

Republic of Iraq
Ministry of Higher Education
and Scientific Research
Al-Nahrain University
College of Science



MOBILE APPLICATION FOR SECURE DATA IN QR CODES

**A Thesis Submitted to the College of Science at Al-Nahrain
University as a Partial Fulfillment of the Requirements for the
Degree of Master in Computer Science**

By

Basheer Nahiz Ameen

(B.Sc. in Computer Science, 2013)

Supervisor

Dr. Sawsan Kamal Thamer

Jamadi Al'wlaa
January

1438 A. H.
2017 A. D.

Abstract

QR_Code (Quick Response Barcode), is two dimensional barcode that encode special information such as (numeric, URLs, alphanumeric and byte or binary). QR Codes are mainly used to carry or store messages because they have higher or large storage capacity than any other normal conventional 'barcodes'.

In this work, the motivation is to use QR codes for security purposes or to use QR codes to send and receive data in a secret manner and that is what is newly discussed in the presented thesis.

Three algorithms (2 for ciphering-QR code and 1 for hiding-QR code) are introduced. In the first ciphering algorithm, QR code is generated for the secret message and the secret key, the encryption process is done by XOR part (series of pixels) of QR code image for secret message with the same part of QR code image for secret key, and then embedding the key into the resulted QR to produce ciphered QR code.

In the second ciphering algorithm, QR code is generated for secret message only; the encryption process is done by inverting two important areas in QR code image for secret message to produce ciphered QR code.

Finally, the third hiding algorithm, two QR codes are generated. One for secret message and the other for the cover of secret message. The hiding process is done by hiding the first QR code image of the secret message into second QR code image of the cover to produce stego-QR that provides a normal data without any suspicious when one scanned the resulted QR code.

In this thesis, the resulted QR code can be kept with the person all the time and the QR Code scanner with a special program, which can decode or extract the important information with an authentic data and saved in it.

Each stage aims to design and implement a mobile application that uses QR_Code technique to encrypt or hide data for security, such application requires first design the software part, which is a program written in android studio software. This software provides an application (of extension .apk) files which are accepted on mobile.

The results obtained in this thesis are as follows. The first and second methods in the process of encoding and decoding are given the MSE 0 and the PSNR infinite and the third method gives low percent in MSE and high percent in PSNR.

List of Abbreviations

<i>Abbreviation</i>	<i>Meaning</i>
HDD	Hard Disk Drive.
IDE	Integrated Development Environment.
IEC	International Electrotechnical Commission.
ISO	International Standard Organization.
LSB	Least Significant Bit.
MSD	Mean Squared Deviation.
MSE	Mean Squared Error.
OS	Operating System.
PSNR	Peak Signal-to-Noise Ratio.
QR	Quick Response.
RAM	Random Access Memory.
RMSE	Root-Mean-Square Error.
RMSD	Root-Mean-Square Deviation.
ROI	Region Of Interest.
SDK	Software Development Kit.
SQRC	Secure Quick Response Code.
URL	Uniform Resource Locator.
VGA	Video Graphics Array.
VQ	Vector Quantization.
XML	Extensible Markup Language.

Table of Contents

Abstract	i
List of Abbreviations	iii
Table of Contents	iv
List of Figures	vii
List of Tables	ix

Chapter One

Introduction

1.1	Introduction	1
1.2	Data Hiding	1
1.3	Android operating system	3
1.4	Related Work	4
1.5	Aim of Thesis	6
1.6	Thesis Layout	6

Chapter Two

Theoretical Concepts

2.1	Introduction	8
2.2	Data Security	8
2.2.A	Cryptography	9
2.2.B	Steganography	10
2.3	QR Code	10
2.3.1	Fundamentals of QR Code	11
2.3.2	QR Code characteristics	12
2.3.3	Types of QR Code	17
2.3.4	QR Code Technologies	22
2.4	Quality Measures	28

Chapter Three Practical Work

3.1	Introduction	31
3.2	Ciphering Special Versions of QR code with XOR Application	34
3.2.1	Ciphering Algorithm	35
3.2.2	Sender Side Algorithm	36
3.2.3	Receiver Side Algorithm	39
3.3	Ciphering General Versions of QR code Application	41
3.3.1	Ciphering Algorithm	41
3.3.2	Sender Side Algorithm	42
3.3.3	Receiver Side Algorithm	44
3.4	Hiding Information Using QR code Application	46
3.4.1	Hiding Algorithm	48
3.4.2	Extraction Algorithm	50

Chapter Four Systems Interfaces and Test

4.1	Introduction	52
4.2	Application's Requirements	52
4.3	Software Programming Requirements	53
4.4	Ciphering Special Versions of QR code with XOR Application Interfaces and Result	53
4.4.1	Results	54
4.4.2	Discussion	55
4.5	Ciphering General Versions of QR code Application Interfaces and Result	56
4.5.1	Results	56
4.5.2	Discussion	57

4.6	Hiding Information Using QR code Application Interfaces and Result	58
4.6.1	Result	58
4.6.2	Discussion	59

Chapter Five

Conclusions and Suggestions for Future Work

5.1	Conclusions	60
5.2	Suggestions for Future Work	60

<i>References</i>	62
<i>Appendices</i>	

List of Figures

<i>Figure</i>	<i>Title</i>	<i>Page</i>
Figure (1.1)	Block Diagram of Digital Image Steganography.	2
Figure (1.2)	The Android layers.	3
Figure (2.1)	The cryptography diagram.	9
Figure (2.2)	Image Steganography processes.	10
Figure (2.3)	An example of the QR Code encodes numeric and alphabetic characters.	13
Figure (2.4)	Kanji and Kana is converted to QR Code.	14
Figure (2.5)	Dirt and damage of QR Code.	15
Figure (2.6)	Structured appending of QR Code.	17
Figure (2.7)	The symbol of Micro QR Code and QR Code.	18
Figure (2.8)	The rectangular modules of IQR Code.	19
Figure (2.9)	Comparison of regular QR Code and IQR Code.	20
Figure (2.10)	An example of LogoQ.	21
Figure (2.11)	Version of QR code symbols.	23
Figure (2.12)	Structure of QR Code.	24
Figure (2.13)	QR Code decoding process.	27
Figure (3.1)	Block diagram of three applications.	34
Figure (3.2)	Example of encryption process.	35
Figure (3.3)	Example of decryption process.	36
Figure (3.4)	Flowchart of encryption process.	36
Figure (3.5)	Flowchart of decryption process.	39
Figure (3.6)	Example of above algorithm.	42

Figure (3.7)	Flowchart of encryption process.	43
Figure (3.8)	Flowchart of decryption process.	45
Figure (3.9)	Hiding (Steganography) system.	46
Figure (3.10)	Least Significant Bit (LSB) substitution.	47
Figure (3.11)	simple effective of red color.	47
Figure (3.12)	Flowchart of hiding process.	48
Figure (3.13)	Flowchart of extracting process.	50
Figure (4.1)	Systems work.	52
Figure (4.2)	The Changing of MSE & PSNR by increase the Message Size.	59

List of Tables

<i>Tables</i>	<i>Title</i>	<i>Page</i>
2.1	Types of QR code.	17
3.1	Embed key process in sender side example.	35
4.1	Special ciphering QR code application results.	55
4.2	Special ciphering QR code application results.	57
4.3	Hiding Information Using QR code application Results.	58

CHAPTER ONE

GENERAL INTRODUCTION

1.1 Introduction

In nowadays world, security is a large topic and securing significant data is very necessary, so that the data cannot be interrupted or misused for any sort of illegal use. The hackers and interlopers are always stand by to get through individual data or important data of a person or an organization, and abuse them in various ways. For this reason, the field of cryptography is very vital and the cryptographers are trying to introduce new cryptographic routines to secure the data as much as possible. The protection of his/her worthy data like passport information, bank statements, social security number, etc. with himself/herself all the time, but one is always afraid of doing so because these information are intimidated and can be easily intercepted by outsiders for misuse. Another example is shown; a bank manager wants to enjoin his subordinates about the operation of a huge transaction. If this data is not encrypted duly, then it can be regained by a hacker to redirect the transaction process to credit a different account. For this purpose, encryption of data and disguise data from unauthentic usage is very important. One can disband this matter by encoding data into QR code and encrypting or hiding the QR Code [Kie10] [Dey13] [Dey12].

1.2 Data Hiding

The popular saying ‘a picture is worth a thousand words’ was certainly true until last decade but, the growing research interests in the field of digital image processing during the last decade have changed this estimation about a picture. Now pictures in their digital representations speak much more than a thousand words. For example, the block diagram given in figure (1.1)

explains the process of Steganography in which generally some secret message is embedded into an innocuous looking simple image (called as the cover image) and create a Stego image. The Stego image visually seems to be indifferent from the original cover but hides the secret message inside it and is transmitted to the desired recipients over the communication channels without creating any suspicion in the minds of the intermediately sniffers or/and receivers [Cha11].

When the authorized recipient receives the image, one follows the extraction procedure to retrieve the secret message. To increase the secrecy or security of the hidden message, some keys may be involved in this process of embedding and extraction. At the transmission end, during embedding, the message can suitably be encrypted using one or more encryption techniques. These encryption standards can be key based encryptions or non-key based and in key based techniques, they again can be public or private or a mix. Depending upon the encryption method used during the embedding process, the receiver needs to execute certain decryption algorithms to retrieve the correct message. If any decryption algorithm or the keys used for the procedure or the sequence is not known to the receiver then the extraction fails and the receiver cannot retrieve the message [Cha11].

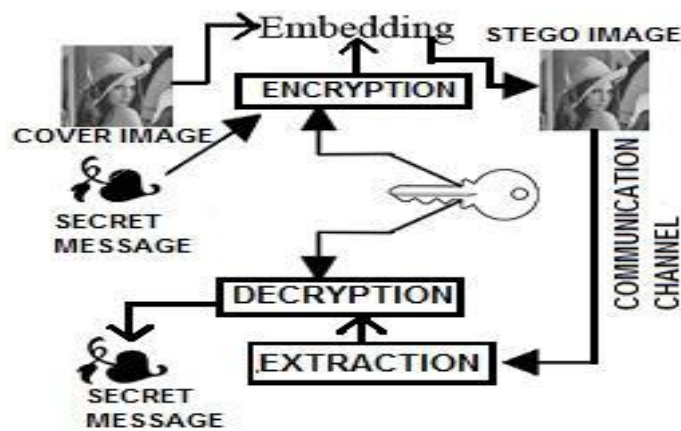


Figure 1.1: Block Diagram of Digital Image Steganography [Cha11]

1.3 Android operating system

The Android operating system was designed for mobile devices and tablets; it is one of the most widely used operating systems for smartphones today, the company that founded this O.S. in the 2003 called the Android Inc. Then in 2005 specifically, Google acquired this operating system and launched it in 2007. This O.S. based on the Linux kernel, has its own virtual machine and is used to execute its applications. The advantages of the Android operating system the continuous improvement on this operating system by Google Inc additional to the higher speed to access to the internet. The Android operating system consists of four layers as shown in Figure (1.2) [Nar16].

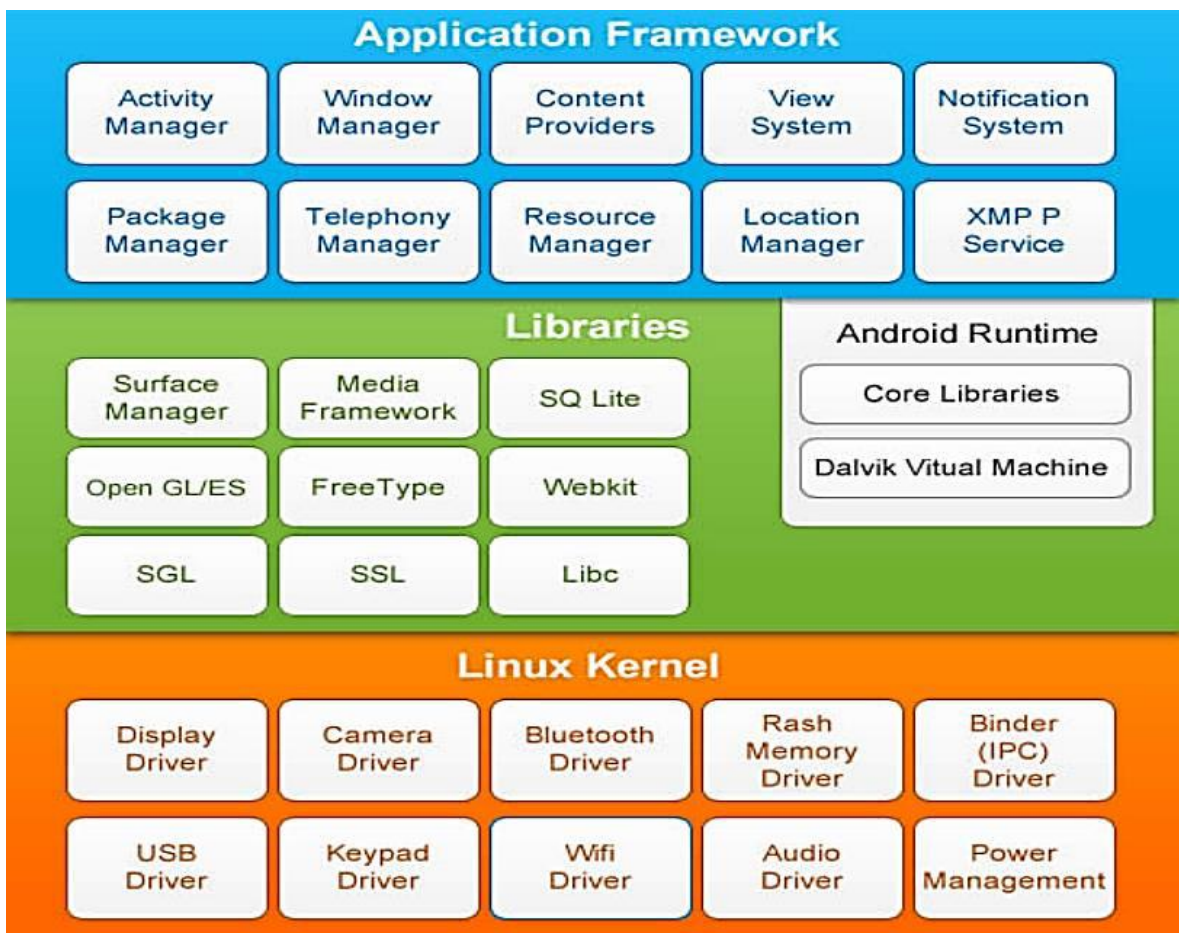


Figure 1.2: The Android layers [Nar16].

- 1- Linux Kernel: This layer does not provide the ability to interact with developers and users, it providing compatibility between the hardware component and upper layers.
- 2- Libraries: Set of libraries that written in the native of c/c++ language that support various components (SQLite, OpenGL, Wib Kit, etc.), additionally it is provide the Android Runtime, it is provide the Dalvik Virtual Machine (DVM), that used to execute its applications.
- 3- Application Framework: This layer provides services (Activity- Manager, Telephony-Manager, etc.) to developers and is of a higher level for applications in the form of Java.
- 4- Applications: it is provide the interaction between the device and the user.

1.4 Related Work

The most related work are presented in the following:

- **In 2012, A.Somdip Dey**, [Som12] were presented a different data-hiding algorithm, where a secret text is encrypted with combined cryptographic method (TTJSA) and then converting the encrypted data in a QR Code. In this method, only the encryption of text messages is shown when the QR code is scanned. The implication of QR Code adds an additional level of safety to the encrypted text and the recipient can access the original text very speedily, just by scanning the QR Code and deciphering it using software.
- **In 2013, Wu, Wen-Chuan**, [Wen13] Suggested easy and effective QR code steganography technique. It combines the advantage of edge detection and Vector Quantization (VQ) to camouflage the look of a QR code. First is to elicit the Region of Interest (ROI) area out of the original image. Next, an edge map is created in order to discover convoluted

image blocks. Only these blocks are concealed into bit stream of QR code by using VQ coding. As presented in the trials or test, the work results of the proposed scheme in both quality and efficiency pleasurable. Also results showed that the suggested method hides the QR code successfully. Moreover, the quality of stego-images is almost 30 dB.

- **In 2014, Muthaiah.RM & Krishnamoorthy. N,** [Mut14] introduced a new algorithm for hiding information backwards the images or any other digital media for security matter. The authors conceal the data behind any digital media, here behind an image and to have its existence discreet, they put the image with hidden data into a QR Code. Then cipher the QR code for more security. The ciphered QR is sent to receiver side. The receiver applies the opposite steps to get the secret message. All used Techniques are for increasing secrecy of transmitted data. This method could be used in the transmission of the big data overcoming the shortfalls of the digital media and the associated security techniques.
- **In 2014, Katharina k.,** [Kro14] in this research, the authors described the multiple use cases of QR codes. Moreover, they construed the most considerable attack scenarios with respect to the particular use cases. Finally, they suggest a design requirements with respect to the QR code itself, the reader software and usability features in order to support further research into making QR code handling both secure and usable.
- **In 2015, Akshara G. & K.R.Singh,** [Gai15] introduces the concept of QR codes, an automatic method to hide information using QR codes and to embed QR codes into color images with bounded probability of detection error. The embedding methods are designed to be compatible with standard decoding applications and can be applied to any color or gray scale image with full area coverage. The embedding method consists of two components. First is the use of halftoning techniques for

the selection of modified pixels to break and reduce the coarse square structure of the QR code and second is the luminance level to which the pixels are to be transformed in such a way that it should not be visible to the naked eye on the color image. Further to decode the QR code from the color image with minimum errors.

1.5 Aim of Thesis

This thesis aims to build an application to encrypt the image of the QR code, as well as to add another function that includes hiding the data in it. The work depends on the features of QR code and its characteristics to use them in a security manner. Each method in this work depends on a feature in QR code to satisfy security for it.

