LCC-LSB-FPGA Stego-A Reconfigurable Security

Balakrishnan Ramalingam, Rengarajan Amirtharajan and John Bosco Balaguru Rayappan
School of Electrical and Electronics Engineering, SASTRA University,
Thanjavur, Tamil Nadu, India

Abstract: Information touting implants the secret info within a cover file making the former imperceptible. This study suggests an unexamined steganography construct to embed encrypted message bit in a (digital) color image taking up Linear Congruential Generator (LCC) for producing the arbitrary plot in support of pixel preference and elementary Least Significant Bit (LSB) routine for data infixing. This proposal also witnesses optimum embedding competence along with minimum computation complexity and superior image excellence. In order to prove the complexity and superiority of the proposed algorithm, Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR) of the stego image has become calculated and results are reported. The proposed algorithm was implemented in ALTERA cyclone ii FPGA for which design and performance results are presented in the study.

Key words: Linear congruential generator, FPGA, hardware steganography

INTRODUCTION

With the advancement of internet and communication technologies, information sharing has become easier and faster. However, maintaining secrecy and confidentiality of the information being shared between two entities has turned into a specialized area requiring continuous research and development. Cryptography (Schneier, 2007; Salem et al., 2011) and steganography (Cheddad et al., 2010; Karzenbeisser and Pericolas, 2000; Amirtharajan et al., 2012, Provos and Honeyman, 2003; Petitcolas et al., 1999; Praveen Kumar et al., 2012a, b, 2013a-j; Amirtharajan and Rayappan, 2013) are two commonly used methods employed for securing the privacy of the information transmitted between the nodes of the network.

In cryptography, the confidential information is being transmitted is scrambled so that it can be read by only by the intended recipient by unscrambling. Steganography, on the other hand offers solution by hiding the secret data in a multimedia object such as, text, image, audio and video files without being noticed by eavesdroppers. The drawback of the cryptography is that the cryptic messages draw attention of the hackers.

Steganography on the other hand does not reveal the presence of hidden secret information. If the presence of hidden information is revealed or even suspected, the purpose of steganography is partly defeated. In order to provide another layer of security in the steganography, the secret data is encrypted either in spatial or frequency domain, prior to hiding it in the cover object. For embedding in spatial domain several schemes such as Least Significant Bit (LSB) (Chan and Cheng, 2004; Amirtharajan and Balaguru, 2009; Amirtharajan et al., 2011; Amirtharajan and Rayappan, 2012a, c; Amirtharajan et al., 2013a), Pixel indicator method (Gutub, 2010; Amirtharajan and Rayappan, 2012b; Amirtharajan et al., 2013b) and Pixel Value Differencing (PVD) (Amirtharajan et al., 2010) have been proposed to encrypt the pixels hidden in the cover image. For frequency domain encryption, the cover image is first transformed using Discrete Cosine Transform (DCT) (Song et al., 2012) or discrete wavelet transform (Chen and Lin, 2006) or Integer Wavelet Transform (IWT) (Amirtharajan and Rayappan, 2012a, b; Thanikaiselvan et al., 2013; Ramalingam et al., 2014) and the confidential data is hidden in the transformed coefficients of the cover.

Most encryption and data hiding algorithms have been implemented in software, due to the flexibility. However, software implementation is vulnerable to the security breaches of the operating system in addition to the less throughput arising from the non-dedicated nature of the platform. Reconfigurable devices like FPGAs are attractive solutions for embedding cryptographic and steganography applications and offer higher throughput and security (Ramalingam et al., 2014; Rajagopalan et al., 2012a, b, Jankiraman et al., 2012).

Corresponding Author: Balakrishnan Ramalingam, School of Electrical and Electronics Engineering, SASTRA University, Thanjavur, Tamil Nadu, India
The FPGA has the advantage of low investment cost and provides moderate processing speed. Knowing the review on steganography, this FPGA work is organized as follows hardware work presents reconfigurable hardware architecture for random image steganography in materials and methods with proposed method. There is no implementation hardware LCC PRNG generator enhance security of the algorithm without compromising the computation space and time in existing review. Results and discussion is presented as next section. Final section concludes the proposed method offers better security of the proposed implementation without compromising the computation space and time.

**MATERIALS AND METHODS**

Functional block diagram of proposed scheme is shown in Fig. 1. Here, data is hidden by LCC random generator and LSB data hiding technique.

*Cover and secret data:* Here cover file and secret data is chosen, respectively as BMP gray image or color image and encrypted message.

*LSB embedding process:* LSB embedding technique inserts the secret data bits straight to the cover image’s LSB plane in a settled order employing both color and grey images. Because of 8 bit representation, 1 or 2 bits can be buried in grey and monochrome images. 3 bits can be embedded in all the three color components if it is a 24 bit image.

*Linear Congruential Generators (LCG):* LCG-Linear Congruential Generator method generates pseudo random numbers up to a certain series, after which the sequence starts recurring. This is known as “seed”. The level of uncertainty is determined by the values m, a and c:

\[
X_{n+1} \equiv (ax_n + c) \mod M \\
M, 0 < M
\]  

Where, \(M\), modulus input \(A\), multiplier and \(c\), constant.

According to the seed value and constraints, LCC PRNG produces two random progressions founded on Eq. 1. First sequence is to randomly choose \(M \times N\) image block, second is for selecting the image pixels \(M \times N\) matrix.

