



Improvement of Image Steganography Using Physical Unclonable Function

By

Noor Mueen Mohammed Ali Hayder

University of Babylon, Iraq
noormouen@yahoo.com

Abstract

Image steganography stills a hot topic for researchers because of a huge number of images that are exchanged via the Internet. In this paper, image stenography system with least significant bit (LSB) method is designed and implemented to hide a text message. The proposed system differs from other LSB systems by depending on completely random keys in choosing hiding positions of the cover image. These keys are hardware random numbers (HRN) which are generated by physical unclonable function (PUF).

The proposed PUF is designed and implemented using microcontroller chip (PIC32MX795F512L) with Ring Oscillator PUF (ROPUF). The output (HRN) of ROPUF is generated without requiring any seed inputs. The HRN which is created by the proposed random number generator (RNG) is passed most NIST tests successfully, this proves its randomness. The generated HRN also have high entropy value (0.9999999) and it have correlation value (-0.1514535) which means it have no correlation. The PSNR of the stegocovers are between (73.634924422 - 95.912316112).

Keywords:

Physical Unclonable Function (PUF), Hardware Random Numbers (HRN), Least Significant Bit (LSB), Random Number Generator (RNG).

1. Introduction

Information protection nowadays is one of the majority vital factors of information processing and correspondence; the reason goes back to a big increase of the (WWW) and the copyright rules. There are many important technologies are utilized for information safety of digital reality for today. The steganographic technology is one of the most important technologies

used to protect information in security, this technology depends on hiding and covering the information to protect it [1].

Randomness concept is utilized widely in this field; also the power and strong argument of any encoding algorithm, build upon the encoding secret code attributes; its length and randomness. The protection of all applications of this field depends essentially on making unpredictable secret code [1].

In the field of security, PUF is a simple physical unit to build but approximately hard to make the second one, even known the accurate developed process that produced it. The strong argument of PUF is its distinguishing characteristic; because it is so hard to make a copy of the circuit as it is not possible to control the developed process variations [2].

Also one of the significant characteristics of PUF is the Low cost of production random numbers (RNs). Therefore, this paper focuses on developing a new security system depending on random secret codes which are generated by PUF. These random secret codes are utilized for hiding.

This paper is organized as follows. In section (2) the steganography and PUF function and its characteristics are presented. The proposed system is explained in section (3). Experimental results and analyses are shown in section (4). Finally, the conclusions are remarked in section (5).

2. Background

This part is concerned with selected topics, which are considered as the background for this paper.

2.1 Steganography

It means “covering writing” in Greek. It is the art of hiding data by embedding message within another file which called cover; this grouping is known as stego-file. Several algorithms have been produced in this field; one of them is Least Significant Bit (LSB) [3].

LSB is a common and simple technique to embed information in an image file. In this method, the LSB of a byte is replaced with an M's bit. It works very well with image steganography. A high-quality and resolution image is an easier to hide information inside it. Although 24 Bit images are best for hiding information due to their size [3]. Formula (1) shows a very generic description for the process of steganography technique [4].

$$\text{Hidden information} + \text{Cover-image} = \text{Stego-image} \quad (1)$$

2.2 Physical Unclonable Function (PUF)

PUF (also called Physical Random Function) is a new class of security, in which it has attracted a great deal of attention. The modern cryptographic scheme depends on the use of one-way functions. These are functions that are simple to work in the forward direction but infeasible to compute in the reverse direction without additional information [6].

PUFs are one-way functions, which are easy to evaluate but difficult to invert. These hardware one-way functions are inexpensive to fabricate, difficult to duplicate, grant no compact mathematical representation [6]. PUFs are innovative primitives to deduce secrets from complex hardware characteristics of ICs rather than storing the secrets in digital memory [7].

The use of PUFs for the secret key generation was first proposed in [8]. The proposed PUF in this paper based on internal Ring Oscillator (RO). A ROPUF is composed of an odd series of inverters. The RO frequency is generated from the inverted signal that travels through the RO

loop as shown in Fig. 1 [8]. The presence of process variations inside logic gates and wires causes an uneven delay across the chip.

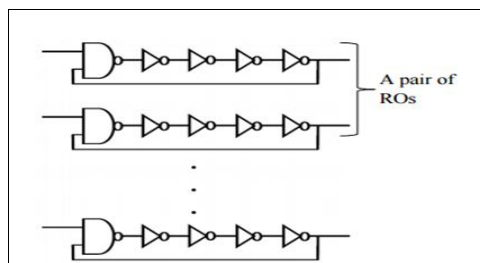


Fig. 1 . Ring Oscillator Physical Unclonable Function (ROPUF)

A pair of ROs could produce two different frequencies because of the presence of process variations.

