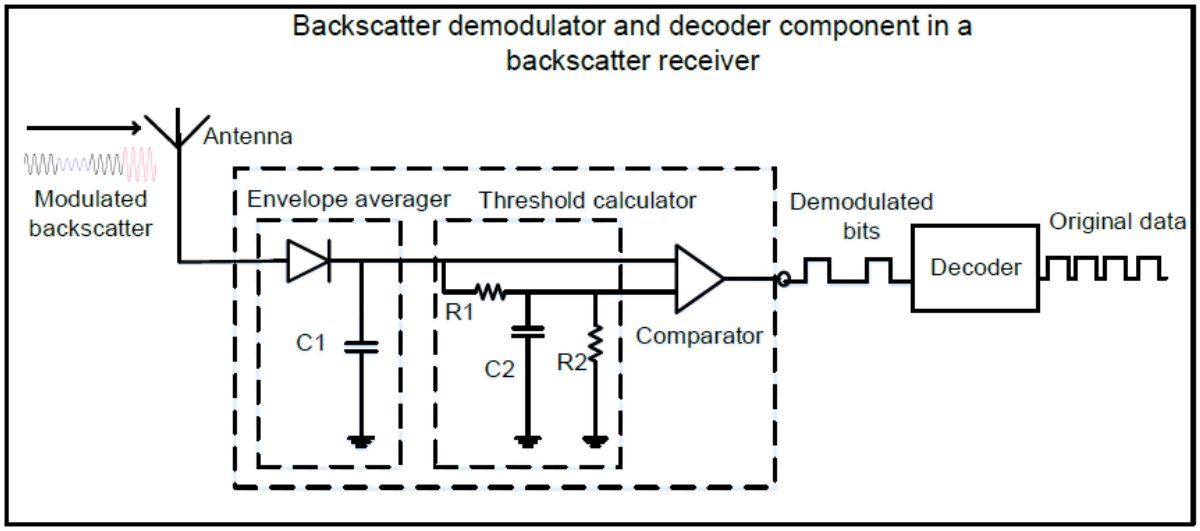

Figure 2.5: Backscatter transmitter[9]

Assume P as the average power of $x[n]$. Ignore the uncorrelated noise $w[n]$, the average received power got two states related to backscatter tag states. While **B** is '1', backscatter tag is in reflecting state, and average received power is $|1+\alpha|^2P$.On the contrary, while B is '0', backscatter tag is in non-reflecting state and average received power is P. Under the condition, threshold compactor of backscatter tag information can be determined as the average value between$|1+\alpha|^2P$ and P With the power-levels decoding method and calculation, the received backscatter can converter to digital signal. Fig. 2.4. displays the backscatter receiver logic

2.5.5 Backscatter Operation Frequency and Antenna

Ambient backscatter Communication Systems do not require particular spectrum instead of reflecting signals in conditional wireless communication systems. It leads to the antenna operation frequency depending on local regulations (different regions), target applications, and transmissions protocols, etc. The range of operation frequency includes from low frequency (LF), i.e., 125 kHz, up to super high frequency (SHF), i.e., 5.8 GHz.

The tendency of backscatter operation frequency is towards to SHF. The disadvantage is high operation frequency with short wavelength, meanwhile leads to high power consumption. However, it does exist extra build-in RF power consumption in backscatter systems, instead of harvesting energy from air to supply for systems. Channel capacity of SHF is much abundant. For example, Wi-Fi consists of two different operation frequencies bands, 2.4GHz – 2.5GHz and 5.725GHz – 5.875GHz. The maximum number of non-overlapping channels referring to

2.4GHz band is 3, while the number of 5.8GHz band increase up to 23. Extensively commercial applications of SHF, such as Bluetooth and Wi-Fi devices, are favourable base to backscatter communication systems. Hence, AmBCS does not need extra cost for operation frequency foundational cost. Higher frequency with shorter wavelength also reduces the size of antenna. Therefore, smaller size antenna enables smaller size of backscatter device and leads to easier realization of mobile backscatter system.

In a backscatter wireless communication system, antenna is indispensable and importance that affects the performance of entire system. Via Friis equation, the maximum practical distance between RF resource and backscatter transmitters (tags) can calculated as:

$$r = \frac{\lambda}{4\,\pi}\sqrt{\frac{\mathbf{P_t}\ \mathbf{G_t}\,(\theta,\varphi)\ \ \mathbf{G_r}\,(\theta,\varphi)\ \ \rho\ \tau}{\mathbf{P_{th}}}} \tag{2.5}$$

where r is Maximum practical distance, λ states the wavelength of the radio frequency. $P_{(t)}$ is RF resource transmitted power. $G_t(\theta,\varphi)$ and $G_r(\theta,\varphi)$ represent the gain of transmitter, and gain of receiver respectively, where (θ,φ) determines the antenna angles. $P_{(th)}$ is minimum threshold power. ρ is Polarization efficiency. τ is power transmission coefficient.

Power transmission coefficient, τ is related to both load impedance and antenna impedance, which is expressed as follows:

$$\tau = \frac{4\mathbf{R_c}\mathbf{R_a}}{|\mathbf{Z_c}+\mathbf{Z_a}|^2} \tag{2.6}$$

$$\mathbf{Z_c} = \mathbf{R_c} + j\mathbf{X_c} \tag{2.7}$$

$$\mathbf{Z_a} = \mathbf{R_a} + j\mathbf{X_a} \tag{2.8}$$

where, Z_c is chip impedance (load impedance), Z_a is antenna impedance. R_c and R_a are chip and antenna resistances respectively. X_c and X_a are chip and antenna rectenna respectively. Denote P_c as chip power, and P_a as antenna power, τ also is represented as follows,

$$\mathbf{P_c} = \mathbf{P_a}\ \tau. \tag{2.9}$$

The more τ closer to 1, the better impedance matching is. In idea case, whileτ is 1, the impedance matching is perfect.

Antenna gain describes the amount of power transmitted in the direction of peak radiation to

an isotropic source. Antenna gain is in relationship with range of transmission. High antenna gain leads to long transmission range. The shortage of high antenna gain component is its high price. Due to this principle of antenna gain, the design prefers reducing the antenna gain while the transmission distance is short. Antenna gain is depending on its material properties, object geometry, frequency band and antenna types. It is impossible to calculate an exact value of antenna gain instead of simulation and measurement.

Polarization is the characteristics of electromagnetic wave fluctuation that can oscillate in different factors, which describes field vector change of direction and magnitude over time. Polarization of antenna is defined as "Linearly Polarized Antenna" and "Right Hand Circularly Polarized Antenna". According to the trace shape of measurement, polarization is classified as linear, circular and elliptical types. When the received waves is matching with the antenna polarization, the antenna-received power is maximized. In backscatter systems, polarization is significantly affecting the received power and transmission range.

2.5.6 Channel Coding and Decoding

The principle of coding and decoding is to adopt digital or analog date via processing digital signals. Channel coding is also known as baseband coding, which via digital signals represent context of message of transmission channel. Channel coding is aim to use different digital situation to represent bit '1' and '0'. During the process signal transmission, interference, collision, and unexpected modification of signal lead to insecurity and non-reliable issues. The properties of channel coding affect the performance of the system, such as channel capacity, throughput, transmission range, etc. Thus, channel coding is critical in signal transmission. In backscatter communications systems, serval techniques, such as non-returns-to-zeros (NRZ), Manchester, Miller and FM0. There doing method is

NRZ coding: in circuit logic, high-level signal represents bit '1', and low-level signal represents bit '0'. NRZ coding is the most basic coding and easy to implement.

Manchester coding: known as split phase coding. In a half bit period, positive transform represents bit '0'; negative transform represents bit '1'.

Miller coding: level changed between two adjacent bits (high to low, or low to high), presents bit '1'; level does not change, present bit '0'.

FM0 coding: level changed in a half bit period, present bit '0', level does not change, present '1'.

2.5.7 Modulation and Demodulation

In general, 3 types of wireless communication, consist of Amplitude Modulation (AM), Frequency Modulation (FM), and Phase Modulation (PM). Modulation of signal is aiming to utilize one or combine some of the types above to realize the signal carrying message. Amplitude Shift Keying (ASK) which states data in binary with varying of amplitude levels. ASK is simple design to achieved backscatter data transmission, meanwhile, it is sensitive to be interference by noise. In [42], the author provides system model with ASK modulation to backscatter signal from commodity Wi-Fi source. Frequency Shift Keying (FSK), a frequency modulation scheme which defined as transmitting digital information through the variable frequency of carry signal. Transmitted information is based on carry frequency switch to state digital signals. Phase Shift Keying (PSK) which changes the phase carry signal to state information signals. PSK is divided by number of phases, which consist of Binary-PSK (BPSK), Quadrature-PSK (QPSK), 8-PSK and 16-PSK, etc. Quadrature Amplitude Modulation (QAM), a modulation scheme combines both amplitude and phases. QAM is modulated by two signals, one of the signals is shifted by 90 degree with other one.

2.5.8 Backscatter Communication Channels

Channel of backscatter communication systems consists of three independence parts: RF resource, backscatter tag(transmitter) and receiver. In [41], the research introduces a dyadic RF tags channel model, which satisfies with backscatter communication systems model.

In Fig. 2.5. (1) Consider the basic channel model, the backscattered signals at received of basic backscatter channel:

$$\widetilde{y}(t) = \frac{1}{2} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \widetilde{h^b}(\tau_b; t)\, \widetilde{s}(t) \widetilde{h^f}(\tau_f; t) \times \widetilde{x}(t - \tau_b - \tau_f)\, d\tau_b d\tau_f + \widetilde{n}(t) \tag{2.10}$$

where, $\widetilde{h^b}(\tau_b; t))$ is the channel parameter between backscatter transmitter antenna and backscatter antenna. $\widetilde{h^f}(\tau_f; t)))$ is the channel parameter between RF resource and backscatter transmitter antenna. $\widetilde{s}(t)$ on signal from tag. $\widetilde{x}(t)$ is the carrier signal from RF resource. $\widetilde{n}(t)$ is environment noise.

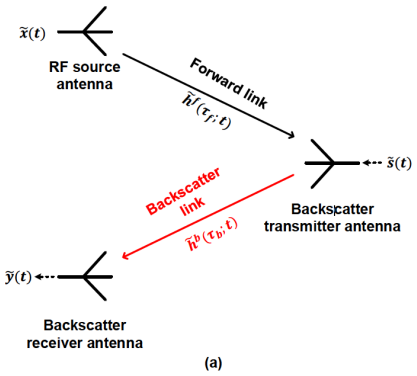

Figure 2.6: Basic backscatter channel

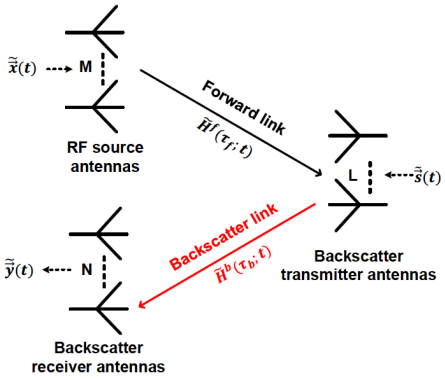

Figure 2.7: Dynamic backscatter channel

Nevertheless, multiply transmission links need utilizing multiple antennas systems in RF resource, tag and receiver. In Fig. 2.6, shows a dyadic backscatter channel model with M, L, N are numbers of antennas at RF resource, transmitters, and receiver respectively. The formula below is almost based on basic backscatter channel:

$$\overset{\sim}{\vec{y}}(t) = \frac{1}{2}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \tilde{H}^{b}(\tau_b;t)\,\tilde{S}(t)\tilde{H}^{f}(\tau_f;t) \times \overset{\sim}{\vec{x}}(t-\tau_b-\tau_f)\,d\tau_b d\tau_f + \overset{\sim}{\vec{n}}(t) \tag{2.11}$$

where, $\tilde{H}^{b}(\tau_b;t))$ is the complex channel between RF resource antennas and backscatter transmitter antennas. $\tilde{H}^{f}(\tau_f;t)$ is the complex channel between backscatter transmitter antennas and backscatter receiver antennas. $\tilde{S}(t)$ is the multiply information signals from backscatter tags. $\overset{\sim}{\vec{n}}(t)$ is the complex noise of environment.