The motivation for using QR code images is its new and fast entrance to the digital communication technologies that are used in our daily lives, which necessitated its application in the environment of Smartphone operating systems such as Android.

1.6 Thesis Layout

Besides chapter one, the thesis contains four other chapters. Chapter one is an introduction to the fields of mobile application and data hiding, and introduces the aim of the present thesis. Other chapters are presented in the following:

1. Chapter Two: Entitled "*Theoretical Concepts* "

In this chapter, the theoretical basis of the QR codes and some specific subjects that related to the field of interest are mentioned briefly. This chapter contains all the significant terms mentioned in following chapters, and introduces the theories of the related concepts.

2. **Chapter Three:** Entitled "*Practical Work*"

In this chapter presents the design and implement of three stags which used QR-code in their work. Three new methods are presented with their algorithm steps.

3. **Chapter Four:** Entitled "*Results and Discussion*"

The implementation of the suggested systems is shown in this chapter. Then, the results are presented digitally and shapes. Moreover, brief discussion related to system behavior is documented to describe the performance of the proposed system.

4. **Chapter Five:** Entitled "*Conclusions and Suggestions for Future Work*"

This chapter contains just two subjects: the conclusions and future work. The conclusions are a numerously mentioned to summarize proposed system achievements, whereas the future work are the suggestions may be useful for anyone want to improve the proposed system in the future.

CHAPTER TWO

THEORETICAL CONCEPTS

2.1 Introduction

Due to the increased use of the Internet, which has become the main element of everyday life, and as a result to that the Internet has become an important tool in the scientific and business life and various fields. It appeared the urgent need for a means to ensure security of data stored and transmitted on the Internet. Therefore, the researchers focused on devising different ways to protect the data. This chapter is organized to present some issues relevant to thesis work. It covers the following: the basic concepts of security first; it includes the definition of security, the fundamentals of security and its fields (Cryptography, Steganography). Second, present the concept of the QR-codes (Quick Response codes), fundamentals, characteristics, types and technologies.

2.2 Data Security

The importance of information security has emerged in recent years, due to the spread of computing system in all aspects of life. Therefore, researchers focused in this area on how to keep this information from exposure to theft or loss or change. It has become the field of information security of the most important areas that are being studied and developed, and is defined " Is to provide protection for any automated system that specializes in managing and storing and providing information" [Wil15].

There are properties must be provided by information security to the information managed by the system, which includes [Kra09]:

- Confidentiality: Information is available only to authorized persons.
- Integrity: Unauthorized changes to the Information are rejecting.

- Availability: Information must be available all the time to people authorized to access them when needed.

In order for the system to achieve the above properties, some measures must be taken, these measures are classified as follows [Die11]:

- 1- Prevention: Measures taken to protect information from any damage.
- 2- Detection: Measures taken to detect the damage to information, how it was damaged and what is causing the damage.
- 3- Reaction: Measures taken to repair the damaged information or to restore it to pre-damage.

There are many techniques that are developed to achieve the data security; the most common techniques are cryptography and steganography [Kes03]:

A- Cryptography: which is the science of writing in secret code and is an ancient art. The first documented use of cryptography in writing dates back to Circa 1900 B.C. when an Egyptian scribe used non standard hieroglyphs in an inscription. Encryption is a method of transforming original data, called *plaintext* or *clear text*, into a form that appears to be random and unreadable, which is called *cipher text*, and the reverse process called decryption that produce the *plaintext* from *cipher text* [Kes03] figure(2.1) shows the simple cryptography diagram.

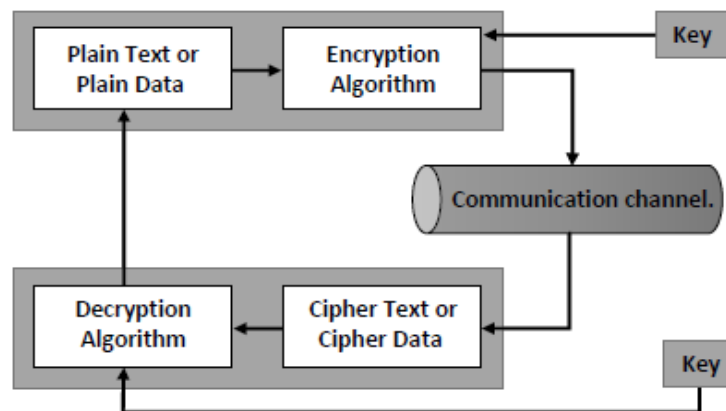


Figure 2.1: The cryptography diagram

B- Steganography: which can be defined as the science and art of hiding information in a way that prevents to be detected. Some familiar stenographic methods include the use of invisible ink or forming a message using the second letter of each word in a large body of text. In the twenty one century, steganography allows people to hide information within images or audio files. The process to hide information called embedding to produce the Stego Cover, and the reverse process called Extraction process to produce the *original information* [Kes15] as shown in figure (2.2).

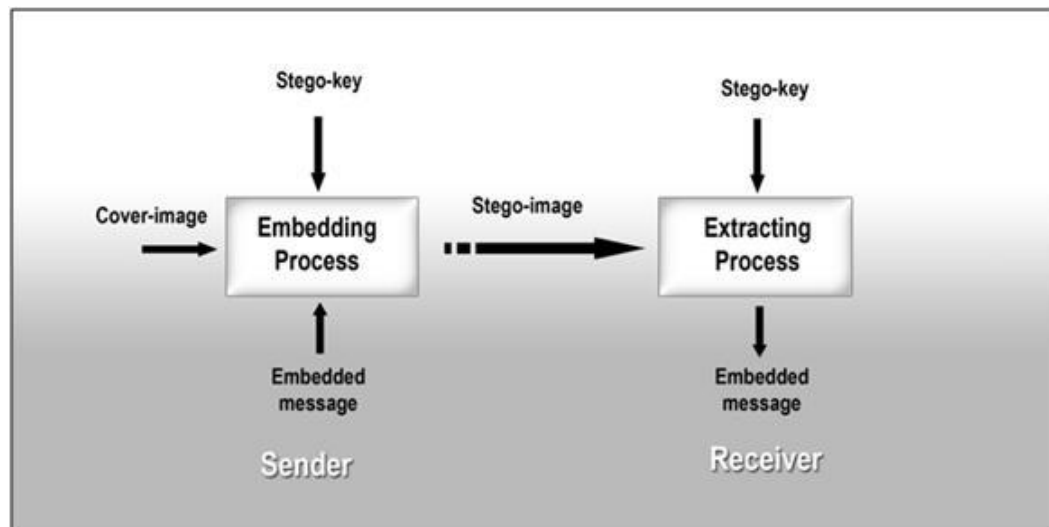


Figure 2.2: Image Steganography processes [Kes15].

2.3 QR Code

QR is the abbreviation of Quick Response, the aim of the QR is expressing the development notion of QR code, which means emphasizing on the high-speed reading. The essential information of QR Code is explored in the following sub sections, with defining QR Code and its characteristics. Furthermore, types of QR Code are described and discussed in this chapter as well [Dav14].

2.3.1 Fundamentals of QR Code

A QR Code is a two-dimensional barcode is scanned in two directions: the vertical and horizontal. QR Code enables storing more data than one dimensional barcode than can be illustrated. Implies; the QR Code requires a more sophisticated reader[Qji14].

QR Code is defined by the ISO/IEC 18004 industrial standard. Nevertheless, QR Code is created and protected by the Japanese company Denso Wave in 1994. The main objective of QR Code development is encoding and reading easily for user[*Sin16*].

Denso Wave announced QR Code is released to the world in 1994, whereas Denso Wave retains the patent right of the QR Code. In accordance with the intent of developers, from the beginning of QR Code development, it could be used by as many people as possible. Under this circumstance, QR Code can grow into “public code” used by individuals and enterprises without cost and do not worry about the potential problem [*Dav14*].

Today, QR Code obtains central commercial popularity thanks to mobile technology. In 2001, the penetration ratio of mobile devices is 15.6%, until 2010 it increases to 74.9% globally[*Cat13*].

QR Code can be read by almost all mobile phones in Japan. South Korea is similar to Japan, Europe is catching up, and the United State of America is comparatively new to the game. The iphone, Google’s Android operating system, and Windows Mobile phone, and Nokia’s phones all provide Internet access and camera, because they have the ability to scan and decode QR Code [*Qji14*].

The QR Code serves as a bridge which links the virtual and real world to digital domain is altering the method of marketing. Additionally, the QR Code also offers an opportunity to Interact with consumers and attract

consumer with their brand. Previous research and investigation statistics brand needs an appropriate implementation of marketing campaign to create consumers' awareness via QR Code. There are two cases of QR Code implementations are applied to demonstrates two frameworks. One of the frameworks is the marketing strategies in accordance with the layer of product or service involvement. Another one is the marketing communication system used [Cat13].

At present, QR Code is becoming an increasingly standard way when communicating with potential customers via print media in most countries. QR Code is being located on e.g. stickers, booths, business cards and advertisement vehicle. When an audience of tradeshow walks past a booth and the QR Code catches audience attention. And the audience use self smart-phone to scanning the QR Code, the QR Code will automatically link to the company's webpage. The audience will clear the detail information of the company[Wei10].

The reason why use QR Code is that QR Code is new and unique. In addition, QR Code can immediately connect people to virtual environment of information and entertainment. In addition, convenient and fast features of QR Code also attract people to use it. Besides, QR Code can send information to the mobile phone instantly, whatever someone's location [Qji14].

2.3.2 QR Code Characteristics

The characteristics of QR Code are described in the following. The QR Code features are help to understand the basic theory of it.

A- High capacity encoding of data

A barcode is one-dimensional, which means that scanners use only horizontal direction to scan the barcode, no matter what the height of barcode is. Since barcode is one-dimensional, the storage of information capacity is limited and the barcode can store less than 20 characters [Qji14].

A QR Code possesses a high capacity of storing information. While a traditional barcode is able to store highest of approximately 20 digits, a QR Code enables to store several information that is hundred times more than the capacity of traditional barcode stores information. The QR Code is capable of storing various types of data, e.g. numeric and alphabetic characters, the languages of (Korean, Japanese, and Chinese) (kanji, kana, hiragana), symbols, binary, and control codes. Additionally, QR Code can store maximum 7,089 characters in one symbol [Sha13].

Figure (2.3) demonstrates that the QR Code can store encoding numeric and alphabetic characters.

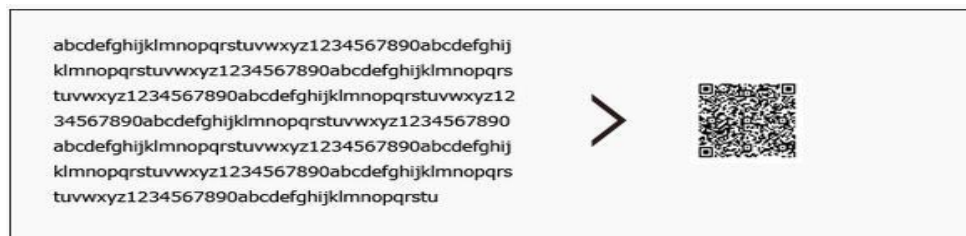


Figure2.3: An example of the QR Code encodes numeric and alphabetic characters [Qji14].

Figure (2.3) displays that the different alphabetic characters and numerics are encoded in a QR Code. Although the size of the symbol is small, it can store amount of information.

B- Small printout size

Compared with one dimensional barcode, a QR Code can hold information in both horizontal and vertical direction. When the number of

data is the same, the space of QR Code information storage only accounts for 25% space of the one dimensional barcode information storage [Qji14].

C- Kanji and kana capability

QR Code transfers the information into an image for the sake of saving the space. Due to QR Code designed in Japan that is the reason why QR Code is suitable to kanji and kana setting. QR Code encoding focuses on the Japanese industrial standards level1 and level 2 of kanji character set. In terms of kanji encode one kana or kanji character is efficiently encoded in 13 bit. Compared with other two dimensional code, QR Code can be stored more than 20% data [Qji14]. Figure (2.4) displays the kinds of kanji convert to QR Code, in which various kanji and kana can be transferred into data and they are stored in QR code symbol.



Figure 2.4: Kanji and Kana is converted to QR Code [Qji14].

D- Capacity of restoring and error correction of QR code

The capacity of QR Code relies on several factors, which contains the version of QR Code, the size of the version, the level of error correction and the categories of encoded data both impact the ability. The significant parts of QR Code are data part and error correction code-words. Data part combines several of segments using different encoding; each part has its unique mode. Moreover, the segment of data code-words and part of error

correction code-words are easy to identify except decoding QR Code when version and error correction level are given. Besides, the length of the information part is not based on the real length of the data. The length of the information is filled up with padding patterns to the whole length [Qji14].

QR Code has the error correction capability; therefore, QR Code can store the code-words of maximum 30% when the image is dirty and damaged. When the image of QR Code is contaminated, the error detecting can focus on the place of correct information. Data can be recovered even though a part of the code is dirty and damaged in general situation. Nevertheless, in some situation of the image is dirty and/or damaged, data may not be restored [Sha13].

Figure (2.5) illustrates that the Dirt and Damage of QR Code. In this work, this problem is exceeded by using digital QR code.

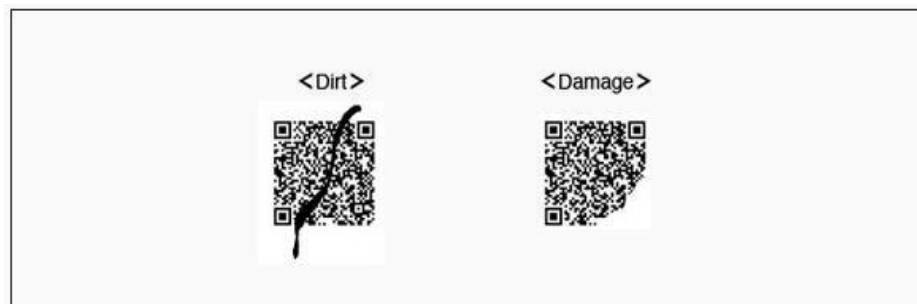


Figure 2.5: Dirt and damage of QR Code [Qji14].

The QR Code can be corrected based on these situations of dirty QR Code and damaged QR Code. Such that, Data can be easily restored even if a segment of the code due to whatever reason lead to the code unreadable. There are four different levels of error detection that can be chosen for the incomplete QR Code corrected. In addition, the elements of considering an error correction level selection contain the size of QR Code, the performing situation, the real-estate it will have and the degree to what kind of

environment can be controlled. The four levels of Reed-Solomon error correction of QR Code refers to L, M, Q and H in increasing order of capacity as follows:

- 1- Level L is approximately 7% or less error enables corrected.
- 2- Level M is approximately 15% or less error enables corrected.
- 3- Level Q is approximately 25% or less error enables corrected.
- 4- Level H is approximately 30% or less error enables corrected.

In accordance with the level of error correction, the capacity of Level L is the weakness one, the capacity of level H is the stronger one [Qji14].

E- Readable from any direction in 360 degrees

QR Code has a characteristic that it can be read in 360-degree direction. Nonetheless, the traditional one-dimensional barcode recognizes the information only plus-minus ten degrees which is relatively smaller than QR Code. QR Code not only can be read in 360-degree direction, but QR Code also can be read in high speed. The secret of QR Code reading direction in 360 degrees is that position the detection patterns located at the three corners of the symbol can locate the QR Code. Therefore, QR Code can read quickly and circumvent the effects of background interference [Sha13].

F- Structured appending Feature

QR Code is capable of classifying a variety of data areas. On the contrary, a lot of information is stored in various QR Code symbols can compose a QR Code symbol. One data symbol allows dividing into maximum 16 symbols that is providing convenience to print. Figure (2.6) reveals that the structured appending features of QR Code and shows that a QR Code can be divided into more than one QR Code and all the QR Codes also can be stored in one QR Code [Qji14].

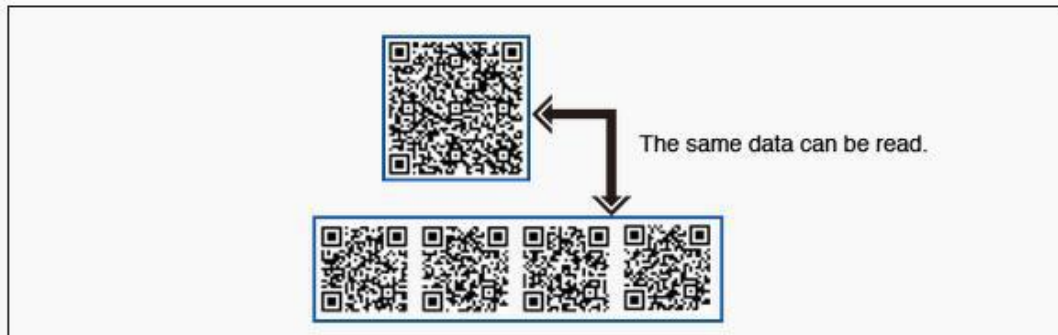












Figure 2.6: Structured appending of QR Code [Qji14].

2.3.3 Types of QR Code

Different types of QR Code are found, Table (2.1) displays five types of them. In agreement with table (2.1), there are six types QR Code e.g. Model one, Model two, Micro QR Code, IQR Code, SQRC and LogoQ.

In this thesis, Model one and Model two QR codes are used in the work.