3. The Proposed System

This proposed system combines the notion of PUF as random number generator with steganography.

The proposed secure system contains HW and SW designs. The HW design involves ROPUF depend on microcontroller chip (PIC32MX795F512L). The SW design is about text in image steganography scheme using LSB technique. Figure (2) shows the general block diagram of the proposed system at the transmitter side. While the general block diagram of the proposed system at the receiver side is shown in figure (3).

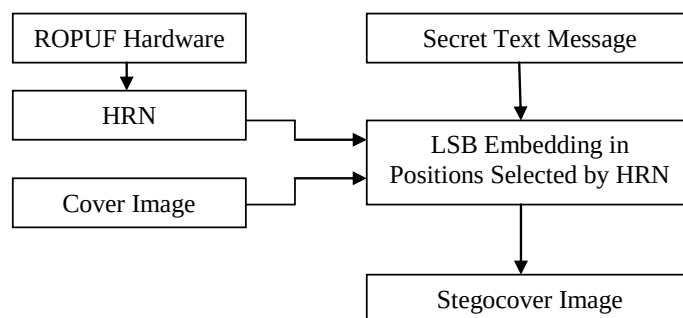


Fig. 2 . Block Diagram of the Proposed System in Transmitter Side

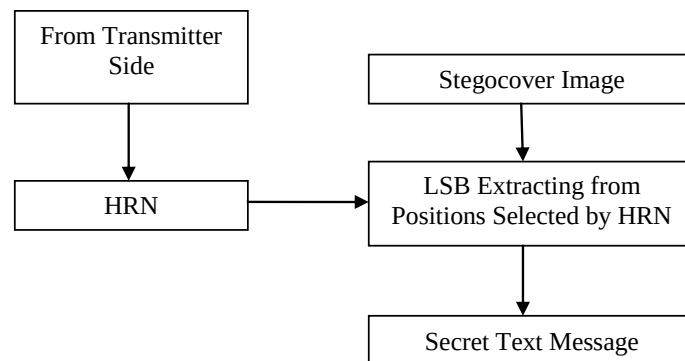


Fig. 3. Block Diagram of the Proposed System in Receiver Side

3.1 Transmitter Side

A. Hardware Design (Design of ROPUF)

This is the first step at the transmitter side of the proposed system as in figure (2). The purpose of this step is to generate HRNs. ROPUF is explained previously in subsection (2.2). The hardware design of the proposed system is shown in Fig. 4. The hardware electronic circuit contains a power supply and two integrated circuits, first one (IC2 7805) is (5v) regulator. While the second one (IC3 TC1262-3.3) is a programmable regulator.

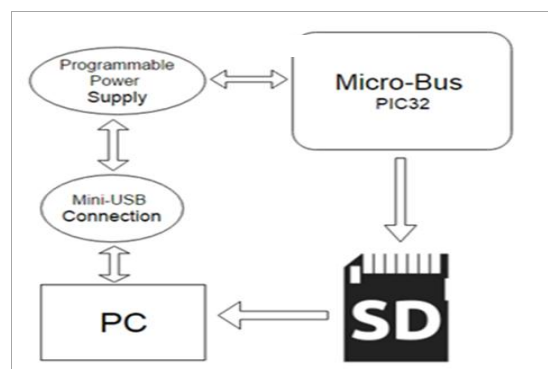


Fig. 4. Block Diagram of the Hardware for the Proposed System

Fig.5. shows the programmable power supply (TC1262) which is used to give the variable voltage approximately (1.5-5 v) for the microcontroller and 3.3v to signalization LED.

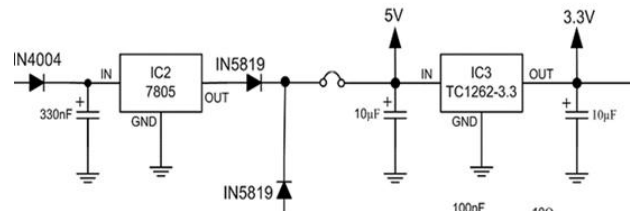


Fig. 5 . Power Supply of the Hardware Proposed System

The Micro-Bus technique is used as a variable way to send data to SD and PC, in which it connects (8) pins.

The algorithm of the written program inside microcontroller chip PIC32MX795F512L is shown in algorithm (1). This program is written in High-level Micro-C language.

Algorithm (1) Writing Random Numbers on PC and SD

Input: Libraries (SD, SPI, and UART) ; Counter (X).