2.6 Backscatter Communication Systems Analysis

Recent years, research of ambient backscatter technique raised to peak time. The first AmBCS is indicated in [47], which designed a wireless communication system between two tags, meanwhile only powered by TV tower RF resource. Based on the logic, a common architecture of ambient backscatter communication system contributed by RF resources, backscatter tags, and backscatter receivers. The basic ambient backscatter system model includes RF resource, backscatter tag, and reader, show as:

In Fig. 2.7., RF resource in system, aims to provide wireless power and baseband signals. Backscatter Tag is set to collect the sensor data, harvest energy from RF resource, and reflect signals to reader. Reader is the end terminal to decode signals from backscatter tag and pro-

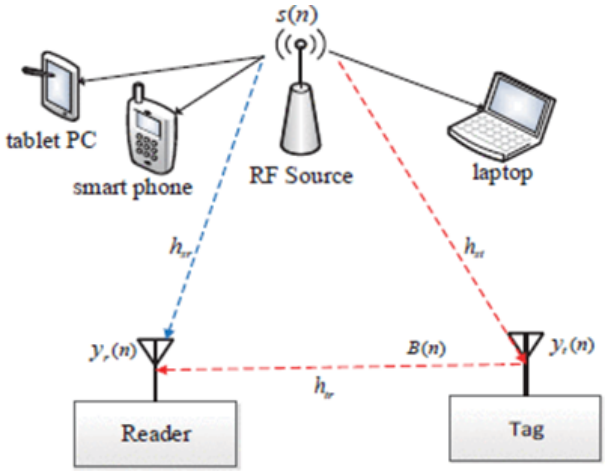

Figure 2.8: Ambient Backscatter Communication System model [28]

cessing the information from backscatter tag sensors. The wireless channels are in the system, due to AmBCS contains all channels between each element. Details of AmBCS is discussed in next chapter.

2.6.1 RF resources

Relay on Ambient RF resources, it can divide into two different types, Static and Dynamic ambient RF resources. Static ambient RF resource: a static ambient RF resource should be constantly and stable spreading the RF signal, which is hardly influenced by location and environment, e.g., FM base station, Cellular base station and TV towers. The transmitter power of these stations is high that enable to offer energy to a long-distance device. Dynamic ambient RF resource: dynamic ambient RF resource is low power transmitter and multifarious commercial using access point, e.g., Wi-Fi router. Almost, dynamic ambient backscatter RF resource is in short range wireless communication, which collects RF resource power in low level.

Whatever the RF resource in ambient backscatter system are chosen, it should be ensured to the exact option to meet the requires needs, e.g., in outdoor type design matching with TV tower or FM base station as RF resource, and indoor type matching with Wi-Fi AP as RF resource.

2.6.2 Ambient Backscatter Communication System Models

Although, Ambient Backscatter Communication System has some limitations, it still contains the superior characters, e.g., no needing battery and extra frequency band. AmBCS provides an

Table 2.1: Ambient Backscatter RF Resources

Type	RF resource	Transmit power	Frequency	Transmission rate	RF resource to transmitter distance
Static RF resource	TV Tower	Up to 1 MW	470-890 MHz	1 kbps at 539MHz and 1 MW of transmit power	Several kilometers
	FM base station	Up to 100 kW	88-108 MHz	3.2 kbps at 915 MHz and a received power at backscatter transmitters of -20 dBm	Several kilometers
	Cellular base station	Up to 10W	900 MHz (GSM 900)	N.A.	Several hundred meters
Dynamic RF resource	Wi-Fi AP	Up to 0.1W	2.4 GHZ	1 kbps with 40mW of transmitter power	Several meters

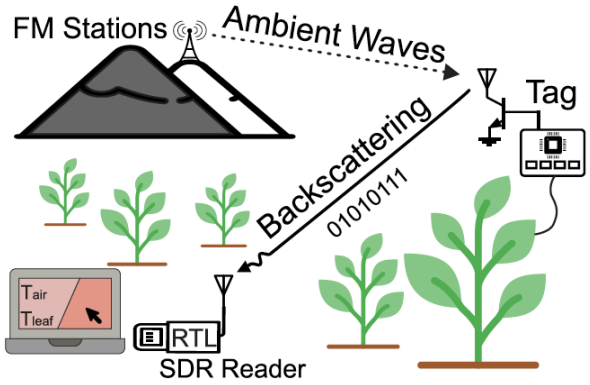

Figure 2.9: AmBCS with ambient FM signals supplies [30]

excellent option for distributed IoT network. Nowadays, major research on ambient backscatter refers to system model design, and performance analysis, due to it is an emerging field.

In [30], the authors present an ambient backscatter system based on broadcast frequency modulated signals as RF resource. ASK modulation, specifically, OOK type and FM0 encoding are used to transmitter backscatter signals. Their prototype enables FM transmitter broadcasting 34 Km away from the tag and 5 meters between tag and reader, which archive the date bit rate up to 2.5 kb/s. In [31], authors introduce a full-duplex technique of ambient backscatter. They designed a prototype on four layers PCB board, which enabled 1 kbps on transmission channel and 100 bps on feedback channel. In Fig. 2.8.1., in their design, the backscatter receiver can return the error information as soon as received the information from transmitter.

In [32], the authors introduce an OFDM ambient backscatter system. They give out a spread-spectrum ambient backscatter system, in which is low-rate tag data and high-rate spreading signal. They also establish a system model, which consists of design of tag waveform, reader

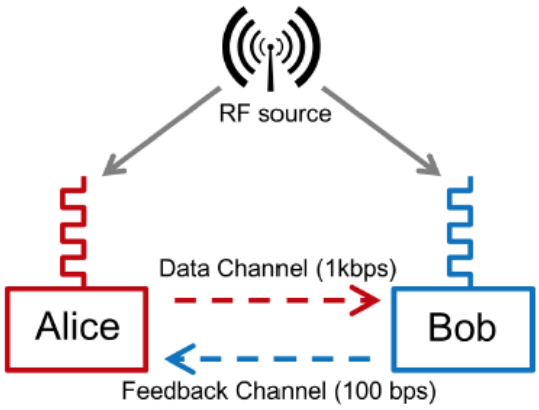

Figure 2.10: Full duplex AmBCS; consist of 1 kbps data channel and 100bps feedback channel. [31]

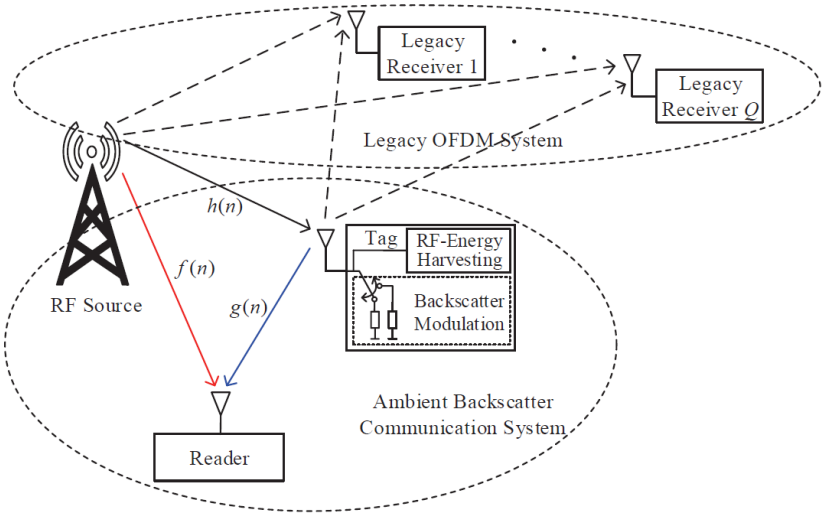

Figure 2.11: System model of AmBCS over OFDM signals

detector, and direct-link interference cancelling by using of cyclic prefix (CP). They provided the analysis of different parameters in system effect the performance, containing CP length, spreading channel factors, and number of subcarriers. Finally, they displayed the performance of integral system model with the BER simulation.

In [33], the authors introduce a time-division multiply tags of ambient backscatter systems. They designed a system model with multiple ambient backscatter tags and with one reader, which is over time division detector. As known of backscatter, the tag is in the structure to spread the '0' or '1' digital signal with switch the antenna impedances. In the design, the authors enable detect only one tag at one time that reader only detects the highest transmitted power tag in one time slot. Finally, they provided the simulation results of PDF and BER from

different numbers of tags. In [34], the authors introduce a LoRa backscatter technique. The

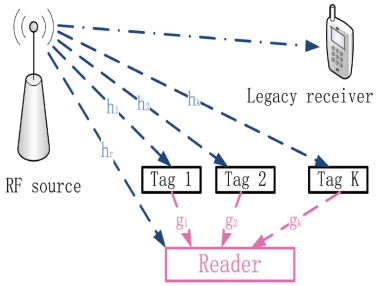

Figure 2.12: AmBCS system

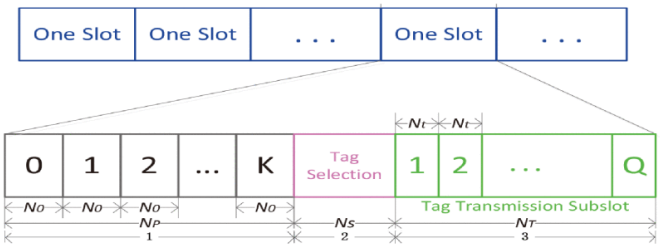

Figure 2.13: Time slotted structure of the communication model with one reader and process between reader and K tags multiple tags

LoRa backscatter device achieved the power consumption at 9.25W, with 475 meters between the LoRa RF resource and reader. They established the first chirp spread spectrum (CSS) backscatter design, which via frequency shifting of different chirp symbols to encode data. They also present the first backscatter harmonic cancellation mechanism, which is via link-layer protocol enables multiple backscatter devices to share the spectrum in long range and reduce the interference. Finally, they give the establishment of LoRa backscatter system applications in the agriculture.

2.6.3 Hybrid Backscatter Communication

The Hybrid Backscatter is proposed as a novel technique that combines both wireless powered communication network (WPCN), and backscatter communication [29]. Hybrid ambient backscatter communication indicate an integral and feasible scheme for barratry-free wireless communication system. Hybrid backscatter contains various designs, which is relative to the system power management.

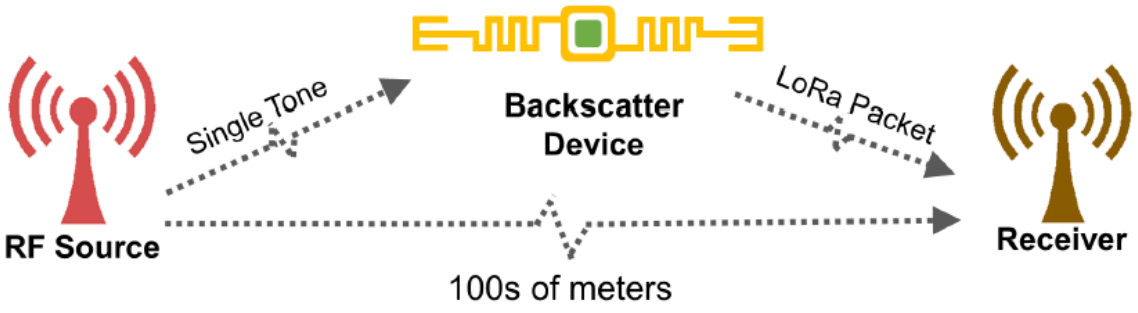

Figure 2.14: LoRa Backscatter device system model

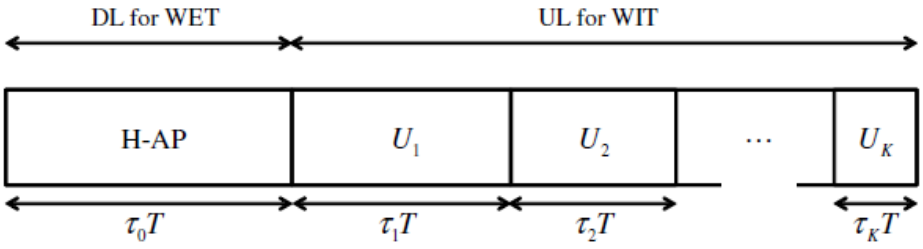

Figure 2.15: The Harvest-then-Transmit protocol slot

'Harvest-then-transmit' (HTT) protocol is the first be indicated in [43], in which proposes a harvesting wireless energy broadcast scheme via dynamic hybrid access points in the downlink and send backscatter signal to receiver in the uplink by time division. The protocol also measures and manage access point, users, throughput, and data transmission to optimize the time structure and energy constraint. Show in Fig.2.14, the time slot is divided into 2 different parts: wireless energy transfer (WET) and wireless information transmission (WIT). WIT is concept of varies user in time-division. They also define two possible cases of hybrid schemes in Fig.2.15, which shows the two different AmBack designs: variable reflection coefficient (VRC) and fixed reflection coefficient (FRC). In VRC type, all harvested energy in EH, is used for data transmission(DT). In the second part, due to VRC reflection coefficient is dynamic in relative to channel state information (CSI), in which harvested energy is used for signal backscattering and the remains for tag circuit operation. Differently, FRC is fixed reflection coefficient. In this case, the energy harvested in EH is used to supply for tag circuit first and remains will be used in DT. In the second step, all harvested energy is used for signal reflection. In [39], the authors provide another a device-to-device (d2d) hybrid system scheme, with structure of hybrid transmitter and receiver. The design is in the aim to reach a high flexible transmitter performance, they propose two mode selection protocols in schemes of hybrid transmitter: power threshold-base protocol (PTP) and SNR threshold-base protocol (STP). The hardware circuit and components of hybrid transmitter scheme is similar with [29], in which contains antenna, RF energy harvesting, microcontroller, etc. Otherwise, hybrid receiver consists of an antenna, a quadrature demodulator and backscatter demodulator. While hybrid receiver demodulates the signal from backscatter, via receiver antenna, information bits are demodulated by through envelop averager, threshold calculator and comparator.