Table 2.1: Types of QR code

 QR Code Model 1 and Model 2	 Micro QR Code	 iQR Code	 SQRC	 LogoQ
				
<p>[Feature] Model 1 is the original QR Code. The largest version of this code is 14 (73 x 73 modules), which is capable of storing up to 1,167 numerals. Model 2 is an improvement on Model 1 with the largest version being 40 (177 x 177 modules), which is capable of storing up to 7,089 numerals. Today, the term QR Code usually refers to this type.</p>	<p>[Feature] Only one orientation detecting pattern is required for this code, making it possible to print it in a smaller space than before. This code can be viable even if the width of its margin is 2 module-worth (QR Code requires a margin of 4 module-worth at least around it). The largest version of this code is M4 (17 x 17 modules), which can store up to 35 numerals.</p>	<p>[Feature] Code that can be generated with either square modules or rectangular ones. Can be printed as a turned-over code, black-and-white inversion code or dot pattern code (direct part marking). The maximum version can theoretically be 61 (422 x 422 modules), which can store about 40,000 numerals</p>	<p>[Feature] QR Code that has a reading restricting function. Can be used to store private information or manage a company's internal information) Its appearance is no different from the regular QR Code.</p>	<p>[Feature] QR Code that can incorporate high-levels of design features such as illustrations, letters and logos. Since proprietary logic is used in generating this type of code, its readability is not compromised.</p>

A. QR Code Model one and Model two

Model one is the primary QR Code and it enables encoding 1,167 numerals and its highest version being 14. Model two is the edition of

Model one promotion, thus Model two can be read smoothly even though it is distorted in some way. Model two can store more than 7,089 numerals with its maximum version being 40 [Qji14].

B. Micro QR Code

A traditional QR Code has three finder patterns which are placed on the three corners of the QR Code image. Compared with the traditional QR Code, Micro QR Code has merely one finder pattern for positioning. On the other hand, a normal QR Code needs no less than four-module wide margin within a symbol. Nevertheless, Micro QR Code only requires a two-module wide margin. Under this circumstance, Micro QR Code permits printing in areas smaller than QR Code. Figure (2.7) illustrates the difference between QR Code and Micro QR Code [Qji14].

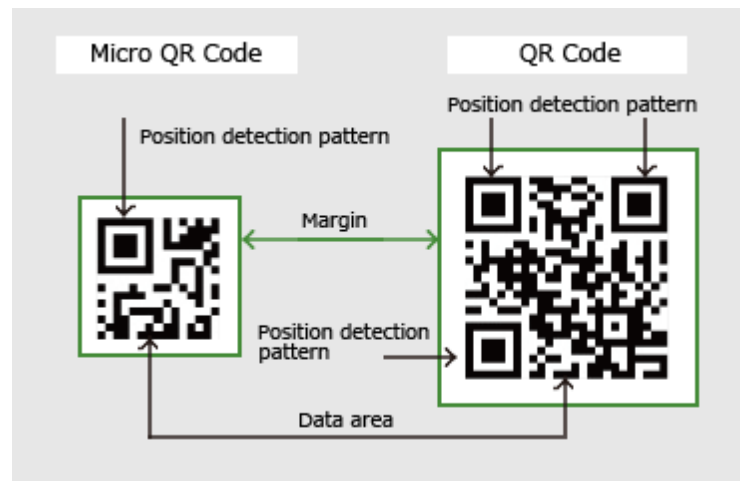


Figure 2.7: The symbol of Micro QR Code and QR Code [Qji14].

Figure (2.7) shows that the Micro QR Code feature and the QR Code feature in details. In accordance figure (2.7), the Micro QR Code has a finder patterns, whereas QR Code has three finder patterns. Moreover, the Micro QR Code's wide margin is smaller than QR code. Furthermore, for the capacity of data storage and the size of code, the data can be stored by

Micro QR Code in less than 35 numerals data. Micro QR Code not only enables to encode data more efficiently than the regular QR Code, but the size of Micro QR Code also does not need to be made much larger when the numbers of data stored rising. In addition, the standardization of Micro QR Code is made publicly available similarly to QR Code [Qji14].

C. IQR Code

IQR Code is a matrix-type two dimensional barcode and its position and size is read easily. Using IQR Code can be generated more extensive two dimensional barcode. The new two dimensional barcode can be smaller than the normal QR Code and Micro QR Code. Moreover, the new two dimensional barcode also can be a large size two dimensional barcode. Furthermore, IQR Code is able to printout as a rectangular code, and it supports for turned-over code, black-and-white inversion code and dot pattern code. IQR Code permits a wide range of applications in several fields. Because IQR Code can be generated as rectangular modules, IQR Code enables replace the one dimensional barcode. IQR Code can maintain the code's readability while it printed on cylindrical products, even though square modules are difficult to print on cylindrical. Figure (2.8) shows that the sample of code with rectangular modules [Qji14].

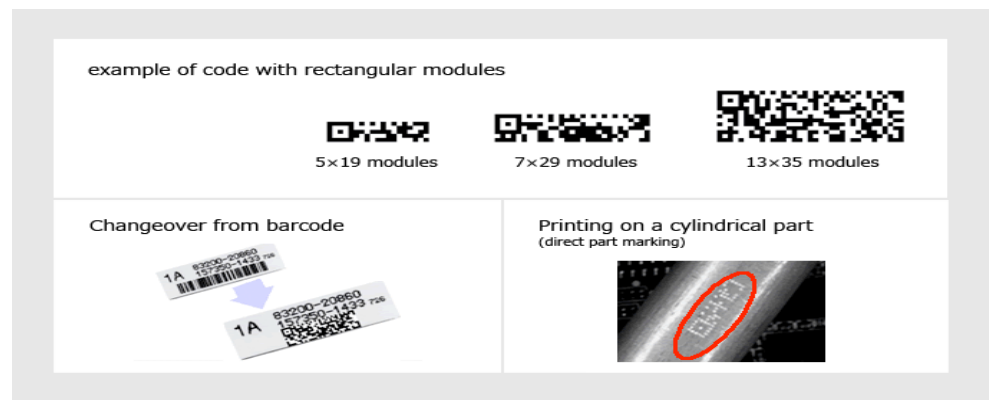


Figure 2.8: The rectangular modules of IQR Code [Qji14].

In agreement with figure (2.8) IQR Code has different size of version. Additionally, IQR Code can instead one dimensional code to print out on product. IQR Code can store more information than the ordinary QR Code. If the size of symbol is same, compared with the ordinary QR Code, the IQR Code capacity of storing information increases to 80% regular QR Code. If the same amount is stored, an IQR Code can be made 30% smaller than the regular QR Code. Figure (2.9) demonstrates the situation of same size and same amount within the regular QR Code and IQR Code [Qji14].



Figure 2.9: Comparison of regular QR Code and IQR Code [Qji14].

Figure (2.9) reveals that the size of IQR Code reduces when IQR Code has same amount data as traditional QR Code. Moreover, Figure (2.9) displays IQR Code possesses high data capacity. When the characters are all numerals, the highest version of QR Code can store 7,000 characters. By contrast, the number of characters that IQR Code can hold in its biggest version is approximately 40,000. Besides, IQR Code has high restoration capability which is higher than traditional QR Code. The QR Code error correction highest level recovered no more than 30% of the error in a QR code. However, compared with the QR Code, the error correction level of IQR Code is improved to 50% [Qji14].

D. SQRC

SQRC (Secure Quick Response Code) is a particular QR Code and it is embedded into reading restricting function. The SQRC concentrates on private data storing and internal data of enterprise managing, nevertheless, this function does not ensure securing of coded data. The aspects and properties of SQRC are similar to the traditional QR Code. In addition, SQRC can be locking up of encode data, merely specific scanners can read it. Besides, data for SQRC includes public segment and private segment, different layer of information can be stored in one SQRC [Qji14].

E. Logo Q

A new style of QR Code is LogoQ which combines a QR Code with a picture. LogoQ is designed for the sake of boosting the recognizable ability of vision. Figure (2.10) reveals the sample of LogoQ.

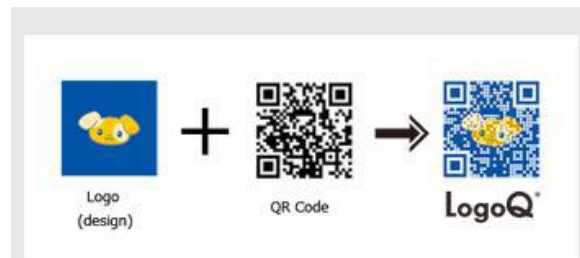


Figure 2.10:An example of LogoQ [Qji14].

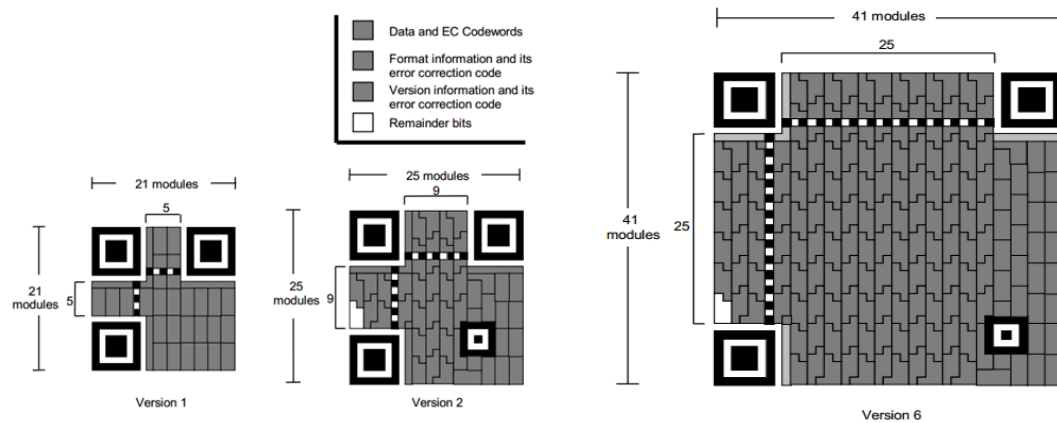
Figure (2.10) displays that colorful combination guides people easy to understand the code base on personal intuition. Because of LogoQ is used an exclusive logic in generating, it possesses design ability and readability. What is more, since LogoQ has highly designable feature and it is different from the ordinary QR Code [Qji14].

2.3.4 QR Code Technologies

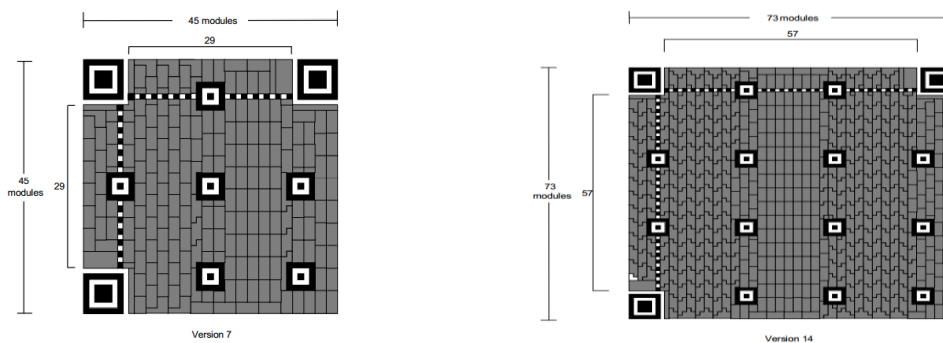
The symbol of QR Code is described and explored to guide the people to understand the QR Code structure clearly. As well as encoding and decoding procedures are analyzed in this section. Additionally, the guidebooks of implementation of QR Code generating process and reading process are listed in following sections:

A. QR Code symbols

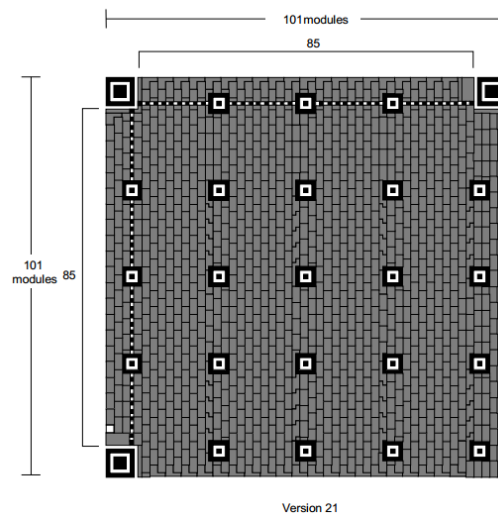
QR Code has forty sizes symbol e.g. version 1, version 2 and version 40. When the version increases one, the side of version plus 4 modules, such as a side of version 1 is 21 modules, a side of version 2 is 25 modules and the side of version 40 is 177 modules. Figure (2.11) illustrates the structure of versions 1 and version 2 [Qji14].



(a)



(b)



(c)

Figure 2.11 :Version of QR code symbols.

(a): Version 1, version 2 and version 6 symbol, (b): Version 7 and version 14 symbol and (c): Version 21 symbol [Qji14].

In accordance with the Figure (2.11-a), the area of version 1 is 21 modules x 21 modules, and the area of version 2 is 25 modules x 25 modules, and the area of version 6 is 41 modules x 41 modules, and the distance between two finder patterns is 25 in version 6. Figure (2.11-b) demonstrates the structure of Version 7 is 45 modules x 45 modules, and the distance between two finder patterns is 29 in version 7, and the structure of version 14 is 73 modules x 73 modules, and the distance between two finder patterns is 57 in version 14. In accordance with figure (2.11-c), the area of Version 21 is 101 modules x 101 modules, and the distance between two finder patterns is 85 in version 21. Concerning the structure of QR Code, each QR Code symbol is constructed by square. The regular square consists of an encoding region and function patterns. The function patterns focus on the positioning and the encoding region concentrates on data encoding [Qji14].

B. QR Code structure

Figure (2.12) demonstrates the structure of QR Code, which is divided into two segments. In terms of function patterns which are composed by finder patterns, separators, timing patterns and alignment patterns. The Finder patterns are three common structures that are located in QR Code's three corners. Finder pattern is used for positioning the symbol, recognizing the symbol and deciding the correct orientation. Separators surround the finder pattern that can promote identification of the finder pattern. Timing patterns enable the decoder software to judge the side of module. Alignments patterns sustain decoder software in correcting for reducing the image distortion. Version one QR Code does not have alignment pattern. With the size of the version increasing, alignment pattern is added at the same time. For encoding region, format information appears in all sizes of version that used to store formatted data and select masking pattern. Data is transferred into a bit stream and stored in 8 bit parts in data section. And error correction codes are stored in error correction section [Qji14].

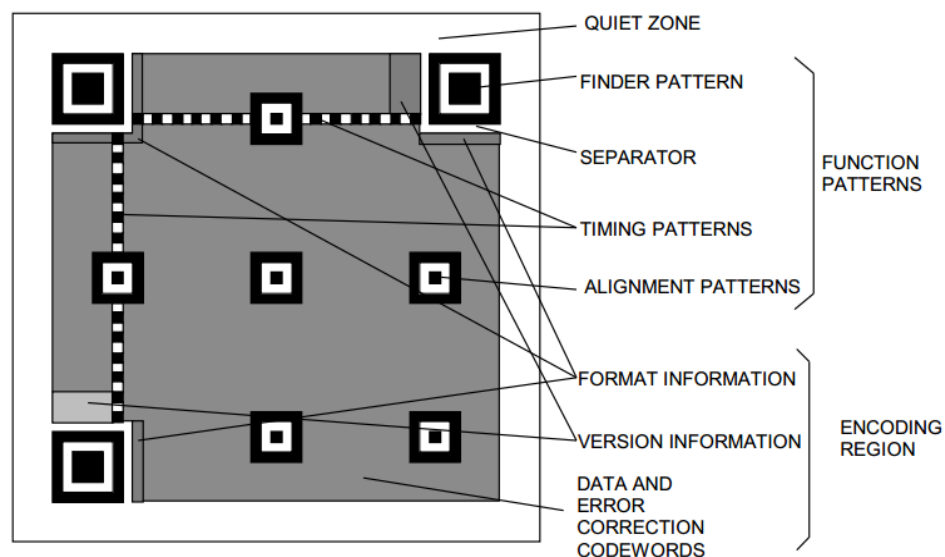


Figure 2.12: Structure of QR Code [Qji14].

C. Encoding procedure overview

The encoding procedure of QR Code plays an important role in QR Code study. “The QR Code has highly recognition rate and decryption the data in a short time” In the following, the encoding procedure overview of QR Code is displayed [Sha13]:

- Step 1 is data analysis, which aims at analyzing the input data stream to recognize the plenty of characters for data encoded. The Extended Channel Interpretation feature is offered by QR Code format except for Micro QR Code format that is capable to encode different types of data. QR Code has variety of modes for transfer the characters into a symbol in efficiently. The modes includes numeric mode, alphanumeric mode, byte mode, kanji mode, extended channel interpretation mode, structured append mode and Fnc1 mode. Modes switch during transferring characters as needed in order to convert data into a binary string rapidly. If the user does not require specific symbol version, the smallest version can be the best choice which accommodate the data. Table (B.1) in appendix B provides completed information concerning symbol versions and capacities of QR Code [Sha13].
- Step 2 is data encoding which aims at transferring input data. Data transformation through matching modes into a bit stream on the basis of the rules for the mode respectively and the bit stream sequence is composed of one or more modes.
- Step 3 is error correction coding. In appliance with the version of image and the layer of error correction, dividing the data sequence into plenty of blocks for the sake of apply error correction coding. After the error correction code-words of every block are generated, the code-words are added at the end of the data sequence.

- Step 4 is arranging the data and error correction code-words from different block.
- Step 5 is placing codeword modules in the matrix with the finder patterns, separators, timing patterns, and alignment patterns.
- Step 6 is data masking. The data masking patterns in the encoding region of the symbol to optimize the dark and light module balance and minimize the wrong patterns appearance.
- Step 7 is relevant to generate format information and version information and complete the symbol.

D. Decoding procedure overview

Visual appearance of QR Code is different from the one dimensional barcode. The QR Code has nubby patterns, high speed, two dimensional graphic images and the QR Code can be read immediately by scanners and smart-phones which have QR reader application. As a result, the QR code decoding maybe relevant to the blocky patterns, in this section, the decoding procedure is introduced. Encoding procedure is opposite to the decoding steps which are reading a QR Code symbol to outputting data characters. Figure (2.13) shows a flow chart of the decode process of QR Code [Qji14].