Output: Writing random numbers on PC and SD.

1. First step: is calling the library of SD, SPI, and UART. Then a variable (x) is chosen to be as a counter for counting the binary numbers. In this paper, the maximum value of (x) is 99393. Of course, this value can be changed as required.
2. Second step: is using the Real Time Clock (RTC) system in order to serially writing data on PC and in SD.
3. Third step: is checking the data that must be written on SD when these data are displayed on PC. Otherwise, go to step (2).
4. Fourth step: is repeating the condition of step (3) for 24 times which equal to a number of binary bits in the proposed system.
5. Fifth step: is stopping write data on PC and SD card when $x > 99393$.

B. Software Design

Choose Secret Text Message and Cover Image

This is the first step in the proposed software design of the transmitter side, in which a suitable secret text message and cover image are chosen. The size of secret text message must be according to Eq. 1.

$$\text{Size of secret text message} \leq \text{Size of cover image} / 8 \quad (1)$$

Convert Format of Secret Text Message

This is the second step in the proposed software design of the transmitter side. In this step, the characters of the secret text message are converted to decimal format by using ASCII value for each character. Also, in this step cover image pixels are converted into decimal format.

Embedding Secret Message using LSB

This is the third step in the proposed software design of the transmitter side. LSB algorithm is described in subsection (2.1). In LSB insertion method with secret code, a random number generator is used to hide bits of secret message in the least significant bit of cover image pixels randomly. In this approach, the random numbers are obtained from ROPUF in which is described in subsection (3.1 A). The implemented LSB method is clarified in algorithm (2).

Algorithm (2) LSB Algorithm for Embedding Message in Cover Image

Inputs: Cover image, HRN-secret code, and secret message.

Output: stego image.

```
{
  1. Convert the secret message into equivalent binary value into an 8 bit integer array.
  2. Read the RGB color image (cover image) into which the secret message is to be
    embedded (in Blue) and then convert it to binary.
  3. While not end of secret message
    {
  4. Read secret code from ROPUF file and check it if secret code is in range of image size
    then
  5. Take a bit from binary ciphered secret message and hide it in LSB of the selected blue
    pixels.
  6. Write the pixel of step (5) into stego image file.
    } end while.
  7. Write the rest of the cover image pixels in stego image file.
} End program.
```

Convert Format of Stego Image

This is the last step in the proposed software design of the transmitter side . In this step, the stego image data are converted into image format.

3.2 Receiver Side

The receiver side of the proposed system consists of software design only. This is because of the HRNs file (which is created in transmitter side subsection (3.1 A)) is handed down formerly to the receiver side or it is ciphered and then it is transmitted to the receiver side. Note that ciphering of HRNs file is not the key point of this paper. Therefore, the software design of the receiver side of the proposed system includes the following steps.

i. Convert Format of Stego Cover Image

This is the first step at the receiver side of the proposed system. In this step, the stego cover image is converted to the decimal numbers of RGB colors.

ii. Extraction of Ciphered Secret Message using LSB

In LSB extraction algorithm, the values of the selected locations are taken from HNR file which is created in subsection (3.1 A). Then, after converting pixel values to binary form, these bits are taken from least bits and ordered sequentially in order to form the binary bytes of the ciphered secret message. The extraction of the ciphered secret message is shown in the algorithm (3).

Algorithm (3) Extraction of Ciphered Secret Message using LSB Technique

```

Inputs: Stego-image, HRN file.
Output: secret message file.
{
    1. Open the stego image as decimal numbers.
    2. While not end of (HRN file)
    {
        3. Take a pixel according to the selected location.
        4. Take the least bit of the selected pixel and put it
           into extracted secret message file sequentially.
    }
}

```

iii. Convert Format

This is the last step at the receiver side of the proposed system. In this step, the ASCII values of the secret message are converted to characters form. These characters are ordered sequentially, then the secret text message is ready to be read.

4. Results

Since the proposed system consists of two phases (HRNs generation phase, and hiding phase), therefore different measures are used to evaluate these phases.

4.1 HRNs (PUF Keys)

Table (1) shows a sample of (256) byte HRN keys (hardware ROPUF random numbers).