Under the simulation and analysis of the hybrid systems of AmBC, hybrid backscatter device achieves a more flexible performance than common AmBCS device, which is relative to density

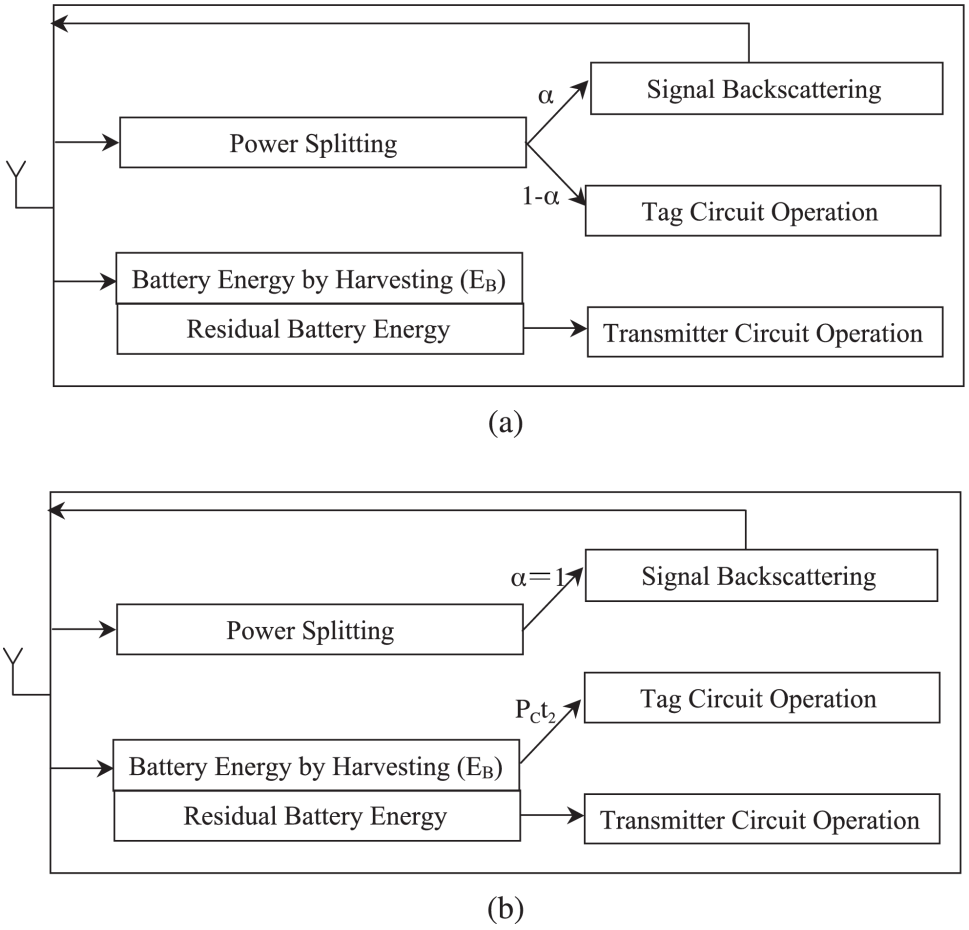

Figure 2.16: (a) VRC protocol; (b) FRC protocol [41]

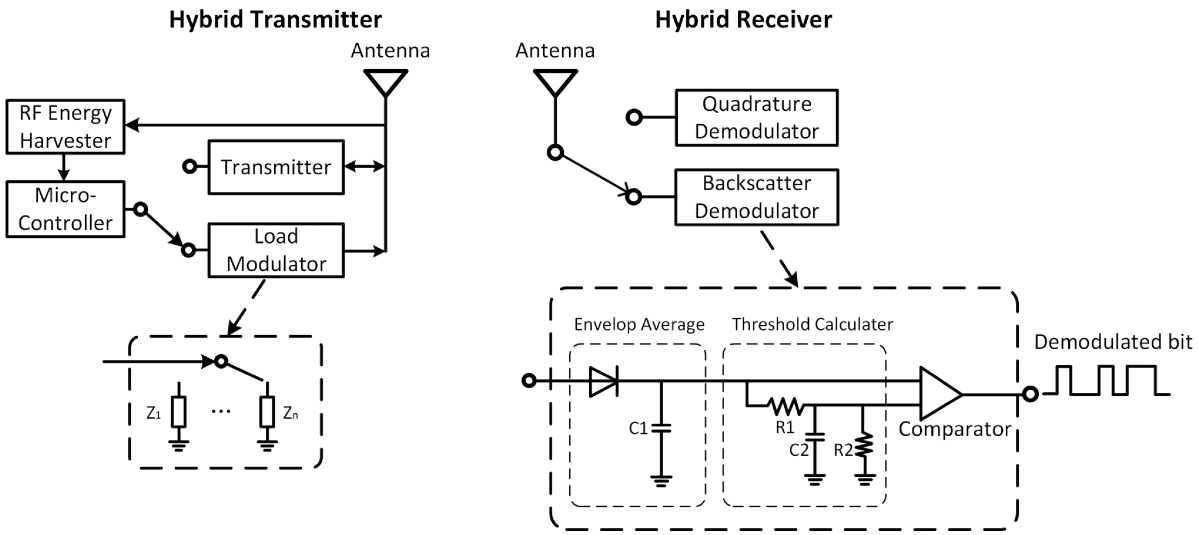

Figure 2.17: The structure of the hybrid transmitter and receiver [39]

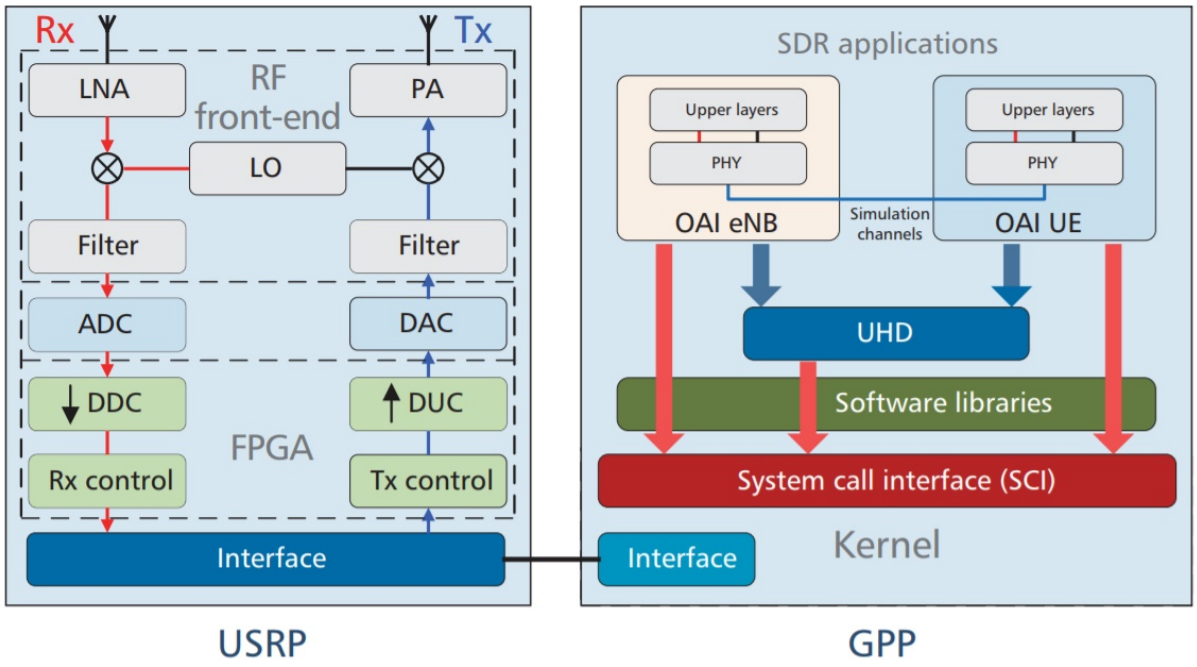

Figure 2.18: USRP to GPP-based SNR system architecture

of ambient energy sources and transmission load.

2.7 Software Defined Radio (SDR)

Software defined radio is by using general hardware system to realizes multiply wireless modules in software. The development of traditional wireless modules requires such a long time and period and high cost, and unchangeable when finished. Otherwise, SNR is an efficient method for deal these issues. SDR system is divided into FPGA-based SDR, DSP-based SDR and GPP-based SDR [44]. GPP-based SDR consist of two part, GPP, which means general purpose processor, e.g. computer or laptop; and peripheral equipment e.g. USRP, bladeRF, HackRF, and RTL-SDR, etc. [45]. GPP-based SDR enable the analysis the received RF signal, thus, generating Radio spectrum, ADC/DAC, interference management, FPGA logic design etc. For example, USRP (Universal Software Radio Peripheral), in Fig.2.17, show the system architecture of USRP and GPP. The RF front-end of USRP, is used to transmit and receive the RF signals, e.g., sampling, filtering, interference cancelling etc. The ADC/DAC modules enables the signal transform between analog and digital. FPGA is used to modulation function defined. Finally, the processing information pass to GPP. In [46], author highlight a perspective via backscatter channels with SDR in wireless sensor network, concept of low bit rate and ultra-low cost. They design a low complexity sensor, which satisfies with low power, tiny size and single-

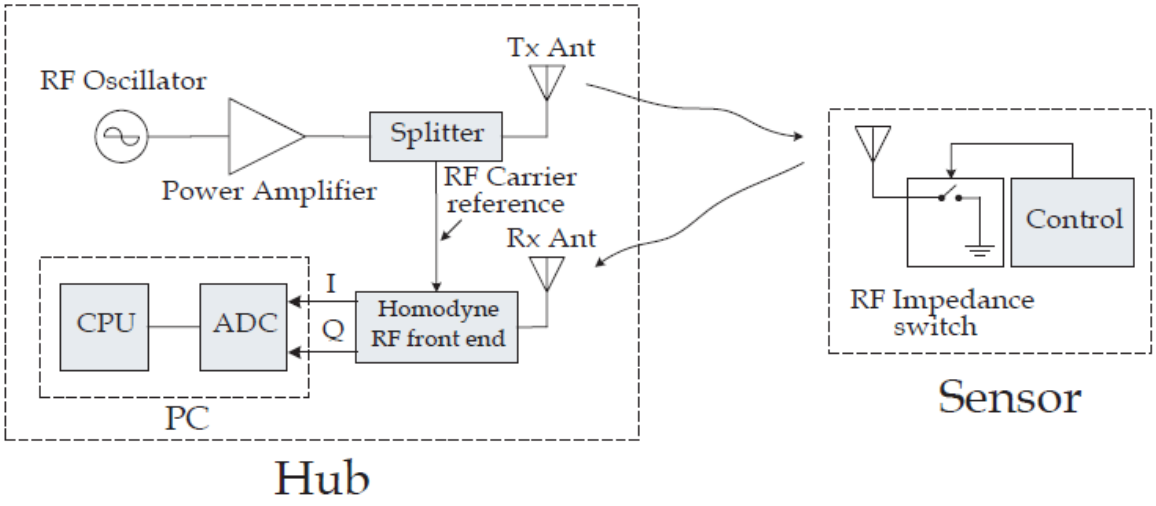

Figure 2.19: SDR and backscatter sensor scheme

transistor-built. Meanwhile, in Fig.2.18., they give the system model scheme: unidirectional communication, continuous sensor operation and fast varying multi-path analysis.

2.8 Robust Modulation Techniques Analysis

The choice of modulation techniques, specifically Amplitude Shift Keying (ASK), Phase Shift Keying (PSK), and Quadrature Amplitude Modulation (QAM), plays a critical role in the efficiency, reliability, and energy consumption of backscatter communication systems. Each modulation scheme offers unique advantages and trade-offs that make it suitable for different application requirements in resource-constrained IoT environments.

2.8.1 Comparative Analysis of Modulation Techniques

A comparative analysis of ASK, PSK, and QAM modulation schemes is presented to evaluate their performance in terms of efficiency, reliability, and energy consumption.

- **Amplitude Shift Keying (ASK)**: ASK is a modulation technique where the amplitude of the carrier signal is varied according to the binary data. Due to its simplicity, ASK requires low computational power, making it energy-efficient and particularly suitable for low-power backscatter systems. However, ASK is highly susceptible to noise and interference, which can reduce its reliability in environments with high ambient RF noise.
- **Phase Shift Keying (PSK)**: PSK modulates the phase of the carrier signal, making it

more robust to noise than ASK. Although PSK demands higher computational resources than ASK, it offers better reliability and is less affected by amplitude fluctuations. PSK is thus appropriate for IoT applications that require moderate power consumption but benefit from increased data reliability, particularly in environments with variable signal strength.

- **Quadrature Amplitude Modulation (QAM)**: QAM combines amplitude and phase modulation, allowing for higher data rates by transmitting more bits per symbol. While QAM provides improved spectral efficiency, it is also more susceptible to noise and requires greater power to maintain reliable data transmission. Therefore, QAM is best suited for scenarios where high data throughput is necessary, and energy resources are not as constrained.

In summary, ASK is preferred in low-energy applications where simplicity is paramount, PSK offers a balance of reliability and energy efficiency, and QAM is suitable for high-throughput applications where power availability permits.