Figure (2.13) displays that the reader focus on positioning the image of QR Code when the reading process is starting.

- Step 1 reader recognizes the three finder patterns and identifies white and dark blocky.
- Step 2 is the format information is decoded. In this stage, the masking patterns are released and error correction is operated on the format information part.

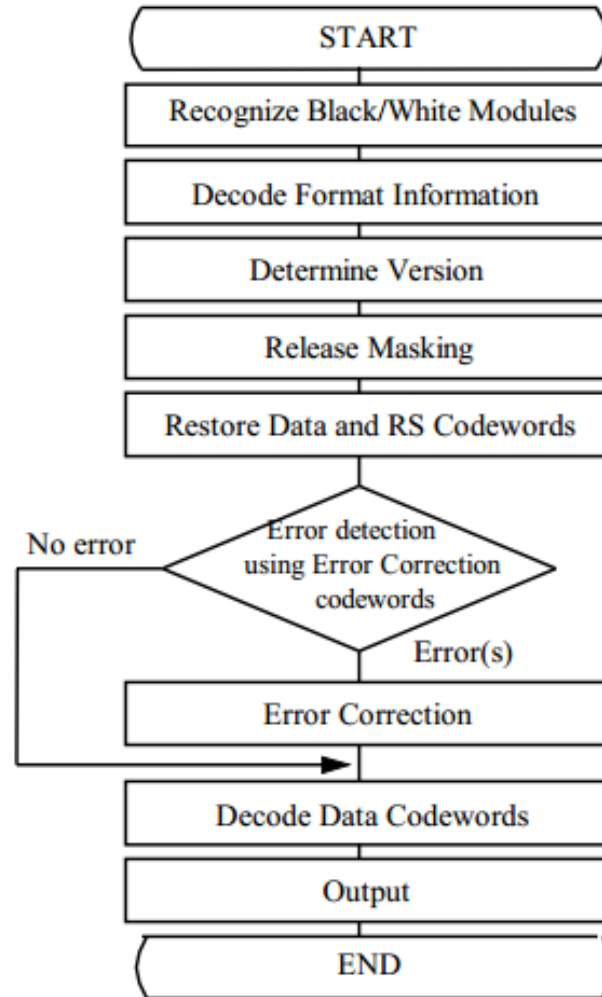


Figure 2.13: QR Code decoding process[Qji14].

- Step 3 Symbol is in general guiding when successful, if not, the mirror image decoding of format information is attempted, the error correction is used for helping to decode.
- Step 4 is determining the version of QR Code. In this step, version information is read and the version of the QR Code is verified. Later, the data masking is released.
- Step 5 & Step 6 are reading the characters, detecting the error and restoring data. These steps utilize the error correction codeword to correct

the error. The error will be amended when any error detected. The seventh step is classifying the data code-words into two parts in the light of the indicators of mode and indicator of character count. Finally, decode the data character base on one or more mode and result in the original data [Qji14].

2.4 Quality Measures

In this section, the evaluation method that used to assess the difference between two images, first image represent the original QR code and second image represent the resulted QR code from the processes on the original image:

- Mean Squared Error (MSE)

In statistics, the Mean Squared Error (MSE) or Mean Squared Deviation (MSD) of an estimator (of a procedure for estimating an unobserved quantity) measures the average of the squares of the errors or deviations that is, the difference between the estimator and what is estimated. MSE is a risk function, corresponding to the expected value of the squared error loss or quadratic loss. The difference occurs because of randomness or because the estimator doesn't account for information that could produce a more accurate estimate" [Tsa14]. "The MSE is a measure of the quality of an estimator it is always non-negative, and values closer to zero are better. The MSE is the second moment (about the origin) of the error, and thus incorporates both the variance of the estimator and its bias. For an unbiased estimator, the MSE is the variance of the estimator. Like the variance, MSE has the same units of measurement as the square of the quantity being estimated. In an analogy to standard deviation, taking the square root of MSE yields the Root-Mean-Square Error or Root-Mean-Square Deviation (RMSE or RMSD), which has

the same units as the quantity being estimated; for an unbiased estimator, the RMSE is the square root of the variance, known as the standard deviation”.

$$MSE = \frac{1}{H * W} \sum_{x=1}^W \sum_{y=1}^H (Im_{original}(x,y) - Im_{affected}(x,y))^2 \quad (2.1)$$

- Peak Signal-to-Noise Ratio (PSNR)

Peak Signal-to-Noise Ratio, often abbreviated PSNR, is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed in terms of the logarithmic decibel scale. PSNR is most commonly used to measure the quality of reconstruction of lossy compression codecs (e.g., for image compression). The signal in this case is the original data, and the noise is the error introduced by compression. When comparing compression codecs, PSNR is an approximation to human perception of reconstruction quality. Although a higher PSNR generally indicates that the reconstruction is of higher quality, in some cases it may not. One has to be extremely careful with the range of validity of this metric; it is only conclusively valid when it is used to compare results from the same codec (or codec type) and same content [Huy08]. PSNR is most easily defined via the mean squared error (MSE). Given a noise-free $m \times n$ monochrome image I and its noisy approximation K . MSE is defined as: PSNR is an approximation to human perception of reconstruction quality. Although a higher PSNR generally indicates that the reconstruction is of higher quality, in some cases it may not. One has to be extremely careful with the range of validity of this metric; it is only conclusively valid when it is used to compare results from the same codec (or codec type) and same content” [Huy08]. “PSNR is most easily

defined via the mean squared error (MSE). Given a noise-free $m \times n$ monochrome image I and its noisy approximation K . MSE is defined as”:

The PSNR (in dB) is defined as:

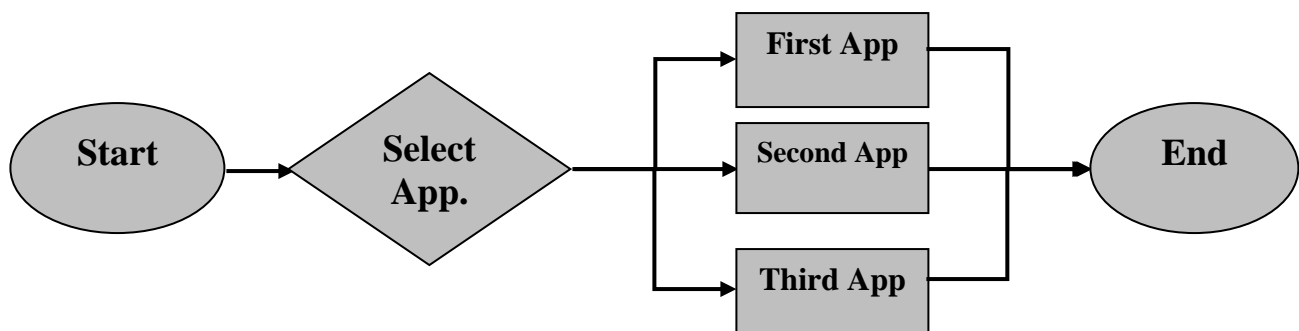
$$PSNR = 10 \log_{10} \left(\frac{\left(\max_{x,y} I_{original}(x,y) - \min_{x,y} I_{affected}(x,y) \right)^2}{MSE} \right) \quad (2.2)$$

CHAPTER THREE

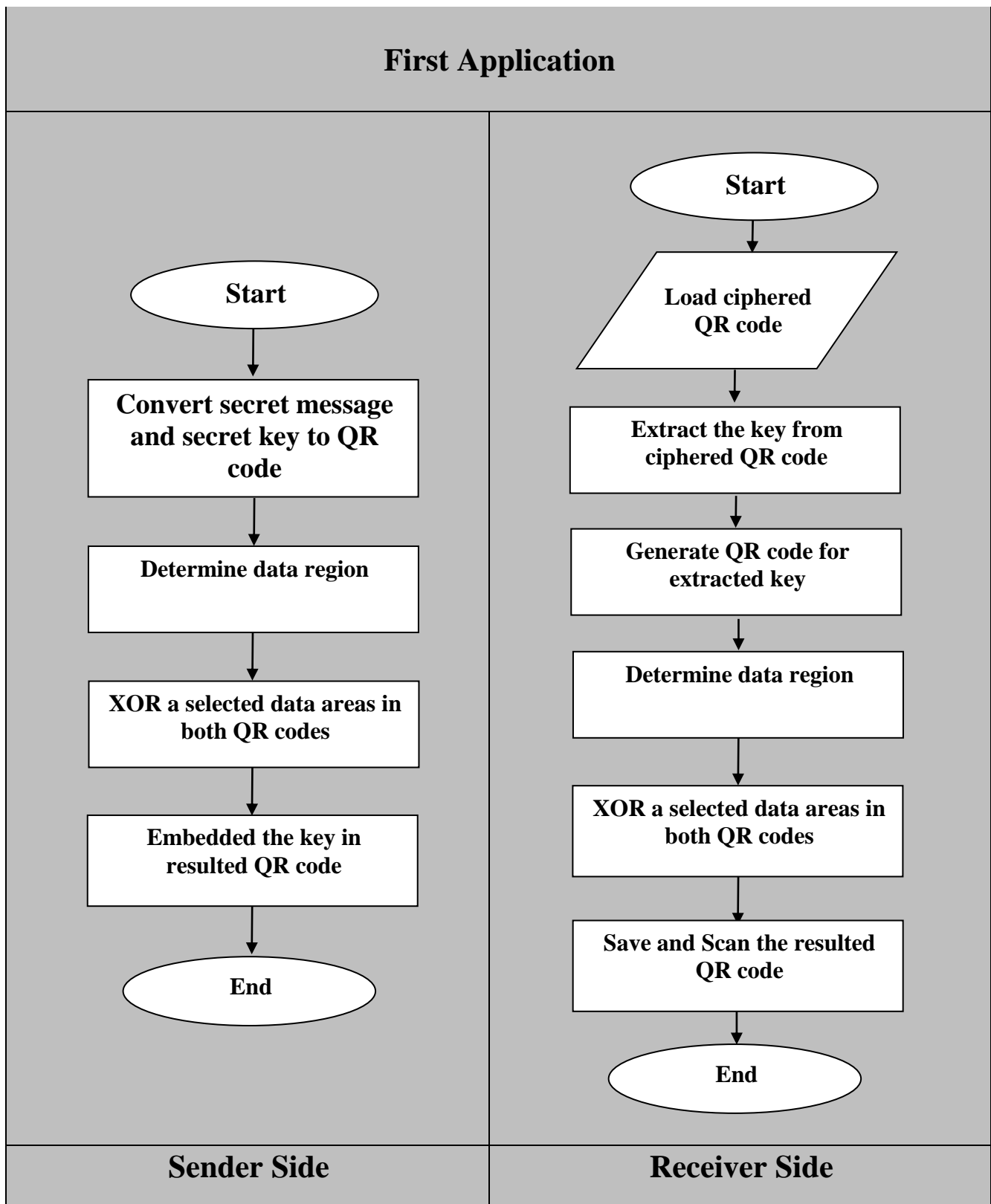
PRACTICAL WORK

3.1 Introduction

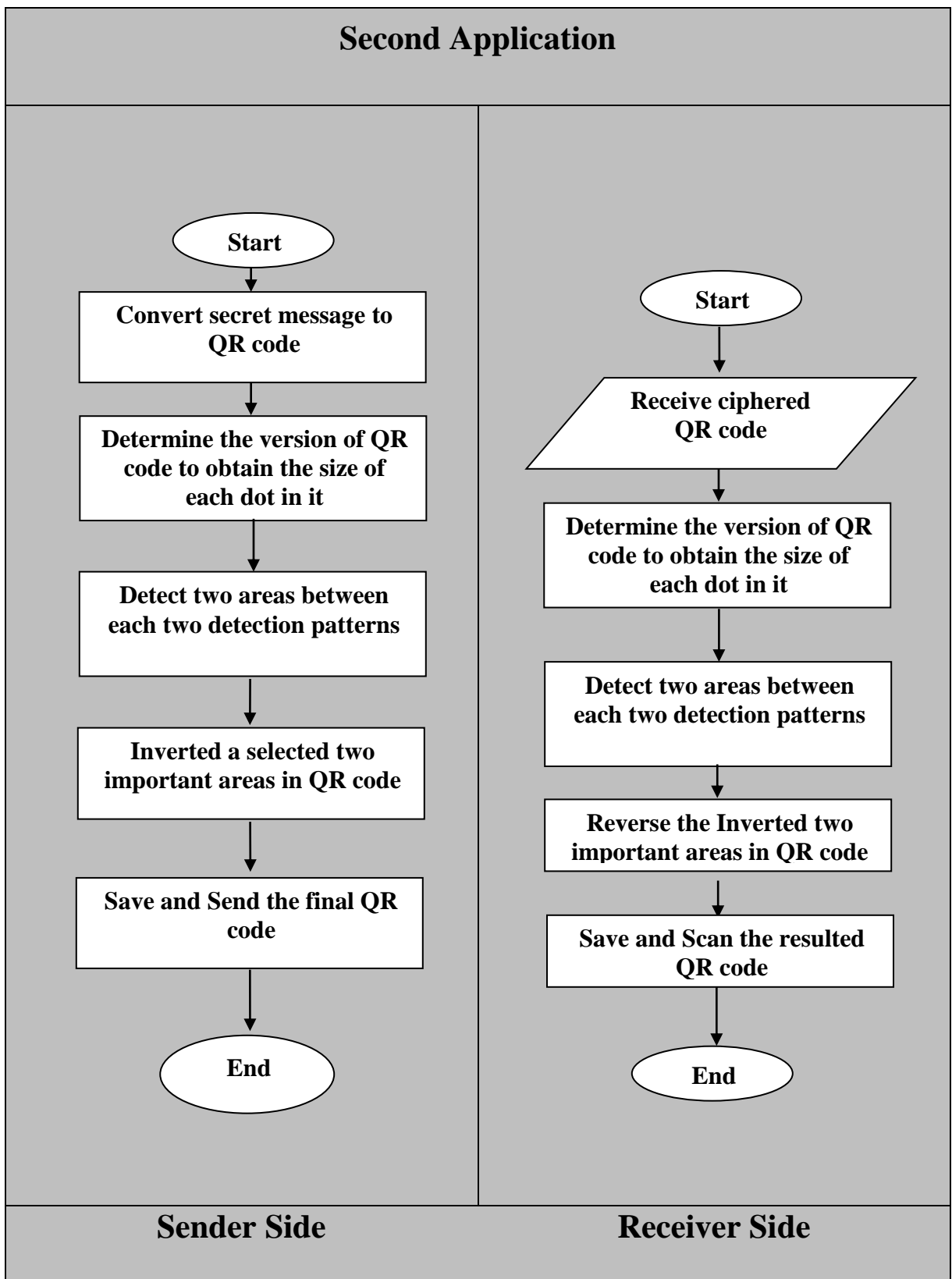
In this chapter, the design of information security based on QR codes applications will be presented. Three applications are proposed to adapt three new methods for security by using QR codes technique. Each of the proposed applications (**Ciphering Special Versions of QR code with XOR Application, Ciphering General Versions of QR code Application and Hiding Information Using QR code Application**) include a method for hiding and/or encrypting the QR's data. The First method takes two QR codes. One of them represents the secret message and the other represents a secret key; this method will be described in details in section (3.2). The second method takes one QR code which represents the secret message as a QR. Then encrypts it to a secret QR code. This method will be described in details in section (3.3). Finally, the third method takes two QR codes; one of them represents the secret message and the other represents the cover for the first QR code, and then hides the first one into the second. This method will be described in details in section (3.4). Figure (3.1) shows the block diagram of each application.



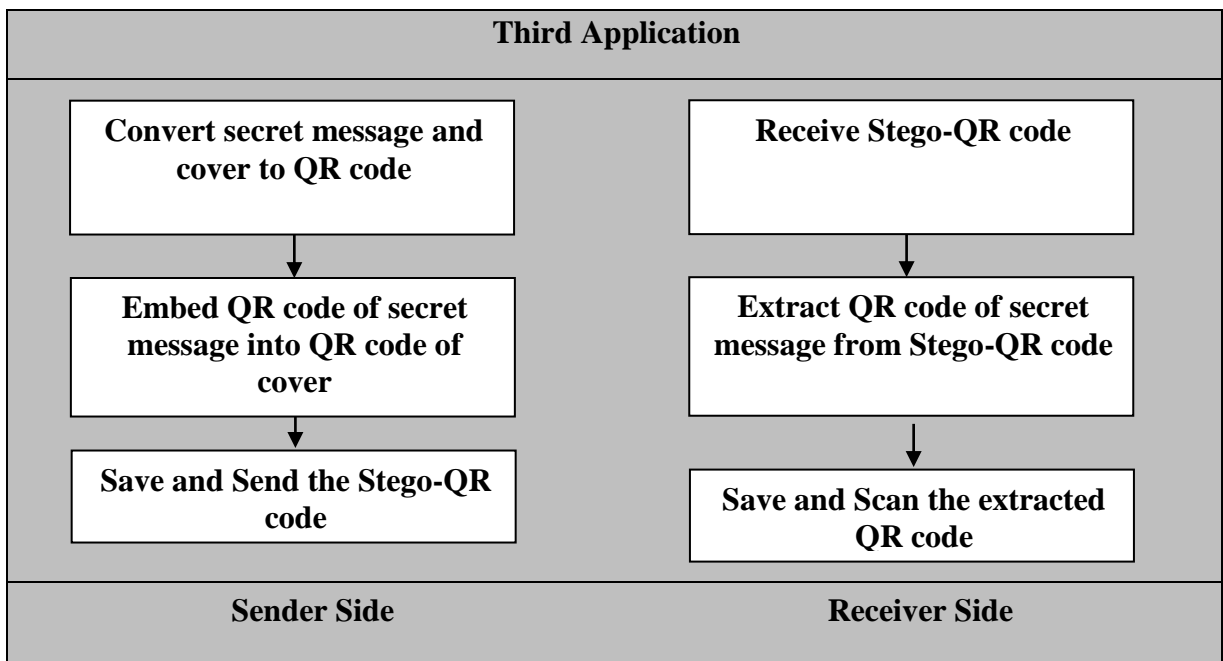
(A)



(B)



(C)



(D)

Figure (3.1): (A), (B), (C) and (D) Block diagram of three applications.