Table (1) (256) Bytes HRN Keys

256 Bytes HRN Keys
56,55,13,96,43,182,29,30,52,69,124,49,74,189,255,117,122,202,143,27,186,198,118,38,27,21 9,194,125,205,159,67,218,69,10,247,180,138,178,178,112,152,114,223,228,239,148,59,133,9 6,250,254,61,98,167,38,57,244,244,133,162,231,106,170,50,144,55,41,53,235,224,172,139,40 ,93,86,118,121,12,99,253,242,207,69,129,173,195,98,192,15,46,74,153,128,187,128,149,192, 230,2,97,141,41,37,112,191,180,26,40,192,131,154,239,168,188,59,113,32,173,133,199,96,18 7,207,174,60,35,83,18,90,66,10,212,135,73,159,56,190,52,18,161,144,184,141,50,208,108,35, 30,104,139,250,59,146,18,93,166,87,48,56,73,197,131,20,244,237,234,186,21,236,231,137,23 1,29,179,69,74,8,246,173,146,25,84,193,43,90,42,30,17,224,56,199,154,20,231,229,148,54,93 ,205,94,229,208,230,150,57,246,137,97,57,152,151,188,116,85,15,65,66,178,56,133,154,22,4, 117,65,109,234,14,198,54,35,59,232,126,212,106,194,157,130,163,38,142,115,245,105,140,2 8,1,66,175,234,161,228,65,123,37,220,64,

4.2 Randomness Tests of HRNs (PUF Keys)

Table (2) shows the results of the NIST tests on the HRNs which is generated by the proposed PUF circuit.

Table (2) Results of NIST Statistical Test Suite on the generated HRNs

Test name	Result	Lowest success ratio
Frequency Test	Successful	99%
Approximate Entropy test	Successful	100%
Block Frequency Test	Successful	100%
Serial Test	Successful	98%
Cumulative Sums Test (Forward)	Successful	100%
Runs Test	Successful	98%
Longest Run of One's test	Successful	100%
FFT Test	Successful	100%
Rank Test	Successful	100%
Nonperiodic Templates Test	Successful	79%
Overlapping Template of all One's Test	Successful	100%
Lempel-Ziv compression Test	Successful	100%

4.3 Distribution of Selected Locations

Figure (6) and figure (7) show the distribution of selected locations by (PUF) for hiding with different sizes of the secret message which is to be embedded in Al-Shaheed cover image.

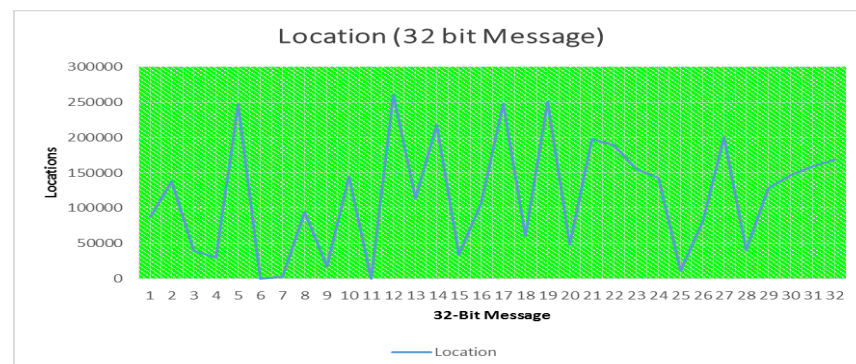


Figure (6) Distribution of Selected Locations for Hiding 32-bit Secret Message in Al-Shaheed Cover Image

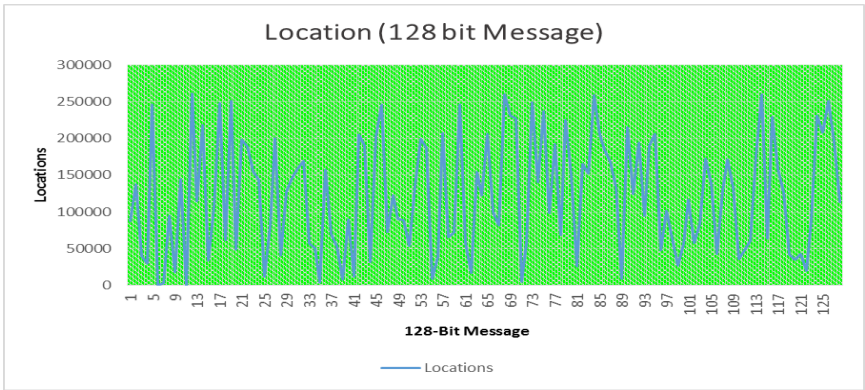
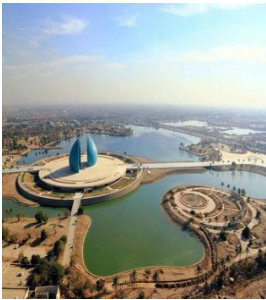
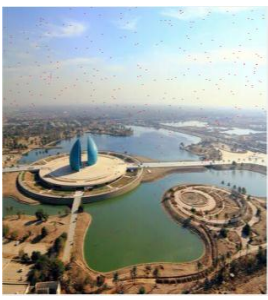


Figure (7) Distribution of Selected Locations for Hiding 128-bit Secret Message in Al-Shaheed Cover Image

4.4 Stegocover Images

Table (3) shows a comparison between the two stegocovers with and without hided locations for 512-bit secret messages.