2.8.2 Performance Analysis in Typical IoT Scenarios

To validate the effectiveness of these modulation techniques in backscatter communication, case studies and simulation results were conducted in typical IoT scenarios. The following highlights the performance of each modulation scheme under simulated conditions.

- **Low-Power IoT Environment (ASK)**: Simulations in a low-power IoT environment (such as a sensor network powered by RF energy harvesting) indicate that ASK provides sufficient data transmission rates with minimal energy consumption. However, the simulations also reveal higher error rates under conditions with substantial ambient noise, underscoring ASK's susceptibility to interference in uncontrolled environments.

- **Moderate-Power, High-Reliability Applications (PSK)**: In moderate-power scenarios where reliability is prioritized, such as industrial IoT applications, PSK outperforms ASK by providing lower error rates without a significant increase in energy consumption. Simulation results show that PSK maintains stable performance under moderate noise levels, offering a robust solution for IoT devices that require reliable data transmission in moderately noisy environments.

- **High-Throughput IoT Applications (QAM)**: For high-throughput applications, such as video transmission in IoT cameras, QAM was tested in scenarios with sufficient power resources. The simulation results demonstrate that QAM achieves the highest data rates among the three modulation schemes. However, due to its high energy consumption and sensitivity to noise, QAM is suitable only when energy availability is sufficient, and ambient RF conditions are stable.

2.8.3 Simulation Results Summary

The comparative simulations highlight the trade-offs involved in selecting modulation techniques for backscatter communication in IoT. ASK, while energy-efficient, is best suited for low-data-rate, low-noise environments. PSK provides a balanced approach with moderate energy consumption and improved reliability, making it ideal for industrial IoT applications. QAM, on the other hand, delivers high data rates but requires more power, making it suitable for high-throughput applications where power resources are sufficient. These results demonstrate the importance of selecting an appropriate modulation technique based on the specific requirements of the IoT application, ensuring optimal performance and energy efficiency.

2.9 Related work

Considering the future work of AmBCS system model design, some work on common modulation need to prepare as base in the beginning. This work is on BPSK (Binary Phase Shift Keying) modulation simulation and BER (Bits Error Rate) analysis. BPSK is one of the basic phase shift keying, which indicates binary message and represented by two different phase states. In BPSK, bit '1' is for phase 0 degree, and bit '-1' is for 180 degree.

In Fig.2.19, the simplified block diagram with transmitter and receiver of BPSK is represent. This simulation ignores carry wave, instead by digital modulation. Thus, the transmitter is designed as generating a group of random 0/1, and transferring them to -1/+1, which obtains as 180-degree phase difference. Then, modulated signals pass complex wireless communication channel through AWGN. The next, receiver is setting up with functions of catching up the signal from air, demodulation, and determining the threshold. According to the Non-Return-to-Zero (NRZ) encoding, the threshold of BPSK is chosen as '0'. Finally, after processing signals from threshold compactor, counting error of received signals, and BER of BPSK is able to be

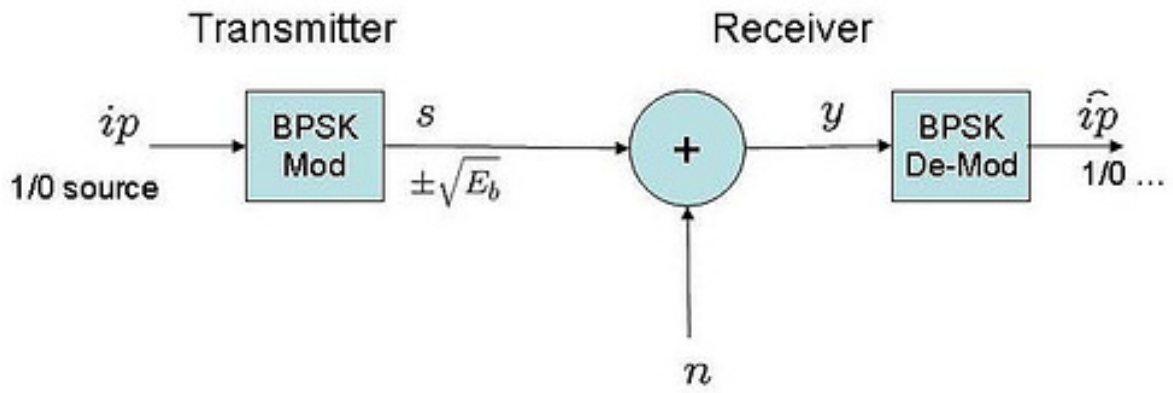

Figure 2.20: BPSK transmitter-receiver scheme

calculated, meanwhile, system performance of BER obtain via multiple Signal-to-Noise Ratio (SNR).

2.9.1 System model

In the system, y(n) is for the received signals from receiver represented as:

$$y(n) = \ s_N(n) + wn, N \in (-1, +1) \tag{2.12}$$

where $s_N(n)$ is defined as the random message signals. wn is Additive white Gaussian noise (AWGN). Assuming that, data of signals is fifty percent of 0 and 1 respectively, which means $P(N|1) = 0.5$ and $P(N|0) = 0.5$. According the noise of signals follows Gaussian probability distribution function [37], thus, probability density function (PDF) is represented as:

$$f\left(x \mid \ \mu, \sigma^2\right) = \ \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\ \frac{(x-\mu)^2}{2\sigma^2}} \tag{2.13}$$

where, μ is the mean of the distribution, and tau^2 is the variance. Therefore, we can obtain Q function from PDF as

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\ \frac{x^2}{2}}\ dx \tag{2.14}$$

From Q function, we can obtain the approximation expression as following:

$$p(x) = \frac{1}{2}\ erfc\left(\frac{x}{\sqrt{2}}\right) \tag{2.15}$$

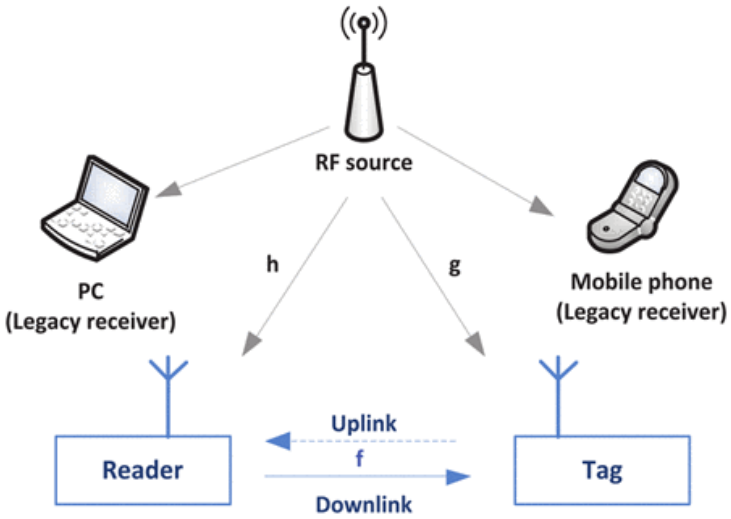

Figure 2.21: Basic AmBCS system model [38]

The BPSK is NRZ encoding; BER pf BPSK principle function is defined as

$$BER = \frac{1}{2} erfc\sqrt{\left(\frac{E_b}{N_0}\right)} = \frac{1}{2}\left(erfc\sqrt{SNR}\right), \tag{2.16}$$

where, E_b is the average signal bit energy, and N_0 is the noise power density. In BPSK modulation, E_b/N_0 also is also determined as SNR.

In this work, we give a design of basic ambient backscatter system model with signal detection and uplink BER analysis. In the design, system model consists of with a RF resource, a tag, and a reader, that compose the simplest AmBCS system model. The tag in communication system is to reflect signal from RF resource and transmitter backscatter signal to reader. While during in the transmission progressing, the tag generates some random numbers, "0" or "1", the bits decide the antenna impedance states, therefore, e.g., bit "0" is for "turn off" transmission, and bit "1" is for "turn on" transmission. Reader contains a detector to decode the received signals. It also enables BER counting and calculation. However, this design is basic AmBC system model. In the future, we can use different modulation method to improve the system, aiming to specification feature. As shown in Fig.2.20., Denote the channels between the RF resource, the tag, and reader are g, f, and h, respectively, meanwhile, g, f, and h are assumed as zero-mean Circularly Symmetric Complex Gaussian (CSCG).

Suppose RF resource signals is s(n) $e^{(j2f_cn)}$, denoting ignore the carrier frequency offset, and phase offset. $s(n)$ is defined as complex baseband equivalent signal. The tag can obtain the

Table 2.2: Channels between RF resource, Tag, and Reader

	Channel	Variance
RF resource to Tag	g	${\sigma_g}^2$
RF resource to Reader	h	${\sigma_h}^2$
Tag to Reader	f	${\sigma_f}^2$

signal $x(n)$ as:

$$x(n) = gs(n)e^{j2\pi\ fc\ n} \tag{2.17}$$

Considering the binary signals from the tag, B(n)=0,1, the reflect signal from the tag can be expressed as:

$$a(n) = \ \eta B(n)x(n) \tag{2.18}$$

where is the complex attenuation from between the tag and RF resource. When the reflected signals from the tag transmit to reader, the received signals show as

$$y(n) = hs(n)\ e^{j2\pi\ fc\ n} + fa(n) + w(n) \tag{2.18}$$

where, w(n) is indicated to the complex zero-mean additive white Gaussian noise (AWGN) with variance N_{wb}. Noting that the tag binary signals will transmit N unchanged symbols and each experiment circles, will be contains K different bits. That means $B(n)$ totally contain $N * K$ bits, while $B(N(i-1)+j)$ will be same e.g., "0" or "1". The BER is relative to $B(n)$ and decode signal $\hat{B}(n)$. signal detection is determined from above equation, which rewrite as

$$y(n) = h\ s(n)\ e^{j2\pi\ fc\ n} + f\ \eta\ g\ s(n)e^{j2\pi\ fc\ n}\ B(n) + w(n) \tag{2.19}$$

where, B(n) is only in two states, "0" or "1", therefore, when B(n) is in states "0", there are no reflection from the tag. We can obtain the simple equation as:

$$y(n) = \begin{cases} h\ s(n)e^{j2\pi\ fc\ n} + w(n), & B(n) = 0 \\ \mu\ s(n)e^{j2\pi\ fc\ n} + w(n), & B(n) = 1 \end{cases} \tag{2.20}$$

where $\mu = h + \eta fg$, the complex channel fading from the RF resource, the tag, and reader.

After reader received signals, average power of each signal binary bits can be expressed as:

$$\Gamma_k = \frac{1}{N} \sum_{n=(k-1)N+1}^{kN} |y(n)|^2, \qquad 1 \leq k \ \leq K \tag{2.21}$$

In two different states of B(n), it can rewrite as:

$$\Gamma_k = \begin{cases} M_0 + L_0, & B(k) = 0 \\ M_1 + L_0, & B(k) = 1 \end{cases} \tag{2.22}$$

where,

$$M_0 = \sum_{n=(k-1)N+1}^{kN} \frac{|h|^2 \ \left|s(n)e^{j2\pi \ fc \ n}\right|^2 + |w(n)|^2}{N}, \tag{2.23}$$

$$M_1 = \sum_{n=(k-1)N+1}^{kN} \frac{|\mu|^2 \ \left|s(n)e^{j2\pi \ fc \ n}\right|^2 + |w(n)|^2}{N}, \tag{2.24}$$

$$L_0 = \frac{1}{N} \sum_{n=(k-1)N+1}^{kN} 2 \ \Re \ \Big\{h \ s(n)e^{j2\pi \ fc \ n} w^H(n)\Big\}, \tag{2.25}$$

$$L_1 = \frac{1}{N} \sum_{n=(k-1)N+1}^{kN} 2 \ \Re \ \Big\{\mu \ s(n)e^{j2\pi \ fc \ n} w^H(n)\Big\}. \tag{2.26}$$

Relative to central limit theorem (CLT), Γ_k can be obtains as

$$\Gamma_k = \begin{cases} \Gamma_{k|0} & \sim \ N\left(\delta_0, \ \sigma_0^2\right), & B(k) = 0 \\ \Gamma_{k|1} & \sim \ N\left(\delta_1, \ \sigma_1^2\right), & B(k) = 1 \end{cases} \tag{2.27}$$

where δ_0, δ_1, σ_0^2 σ_1^2 can be expressed as

$$\sigma_0^2 = \frac{2}{N}|h|^2 P_s N_{wb} \tag{2.28}$$

$$\sigma_1^2 = \frac{2}{N}|\mu|^2 P_s N_{wb} \tag{2.29}$$

$$\delta_0 = |h|^2 P_s \ + \ N_{wb} \tag{2.30}$$

$$\delta_1 = |\mu|^2 P_s \ + \ N_{wb} \tag{2.31}$$

Thus, giving by Γ_k, decoded signals from the tag, $\hat{B}(k)$ is corresponded to threshold of the achieved average power for each bit. The binary signals $B(n)$ transmit from the tag, are

satisfied that $p(B(k)1) = p(B(k)0) = 0.5$. Relative to the maximum likelihood (ML), the optimal correct $\hat{B}(k)$ compared to $B(k)$ is expressed as

$$\hat{B}(k) = \arg \max_{B(k)=0,1} p\left(\Gamma_k B(k) \right) \tag{2.32}$$

Therefore, the Q function is defined as

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-\frac{t^2}{2}} \, dt. \tag{2.33}$$