3.2 Cipherring Special Versions of QR code with XOR Application

In this method, a new data-cipherring algorithm is presented, where two QR_code images are used. The first image represents the converted text of the secret message to QR_code, and the second image represents the converted text of the secret key to QR_code. Since the QR_code image is a black/white image, so each dot in the image acts as 0 or 255 bitwise values that provides series or string of 0s and 1s.

To guarantee an exact retrieving to the original message from the encrypted QR, one of the reversible logic operators is used. The selected one is the (XOR) operator which provides this feature to obtain from cipher the original plain information, more details are explained in the following subsection:

3.2.1 Cipherring Algorithm

This method depends on XOR logic operator. Two QR codes are used, to obtain one QR as a result. The two QRs are XORing in a special selected area to generate one ciphered QR code as an encryption process as shown in figure (3.2). And guarantee that the QR standard shape will not be changed. Then the key as a word is embedded in the resulted QR. The embed process is done by change each character in the string of key to corresponding ASSCI code and then define a string that is filled by series of 0s and 1s by concatenation the binary representation of the ASSCI code, the final string was embedded in resulted QR code as shown the table (3.1).

Table (3.1): embed key process in sender side example

Digit	Pixel color digitally	Result	Color in Resulted QR code image
1	255	254	White color
1	0	1	Black color
0	255	255	White color
0	0	0	Black color

In receiver side, the process is reversed. After extracting the key from encrypted QR (cipher) in decryption process; where the white module (dot) in QR represents (255) and the black module (dot) in QR represents (0) when XORed. The two QRs produced one QR image as shown in figure (3.3).



Figure 3.2: Example of encryption process



Figure 3.3: Example of decryption process

3.2.2 Sender Side Algorithm

Two QR codes are generated, one for the secret message and the other for the secret Key. Then, the two QRs are XORed in a specific part to encrypt the message. Then the key is embedded inside the resulted QR Code. Figure (3.4) summarize the encryption process.

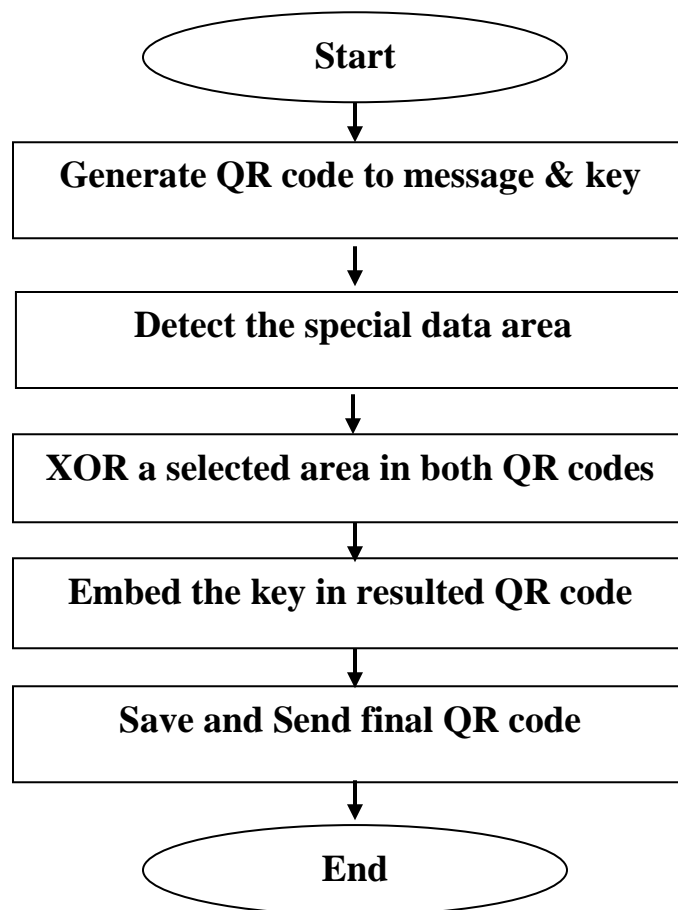




Figure 3.4: Flowchart of encryption process

- **Algorithm (3.1): Generating QR code for message and key**

Input:	The text message or key.
Output:	The corresponding QR code image for text message or key.
Steps:	<p>Step1: Input (read) message/key (text).</p> <p>Step2: generate QR code for the message and key.</p> <p>Step3: save QR image as P/k.</p> <p>Step4: end.</p> <p>Where P as plaintext, k as key.</p>

- **Algorithm (3.2): Detecting the data area indices**

Goal:	The start indices of the selected area that will be XORed.	
Input:	The QR code image.	
Output:	The indices of the data area in QR code image.	
Steps:	<p>Step1: Loop statement i=0 to width Loop statement j=0 to height</p> <p>Step2: If pixel(i,j)= black then Break; Else Continue; Next j,i;</p> <p>Step3: Loop statement k=I to Wd If pixel(k,j) = white then Break; Else Continue; Next k;</p> <p>Step4: //Calculate the size of the dot X=(k-i)-1; S=x/7; //size of dot in QR code image the number seven is</p>	

Step5:	the size black rectangle of detection pattern; //get the indices of the selected area that will be XOR
Step6:	Start_s=wd/2; start_y=Hg/2; end;

• **Algorithm (3.3) Encryption process**

Input:	The QR code image of secret message and QR code Image of secret key.	
Output:	The ciphered QR code image.	
Steps:	<p>Step1: load QR image as P. Step2: load QR image as k. Step3: define cipher as bitmap file with dimensions width (wd) & height (hg). Step4: Call sub routine (Algo. 3.2) detect indices of data area. Step5: Copy P image in cipher image except the data region. Step6: loop statement x=(start_x) to no. of blocks } Begin of data area loop statement y=(start_y) to no. of blocks } area cipher(x)(y)=P(x)(y) XOR k(x)(y) next y,x. Step7: end.</p>	

• **Algorithm (3.4): Embedding Key in resulted QR**

Input:	The text of secret key, cipher QR code.
Output:	Cipher QR code image which contains the embedded text of secret key.
Steps:	<p>Step1: binarization each character or number in key as 8bit. Step2: start the embedment process from the dimension (0,0) in resulted QR code image; Step3: loop statement i<key (length) If (key(i)=1) Key(i)=254; Else</p>


```

        Key(i)=1;
    End if.
Next i
Step4: if statement (key(i)=255)
        Key(i)=253;
Else
        Key(i)=2;
End if.
Step5: end.

```

3.2.3 Receiver Side Algorithm

Look to figure (3.5) that summarize the decryption process.

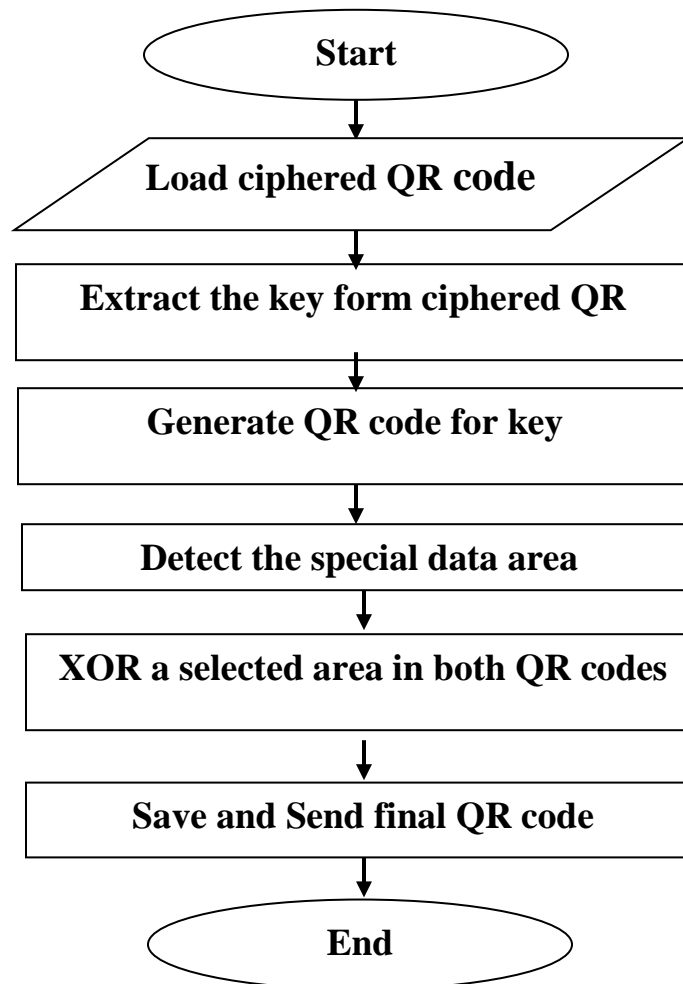


Figure 3.5: Flowchart of decryption process

Since the used operator (XOR) is reversible then, when destination receives the encrypted QR code, one extracts the key from it and generates QR code for this key and then XORing with encrypted QR code to obtain the QR code for secret message.

• **Algorithm (3.5): Extract the embedded key**

Input:	The ciphered QR code image with the text of secret key embed in it.
Output:	The characters of secret key (K).
Steps:	<pre> Step1: loop statement until key(l)=253 or 2 If (key(l)=254 or 1) Str=concat(str,'1'); Else Str=concat(str,'0'); End if Next i Step2: collect each 8bit in str and get character of this collection. Step3: end. </pre>

• **Algorithm (3.6): Decryption process**

Input:	The ciphered QR code image (C) and QR code image of the secret key (K).
Output:	The pure QR code image of the secret message (P).
Steps:	<pre> Step1: start. Step2: load QR image C. Step3: get width (wd) & height (hg) of C. Step4: define plain as bitmap file with dimensions width (wd) & </pre>

height (hg).

Step5: Call sub routine (Algo. 3.2) to detect indices of data area.

Step6: Copy cipher image in plain image except the data region.

Step7: loop statement $x=(start_x)$ to no. of blocks }
loop statement $y=(start_y)$ to no. of block } Begin of data
area

$plain(x)(y)=cipher(x)(y) \text{ XOR } key(x)(y)$

next y,x.

Step8: end.

3.3 Cipherying General Versions of QR code Application

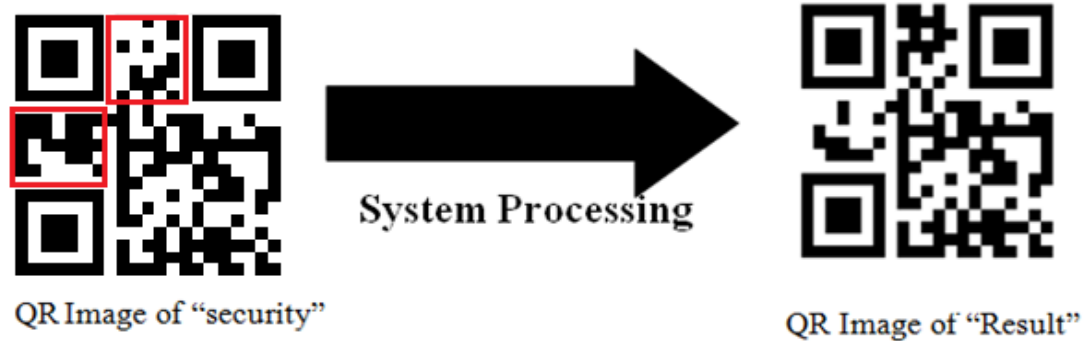
In this method one QR_code image and the logic operation inverter (NOT) are used. The image represents the converted text of the secret message to QR_code then changing the inverse of dot's color in two special important areas (between the detection patterns); i.e, the white dot is substituted with black dot and visa versa.

To guarantee an exact retrieving to the original message from the encrypted QR, one of the reversible logic operators is used. The selected one is the inverter operator which provides this feature and suites to obtain from cipher text the original plain text.

3.3.1 Cipherying Algorithm

This method depends on Inverter logic operator. One QR code is used as input, to obtain one QR as a result. The QR is inverting in two special selected areas to generate one ciphered QR code as in sender side (encryption process). In receiver side, the process is reversed only by repeat the previous process by inverting the same two special selected areas in encrypted QR (cipher); where the white (dot) in QR represent (1) and the

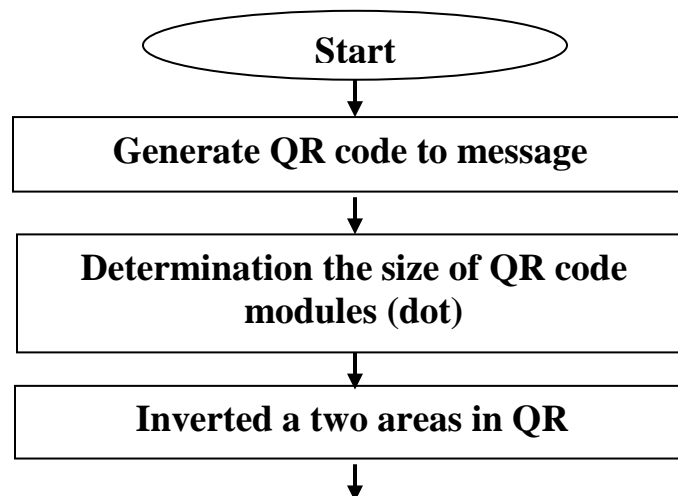
black module (dot) in QR represent (0) when inverted. The QR produced one QR image; the two special selected areas are content an important information about QR code like error corrections, format masking and reminder bits so that if these areas are changed make the QR code unreadable code that is shown in figure (3.6).



3.6: Example of above algorithm

3.3.2 Sender Side Algorithm

One QR code is generated for the secret message. Then, the QR is inverted in a specific two parts to encrypt the message. Figure (3.7) summarizes the encryption process.



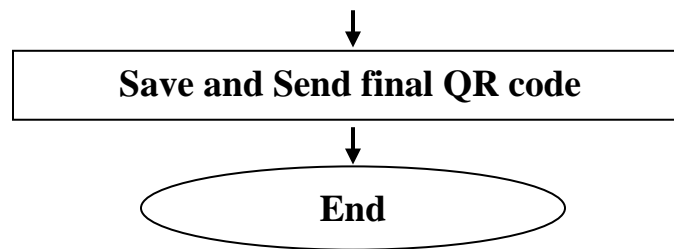


Figure 3.7: Flowchart of encryption process

• **Algorithm (3.7): Generating QR code for message**

Input:	The text message.
Output:	The corresponding QR code image for text message.
Steps:	
Step1: write message. Step2: generate QR code for the message. Step3: save QR image as P. Step4: end. Where P as plaintext.	

• **Algorithm (3.8): getting size of dots in QR**

Input:	The QR code image of the secret message.
Output:	The size of the black square in QR code image.
Steps:	
Step1: Do loop to get beginning of detection pattern on the top left in QR plain.bmp with width i and height j. Step2: Do loop to get the ending of detection pattern on the bottom left in QR plain.bmp whit width i1 and Height j2. Step3: Calculate the size of dot from i,i1 and j,j2. Step4: end.	

- **Algorithm (3.9): Encryption process**

Input: The QR code image of the secret message.

Output: The ciphered QR code image (encrypted QR).

Steps:

Step1: load QR image P.

Step2: define cipher as bitmap file with dimensions width (wd) & height (hg).

Step3: Call (Algo. (3.8)) sub routine to get first region indices (i,j).

Step4: Copy P image in cipher image except two important regions.

Step5: loop statement $x=i$ to end first region } Begin of the first
 loop statement $y=j$ to end first region } region

$\text{cipher}(x)(y)=\text{NOT}(P(x)(y))$

next y,x.

Step6: Call(Algo.(3.8)) sub routine to get second region indices (i1,j1).

Step7: loop statement $x=i1$ to end second region } Begin of the second
 loop statement $y=j1$ to end second region } region

$\text{cipher}(x)(y)=\text{NOT}(P(x)(y))$

next y,x.

Step9: end.

3.3.3 Receiver Side Algorithm

Since the used operator (NOT) is reversible, when destination receives the encrypted QR code, one only repeats the same process that used in encryption process for encrypted QR as input and the result the QR code for the secret message by inverting the selected two areas. Figure (3.8) explains the steps of work in Receiver side.

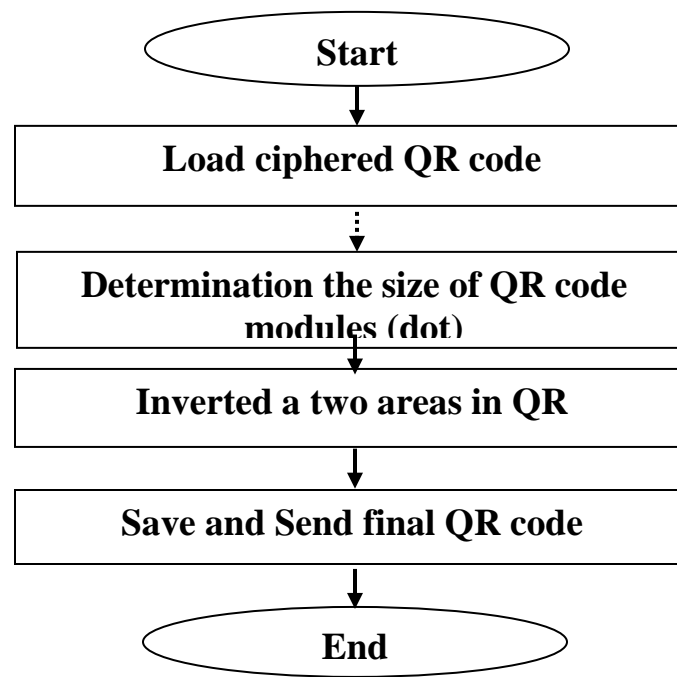


Figure 3.8: Flowchart of decryption process

- **Algorithm (3.10): Decryption process**

The same algorithm of encryption is used to decryption process.