Table (2) Comparison between the Stegocovers with and without Hided Locations for 512-bit Secret Messages

Cover image Name	Cover image	Stego cover image	Stego image With Hiding Locations for 512 Secret bits
Al-Shaheed			



4.5 Quality Metrics Results

The results of the quality metrics which are used to evaluate the proposed system are shown in the table (4) and table (5).

Table (4) Quality Metrics Results of the Proposed System for AL-Shaheed image

Image	Measure	Message size					
		16 bit	32 bit	64 bit	256 bit	512 bit	1024 bit
AL-Shaheed	MSE	0.0000362537	0.000090634	0.00018731	0.000743202	0.001540785	0.0028157099
	PSNR	92.53727108	88.55787099	85.405166651	79.41973275	76.25338178	73.634924422
	NPCR	99.99637435	99.99093655	99.9812688	99.92567975	99.84592145	99.718429003
	UACI	1.416163199	3.540407854	7.31684290	0.000290313	0.000601869	0.0010998867

Table (5) Quality Metrics Results of the Proposed System for Angels image

Image	Measure	Message size					
		16 bit	32 bit	64 bit	256 bit	512 bit	1024 bit
My Angels	MSE	0.0000066666	0.00002063908	0.00004370629	0.000283333	0.000541666	0.001089583
	PSNR	95.912316112	94.98389960	91.72536380	83.60782689	80.79348250	77.75819909
	NPCR	99.998333333	99.99793609	99.99562937	99.97166666	99.94583333	99.89104166
	UACI	6.5104166666	1.302083333	1.707277097	0.000110677	0.000211588	0.000425618

5. Conclusions

In this research, a proposed hardware random numbers generator is designed and implemented using ROPUF inside microcontroller chip PIC32MX795F512L to output random numbers which are used to hide secret message in cover image. The results indicate that the generated HRNs has a true randomness, and its generator does not need seed value.

In spite of LSB steganography is commonly used, but the proposed system improves LSB technique by hiding secret message in random positions of cover image. Table (4) and table (5) show that the proposed system is successfully used as a text in image steganographic system.

References

- [1] Abdulzahra, Hayfaa, R. O. B. I. A. H. Ahmad, and NorlizaMohd Noor, "Combining cryptography and steganography for data hiding in images", Applied Computational Science : 128-135, 2014.
- [2] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, "Controlled physical random functions", Proceedings of 18th Annual Computer Security Applications Conference, December 2002.

- [3] S. Gupta, G. Gujral, and N. Aggarwal, "Enhanced least significant bit algorithm for image steganography" ,IJCEM International Journal of Computational Engineering & Management 15.4 : 40-42, 2012.
- [4] B. Singh, T. Kumar, S. Kataria, and N. Singh Shekhawat, "A Steganography Algorithm for Hiding Secret Message inside Image using Random Key", International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 3 Issue 12, December-2014.
- [5] S.Sivasankari, "Fpga Implementation Of Invisible Video Watermarking Using Dwt Technique", International Journal of Innovations in Scientific and Engineering Research (IJISER), Vol 1 Issue 1 JAN 2014, pp.7-12.
- [6] PappuSrinivasaRavikanth, "Physical one-way functions", Ph.D. thesis, Massachusetts Institute of Technology, March 2001.
- [7] J.-W. Lee, D. Lim, B. Gassend, G. E. Suh, M. van Dijk, and S. Devadas, " A technique to build a secret key in integrated circuits with identification and authentication applications", Proceedings of the IEEE LSI Circuits Symposium, June 2004.
- [8] Suh, G. Edward, and SrinivasDevadas, "Physical unclonable functions for device authentication and secret key generation." Proceedings of the 44th annual Design Automation Conference. ACM, 2007.
- [9] Andrew Rukhin, Juan Soto, James Nechvatal, Miles Smid, Elaine Barker, Stefan Leigh, Mark Levenson, Mark Vangel, David Banks, Alan Heckert, James Dray, San Vo, "A statistical test suite for random and pseudorandom number generators for cryptographic applications", NIST Special Publication 800-22 (with revisions dated May 15, 2001).