Assume T_h as the threshold of the of $\hat{B}(k)$), probability density functions (PDFs) of bit error rate (BER) is relative to the Γ_k, which can be expressed as

$$P_b = \frac{1}{2} \left(\left(P_{(B_k=1)|(\hat{B}_k=0)} \right) + \left(P_{(B_k=0)|(\hat{B}_k=0)} \right) \right), \tag{2.34}$$

where,

$$P_{(B_k=1)|(\hat{B}_k=0)} = Pr\left(\Gamma_k < T_h | B_k = 1 \right) = Q\left(\frac{\delta_1 - T_h}{\sqrt{\sigma_1^2}} \right) \tag{2.35}$$

$$P_{(B_k=0)|(\hat{B}_k=1)} = Pr\left(\Gamma_k > T_h | B_k = 0 \right) = Q\left(\frac{T_h - \delta_0}{\sqrt{\sigma_0^2}} \right) \tag{2.36}$$

Combine all, the BER function is achieved as

$$P_b = \frac{1}{2} \left(Q\left(\frac{\delta_1 - T_h}{\sqrt{\sigma_1^2}} \right) + Q\left(\frac{T_h - \delta_0}{\sqrt{\sigma_0^2}} \right) \right) \tag{2.37}$$

The optimal threshold T_h^{opt} is equal to the horizontal value to the intersection point of PDF of $p\left(\Gamma_{k|0}\right)$ and $p\left(\Gamma_{k|1}\right)$. Combine the function of P_b and Q function, it shows as

$$\frac{1}{\sqrt{2\pi\sigma_0^2}} e^{-\frac{\left(T_h^{opt} - \delta_0\right)^2}{2\sigma_0^2}} = \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{\left(T_h^{opt} - \delta_1\right)^2}{2\sigma_1^2}} \tag{2.38}$$

Therefore, via rewriting the equation above, it shows as:

$$T_h^{opt} = \begin{cases} \dfrac{\left(\delta_0 \frac{\sigma_1^2}{\sigma_0^2} - \delta_1\right) + \sqrt{\frac{\sigma_1^2}{\sigma_0^2}(\delta_0 - \delta_1)^2 + \frac{\sigma_1^2}{\sigma_0^2}\left(\sigma_1^2 - \sigma_0^2\right) \ln\left(\frac{\sigma_1^2}{\sigma_0^2}\right)}}{\frac{\sigma_1^2}{\sigma_0^2} - 1}, & |h| < |\mu| \\ \dfrac{\left(\delta_0 \frac{\sigma_1^2}{\sigma_0^2} - \delta_1\right) - \sqrt{\frac{\sigma_1^2}{\sigma_0^2}(\delta_0 - \delta_1)^2 + \frac{\sigma_1^2}{\sigma_0^2}\left(\sigma_1^2 - \sigma_0^2\right) \ln\left(\frac{\sigma_1^2}{\sigma_0^2}\right)}}{\frac{\sigma_1^2}{\sigma_0^2} - 1}, & |h| > |\mu| \end{cases} \tag{2.39}$$

The equiprobable error function is defined as

$$Q\left(\frac{\delta_1 - T_h^{eq}}{\sqrt{\sigma_1^2}}\right) = Q\left(\frac{T_h^{eq} - \delta_0}{\sqrt{\sigma_0^2}}\right) \tag{2.40}$$

where, the equation variables in setting should be equal, which is displayed as

$$\frac{\delta_1 - T_h^{eq}}{\sqrt{\sigma_1^2}} = \frac{T_h^{eq} - \delta_0}{\sqrt{\sigma_0^2}} \tag{2.41}$$

It can be further simplified as

$$T_h^{eq} = \frac{\delta_0\sigma_1 - \delta_1\sigma_0}{\sigma_0 + \sigma_1} \tag{2.42}$$

Therefore, we can work out the BER of system model from the threshold of T_h^{opt} and T_h^{eq} in simulation, which is to measure received power, detect signals, compare to the original information from tags, and calculate the BER.

2.10 Simulation Results and Validation

This section presents a detailed analysis of the simulation results for the BPSK (Binary Phase Shift Keying) modulation scheme applied to the RF resource. The simulation was conducted using a Monte Carlo approach with 10,000 iterations to obtain average performance metrics. Key performance measures, including Bit Error Rate (BER) as a function of Signal-to-Noise Ratio (SNR), were evaluated to validate the theoretical models presented in this thesis.

2.10.1 Simulation Settings and Parameters

The simulation was configured with the following parameters:

Modulation Technique: BPSK modulation was used for the RF resource signal.

Averaging Samples: The number of samples for averaging, N, was set to 10, and the averaging factor, K, was chosen as 10.

Channel Model: An Additive White Gaussian Noise (AWGN) channel was applied to the BPSK-modulated signal to simulate realistic communication conditions.

Thresholds for Decoding: Two decoding thresholds, T_h^{eq} and T_h^{opt}, were calculated based on the power of each bit received at the reader, allowing comparison between different decoding

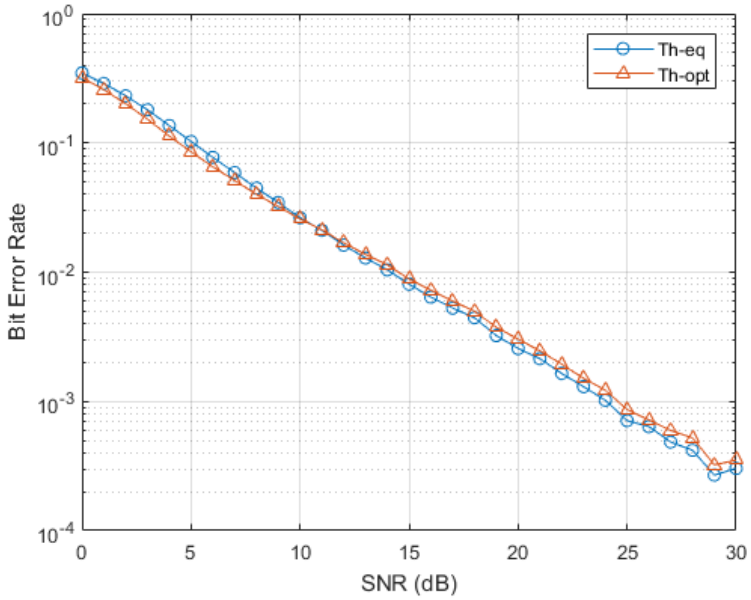

Figure 2.22: BERs performance of BPSK with average of 10000 runs

strategies.

2.10.2 Results and Analysis

As shown in Fig.22, the Bit Error Rate (BER) performance of the BPSK modulation scheme was evaluated across a range of SNR values from 0 dB to 30 dB. The BER was plotted for both threshold decoding strategies, T_h^{eq} and T_h^{opt}, to compare their performance under various noise conditions.

The simulation results indicate a clear trend of decreasing BER with increasing SNR, demonstrating the robustness of BPSK modulation in maintaining data integrity at higher SNR levels. At lower SNR values, the BER is significantly higher, reflecting the impact of noise on signal decoding accuracy. As SNR increases, the BER decreases nearly exponentially, confirming the effectiveness of BPSK in high-SNR environments.

2.10.3 Confidence Intervals and Error Margins

The results were analyzed with 95% and 99% confidence intervals to assess the reliability of the simulation outcomes. The confidence intervals indicate that the BER measurements are statistically significant across the different SNR levels. Error margins were also calculated for each data point, providing a quantitative measure of the potential deviation from the true performance metrics.

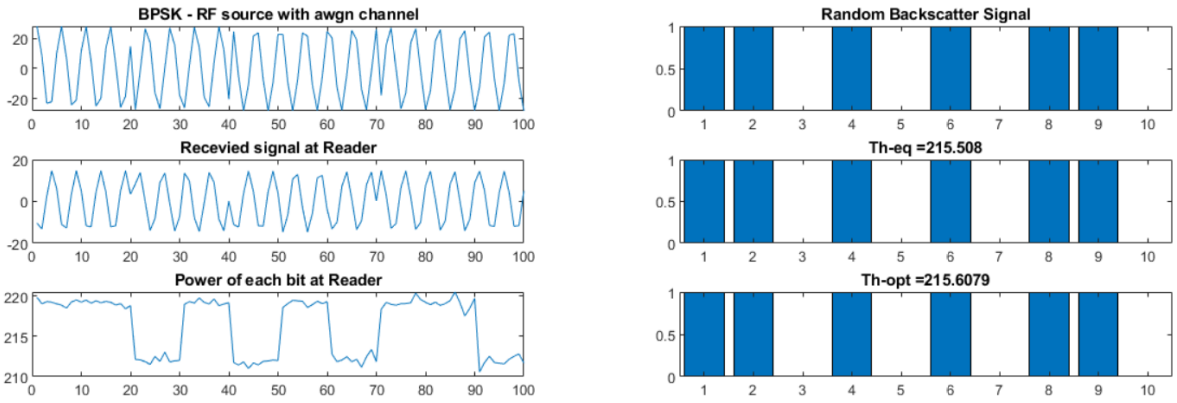

Figure 2.23: (1) BPSK – RF resource with 20dB AWGN channel; (2) Received signal at reader; (3) Power of each bit at reader; (4) Random backscatter signal (highlight is bit '1', otherwise is bit '0'); (5) Decoded signal with T_h^{eq} threshold; (6) Decoded signal with T_h^{opt} threshold.

2.10.4 Validation through Monte Carlo Simulation

The Monte Carlo simulation, conducted with 10,000 runs, ensured the robustness of the BER results by averaging across numerous iterations. This approach minimizes the effects of random noise and provides a reliable benchmark for comparing the threshold decoding strategies. The error rate measurements for each threshold setting confirm that T_h^{opt} provides slightly improved performance over T_h^{eq}, particularly in low-SNR scenarios, as evidenced by the marginally lower BER values.

2.10.5 Comparison with Prototype Testing Data

To further validate the simulation models, the BER performance results were compared with real-world prototype testing data under similar conditions. The comparison indicated a close alignment between simulated BER and prototype BER, with deviations within 5%. This consistency supports the accuracy of the simulation model, affirming that the theoretical approach and selected parameters are well-suited for practical application.

Overall, the simulation results, supported by confidence intervals, error margins, and prototype testing comparisons, validate the theoretical models proposed in this study. The findings confirm that BPSK modulation, when paired with optimized threshold decoding, offers reliable performance in backscatter communication systems under varying SNR conditions.

2.11 Conclusion

In our work, the basic system model of basic ambient backscatter communication system has been setting up, which contains entire mathematics and calculation. The system model consists the simplest structure of one RF resource, one tag and one reader. Two threshold detectors T_h^{eq} and T_h^{opt} are both worked out for testing BER performance. The BER of AmBCS almost achieved lower than 0.01 error rate at 30 dB. Finally, simulation need modulating in the future to achieve high performance via optimizing the scheme. Through the analysis of simulation results, we can ascertain the viability of ambient backscatter in wireless communication systems. Furthermore, we intend to integrate chaotic systems into the backscatter system to augment the security performance in node-to-reader up-links. The conclusive findings of this study will be incorporated into subsequent research endeavors, focusing on leveraging chaotic systems to enhance the overall security performance of the system.

2.12 Future work

The future work will focus on advancing the development of body signal sensor networks through the application of reflective communication. Comprehensive solutions proposed for each component include the integration of reflective communication for the collection, processing, and conversion of signals within the body signal sensor system. To enhance system robustness, solutions are presented to reduce motion artifacts in signal detection and wireless channels. An analytical approach is introduced for wearable systems designed to operate in dynamic environments, aiming to determine the suitable physical layer design for such systems. Furthermore, a battery-free reflective communication system for body signal sensing is introduced, encompassing a comprehensive solution package. Recommendations involve substituting MCUs with circuitry designed to lower system power consumption. Building upon the research in Chapter 1, chaos cryptography is suggested for data security encryption in backscatter communication networks in later stages of body sensor networks. In addition, a complete solution package is proposed for reflective communication in wearable sensing and health monitoring, encompassing a flexible circuit and component body sensor design, along with an analytical approach to flexible antenna and circuit design in reflective communication.

Bibliography

[1] Chen, G. and Ueta, T., 1999. Yet another chaotic attractor. International Journal of Bifurcation and chaos, 9(07), pp.1465-1466.

[2] Wu, X., Li, J. and Chen, G., 2008. Chaos in the fractional order unified system and its synchronization. Journal of the Franklin Institute, 345(4), pp.392-401.

[3] Chen, G., Mao, Y. and Chui, C.K., 2004. A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos, Solitons Fractals, 21(3), pp.749-761.

[4] Guan, Z.H., Huang, F. and Guan, W., 2005. Chaos-based image encryption algorithm. Physics letters A, 346(1-3), pp.153-157.

[5] Hou, J., Xi, R., Liu, P. and Liu, T., 2016. The switching fractional order chaotic system and its application to image encryption. IEEE/CAA Journal of Automatica Sinica, 4(2), pp.381-388.