Input:	The ciphered QR code image (encrypted QR).
Output:	The pure QR code image of the secret message.
Steps:	
Step1: load encrypted QR image C.	
Step2: define plain as bitmap file with dimensions width (wd) & height (hg).	
Step3: Call (Algo.(3.8)) to get first region indices (i,j).	
Step4: Copy C image in plain image except two important regions.	
Step5: loop statement x=i to end first region	} Begin of the first region
loop statement y=j to end first region	
plain (x)(y)=NOT(C(x)(y))	
next y,x.	

Step6: Call (Algo.(3.8)) to get second region indices (i1,j1).

Step7: loop statement $x=i1$ to end second region } Begin of the second
 loop statement $y=j1$ to end second region } region

plain (x)(y)=NOT(C(x)(y))

next y,x.

Step8: end.

3.4 Hiding Information Using QR code Application

New data-hiding is used with different scenarios from the two previous applications. This system uses two QR codes, one for the secret message and the other acts as a cover to the secret message, i.e, the first QR image should be embedded or hidden in the second QR so if an opponent receives the QR, that does not notice anything fishy only, an unsuspecting text or URL of cover. Figure (3.9) provides an illustration of a Hiding (Steganography) model or process. The cover carrier and the embedded message create a stego-carrier.

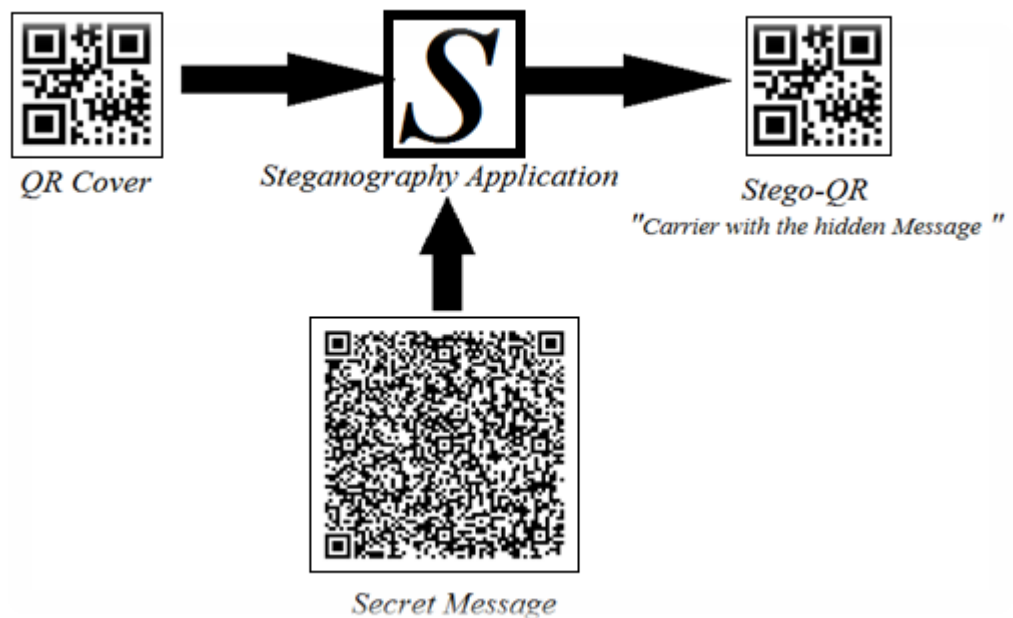


Figure 3.9: Hiding (Steganography) process

LSB (Least Significant Bit) Substitution is a type of data hiding method that modifying the least significant bit (right most binary digit) of the pixel of the carrier image. Figure (3.10) shown the (LSB) substitution in this method the (LSB) is the main idea of hiding process

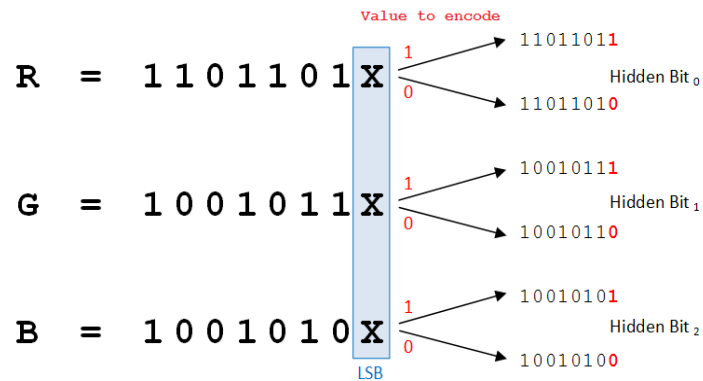


Figure 3.10: Least Significant Bit (LSB) substitution

Since the eyes of humans do not recognize the simple effective of color in RGB of colored image for example the black as shown in figure (3.11).

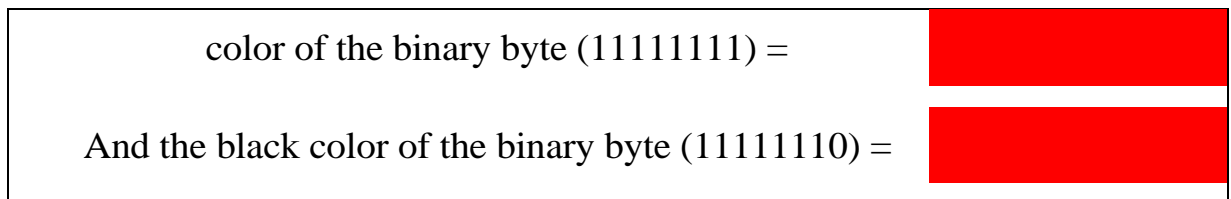


Figure 3.11: simple effective of red color

The same effect for the green and blue color, in digitally this provides benefits to change or replace the 0 to 1 or 1 to 0.

In this method, a new data-hiding is applied, where two QR code images are used, one for secret message and the other is a cover for secret message.

3.4.1 Hiding Algorithm

In this side, the sender imposes the QR code image of the secret message as binary image with 0 value for black region and 1 value for white region, and one imposes the QR code image of cover as binary image with 0 value for black region and 255 value for white region. Now if the region is black of the first QR the value of corresponding region in the cover image stall as 0 or 255, but if region is white of the first QR the value of corresponding region is become 1 if the value of corresponding region is 0 and 254 if the value of corresponding region is 255. Figure (3.12) explains the steps of work in Sender side.

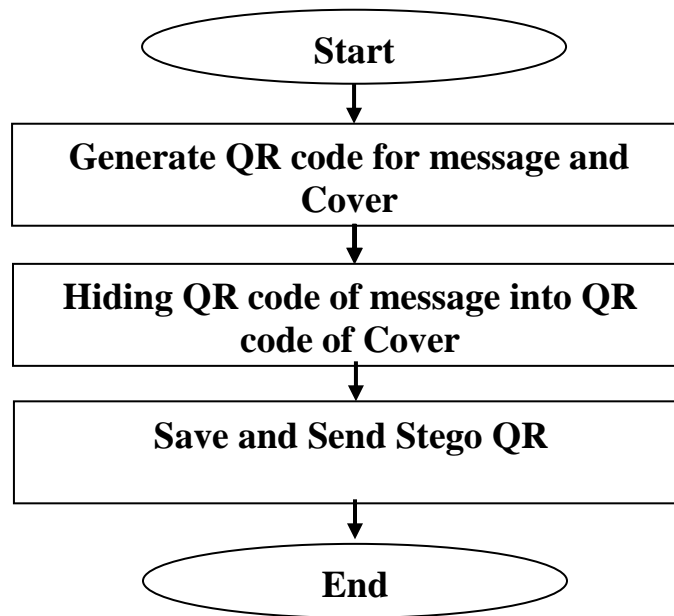


Figure 3.12: Flowchart of hiding process

- **Algorithm (3.11): Generation QR code for message and cover**

<p>Input: The text of the secret message and the text of the cover.</p> <p>Output: The QR code image of the message and the QR code image of the cover.</p>

Steps:

- Step1: write message (any text).
- Step2: generate QR code for the message (M).
- Step3: write cover (any text).
- Step4: generate QR code for the cover (Cover).
- Step5: save QR images as M and Cover.
- Step6: end.

• **Algorithm (3.12): Hiding process**

Input: The QR code image of the message and the QR code image of the cover.

Output: The Stego QR code (message embedded inside cover).

Steps:

- Step1: define S as bitmap image.
- Step2: load QR image P and C.
- Step3: loop statement $i=0$ to Wd of image
 - Loop statement $j=0$ to Hg of image
 - If $M(i,j)=255$ and $C(i,j)=0$ then
 - $S(i,j)=1$
 - Else
 - If $M(i,j)=255$ and $C(i,j)=255$ then
 - $S(i,j)=254$
 - next i,j .
- Step4: save S image.
- Step5: end.

3.4.2 Extraction Algorithm

In this side, the receiver receives an image that acts as a stego image (that handled the message). when the receiver loads the image and if he/she finds a white region and the value of this region is 254 digitally the receiver makes the value in the corresponding region of message image 255, and if the receiver found a black region and the value of this region is 1 then the receiver makes the value in the corresponding region of message image 255; and if the receiver found white region with 255 value and black region with 0 value the receiver makes the value in the corresponding region of message image 0. As a result the receiver will get an image of QR code if he/she scanned this QR; he/she will get the message. Figure (3.13) explains the steps of work in Receiver side.

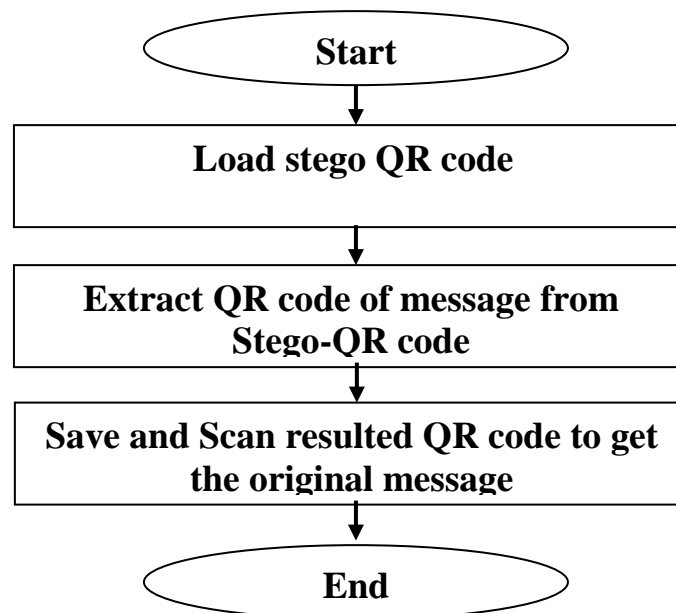


Figure 3.13: Flowchart of extracting process

- **Algorithm (3.13): Extracting message**

Input: The Stego QR code image (QR of Cover).

Output: The QR code image of the secret message.

Steps:

Step1: define M as bitmap.

Step2: load stego image as S.

Step3: loop statement $i=0$ to width of image

 loop statement $j=0$ to height of image

 if $S(i,j)=254$ or $S(i,j)=1$ then

$M(i,j)=255$

 else

$M(i,j)=0$

 next i,j .

Step4: save M image.

Step5: end.

CHAPTER FOUR

SYSTEMS TEST AND RESULTS

4.1 Introduction

This chapter includes a review about user interfaces of each application and applications' execution which are mentioned in the previous chapter. The review associated with an explanation related to the behaviors of the applications; also these behaviors are analyzed and discussed in details. The first application's tests and results are explained in section (4.4), the second application's tests and results are explained in section (4.5) and the third application's tests and results are explained in section (4.6).

4.2 Application's Requirements

Each security application has a method based on an algorithm that is applied the security properties. Since the implemented applications based on QR code so each application that presented in this chapter contain input part that is QR code of secret information (message only or message and key), application processing that applied an algorithm steps on input information to produce an output that represent ciphered QR code as shown in figure (4.1).

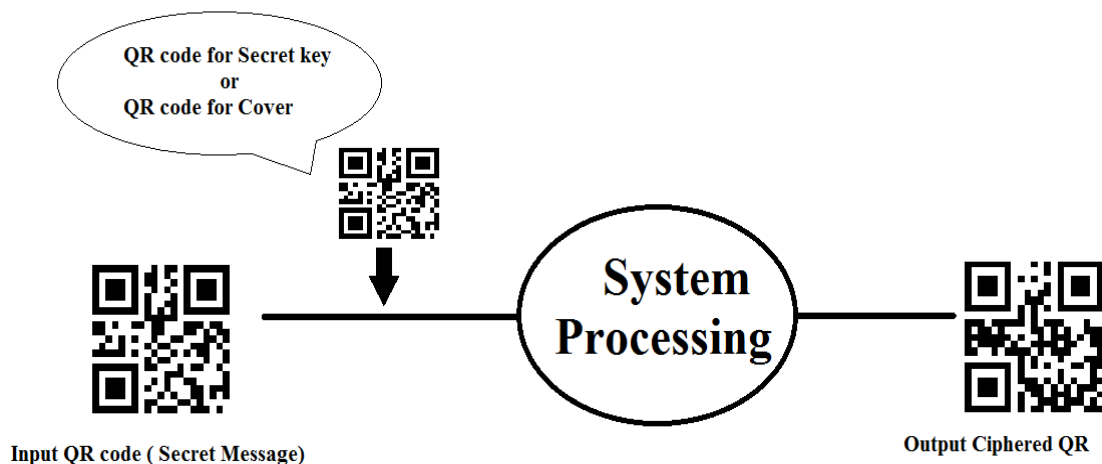


Figure 4.1: Systems work

The input QR code is generated by ZXing functions that is library provide by Google.

The output QR code read from any device that can scan this code such as scanner or mobile camera.

4.3 Software Programming Requirements

In this present thesis, the user uses “Java” programming language to represented the system programmatically, and using “XML” tool to programmed system’s interfaces, and both “Java” and “XML” are using under Android studio IDE (Integrated development environment) as software to represented these systems.

And this IDE is installed on operating system posed by the Micro Soft Company known as (Windows 10, 64-bits OS), since there is a software that will be installed on certain hardware to programmed and executed this software on it. Here the programmer uses Dell laptop with intermediate requirements that android studio IDE needs it and emulator, which will be performs .apk file on it. And the hardware (physical components) of a computer that has been working on it, they are:

- 1- CPU (Intel core i5 2.400 GHz chipset).
- 2- Memory (RAM 4GB).
- 3- Graphics card (VGA 1GB).
- 4- Hard Disk Dirve (HDD 500GB).





















4.4 Ciphering Special Versions of QR code with XOR Application Interfaces and Result

The main interfaces (Activities) for this system are illustrated in details in appendix (A) and described all its features.

4.4.1 Results

When implementing this system on several versions of QR codes with the same or different keys and calculating the MSE and PSNR, the obtained results are shown in table (4.1) in the next page.

Table 4.1: Special ciphering QR code application results

case no.	Message	Key	Result	Result ⁻¹	(Column 2 & column 4)	PSNR	MSE (Column 2 & column 3)
1					0	∞	0.972
2					0	∞	3.377
3					0	∞	1.731
4					0	∞	3.635
5					0	∞	1.753

4.4.2 Discussion

There are many versions of QR codes which are used in practical work. These versions determine the capacity of data size that QR code can handle. The size of the text data which is converted to QR code controls the size of black squares (modules). When the size of the data is big the size of black square (module) area in the QR code image is small, and when the size of the data is small the size of black square (module) area in the QR code image for secret message is big as a result. One observes in the case (1, 3, and 5) in table (4.1) that be affected more than when the size of black square of the secret message QR code is small. As a result, that generates a ciphered QR code with a low MSE and high PSNR, between QR images before and after encryption process. And when the size of black square (module) of the secret message QR code is big as a result, that generates a ciphered QR code with a small MSE and high PSNR, that observed in the case (2 and 4) in the table (4.1) that was not affected in QR code image.

















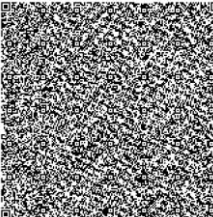
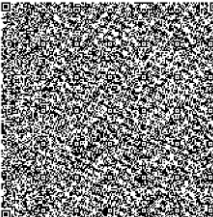
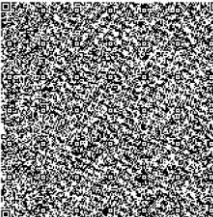

4.5 Ciphering General Versions of QR code Application Interfaces and Result

The main interfaces (Activities) for this system are illustrated in details in appendix (A) and described all its features.

4.5.1 Results

When this system is applied on several versions of QR codes and calculates the MSE and PSNR, the results are shown in table (4.2) in the next page.

Table 4.1: General ciphering QR code application results

Case no.	Message	Result	Result ⁻¹	MSE (column 2 & column 4)	PSNR	MSE (column 2 & column 3)
1				0		1.237
2				0		1.237
3				0		1.720
4				0		2.252
5				0		2.832

4.5.2 Discussion

Table (4.2) shows the results by applying encryption process on different versions of QR codes act as an input to this system. All output QR codes images have the same version of corresponding input images and provided ciphered QR code image with 0% percent of MSE and high percent of PSNR. Also this system is independent on the size of black square of QR code; all inputs have ciphered output (goal of encryption process) and only one QR code used.













