A Comparative Analysis in Hardware Partitioning of a Steganographic based LSB-substitution Algorithm

Shyam Sumukh S R, Raghav Gupta, Jagadish Nayak

Abstract— **Steganography is an amalgamation of art and science complimenting each other where the embedding or hiding of information is done in a digital image. This paper presents a logic level design synthesis (RTL) and IC design is done in Synopsys to visualize the implementation of LSB substitution steganographic algorithm based on 8 bit images, using HDL. This paper intends to visualize and compare the logic level design implementation of a steganographic algorithm as a standalone independent IC or as sub functioning block in a microprocessor (IP block) and conclude on the best possible method to implement steganography in hardware.**

Index Terms— **LSB, Steganographic Algorithm, HDL, Logic level design, IC design**

I. INTRODUCTION

Steganography is a subject which has both artistic and scientific importance, we can summarize that steganography is a concept of writing hidden text or information in a way that no one, apart from the sender and intended recipient, suspects that a hidden information is concealed in it. It is a type of secureness through inconspicuousness [1]. The word is of a Greek dialect meaning "concealed writing" from the Greek words steganos inferring to “protected” and graphei inferred to "writing.

Embedding secret information of high value in images requires exhaustive calculations, and therefore one method is partitioning steganography in a processor as hardware which boosts up steganography [2]. Images are good for steganography conversion because of their size. For instance, a sender will start with an image data and adjust the colour of a random pixel to correspond to a letter in the alphabet; an infinitesimal change occurs and is not visible when seen with the human visual receptors.

II. LITERATURE REVIEW

Securing data is a challenging issue in today’s era and is also of paramount importance. Most of the data travel over the internet and it becomes difficult to make data imperceptible to a third party. So Cryptography was introduced for making data secure [1]. But alone cryptography cannot provide a better security approach because the scrambled message is still available to the eavesdropper.

There arises a need of data hiding. So an amalgamation of steganography and cryptography is used for improving the security. Implementing of steganography with a feistel cipher network (cryptographic encryption for plain text) [1]-[6] helps in integrating it with cryptography. Hardware (FPGA Virtex II pro Platform and Cyclone II platform) implementation of steganographic techniques using Xilinx ISE or ModelSim, using LSB steganography [2]-[4] have been done in the past to study its applications. Images are the most prevalent carriers for steganography, and hence the importance of image steganography is established.

LSB image based steganography is substituting the LSB of carrier image with a bit of the secret message or image [3]. To augment the security of a stego image, the secret image or message bits are first re-arranged and then embedded in a predefined sequence of chosen pixels from the carrier [5]. One major rule is that the secret message size should always be less than that of carrier for efficient data concealing.

III. METHODOLOGY

This methodology focuses on checking how well the hardware performs with respect to embedding the secret data. The current work will compare and will perform an analysis on two ways of implementing LSB substitution steganographic algorithm for 8 bit data. The comparative analysis is a pilot study and will suggest a result on the best suitable way of implementation. This analysis will help decide in partitioning the steganography processing unit as a separate hardware entity (IC) or as a sub processing block in a processor. To check for results, the embedding of information in stego image is done sequentially until all bits in the secret image are embedded in cover image. Here in both the methods embedding is done in a sequential manner. The LSB substitution algorithm is explained in [5].

Method 1: Reading Cover and Secret Images which are 8 bit data (converted to text files, done by the stego block) and are fed as inputs for performing LSB substitution. The current VHDL design is based on file handling and is intended to make the steganography unit as a separate entity and partitioned separately in physical layout architecture as standalone hardware outside the microprocessor. To

Manuscript received June 18, 2014; revised July 23, 2014.

Shyam Sumukh S R is pursuing his graduate studies (M.E) in Microelectronics at Birla Institute of Technology and Science, Pilani – Dubai Campus, Dubai - 345055, UAE, phone: +971-562563956; e-mail: shyamsumukh@gmail.com.

Raghav Gupta is pursuing his graduate studies (M.E) in Microelectronics at Birla Institute of Technology and Science, Pilani – Dubai Campus, Dubai - 345055, UAE, phone: +971-528662095; e-mail: guptaraghavgupta11@gmail.com.

Dr. Jagadish Nayak is an Assistant Professor with the Electrical and Electronics Engineering Department, Birla Institute of Technology and Science, Pilani – Dubai Campus, Dubai - 345055, UAE, phone: +971-554907979; e-mail: jagadishnayak@dubai.bits-pilani.ac.in.

summarize the method it converts image data to binary data and embeds it to the carrier image. This design is intended to function as an independent functioning unit.