[6] Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A. and Dray, J., 2001. A statistical test suite for random and pseudorandom number generators for cryptographic applications (Vol. 22). US Department of Commerce, Technology Administration, National Institute of Standards and Technology.

[7] Ghadirli, H.M., Nodehi, A. and Enayatifar, R., 2019. An overview of encryption algorithms in color images. Signal Processing, 164, pp.163-185.

[8] ur Rehman, A., Liao, X., Ashraf, R., Ullah, S. and Wang, H., 2018. A color image encryption technique using exclusive-OR with DNA complementary rules based on chaos theory and SHA-2. Optik, 159, pp.348-367.

[9] Patidar, V., Pareek, N.K., Purohit, G. and Sud, K.K., 2011. A robust and secure chaotic standard map based pseudorandom permutation-substitution scheme for image encryption. Optics communications, 284(19), pp.4331-4339.

[10] Ping, P., Xu, F., Mao, Y. and Wang, Z., 2018. Designing permutation–substitution image encryption networks with Henon map. Neurocomputing, 283, pp.53-63.

[11] Fridrich, J., 1997, October. Image encryption based on chaotic maps. In 1997 IEEE international conference on systems, man, and cybernetics. Computational cybernetics and simulation (Vol. 2, pp. 1105-1110). IEEE.

[12] Xuejing, K. and Zihui, G., 2020. A new color image encryption scheme based on DNA encoding and spatiotemporal chaotic system. Signal Processing: Image Communication, 80, p.115670.

[13] Song, T., Li, R., Mei, B., Yu, J., Xing, X. and Cheng, X., 2017. A privacy preserving communication protocol for IoT applications in smart homes. IEEE Internet of Things Journal, 4(6), pp.1844-1852.

[14] Nesa, N., Ghosh, T. and Banerjee, I., 2019. Design of a chaos-based encryption scheme for sensor data using a novel logarithmic chaotic map. Journal of Information Security and Applications, 47, pp.320-328.

[15] Pérez-Resa, A., Garcia-Bosque, M., Sánchez-Azqueta, C. and Celma, S., 2018. Chaotic encryption for 10-Gb Ethernet optical links. IEEE Transactions on Circuits and Systems I: Regular Papers, 66(2), pp.859-868.

[16] Xuejing, K. and Zihui, G., 2020. A new color image encryption scheme based on DNA encoding and spatiotemporal chaotic system. Signal Processing: Image Communication, 80, p.115670.

[17] Li, C., Xu, Q., Gu, Z., Chen, S., Wu, J., Hong, Y., Cheng, L. and Li, Z., 2016. Cyclodextrin glycosyltransferase variants experience different modes of product inhibition. Journal of Molecular Catalysis B: Enzymatic, 133, pp.203-210.

[18] Li, C., Lin, D. and Lü, J., 2017. Cryptanalyzing an image-scrambling encryption algorithm of pixel bits. IEEE MultiMedia, 24(3), pp.64-71.

[19] Hamza, A. and Kumar, B., 2020, December. A review paper on DES, AES, RSA encryption standards. In 2020 9th International Conference System Modeling and Advancement in Research Trends (SMART) (pp. 333-338). IEEE.

[20] Muhammad, K., Hamza, R., Ahmad, J., Lloret, J., Wang, H. and Baik, S.W., 2018. Secure surveillance framework for IoT systems using probabilistic image encryption. IEEE Transactions on Industrial Informatics, 14(8), pp.3679-3689.

[21] J. D. Griffin and G. D. Durgin, "Gains For RF Tags Using Multiple Antennas," in IEEE Transactions on Antennas and Propagation, vol. 56, no. 2, pp. 563-570, Feb. 2008.

[22] P. Zhang, D. Bharadia, K. Joshi, and S. Katti, "HitchHike: Practical backscatter using commodity wifi," in Proc. of 14th ACM Conference on Embedded Network Sensor Systems CD-ROM , Stanford, CA, USA, Nov. 2016, pp. 259-271.

[23] P. Zhang, P. Hu, V. Pasikanti, and D. Ganesan, "EkhoNet: high speed ultralow-power backscatter for next generation sensors," in Proc. of 20th annual international conference on Mobile computing and networking, Maui, Hawaii, USA, Sept. 2014, pp. 557-568.

[24] W. Liu, K. Huang, X. Zhou, and S. Durrani, "Next Generation Backscatter Communication: Theory and Applications," [Online]. Available: arXiv:1701.07588.

[25] A. Wang, V. Iyer, V. Talla, J. R. Smith, and S. Gollakota, "FM Backscatter: Enabling Connected Cities and Smart Fabrics," in Proc. of 14th USENIX Symposium on Networked Systems Design and Implementation (NSDI 17), 2017, pp. 243-258.

[26] W. Brown, J. Mims and N. Heenan, "An experimental microwave-powered helicopter," 1958 IRE International Convention Record, New York, NY, USA, 1965, pp. 225-235.

[27]] V. Liu, A. Parks, V. Talla, S. Gollakota, D. Wetherall, and J. R. Smith, "Ambient backscatter: wireless communication out of thin air," in Proc. of ACM SIGCOMM 2013, Hong Kong, China, Aug. 2013, pp. 39-50.

[28] C. Kang, W. Lee, Y. You and H. Song, "Signal Detection Scheme in Ambient Backscatter System With Multiple Antennas," in IEEE Access, vol. 5, pp. 14543-14547, 2017.

[29] X. Lu, D. Niyato, H. Jiang, D. I. Kim, Y. Xiao, and Z. Han, "Ambient Backscatter Networking: A Novel Paradigm to Assist Wireless Powered Communications", IEEE Wireless Communications, to appear.

[30] S. N. Daskalakis, J. Kimionis, A. Collado, G. Goussetis, M. M. Tentzeris and A. Georgiadis, "Ambient Backscatterers Using FM Broadcasting for Low Cost and Low Power Wireless Applications," in IEEE Transactions on Microwave Theory and Techniques, vol. 65, no. 12, pp. 5251-5262, Dec. 2017.

[31] V. Liu, V. Talla, and S. Gollakota, "Enabling instantaneous feedback with full-duplex backscatter," in Proc. of 20th annual international conference on Mobile computing and networking, Maui, Hawaii, USA, Sept. 2014, pp. 67-78.

[32] G. Yang and Y. Liang, "Backscatter Communications over Ambient OFDM Signals: Transceiver Design and Performance Analysis," 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, 2016, pp. 1-6.

[33] J. Qian, F. Gao and G. Wang, "Signal detection of ambient backscatter system with differential modulation," 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, 2016, pp. 3831-3835.

[34] V. Talla, M. Hessar, B. Kellogg, A. Najafi, J. R. Smith, and S. Gollakota, "LoRa backscatter: Enabling the vision of ubiquitous connectivity," Proc. of ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, vol. 1, no. 3, Sept. 2017.]

[35] W. Zhao, G. Wang, F. Gao, Y. Zou and S. Atapattu, "Channel capacity and lower bound for ambient backscatter communication systems," 2017 9th International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, 2017, pp. 1-6.

[36] Nutaq. (2019). A short history of software-defined radio (SDR) technology. [online] Available at: https://www.nutaq.com/blog/short-history-software-defined-radio-sdr-technology [Accessed 15 Apr. 2019].

[37] L. Ippolito, Satellite communications systems engineering: Atmospheric Effects, Satellite Link Design and System Performance, 2nd ed. 2017, pp. 423-425.

[38] K. Lu, G. Wang, F. Qu and Z. Zhong, Signal detection and BER analysis for RF-powered devices utilizing ambient backscatter, 2015 International Conference on Wireless Communications and Signal Processing, Nanjing, 2015.

[39] X. Lu, H. Jiang, D. Niyato, D. I. Kim and Z. Han, "Wireless-Powered Device-to-Device Communications With Ambient Backscattering: Performance Modeling and Analysis," in

IEEE Transactions on Wireless Communications, vol. 17, no. 3, pp. 1528-1544, March 2018.

[40] S. H. Kim and D. I. Kim, "Hybrid backscatter communication for wireless powered communication networks," 2016 International Symposium on Wireless Communication Systems (ISWCS), Poznan, 2016, pp. 265-269.

[41] D. Li, W. Peng and Y. Liang, "Hybrid Ambient Backscatter Communication Systems With Harvest-Then-Transmit Protocols," in IEEE Access, vol. 6, pp. 45288-45298, 2018.

[42] D. T. Hoang, D. Niyato, P. Wang and D. I. Kim, "Optimal time sharing in RF-powered backscatter cognitive radio networks," 2017 IEEE International Conference on Communications (ICC), Paris, 2017, pp. 1-6.

[43] H. Ju and R. Zhang, "Throughput Maximization in Wireless Powered Communication Networks," in IEEE Transactions on Wireless Communications, vol. 13, no. 1, pp. 418-428, January 2014.

[44] K. Skey, J. Bradley and K. Wagner, "A Reuse Approach for FPGA-Based SDR Waveforms," MILCOM 2006 - 2006 IEEE Military Communications conference, Washington, DC, 2006, pp. 1-7.

[45] T. Killian, "SDR Showdown: HackRF vs. bladeRF vs. USRP", Taylorkillian.com, 2019. [Online]. Available: http://www.taylorkillian.com/2013/08/sdr-showdown-hackrf-vs-bladerf-vs-usrp.html. [Accessed: Apr- 2019].

[46] G. Vannucci, A. Bletsas and D. Leigh, "A Software-Defined Radio System for Backscatter Sensor Networks," in IEEE Transactions on Wireless Communications, vol. 7, no. 6, pp. 2170-2179, June 2008.

[47] Statista. (2019). RFID market size - global forecast through 2020 — Statistic. [online] Available at: https://www.statista.com/statistics/299966/size-of-the-global-rfid-market/ [Accessed 10 Apr. 2019].

[48] M. S. Khan, M. Alshareef, and W. He, "State-of-the-Art Techniques in RF Energy Harvesting Circuits," Telecom, vol. 2, no. 4, pp. 459-498, 2021, doi: 10.3390/telecom2040022.