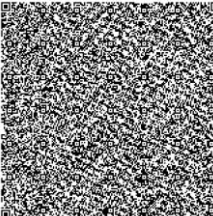


4.6 Hiding Information Using QR code Application Interfaces and Result

The main interfaces (Activities) for this system are illustrated in details in appendix (A) and described all its features.

4.6.1 Result

When implementing this system on several versions of QR codes and calculate the MSE and PSNR, the results are shown in table (4.3).

Table 4.3: Hiding Information Using QR code application Results

Case no.	Message	Cover	Stego-QR	MSE (column 3 & column 4)	PSNR
1				1.6085	46.0665
2				1.5938	46.1064
3				1.5499	46.2275
4				1.4671	46.4661
5				1.4077	46.6454

After showing the last table, that notices when the capacity or (message size) is increased, the MSE is decreased and the PSNR is increased. This result is caused because when the capacity increased, the information square is minimized and the change of pixels will unnoticeable, which increase the security of message. The figure (4.2) shows the changing the PSNR and MSE by increased the message size.

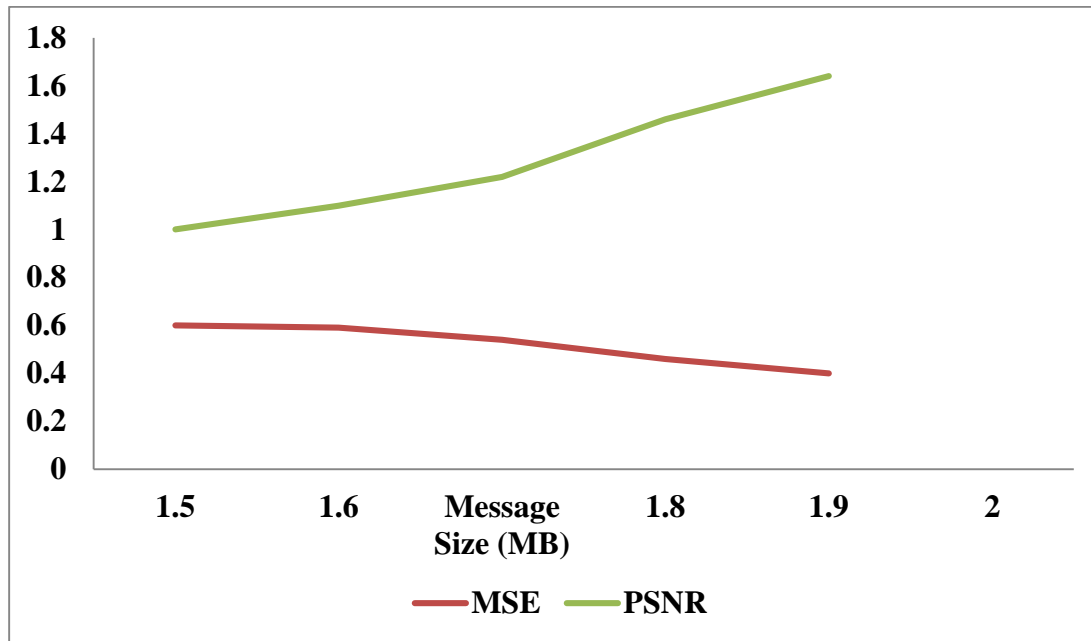


Figure 4.2: The Changing of MSE & PSNR by increase the Message Size.

4.6.2 Discussion

In this system, it has observed the results in table (4.3) from applying hiding process on different versions of QR codes with the same QR code cover. All output QR codes images have the same version of cover QR code that used when implementing hiding process. When the size of the secret message becomes high the MSE became low and the PSNR became high. The reason of the value of MSE becomes minimum due to the number and the small size of the black square in secret message that is affected on the result. There are probable to more than one black square in QR message hide in one black square in QR cover, As we notice, there are irregular increases due to the increase in the details shown in the QR image of the secret message. This is apparent in figure (4.2). The curved slope increased after the use of case images (3, 4 and 5).

CHAPTER FIVE

CONCLUSIONS AND SUGGESTION FOR FUTURE WORK

5.1 Conclusions

From this research, many important points are observed and concluded. the following are the most important ones:

1. The first application is faster than second application, and the second application is faster than third application. (Execution time 0.97sec for first method, 1.08 sec for second method and 2.34sec for third method).
2. The second method gives more acceptable results than the first method because it has been generalizing on all versions of QR codes.
3. The third method is more acceptable than others due to that the resulted QR code is readable.
4. There is a little difference between original QR code of the Cover and stego-QR code after embedment process.
5. The first method failed when dealing with high versions of QR code due to more details shown in the message QR code.
6. The second and third methods can be applied on all versions of QR code in purpose of ciphering or hiding.

5.2 Suggestions for Future Work

The following topics are suggested for future work:

1. Since these systems as mobile applications, a series of updates should be taken in accounts with time.
2. One can mix the work of two or three methods that are discussed in this thesis to produce a new method that leads to more secure results.

3. One can secure the chat room that supports this system to transfer QR code images between all users.
4. This application could be added to one of social media applications for transferring secret messages as QR code images.
5. For more security, one can adapt a middle stage of work that including a conversion of the message to a cipher text before converting it into its QR code.

References

[Cat13]

T. Cata, P. S. Patel, and T. Sakaguchi, "QR code: a new opportunity for effective mobile marketing," *Journal of Mobile Technologies, Knowledge and Society*, vol. 2013, 2013.

[Cha11]

D. Chatterjee, J. Nath, S. Dasgupta, and A. Nath, "A new Symmetric key Cryptography Algorithm using extended MSA method: DJSA symmetric key algorithm," in *Communication Systems and Network Technologies (CSNT), 2011 International Conference on*, 2011, pp. 89-94.

[Dav14]

J. Davies, "On-site digital heritage interpretation: current uses and future possibilities in world heritage sites," Unpublished MA thesis, Department of Archaeology, Durham University, 2014.

[Dey12]

S. Dey, "SD-EQR: A New Technique To Use QR Codes™ in Cryptography," *International Journal of Information Technology and Computer Science (IJITCS)*, 2012.

[Dey13]

S. Dey, S. Agarwal, and A. Nath, "Confidential Encrypted Data Hiding and Retrieval Using QR Authentication System," in *Communication Systems and Network Technologies (CSNT), 2013 International Conference on*, 2013, pp. 512-517.

[Dut12]

Ph. Dutson, "Creating QR and Tag Codes", Sams Publishing, 2012.

[Esl14]

T. Eslinger, "Mobile Magic: The Saatchi and Saatchi Guide to Mobile Marketing and Design", John Wiley & Sons, 2014.

[Gai15]

Gaikwad, Akshara, and K. R. Singh. "information hiding using image embedding in QR codes for color images: A review." *International Journal of Computer Science and Information Technologies*, Vol.6, 2015, P.P. 278-283.

[Huy08]

Huynh-Thu, Quan, and Mohammed Ghanbari. "Scope of validity of PSNR in image/video quality assessment" *Electronics letters*, Vol.44, Issue 13, 2008, p.p. 800-8.

[Kes04]

Kessler, Gary C. "An Overview of Steganography for the Computer Forensics Examiner (Updated Version, February 2015)", *Forensic Science Communications*, Vol.6, 2004.

[Kie10]

P. Kieseberg, et al., "QR code security," in *Proceedings of the 8th International Conference on Advances in Mobile Computing and Multimedia*, 2010, p.p. 430-435.

[Kro14]

Krombholz, Katharina, et al. "QR code security: A survey of attacks and challenges for usable security", *International Conference on Human Aspects of Information Security, Privacy, and Trust*. Springer International Publishing, 2014.

[Mut14]

Muthaiah, R. M., and N. Krishnamoorthy. "An Efficient Technique for Data Hiding with use of QR Codes-Overcoming the Pros and Cons of Cryptography and Steganography to Keep the Hidden Data Secretive" *International Journal of Computer Applications*, Vol.100, (2014).

[Nar16]

Narmatha M., and Venkata K. "***Study on Android Operating System And Its Versions.***" *International Journal of Scientific Engineering and Applied Science (IJSEAS)* 2.2 (2016), p.p. 439-445

[Qji14]

Q. Ji, "Exploring the concept of QR Code and the benefits of using QR Code for companies", Bachelor's Thesis, Lapland University of Applied Science, 2014.

[Sha13]

L. Shao, J. Sun, and X. Hui, "Fuzzy Systems, Knowledge Discovery and Natural Computation Symposium", DEStech Publications, 2013.

[Sin16]

A. Singh and P. Singh, "A REVIEW: QR CODES AND ITS IMAGE PRE-PROCESSING METHOD", 2016.

[Som12]

Dey, A. Somdip, B. Joyshree Nath, and C. Asoke Nath. "A New Technique to Hide Encrypted Data in QR Codes (TM)." *Proceedings on the International Conference on Internet Computing (ICOMP)*. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2012.

[Son14]

D. Sonawane, etal, "QR based Advanced authentication for all hardware platforms", International Journal of Scientific and Research Publications, vol. 4, pp. 1-4, 2014.

[Tsa14]

Tsafarakis, Odysseas, and Wilfried GJHM van Sark. "Development of a data analysis methodology to assess PV system performance." 29th European Photovoltaic Solar Energy Conference and Exhibition. Amsterdam. 2014.

[Wei10]

M. Weir, "QR Codes & Mobile Marketing for the Small Business Owner", Michael Weir, 2010.

[Wen13]

Wu, Wen-Chuan, Zi-Wei Lin, and Wei-Teng Wong. "Application of QR-code steganography using data embedding technique." *Information Technology Convergence*. Springer Netherlands, 2013. 597-605.

APPENDIX

(A)

A.1 Special Cipherring QR code Application Interfaces

The main interface (Activity) for this system is illustrated in figure (A.1) and describes all its features:

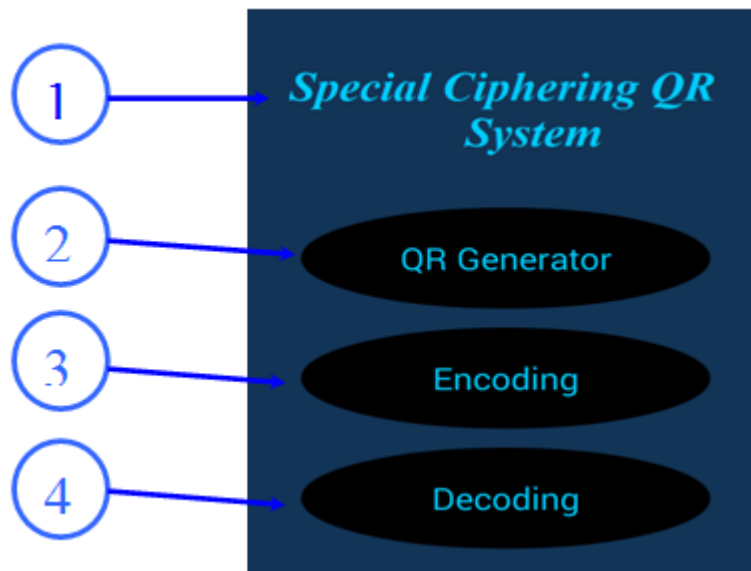


Figure A.1: Main User Interface of System

Four choices are located in the main activity for system jobs. They are:

- 1- The name of the system (special cipherring QR system) located in the top of the screen.
- 2- Button named "QR Generator" when the user clicks it, it moves to a new activity that shown in figure (A.2).



Figure A.2: User Interface of QR Generator Button

- 2.1 Text-Box that allowed the user to type any text either message or key (numeric, alphanumeric, etc).
 - 2.2 Button named “Generate QR”, which after the text is typed by the user in the (text-box) it is pressed this to generate an image of QR code for text that typed.
 - 2.3 Button named “save”, when the user clicks it, the generated image will be saved in the gallery of the device.
- 3- Button named “Encoding”, when the user clicks it, it moves to a new activity that shown in figures (A.3) and (A.4).

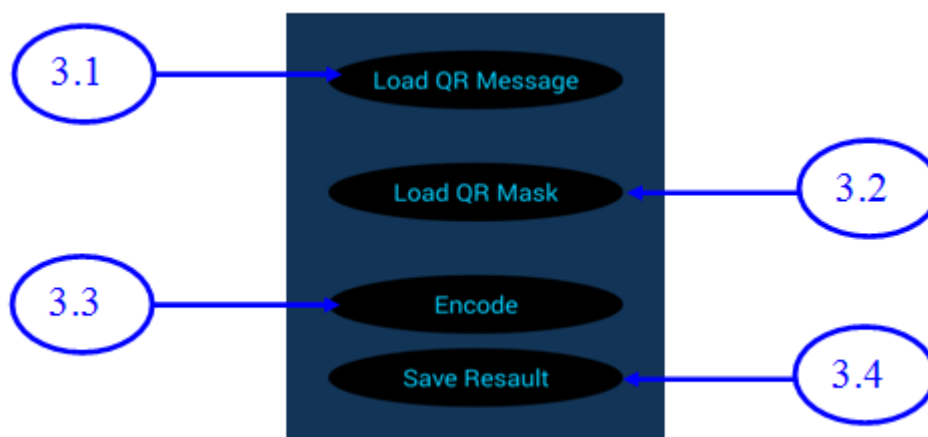


Figure A.3: User Interface of Encoding Button

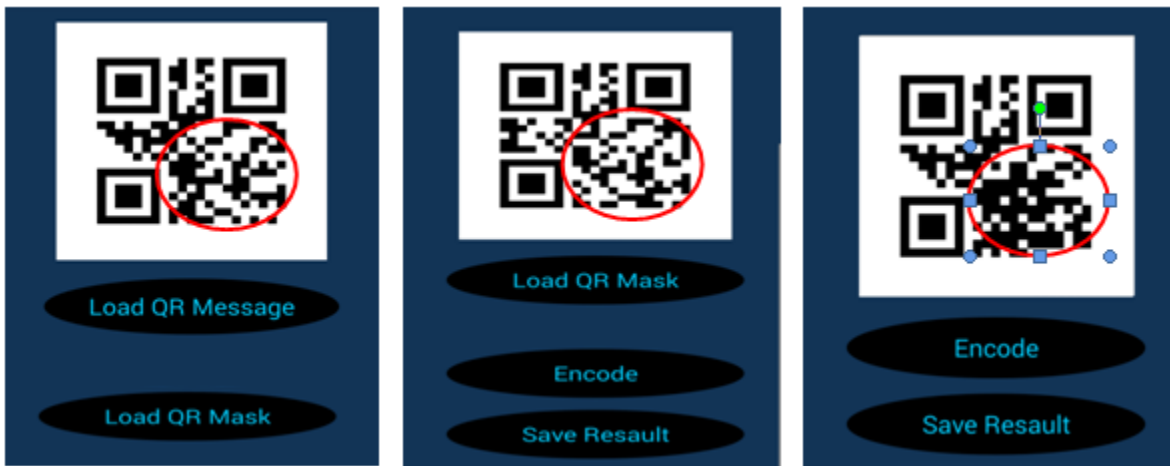


Figure A.4: User Interface of Encoding Button

- 3.1 Button named “Load QR Message”, when the user clicks it, it opens the gallery of the device to let the user chooses the QR image that saved from previous activity.
 - 3.2 Button named “Load QR Mask”, when the user clicks it, it opens the gallery of the device to choose QR image that saved from previous activity.
 - 3.3 Button named “Encode”, when the user clicks it, it performs the encryption process by XORing the two QRs in a specific selected area and embeds the content of the key in the resulted QR code and shows the result in new image view.
 - 3.4 Button named “Save Result”, when the user clicks this button, the resulted image from encryption process is saved in gallery of the device (the work of this button exactly the same as the work of save button of (2.3) in previous section).
- 4- Button named “Decoding”, when the user clicks it, It moves to a new activity that shown in figure (A.5) and (A.6).

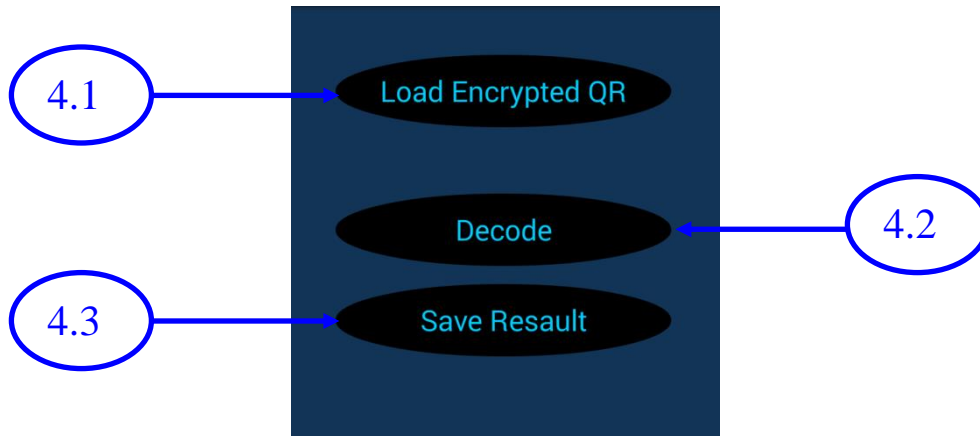


Figure A.5: User Interface of Decoding Button



Figure A.6: User Interface of Decoding Button

- 4.1 Button named “Load Encrypted QR”, when the user clicks it, it opens the gallery of the device to choose encrypted QR image. This button immediately extracts the key from encrypted QR and shows in the new image view.
- 4.2 Button named “Decode”, when the user clicks it, it performs the decryption process by XORing the two QRs in the same specific selected area and shows the result in a new image view.
- 4.3 Button named “Save Result”, when the user clicks this button the resulted image from decryption process saved in gallery of the device (the work of this button exactly the same as the work of save button of (2.3) and (3.4) in previous sections).

A.2 Cipherring General Versions of QR code System Interfaces

The main interface (Activity) for this system is shown in figure (A.7). Then a description of all its contents is declared.