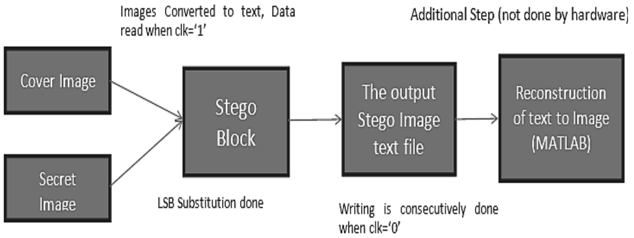
Images Converted to text, Data read when clk='1'
Additional Step (not done by hardware)
Cover Image
Stego Block
The output Stego Image text file
Reconstruction of text to Image (MATLAB)
Secret Image
LSB Substitution done
Writing is consecutively done when clk='0'

Figure 1: Block diagram representation of Method 1

Method 2: Use Cover Image raw data being directly being fed in parallel and the secret image data fed being fed in serially to perform the LSB substitution (here the preprocessing of image pixel data to binary vector is done by microprocessor). To simulate the random 8 bit pixel data, a pseudo random number generator is used (PRSG) and has period of 2^n -1 which is considerably random to simulate a continuous flow of 8 bit data in to the block.

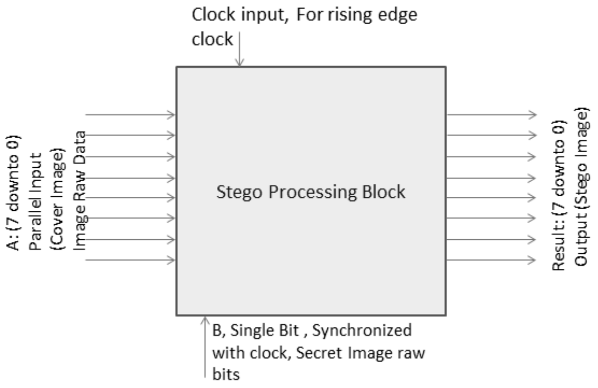
Clock input, For rising edge clock
A: (7 downto 0) Parallel Input (Cover Image) Image Raw Data
Stego Processing Block
Result: (7 downto 0) Output (Stego Image)
B, Single Bit , Synchronized with clock, Secret Image raw bits

Figure 2 Block diagram representation for Method 2

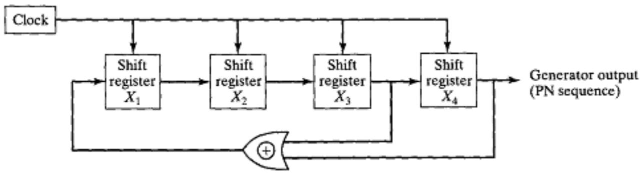

Figure 3: A typical PRSG (shift register concept)

Additionally, to realize hardware implementation of the proposed methods we need to design their physical layouts [6], since the measurable quantities like power and area occupied are derived from physical layout design. Also, to compare effectively, the same secret image or data has been used in both the methods.

IV. RESULTS

Methods 1 and 2 have been designed in the physical layout to derive their simulated characteristics. The power and cell area of both the methods have been tabulated.

Method 1: No. of cells = 32, No. of Nets = 50, No of Combinational Cells = 16, No. of Sequential Cells = 16
Method 2: No. of Cells = 17, No. of Nets = 19, No. of Combinational Cells = 1, No. of Sequential cells = 16

TABLE I. COMPARISON OF POWER IN METHODS 1& 2

Met.	Power Analysis				
	Total Power		Power Group		
	Total Dynamic Power (uW)	Cell Leakage Power (uW)	Sequential (uW)	Combinational (uW)	Total Power (uW)
1	16.5207	9.2036	10.2751 (39.94 %)	15.4492 (60.06%)	25.7243
2	0.5388	7.0066	7.3764 (97.03%)	0.2262 (2.97%)	7.6025

TABLE II. COMPARISON OF AREA (UNITS IN CELL) IN METHODS 1 &2

Met.	Area Analysis				
	Area Group			Area	
	Combinational Area	Non-Combinational Area	Net Interconnect Area	Total Cell Area	Total Area
1	132.71	457.11	18.659	589.824	608.483
2	13.824	398.13	11.529	411.955	423.484

The total power comprises of sequential and combinational values indicated in the power group. It is been noted that the power of the combinational circuit in Method 2 is only of leakage power; both switching and internal powers are insignificant. The tabulated data infers that Method 2 consumes about 3.5 times less power than Method 1.

Area (cell wise) of Method 2 is 1.44 times less than Method 1. The cell area is measured in terms of the technology used.

Synopsys Tools Used: Design Vision, IC Compiler with 90 nm technology

Operating conditions of circuit in virtual environment: Global operating voltage range is 1~1.2 V.