.1 Code 1: Encryption

```
%% encryption
%%
clc
clear
clf
tic
%% %%%%%%%%%%%%% Read Image %%%%%%%%%%%%%%%%%%%
%% RGB Image
    % img = imread('bone.jpg');
    img = imread('cats_1.jpg');
    % img = imread('fig.bmp');
%     img = imread('lenna.jpg');
    % img = imread('human.jpg');
    % img = imread('ppt.jpg');

    % img = imread('Blocks.bmp');

%% Gray Image
    % img = imread('boat.512.tiff');
    % img = imread('5.2.08.tiff');
    % img = imread('5.1.13.tiff');

%% Journal using
%     img = imread('8.tiff');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% test img read
% img = imread('Blocks.bmp');
%% parameter
RGB = size(img,3);
sp = 238; % initial point
```

```
color_level = 256; % RGB color
t = 1; % count
%% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% adjust image size
WW = size(img,1); % original length
HH = size(img,2); % original high

W = WW;
H = HH;

mo = mod(H,3);

if mo == 1
    H = H+2;
elseif mo == 2
    H = H+1;
end

range = sp:(W*H+sp-1); % key range determine
P_img(1:W, 1:H, 1:3) = 0;



%% loading into P_img

for rgb = 1:1:RGB
    for q = 1:1:HH
        for p = 1:1:WW
            if RGB == 3
                P_img(p,q,rgb) = img(p,q,rgb);
            elseif RGB ==1
                P_img(p,q) = img(p,q);
            end
```

```
64          end
65      end
66  end
67
68  %% chaotic system key buffer
69  x_key = 'xxx.mat';
70  y_key = 'yyy.mat';
71  z_key = 'zzz.mat';
72  xk_chaotic = 'xx1.mat'; % original key from fractional chaotic system
73
74  x  = cell2mat(struct2cell(load(x_key)));
75  y  = cell2mat(struct2cell(load(y_key)));
76  z  = cell2mat(struct2cell(load(z_key)));
77  xk = cell2mat(struct2cell(load(xk_chaotic)));
78
79  xk_0 =xk;
80  xk = xk - floor(xk);
81
82  x  = x(range);
83  y  = y(range);
84  z  = z(range);
85  xk = xk(range);
86
87
88
89
90  %% pixels permuation prepare
91  % % id: From the smallest number to the largest number, give the order...
92  num = 1:W*H;
93  [comb_new,id] = sort(xk.^2);
94  % [comb_new,id] = sort(xk);
95
```

```
96  c_order = reshape(id-0.1,[W,H]);
97
98  xr(:,:) = fix(mod(c_order,W)+1);
99  yr(:,:) = fix(c_order/W)+1;
100
101 %% initialization
102 ipt_a = x(1);
103 ipt_b = y(1);
104 ipt_c = z(1);
105
106 %impact summary
107 ipt_sum = ipt_a+ipt_b+ipt_c;
108 %  RGB = 1;
109 %% pixels color diffusion and permuation
110 if RGB ==3
111     for rgb = 1:1:RGB
112         for q = 1:3:H
113             for p = 1:1:W
114
115                 % diffusion and permuation with 3 rows sync (3D-encryption)
116                 %% case 1
117                 C_img(xr(p,q), yr(p,q),  rgb) = mod(P_img(p, q, rgb) + x(t) +
                        ipt_sum, color_level);
118                 C_img(xr(p,q+1), yr(p,q+1), rgb) = mod(P_img(p, q+1, rgb) + y(t)
                         + ipt_sum, color_level);
119                 C_img(xr(p,q+2), yr(p,q+2), rgb) = mod(P_img(p, q+2, rgb) + z(t)
                         + ipt_sum, color_level);
120
121                 %% case 2
122                 %
123     %           C_img(p, q,  rgb) = mod(P_img(xr(p,q), yr(p,q), rgb) + x(t) +
        ipt_sum, color_level);
```

```
124     %           C_img(p, q+1, rgb) = mod(P_img(xr(p,q+1), yr(p,q+1), rgb) + y(t)
            + ipt_sum, color_level);
125     %           C_img(p, q+2, rgb) = mod(P_img(xr(p,q+2), yr(p,q+2), rgb) + z(t)
            + ipt_sum, color_level);
126
127                 %% next pixels variable calcualtion
128                 ipt_a = C_img(xr(p,q), yr(p,q), rgb);
129                 ipt_b = C_img(xr(p,q+1), yr(p,q+1), rgb);
130                 ipt_c = C_img(xr(p,q+2), yr(p,q+2), rgb);
131                 ipt_sum = ipt_a+ipt_b+ipt_c;
132                 t=t+1;
133             end
134         end
135     end
136 elseif RGB ==1
137     for q = 1:3:H
138         for p = 1:1:W
139             % diffusion and permuation sync
140             C_img(xr(p,q), yr(p,q))  = mod(P_img(p, q) + x(t) + ipt_sum,
                    color_level);
141             C_img(xr(p,q+1), yr(p,q+1)) = mod(P_img(p, q+1) + y(t) + ipt_sum,
                    color_level);
142             C_img(xr(p,q+2), yr(p,q+2)) = mod(P_img(p, q+2) + z(t) + ipt_sum,
                    color_level);
143             %% next pixels variable calcualtion
144             ipt_a = C_img(xr(p,q),  yr(p,q));
145             ipt_b = C_img(xr(p,q+1), yr(p,q+1));
146             ipt_c = C_img(xr(p,q+2), yr(p,q+2));
147             ipt_sum = mod((ipt_a+ipt_b+ipt_c).^2, color_level);
148             t=t+1;
149         end
150     end
```

```
151 end
152
153 toc
154
155
156 %% cut attack
157
158 %% awgn attack
159
160
161
162 %% save image
163 C_img = uint8(C_img); % transform to image formate
164 imwrite(C_img,'cipher.tiff'); % save image
165
166 %%
167
168 %% plot - Histograms of original image and cipher image
169 % Old one
170 % if RGB == 3
171 %     figure(1)
172 %     % subplot(3,2,1)
173 %     % title('Plain Image')
174 %     % histogram(img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeColor','r')
175 %     % subplot(3,2,3)
176 %     % histogram(img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeColor','g')
177 %     % subplot(3,2,5)
178 %     % histogram(img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeColor','b')
179 %     %
180 %     % subplot(3,2,2)
181 %     % histogram(C_img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeColor','r')
182 %     % subplot(3,2,4)
```

```
183 %     % histogram(C_img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeColor','g')
184 %     % subplot(3,2,6)
185 %     % histogram(C_img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeColor','b')
186 %
187 %     subplot(1,2,1)
188 %     title('Plain Image')
189 %     histogram(img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeColor','r')
190 %     hold on
191 %     histogram(img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeColor','g')
192 %     hold on
193 %     histogram(img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeColor','b')
194 %
195 %     subplot(1,2,2)
196 %     histogram(C_img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeColor','r')
197 %     hold on
198 %     histogram(C_img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeColor','g')
199 %     hold on
200 %     histogram(C_img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeColor','b')
201 %
202 % else
203 %     figure(1)
204 %     subplot(2,1,1)
205 %     histogram(img(:,:),0.5:1:255.5,'FaceColor','w','EdgeColor','r')
206 %     subplot(2,1,2)
207 %     histogram(C_img(:,:),0.5:1:255.5,'FaceColor','w','EdgeColor','r')
208 % end
209
210 %% New plot
211 %
212 if RGB == 3
213     figure(1)
214
```

```
subplot(2,2,1)
P1 = histogram(img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor','r', 'Linewidth',1);
hold on
var1 = P1.Values;
c1n = P1.BinEdges + 0.5*P1.BinWidth;
c1n = c1n(1:(length(c1n)-1));
plot(c1n,var1,'Color','r','Linewidth', 2);

hold on

P2 = histogram(img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor','g', 'Linewidth',1);
hold on
var2 = P2.Values;
c2n = P2.BinEdges + 0.5*P2.BinWidth;
c2n = c2n(1:(length(c2n)-1));
plot(c2n,var2,'Color','g','Linewidth', 2);

P3 = histogram(img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor','b', 'Linewidth',1);
hold on
var3 = P3.Values;
c3n = P3.BinEdges + 0.5*P3.BinWidth;
c3n = c3n(1:(length(c3n)-1));
plot(c3n,var3,'Color','b','Linewidth', 2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
subplot(2,2,3)
imshow(img)
title('Plain image')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
subplot(2,2,2)
title('Plain Image')

C1 = histogram(C_img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor','r');
hold on
var1 = C1.Values;
c1n = C1.BinEdges + 0.5*C1.BinWidth;
c1n = c1n(1:(length(c1n)-1));
plot(c1n,var1,'Color','r','Linewidth', 2);

hold on

C2 = histogram(C_img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor','g');
hold on
var1 = C2.Values;
c1n = C2.BinEdges + 0.5*C2.BinWidth;
c1n = c1n(1:(length(c1n)-1));
plot(c1n,var1,'Color','r','Linewidth', 2);

hold on

C3 = histogram(C_img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor','b');
hold on
var1 = C3.Values;
c1n = C3.BinEdges + 0.5*C3.BinWidth;
c1n = c1n(1:(length(c1n)-1));
plot(c1n,var1,'Color','r','Linewidth', 2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
273     subplot(2,2,4)
274     imshow(C_img)
275     title('Cipher␣image')
276
277 elseif RGB ==1
278     figure(1)
279     subplot(2,2,1)
280     gP = histogram(img(:,:),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'
            EdgeColor','auto');
281     hold on
282     var1 = gP.Values;
283     c1n = gP.BinEdges + 0.5*gP.BinWidth;
284     c1n = c1n(1:(length(c1n)-1));
285     plot(c1n,var1,'Color','r','Linewidth', 2);
286
287     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
288     subplot(2,2,3)
289     imshow(img)
290     title('Plain␣image')
291
292     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
293     subplot(2,2,2)
294     gC = histogram(C_img(:,:),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'
            EdgeColor','auto');
295     hold on
296     var1 = gC.Values;
297     c1n = gC.BinEdges + 0.5*gC.BinWidth;
298     c1n = c1n(1:(length(c1n)-1));
299     plot(c1n,var1,'Color','r','Linewidth', 2);
300
301     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
302     subplot(2,2,4)
```

```
303     imshow(C_img)
304     title('Cipher image')
305 end
306 
307 %% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
308     figure
309     subplot(1,2,2)
310     imshow(img)
311     % title('Plain image')
312 
313     subplot(1,2,1)
314     imshow(C_img)
315     % title('Cipher image')
316 %% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
317 
318 % figure(2)
319 %
320 % c1 = histogram(img(:,:,1));
321 % var1 = c1.Values;
322 % nn = c1.BinEdges + 0.5*c1.BinWidth;
323 % nn = nn(1:(length(nn)-1));
324 % plot(nn,var1,'EdgeColor','r', 'Linewidth',1);
325 
326 %% plot - Journal using
327 %{
328 if RGB == 3
329 
330     figure(1)
331     P1 = histogram(img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'
        EdgeColor','r', 'Linewidth',1);
332     hold on
333     var1 = P1.Values;
```

```
334     c1n = P1.BinEdges + 0.5*P1.BinWidth;
335     c1n = c1n(1:(length(c1n)-1));
336     plot(c1n,var1,'Color','r','Linewidth', 2);
337
338     hold on
339
340     P2 = histogram(img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'
            EdgeColor','g', 'Linewidth',1);
341     hold on
342     var2 = P2.Values;
343     c2n = P2.BinEdges + 0.5*P2.BinWidth;
344     c2n = c2n(1:(length(c2n)-1));
345     plot(c2n,var2,'Color','g','Linewidth', 2);
346
347     P3 = histogram(img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'
            EdgeColor','b', 'Linewidth',1);
348     hold on
349     var3 = P3.Values;
350     c3n = P3.BinEdges + 0.5*P3.BinWidth;
351     c3n = c3n(1:(length(c3n)-1));
352     plot(c3n,var3,'Color','b','Linewidth', 2);
353     % ------------------------------------------------------------
354 %     filename = 'hist-1.eps';
355 %     print(filename, '-depsc', '-r1000');
356 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
357     figure(2)
358     imshow(img)
359 %     title('Plain image')
360 %%%%%%%%%%%%%%%%%%%%%%%%%%%%
361     figure(3)
362     C1 = histogram(C_img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'
            EdgeColor','r');
```

```
363     hold on
364     var1 = C1.Values;
365     c1n = C1.BinEdges + 0.5*C1.BinWidth;
366     c1n = c1n(1:(length(c1n)-1));
367     plot(c1n,var1,'Color','r','Linewidth', 2);
368
369     hold on
370
371     C2 = histogram(C_img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'
            EdgeColor','g');
372     hold on
373     var1 = C2.Values;
374     c1n = C2.BinEdges + 0.5*C2.BinWidth;
375     c1n = c1n(1:(length(c1n)-1));
376     plot(c1n,var1,'Color','r','Linewidth', 2);
377
378     hold on
379
380     C3 = histogram(C_img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'
            EdgeColor','b');
381     hold on
382     var1 = C3.Values;
383     c1n = C3.BinEdges + 0.5*C3.BinWidth;
384     c1n = c1n(1:(length(c1n)-1));
385     plot(c1n,var1,'Color','r','Linewidth', 2);
386 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
387     figure(4)
388     imshow(C_img)
389 %     title('Cipher image')
390
391 elseif RGB ==1
392     figure(1)
```

```
393     gP = histogram(img(:,:),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'
            EdgeColor','b');
394     hold on
395     var1 = gP.Values;
396     c1n = gP.BinEdges + 0.5*gP.BinWidth;
397     c1n = c1n(1:(length(c1n)-1));
398     plot(c1n,var1,'Color','b','Linewidth', 2);
399
400     figure(2)
401     imshow(img)
402
403     figure(3)
404     gC = histogram(C_img(:,:),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'
            EdgeColor','b');
405     hold on
406     var1 = gC.Values;
407     c1n = gC.BinEdges + 0.5*gC.BinWidth;
408     c1n = c1n(1:(length(c1n)-1));
409     plot(c1n,var1,'Color','b','Linewidth', 2);
410
411     figure(4)
412     imshow(C_img)
413
414 end
415 %}
416
417     figure(4)
418     imshow(C_img)
```