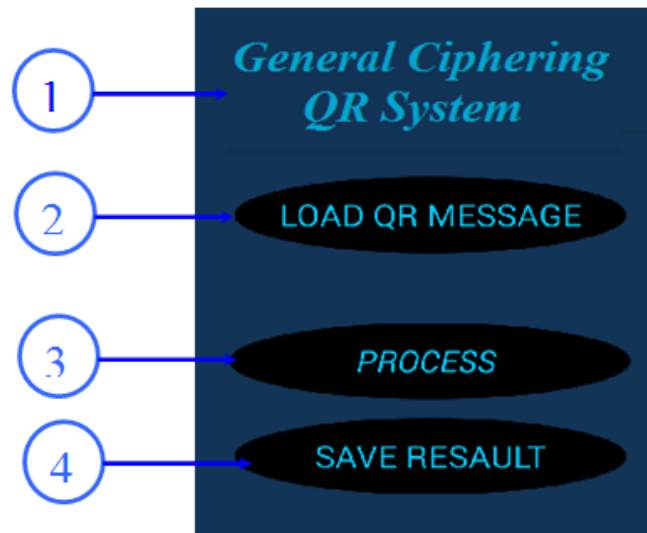


Figure A.7: Main User Interface of System

Four things are located in the main activity. They are:

- 1- The name of the system (general cipherring QR system) located in the top of the screen.
- 2- Button named "Load QR Message", when a user clicks it, it opens the gallery of the device to choose QR image for secret message or for ciphered message that saved in device and shows the in image view as shown in figure (A.8) (a).
- 3- Button named "Process", when a user clicks it, and the user loads QR code image for secret message it performs encryption process on QR code image to produce ciphered QR code image and preview in new image view; when a user clicks the button and the user chooses an encrypted QR code image (ciphered) the result after performed process is pure QR code for secret message (readable) was preview in new image view as shown in figure (A.8) (b).
- 4- Button named "Save Result", when a user clicks this button, the resulted image from previous process is saved in gallery of the device (the work of

this button exactly the same as save button of (2.3), (3.4) and (4.3) in previous sections) as shown in figure (A.8) (c).

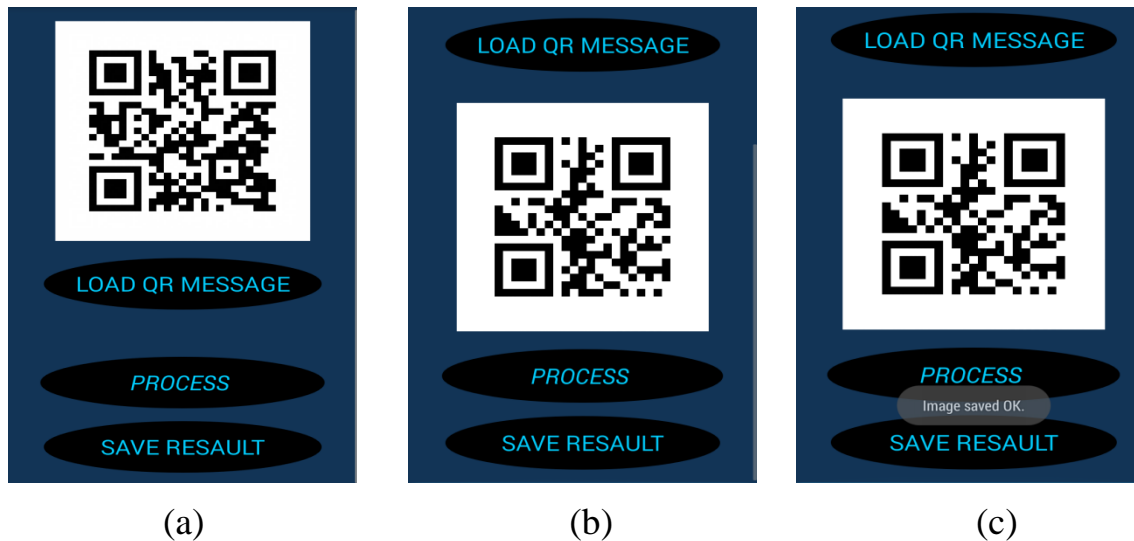


Figure A.8: Main User Interface of System

(a): User interface after load QR image (b): User interface after process done (c):
User interface after saved QR image.

A.3 Hiding Information Using QR code System Interfaces and Result

The main interface (Activity) for this system is shown in figure (4.9). Then a full description of its contents is followed:

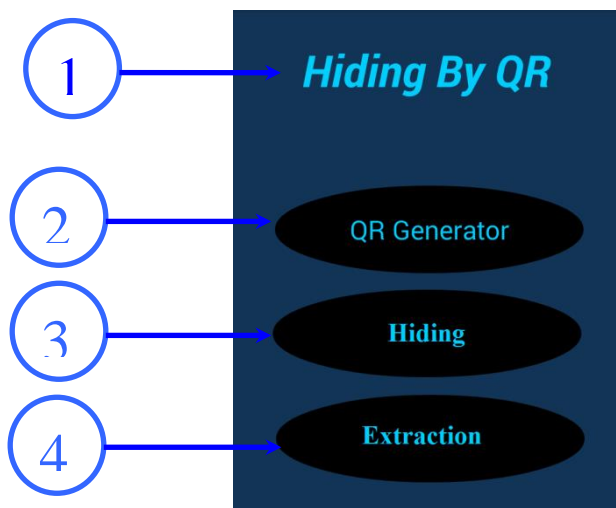


Figure A.9: Main User Interface of System

Four things are located in the main activity. They are:

- 1- The name of the system (Hiding by QR code System) located in the top of the screen.

- 2- Button named “QR Generator”, when a user clicks it, it moves to a new activity which is shown in figure (A.2), and this point have the same properties of the 2nd point in “*Special cipherring QR code system*”.
- 3- Button named “Hiding”, when a user clicks it, it moves to a new activity which is shown in figures (A.10) and (A.11).

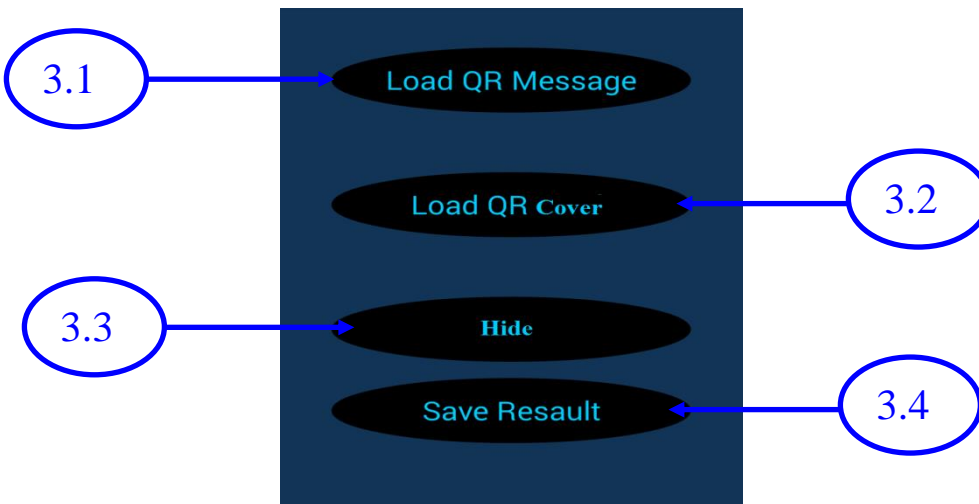


Figure A.10: User Interface of Encoding Button

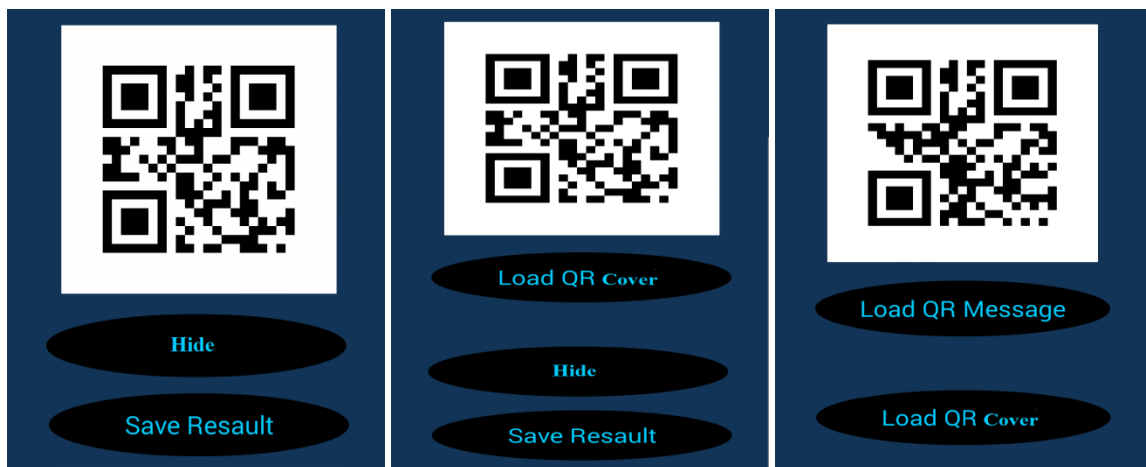


Figure A.11: User Interface of Hiding Button

- 3.1 Button named “Load QR Message”, when a user clicks it, it opens the gallery of the device to choose QR image that saved from activity of QR generator in point (2 and its sections) in pages (3 and 4).

- 3.2 Button named “Load QR Cover”, when a user clicks it, it opens the gallery of the device to choose QR image that saved from activity of QR generator in point (2 and its sections) in pages (3 and 4).
- 3.3 Button named “Hide”, when a user clicks it, it performs the Hiding process between two QRs and shows the result as a new image view.
- 3.4 Button named “Save Result”, when a user clicks this button, the resulted image from Hiding process is saved in gallery of the device (the work of this button is exactly the same as save button of (2.3) in previous section).
- 4- Button named “Extraction”, when a user clicks it, it moves to a new activity that is shown in figure (4.12) and (4.13).

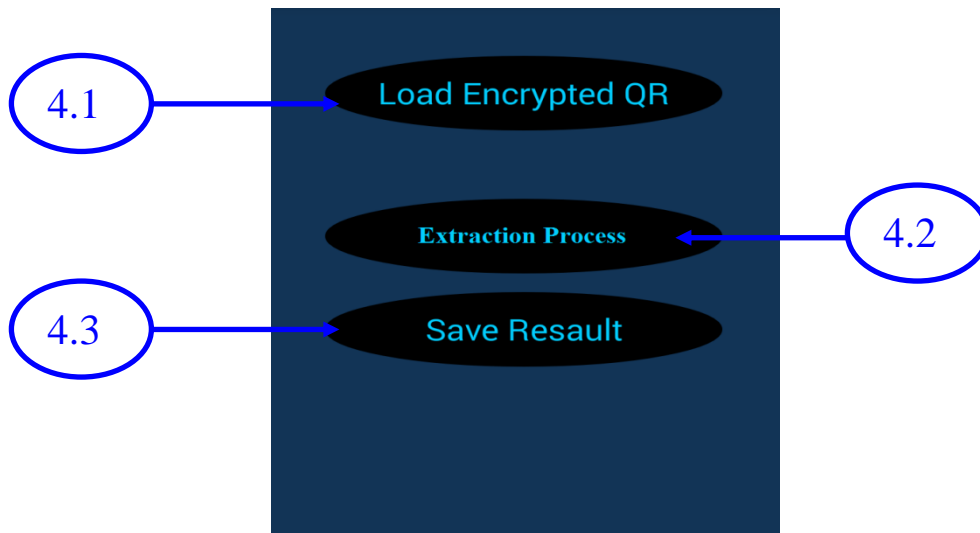


Figure A.12: User Interface of Extraction Button



Figure A.13: User Interface of Extraction Button

- 4.1 Button named “Load Encrypted QR”, when a user clicks it, it opens the gallery of the device to choose an encrypted QR image which is saved from previous activity.
- 4.2 Button named “Extraction process”, when a user clicks it, it performs the extraction process and shows the result in as new image view.
- 4.3 Button named “Save Result”, when a user clicks this button, the resulted image from decryption process is saved in the gallery of the device (the work of this button is exactly the same as save button of (2.3) and (3.4) in previous sections).

APPENDIX

(B)

Table (B.1) reveals that the different versions have different modules which contain function patterns modules, format and version information modules, data modules. In addition, the capability of data is different as well. However, in terms of the remainder bits, some part of versions are same, some is difference.

Table B.1: Codeword capacity of all versions of QR Code [Sha13].

Version	No. of Modules/ side (A)	Function Pattern Modules (B)	Format and version Information modules (C)	Data modules except (C) ($D=A^2-B-C$)	Data capacity [codewords] ^a (E)	Reminder Bits
M1	11	70	15	36	5	0
M2	13	74	15	80	10	0
M3	15	78	15	132	17	0
M4	17	82	15	192	24	0
1	21	202	31	208	26	0
2	25	235	31	359	44	7
3	29	243	31	567	70	7
4	33	251	31	807	100	7
5	37	259	31	1079	134	7
6	41	267	31	1383	172	7
7	45	390	67	1568	196	0
8	49	398	67	1936	242	0
9	53	406	67	2336	292	0
10	57	414	67	2768	346	0
11	61	422	67	3232	404	0
12	65	430	67	3728	466	0
13	69	438	67	4256	532	0
14	73	611	67	4651	581	3
15	77	619	67	5243	655	3
16	81	627	67	5867	733	3
17	85	635	67	6523	815	3

18	89	643	67	7211	901	3
19	93	651	67	7931	991	3
20	97	659	67	8683	1085	3
21	101	882	67	9252	1156	4
22	105	890	67	10068	1258	4
23	109	898	67	10916	1364	4
24	113	906	67	11796	1474	4
25	117	914	67	12708	1588	4
26	121	922	67	13652	1706	4
27	125	930	67	14628	1828	4
28	129	1203	67	15371	1921	3
29	133	1211	67	16411	2051	3
30	137	1219	67	17483	2185	3
31	141	1227	67	18587	2323	3
32	145	1235	67	19723	2465	3
33	149	1243	67	20891	2611	3
34	153	1251	67	22091	2761	3
35	157	1574	67	23008	2876	0
36	161	1582	67	24272	3034	0
37	165	1590	67	25568	3196	0
38	169	1598	67	26896	3362	0
39	173	1606	67	28256	3532	0
40	177	1614	67	29648	3706	0

^a All codewords are 8 bits in length, except in versions M1 and M3 where the final data codeword is 4 bits in length



جمهورية العراق
وزارة التعليم العالي والبحث العلمي
جامعة النهرين
كلية العلوم

تطبيق للهواتف الذكية لأمنية البيانات داخل رموز الاستجابة السريعة

رسالة

مقدمه الى كلية العلوم في جامعة النهرين
كجزء من متطلبات نيل درجة الماجستير
في علوم الحاسوب

من قبل

بشير ناهض عبدالامير

(بكالوريوس علوم حاسبات, 2013)

اشراف

م.د. سوسن كمال ثامر

1438 هـ
2017 م

جمادى الاولى
يوليو

المُلخَص

رمز الاستجابة السريعة هو باركود ثنائي الأبعاد لترميز المعلومات الخاصة مثل (الارقام, الروابط, الحروف الابدجية, و بايت / ثنائي). رموز الاستجابة السريعة تستخدم بصورة رئيسية لحمل او خزن الرسائل لانها تحوي على سعة ذاكرة اكبر او اعلى من الانواع الطبيعية التقليدية الاخرى. في هذا العمل، الدافع هو استخدام رموز الاستجابة السريعة لأغراض أمنية أو هو استخدام رموز الاستجابة السريعة لإرسال واستقبال البيانات بطريقة سرية. حيث تم تطبيق ثلاث خوارزميات جديدة (اثنان لتشفير رموز الاستجابة السريعة, وواحد للاخفاء). في الخوارزمية الاولى, يتم تمثيل الرسالة و المفتاح على شكل الباركود ثنائي الابعاد, والتشفير يتم عن طريق استخدام XOR على منطقة خاصة في الاثنين من اجل تحصيل باركود ثنائي الابعاد مشفر. ثم يتم تضمين مفتاح التشفير في الباركود الثنائي الابعاد المشفر.

في الخوارزمية الثانية, يتم تمثيل الرسالة على شكل الباركود ثنائي الابعاد, والتشفير يتم عن طريق استخدام NOT على منطقتين مهمتين فيه من اجل تحصيل باركود ثنائي الابعاد مشفر. اخيراً, الخوارزمية الثالثة, يتم فيها استخدام رسالة سرية و رسالة ثانية تكون كغطاء للرسالة الاولى. عندما يتم مسح الباركود الناتج ببرنامج القراءة العادي سوف تظهر الرسالة العادية بدون ان تثير اي شبهة.

كل نظام يهدف الى بناء وتنفيذ تطبيق على الهاتف الذكي الذي يستخدم تقنية الباركود ثنائي الابعاد لتشفير او اخفاء المعلومات من اجل السرية. مثل هذه التطبيقات تتطلب تصميم الجزء البرمجي, والذي يمثل برنامج مكتوب باستخدام بيئة اندرويد استديو. هذا البرنامج يزودنا بفايلات من نوع apk. والتي تكون مقبولة للتنفيذ على جهاز الموبايل. البرامج النهائية من الممكن تنفيذها على اي موبايل يتطابق مع نظام الاندرويد.

النتائج التي تم الحصول عليها في هذه الرسالة هي كما يلي: الخوارزميتان الأولى والثانية في عملية التشفير وفك التشفير (صفر بالمائة نسبة مربع الخطأ) ولا توجد اي نسبة ضوضاء، بينما تعطي الطريقة الثالثة نسبة مئوية منخفضة في مربع الخطأ وكذلك في نسبة الضوضاء .