**Algorithm for secret data embedding process:**

- Input: Cover Image (C) and encrypted message (A)
- Output: Stego Cover (S)
- Step 1: Read the secret message and (Color) cover image (C)
- Step 2: Using the first arbitrary sequence, jumble the secret message
- Step 3: Change the messed up data bits into binary row matrix
- Step 4: Segment the image into 8×8×3 blocks namely B1, B2, . . . , Bn
- Step 5: By means of the random progression two, choose 8×8×3 blocks and divide the RGB plane separately
- Step 6: With random series three, arbitrarily choose the RGB pixels in 8×8 matrices to entrench the secret data
- Step 7: Perform variable bit embedding as given by the user
- Step 8: Merge the RGB planes to produce the stego cover

**Hardware architecture:** The reconfigurable hardware architecture of data hiding system is shown in Fig. 2. There are four modules which construct the main parts of the hardware: FSM processing unit, SRAM controller, data Embedding unit and LCC random number generator.

![Fig. 1: Block diagram of the data hiding architecture](image-url)
**FSM unit:** The synthesized state diagram of FSM processing unit is shown in Fig. 3. FSM processing unit generates the control signal for SRAM controller, data embedding unit and LCC random number generator. FSM processing unit consists of state decoding logic circuit, timing generation unit, general purpose storage registers and functional registers. Functional registers are used to store the current state, next state and FSM state and header information of the image. General purpose registers are used to store the initial seeds of LCC and count value of embedded message bits etc. Timing and control logic is made of PLL (Phase Lock Loop) which generates the required clock signal to integrated functional hardware components.

**SRAM controller:** SRAM communication core is used to read or write the data from master device (such as the FPGA). It is composed of 16 bit data bus, 18 bit address bus and four control signals such as read, write and output enable and word or byte mode selection. Timing diagram of SRAM is shown in Fig. 4(a-b). The SRAM Controller supports a clock frequency of 50-200 MHZ. This study uses SRAM Controller to communicate with the 256K×16 asynchronous CMOS static RAM (SRAM) chip on Altera’s DE2/DE1 Boards.

**On chip embedded memory:** FPGA M4K memory bits are configured as dual-port on chip embedded memory which can be used to store the M×N RGB value, encrypted secret message and M×N stego pixels and random key sequence. Schematic diagram of on chip embedded memory shown in Fig. 5. In simple dual-port mode, host can simultaneously carryout read and write operation in memory blocks because dual port memory has separate write enable and read enable signal.

**Data embedding unit:** Function block diagram of embedding module is shown Fig. 6. It consists of function registers A, B and cascaded AND-OR logic module each of 24 bit wide. Function registers are used to store 24 bit pixel value and secret message bits in substitution process. After hiding the data into image pixel, the pixel value is stored into on chip embedded memory.
Fig. 4(a-b): SRAM (a) Read and (b) Write timing diagram

Fig. 5: On chip embedded memory

**LCC random number generator:** LCC random number generator consists of LFSR, XOR feedback, multiplier adder and modules function and storage buffer. A secret key can be used as an initial seed to generate the random number. RTL view of the LFSR is shown in Fig. 7.
Fig. 6: Function block diagram of embedding module

Fig. 7: RTL view LCC random number generator

**HARDWARE SYNTHESIZE AND PERFORMANCE ANALYSIS RESULTS**

Proposed data hiding architecture has been synthesized in ALTERA QUARTUS II Design software Version 9.0 and tested in ALTERA DE2 development board. It is made up of cyclone ii FPGA and 512 KB on-board static SRAM memory. Compilation report of data hiding architecture was given in Table 1. RTL view and design floor planning output of data hiding architecture are shown in Fig. 8(a-b).

**Timing analysis:** Throughput of data hiding architecture is calculated through ZERO plus logic analyzer modules.
Figure 9 shows Logic analyzer timing output. The data hiding architecture consumes 0.8 is for RGB separation, random number generation and embedding process in one 8×8 block.
Table 1: Compilation report of data hiding architecture

<table>
<thead>
<tr>
<th>Family</th>
<th>Cyclone II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device</td>
<td>EP2C35F672C6</td>
</tr>
<tr>
<td>Total logic elements</td>
<td>960/33216 (3%)</td>
</tr>
<tr>
<td>Total combinational function</td>
<td>723/33216 (2%)</td>
</tr>
<tr>
<td>Dedicated logic register</td>
<td>610/33216 (2%)</td>
</tr>
<tr>
<td>Total pins</td>
<td>44/475</td>
</tr>
<tr>
<td>Embedded multiplier 9 bit elements</td>
<td>2/70 (3%)</td>
</tr>
</tbody>
</table>

Quality analysis of embedded stego image: In this present implementation, color digital images Lena and baboon of dimension 128×128 have been taken as cover images, which is shown in Fig. 10(a-d). The stego process has been studied by calculating MSE and PSNR for variable k bit embedding and its results are given in Table 2. Figure 11(a-d) shows the stego image for k = (3, 3, 2) scheme.

Mean Square Error (MSE):

\[
MSE = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} (X_{ij} - Y_{ij})^2
\]  

where, M and N represent the total number of pixels in the cover image and Y, j represents the pixels of the stego-image. Lesser MSE value means higher image quality.

Peak Signal to Noise Ratio (PSNR):

\[
PSNR = 10 \log \left( \frac{I_{max}}{MSE} \right) \text{ dB}
\]  

where, I max is the peak intensity value of each pixel which is equal to 255 for 8 bit gray scale images. Higher value of PSNR superior the image quality cover image.

From the Table 2, it is vivid that proposed algorithm provides high PSNR and low MSE for variable bit embedding.
CONCLUSION

Proposed data hiding architecture enables the new gateway for hardware based color image steganography for audio data hiding. The speed of the hardware has been sufficient for real-time application. It achieves higher throughput for various size color images. LCC PRNG provides the good secure way of hiding the audio in image pixel based on the LSB substitution method. As 30 dB is fixed as the threshold PSNR value for human visual system, the present results possess excellent imperceptibility without noticeable degradation and the same is well supported by the estimated PSNR value for the stego covers.

REFERENCES