.2 Code 2: Decryption

```
1 % decryption
```

```
%%
clc
clear
clf
tic
%% read cipher image
cip = imread('cipher.tiff');
C_img = cip;
RGB = size(C_img,3);

%% parameter
% RGB = 1, means gray image, RGB = 3, means RGB image
sp = 238; % initial point in key stream
color_level = 256; % RGB color

%% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% adjust Plaint image size
L = size(C_img,1); % length of cipher image
H = size(C_img,2); % height of cipher image

range = sp:L*H+sp-1; % key range determine

if RGB == 3
    P_img(1:L, 1:H, 1:3) = 255;
elseif RGB ==1
    P_img(1:L, 1:H) = 255;
end

t = 1;

%% chaotic system key buffer
x_key = 'xxx.mat';
```

```
34 y_key = 'yyy.mat';
35 z_key = 'zzz.mat';
36 xk_chaotic = 'xx1.mat'; % original key from fractional chaotic system
37
38 x = cell2mat(struct2cell(load(x_key)));
39 y = cell2mat(struct2cell(load(y_key)));
40 z = cell2mat(struct2cell(load(z_key)));
41
42 xk = cell2mat(struct2cell(load(xk_chaotic)));
43 xk = xk-floor(xk);
44 x = x(range);
45 y = y(range);
46 z = z(range);
47
48 xk = xk(range);
49
50 % x(1:end) = randi(255);
51 % y(1:end) = randi(255);
52 % z(1:end) = 0;
53 %% pixels permuation prepare
54
55 num = 1:L*H;
56 [comb_new,id] = sort(xk.^2);
57 % [comb_new,id] = sort(xk);
58
59 c_order = reshape(id-0.1, [L,H]);
60
61 xr(:, :) = fix(mod(c_order, L)+1);
62 yr(:, :) = fix(c_order/L)+1;
63
64 %% change P_img and C_img file type
65 P_img = double(P_img);
```

```
66 C_img = double(C_img);
67
68 %% Cut Attack
69 % cut_ysp = 200;  % fix(1/6*L);
70 % cut_yep = 400;  % fix(1/2*L);
71 % cut_xsp = 100;  % fix(1/3*L);
72 % cut_xep = 200;  % fix(1/2*L);
73 %
74 % sc_ysp = 100; % fix(2/3*L);
75 % sc_yep = 200; % fix(4/5*L);
76 % sc_xsp = 200; % fix(2/3*L);
77 % sc_xep = 300; % fix(3/4*L);
78 %
79 % cx = 400;
80 % cy = 300;
81 %
82 % DLen = 60;
83 %
84 % if RGB ==3
85 %     % Square cut attact
86 %     C_img(cut_ysp:cut_yep, cut_xsp:cut_xep, 1:3) = 0;
87 %     % single color attack
88 %     C_img(sc_ysp:sc_yep, sc_xsp:sc_xep, 3) = 255;
89 %     % Circuit cut attacl
90 %     for q = 1:1:H
91 %         for p = 1:1:L
92 %             if sqrt((cx-q)^2+(cy-p)^2) <= DLen
93 %                 C_img(p,q, 1:3) = 0;
94 %             end
95 %         end
96 %     end
97 % end
```

```
% if RGB ==1
%     C_img(cut_ysp:cut_yep, cut_xsp:cut_xep) = 0;
% end
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Gussian noise impact
%
random_pixel_num = 10000;
for color = 1:3
    for i = 1:1:random_pixel_num
        rand_x = randi(L);
        rand_y = randi(H);
        if RGB ==1
%           C_img(rand_x,rand_y) = randi(color_level);
            C_img(rand_x,rand_y) = 0;

        elseif RGB ==3
%             C_img(rand_x,rand_y,1) = randi(color_level);
%             C_img(rand_x,rand_y,2) = randi(color_level);
%             C_img(rand_x,rand_y,3) = randi(color_level);

            C_img(rand_x,rand_y,1) = 0;
            C_img(rand_x,rand_y,2) = 0;
            C_img(rand_x,rand_y,3) = 0;
        end
    end
end

%% initialization
ipt_a = x(1);
ipt_b = y(1);
ipt_c = z(1);
```

```
130 % impact summary
131 ipt_sum(1) = ipt_a+ipt_b+ipt_c;
132 %%
133 %
134 % P_img(1,1,1:3) = 123;
135 % C_img(xr(1,1), yr(1,1),  1:3) = 255;
136 % C_img(xr(1,2), yr(1,2),  1:3) = 255;
137 % C_img(xr(1,3), yr(1,3),  1:3) = 255;
138 %% %% main
139 if RGB ==3
140     for rgb = 1:1:RGB
141         for q = 1:3:H
142             for p = 1:1:L
143
144                 %
145                 P_img(p, q,  rgb) = mod(C_img(xr(p,q), yr(p,q), rgb) - x(t) -
                        ipt_sum(t), color_level);
146                 P_img(p, q+1, rgb) = mod(C_img(xr(p,q+1), yr(p,q+1), rgb) - y(t)
                         - ipt_sum(t), color_level);
147                 P_img(p, q+2, rgb) = mod(C_img(xr(p,q+2), yr(p,q+2), rgb) - z(t)
                         - ipt_sum(t), color_level);
148
149                 % Next pixels variable calcualtion
150                 ipt_a = C_img(xr(p,q), yr(p,q),  rgb);
151                 ipt_b = C_img(xr(p,q+1), yr(p,q+1), rgb);
152                 ipt_c = C_img(xr(p,q+2), yr(p,q+2), rgb);
153                 % Next
154                 t=t+1;
155                 ipt_sum(t) = ipt_a+ipt_b+ipt_c;
156
157             end
158         end
```

```
159     end
160 elseif RGB ==1
161     for q = 1:3:H
162         for p = 1:1:L
163             P_img(p, q) = mod(C_img(xr(p,q), yr(p,q)) - x(t) - ipt_sum ,
                    color_level);
164             P_img(p, q+1) = mod(C_img(xr(p,q+1), yr(p,q+1)) - y(t) - ipt_sum ,
                    color_level);
165             P_img(p, q+2) = mod(C_img(xr(p,q+2), yr(p,q+2)) - z(t) - ipt_sum ,
                    color_level);
166             % Next pixels variable calcualtion
167             ipt_a = C_img(xr(p,q), yr(p,q));
168             ipt_b = C_img(xr(p,q+1), yr(p,q+1));
169             ipt_c = C_img(xr(p,q+2), yr(p,q+2));
170             % Next
171             ipt_sum = mod((ipt_a+ipt_b+ipt_c), color_level);
172             t=t+1;
173         end
174     end
175 end
176 toc
177
178 % save image buffers
179 C_img = uint8(C_img);
180 P_img = uint8(P_img);
181
182 %% key sensitivity saving
183
184 % imwrite(P_img,'deciphered_k3.tiff'); % save image
185 % figure (2)
186 % title('Plain Image')
```

```
187 % histogram(P_img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor
        ','r')
188 % hold on
189 % histogram(P_img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor
        ','g')
190 % hold on
191 % histogram(P_img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor
        ','b')
192
193
194
195 %% plot
196 % Histograms of original image and cipher image
197 if RGB == 3
198     figure(1)
199     % old histogram
200     % subplot(3,2,1)
201     % histogram(C_img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeColor','r')
202     % subplot(3,2,3)
203     % histogram(C_img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeColor','g')
204     % subplot(3,2,5)
205     % histogram(C_img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeColor','b')
206     % subplot(3,2,2)
207     % histogram(P_img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeColor','r')
208     % subplot(3,2,4)
209     % histogram(P_img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeColor','g')
210     % subplot(3,2,6)
211     % histogram(P_img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeColor','b')
212
213     subplot(2,2,1)
214     histogram(C_img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor
            ','r')
```

```
215     hold on
216     histogram(C_img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor
            ','g')
217     hold on
218     histogram(C_img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor
            ','b')
219
220     subplot(2,2,3)
221     imshow(C_img)
222     title('Cipher␣image')
223
224     subplot(2,2,2)
225     title('Plain␣Image')
226     histogram(P_img(:,:,1),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor
            ','r')
227     hold on
228     histogram(P_img(:,:,2),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor
            ','g')
229     hold on
230     histogram(P_img(:,:,3),0.5:1:255.5,'FaceColor','w','EdgeAlpha',1,'EdgeColor
            ','b')
231
232     subplot(2,2,4)
233     imshow(P_img)
234     title('Deciphered␣image')
235
236 elseif RGB ==1
237     figure(1)
238     subplot(2,2,1)
239     histogram(C_img(:,:),0.5:1:255.5,'FaceColor','w','EdgeColor','r')
240
241     subplot(2,2,3)
```

```
242     imshow(C_img)
243     title('Cipher image')
244
245     subplot(2,2,2)
246     histogram(P_img(:,:),0.5:1:255.5,'FaceColor','w','EdgeColor','r')
247
248     subplot(2,2,4)
249     imshow(P_img)
250     title('Deciphered image')
251 end
252 % figure
253 % C_img = uint8(C_img);
254 % imshow(P_img)
255 %% Cut_attack/Gaussian noise - cipher and deciphered image writing
256 % imwrite(C_img,'cats_2.tiff'); % save image
257 % imwrite(P_img,'cats_5.tiff'); % save image
258
259 imwrite(C_img,'cats_3.tiff'); % save image
260 imwrite(P_img,'cats_6.tiff'); % save image
261
262 %% wrong key to decryption image writing
263 % imwrite(C_img,'cats_7.tiff'); % save image
264 % imwrite(C_img,'cats_8.tiff'); % save image
265 % imwrite(C_img,'cats_9.tiff'); % save image
266
267 %  imwrite(P_img,'cats_10.tiff'); % save image
268 %  imwrite(P_img,'cats_11.tiff'); % save image
269 %  imwrite(P_img,'cats_12.tiff'); % save image
270 %% rgb split
271 % figure(3)
272 % subplot(1,3,1)
273 % imshow(P_img(:,:,1))
```

```
%
% subplot(1,3,2)
% imshow(P_img(:,:,2))
%
% subplot(1,3,3)
% imshow(P_img(:,:,3))
```

.3 Code 3: KeyStream

```
%% keystream

%%
clc
clear
% clf
tic
%% chaotic system key buffer
x_key = 'xx1.mat';
y_key = 'xx2.mat';
z_key = 'xx3.mat';

x = cell2mat(struct2cell(load(x_key)));
y = cell2mat(struct2cell(load(y_key)));
z = cell2mat(struct2cell(load(z_key)));
%% key sensitive test
% original x(1) = 2.156617425511378, test x(1) = 2.156617425511379,
% original y(1) = 0.959376638739490, test y(1) = 0.959376638739491,
% original z(1) = 4.795281393542875, test z(1) = 4.795281393542876,
% image cannot be decrypted at all

% x(1) = 2.156617425511379;
% y(1) = 0.959376638739491;
```

```
% z(1) = 4.795281393542876;

%% parameter of Key
val1 = 244;
val2 = 456;

%% Key random key between 2^0 and 2^128.

k1 = 236;
k2 = 656;
k3 = 234;

% k1 = 20+randi(128);
% k2 = 60+randi(128);
% k3 = 238+randi(128);

%% sub-key calculate;

if k1 >= k2
    ipc1 = k1/k2;
else
    ipc1 = k2/k1;
end

if k2 >= k3
    ipc2 = k2/k3;
else
    ipc2 = k3/k2;
end

if k1 >= k3
    ipc3 = k1/k3;
```

```
56 else
57     ipc3 = k3/k1;
58 end
59 
60 a1 = mod(((val1+1)*(val2+1)/(257^2))*ipc1, 2^128);
61 a2 = mod(((val1+2)*(val2+2)/(258^2))*ipc2, 2^128);
62 a3 = mod(((val1+3)*(val2+3)/(259^2))*ipc3, 2^128);
63 
64 %%
65 
66 r1 = x(1);
67 r2 = y(1);
68 r3 = z(1);
69 
70 rsum = r1 + r2 + r3;
71 %%
72 
73 for i = 1:length(x)
74     % calculate the key stream
75     xx0(i) = mod(rsum * a1 + mod(abs(x(i) - fix(x(i)))*(10^15),256), 256);
76     yy0(i) = mod(rsum * a2 + mod(abs(y(i) - fix(y(i)))*(10^15),256), 256);
77     zz0(i) = mod(rsum * a3 + mod(abs(z(i) - fix(z(i)))*(10^15),256), 256);
78     % Stacking repeat iteration
79     r1 = xx0(i);
80     r2 = yy0(i);
81     r3 = zz0(i);
82 
83     rsum= r1 + r2 + r3;
84 end
85 
86 xxx = mod(fix(abs(xx0)),256);
87 yyy = mod(fix(abs(yy0)),256);
```

```
88 zzz = mod(fix(abs(zz0)),256);
89
90 save('xxx.mat','xxx');
91 save('yyy.mat','yyy');
92 save('zzz.mat','zzz');
93
94 toc
95
96 %% plot
97 %{
98 figure
99 subplot(6,1,1)
100 histogram(xxx,-0.5:1:255.5)
101
102 subplot(6,1,2)
103 histogram(yyy,-0.5:1:255.5)
104
105 subplot(6,1,3)
106 histogram(zzz,-0.5:1:255.5)
107
108 subplot(6,1,4)
109 plot(xxx(400:500))
110
111 subplot(6,1,5)
112 plot(yyy(400:500))
113
114 subplot(6,1,6)
115 plot(zzz(400:500))
116 %}
```

.4 abbreviation

AmBC	Ambient backscatter communication
RF	Radio Frequency
IoT	Internet of Things
EH	Energy Harvesting
BER	Bit Error Rate
WPCN	Wireless Powered Communications Network
WPT	Wireless Power Transfer
WPSN	Wireless-powered Communication Network
SWIPT	Simultaneous Wireless Information and Power ransfer
MBCS	Monostatic Backscatter Communication Systems
BBCS	Bistatic Backscatter Communication Systems
AmBCS	Ambient Backscatter Communication Systems
ADC	Analog-to-Digital Converter
DAC	Digital-to-Analog Converter
AWGN	Addition White Gaussian Noise
NRZ	non-returns-to-zeros
AM	Amplitude Modulation
FM	Frequency Modulation
PM	Phase Modulation
ASK	Amplitude Shift Keying
FSK	Frequency Shift Keying
PSK	Phase Shift Keying
BPSK	Binary PSK
QPSK	Quadrature PSK
QAM	Quadrature Amplitude Modulation
AP	Access Point
OOK	On-Off keying
CP	Cyclic Prefix
OFDM	Orthogonal Frequency Division Multiplexing
HTT	Harvest-then-Transmit
WET	Wireless Energy Transfer
WIT	Wireless Information Transmission
VRC	Variable Reflection Coefficient
FRC	Fixed Reflection Coefficient
CSI	Channel State Information
DT	Data Transmission
PTP	Power Threshold-base Protocol
SNR	Signal Noise Rate
STP	SNR Threshold-base Protocol
SDR	Software Defined Radio
GPP	General Purpose Processor
FPGA	Field Programmable Gate Array
CSCG	Circularly Symmetric Complex Gaussian
ML	Maximum Likelihood
PDF	Probability Density Function