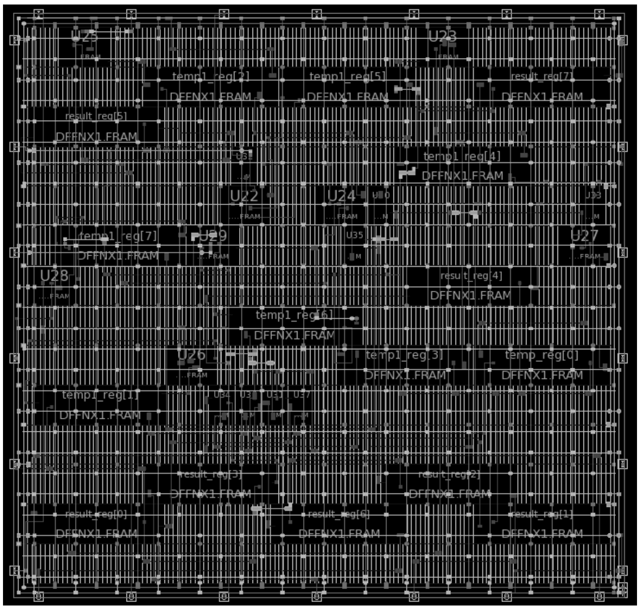

Figure 4: Physical layout of Method 1 from ICC tool

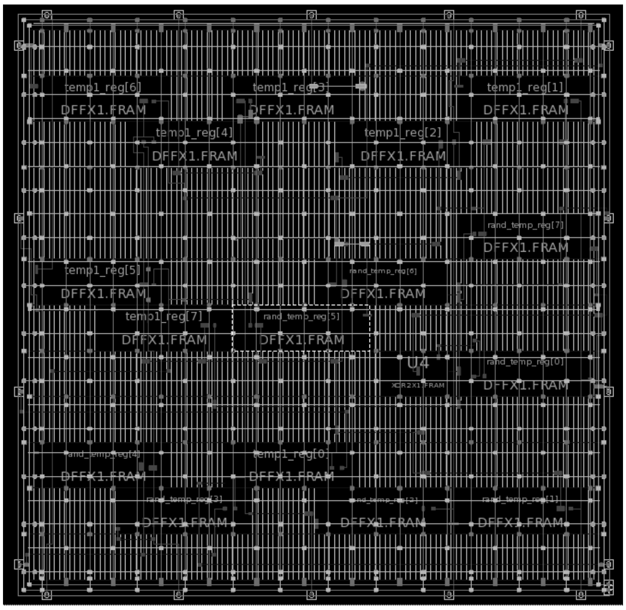

Figure 5: Physical layout of Method 2 from ICC tool

V. Conclusions

Over about half of the bits in the cover image will be modified when embedding the secret image as sometimes the LSB of cover image would be same as the secret bit replacing it. RTL analysis of Methods 1 and 2 have been executed; their area and power reports have been generated and compared. During the literature survey there is a visible trend of implementing steganographic algorithms in software rather than partitioning it in hardware. Not all algorithms presented in the papers can be used for implementing in hardware, due to excess computation time in embedding data in pixels. Method 2 is proven to be more effective in steganographic implementation than Method 1 in both power and area significantly. Method 2 would be an effective IP to be embedded in a processor memory as an added functional sub-unit rather than partitioning it as standalone hardware. Furthermore it will be very insightful to perform such an analysis over a complex algorithm involving more data to be processed.

Appendix

Carrier Image: it is also termed as cover image. The carrier image is one in which a secret file is embedded. The carrier image can be chrome or a grayscale image.

Secret Image/Message: this digital file is one which has to be used stored in the carrier image preferably embedded in a sequential manner, where the secret file bits are kept in the lsb of every pixel in the carrier

Stego Image: termed after embedding secret file with carrier image

Pixel: the minutest controllable component of a picture represented on the screen

References

[1] Ajit Singh, Swati Malik ,"Securing Data by Using Cryptography with Steganography" , International Journal of Advanced Research in Computer Science and Software Engineering, *SES, BPS Mahila Vishwavidhyalaya* India,2013

[2] Bassam Jamil Mohd, Saed Abed and Thaier Al-Hayajneh, Sahel Alouneh, "FPGA Hardware of the LSB Steganography Method", Computer Engineering Department, Hashemite University, German-Jordan University, IEEE, 2012

[3] Alaa A. Jabbar Altaay, Shahrin bin Sahib, Mazdak Zamani, "An Introduction to Image Steganography Techniques", 2012 International Conference on Advanced Computer Science Applications and Technologies, Faculty of Information & Communication Technology, Universiti Teknikal Malaysia Melaka, Universiti Teknologi Malaysia, IEEE, 2012

[4] H.Y. Leung, L.M. Cheng, L.L. Cheng, Chi-Kwong Chan, "Hardware Realization of Steganographic Techniques", Intelligent Information Hiding and Multimedia Signal Processing, 2007. IIHMSP 2007. Third International Conference on (Volume:1), IEEE, Nov 2007

[5] Chi-Kwong Chan∗, L.M. Cheng, "Hiding data in images by simple LSB substitution", The journal for the Pattern Recognition Society, Pattern Recognition 37 (2004) 469 – 474,Department of Computer Engineering and Information Technology, City University of Hong Kong, Hong Kong, 2003

[6] Naveed Sherwani, Algorithms for VLSI physical design automation, Intel Corporation, 2002, pp. 3-7.