Journal of Scientific & Industrial Research
Vol. 73, November 2014, pp. 701-703

Layer Router for Grayscale Stego - A Hardware Architecture on FPGA and ASIC Platforms

S Rajagopalan[1]*, H N Upadhyay[1], R Ragavan[2], R Amirtharajan[1] and J B B Rayappan[1]
[1]School of Electrical & Electronics Engineering, SASTRA University, India
[2]Electrical Engineering, Linkoping University, Sweden

Received 18 July 2012; revised 24 August 2013; accepted 11 June 2014

In the present era of secret communication, steganography has obtained a significant place in information security by means of offering variety of techniques for cleverly hiding the information. In addition to the existing hardware stego algorithms, an adaptive block hardware stego system has been proposed in this paper which follows a shortest path algorithm for performing secret concealment in grayscale images. The proposed image steganographic hardware architecture which adopts traversal procedures predominant in area routing has been implemented both in Stratix III FPGA as well as ASIC Platforms.

Keywords: Hardware Steganography, FPGA, ASIC, Information security.

Introduction

The need for information protection is growing day by day. With the invention of new electronic gadgets and modes of information sharing, the demand for secure communication is at its peak. At this stage one has to look towards the information security providers such as cryptography, steganography[1] and watermarking. Software oriented approaches towards steganography have been reported in a large number in the past [2,3,12]. However there have been some works presented on hardware platforms like FPGA [4,5] and firmware[13] .LSB substitution approach on FPGA (secret key) used Spartan FPGA with a random shuffler block to choose the address of SDRAM before embedding the secret data [6]. Another paper reported LFSR based information hiding[7] on Xilinx Virtex-II FPGA, where the address generation was carried out using a specific LFSR circuit and the stego algorithm was implemented on RGB images. A steganographic context approach has been proposed in another work[8]. Modified Hybrid stego-encryption algorithm[9], hardware LSB stego approach[10] and IWT based stego on Cyclone II FPGA[11] are also present in literature pertaining to hardware steganography.The proposed hardware stego approach follows a shortest path based data embedding. This method uses a twin rule for bit traversal where horizontal and vertical movement are only allowed and this rule is in line with layer routing of ASIC design. Also this approach adopts a data hiding procedure based on the shortest path connecting all the pixels of a 3×3 block. The uniqueness of this 3×3 block embedding algorithm is that there are 16 different possibilities for data embedding for a single block, which increases the complexity of the stego algorithm.

The rest of the paper contains the proposed methodology, FPGA and ASIC implementation results, Chip planner view, sample cover and stego images used in this implementation.

Proposed Methodology:

The suggested block stego approach uses 16 possible traversal paths for embedding the secret information in a grayscale image which has been divided into many 3 × 3 blocks.. The various paths used in this approach are given in Table 1. In the table, P0 to P15 represent the paths Path 0 to Path 15. L0 to L8 represent the pixel locations. For example, if path 7 has been chosen as the embedding path based on shortest path calculation, the data embedding pixel order will be 8 → 5 → 2 → 1 → 4 → 7 → 6 → 3 → 0. Here '0' represents the first pixel (L0) and '8' indicates the ninth pixel (L8) in a 3 × 3 block.The stego system was implemented by carrying out the following steps.

*Author for correspondence
E-mail:raman@ece.sastra.edu

Step1: A 3 × 3 grayscale image block was selected.
Step 2:16 path lengths were computed for the paths P0 to P15. (A path length can be computed by finding the sum of the differences between the pixel values of a 3×3 block).
Step 3: The shortest path was found based on sorting the 16 path lengths.
Step 4: The secret data bit/bits were embedded in the pixels of the 3×3 block following the traversal order of the shortest path.
Step 5: The process was repeated for all the 3×3 blocks of a grayscale image.

FPGA Implementation Results:

The FPGA Implementation of the proposed algorithm was carried out on Stratix III EP3SL340H1152C3 FPGA using Quartus II software. This adaptive stego algorithm was tested on various 90 × 90 grayscale cover images stored in internal RAM of the FPGA. The insystem memory content editor was used as a medium to read and write the required images on the internal RAM of FPGA. Cover images and secret data were imported into the internal RAM with memory initialization file .mif during the testing phase. Table 2 shows the Logic elements consumption by the proposed algorithm for various k-bit embedding where k =1,2,3 & 4.

Fig.1 shows the chip planner image which displays the footprint of the logic elements in stratix III FPGA.

Table 1—Pixel traversal order for 16 paths

Path	L0	L1	L2	L3	L4	L5	L6	L7	L8
P0	0	1	2	5	4	3	6	7	8
P1	0	3	6	7	4	1	2	5	8
P2	2	1	0	3	4	5	8	7	6
P3	2	5	8	7	4	1	0	3	6
P4	6	7	8	5	4	3	0	1	2
P5	6	3	0	1	4	7	8	5	2
P6	8	7	6	3	4	5	2	1	0
P7	8	5	2	1	4	7	6	3	0
P8	4	1	2	5	8	7	6	3	0
P9	4	1	0	3	6	7	8	4	2
P10	4	7	6	3	0	1	2	5	8
P11	4	7	8	5	2	1	0	3	6
P12	4	5	8	7	6	3	0	1	2
P13	4	5	2	1	0	3	6	7	8
P14	4	3	0	1	2	5	8	7	6
P15	4	3	6	7	8	5	2	1	0

Table 2—Hardware consumption on FPGA

k	Combinational ALUTs Used	Registers Used	Block Memory Bits Used	Logic Utilization (%)
1	13130	416	64800	5
2	13130	415	64800	5
3	13115	414	64800	5
4	13235	395	64800	5

ASIC Implementation Results:

The ASIC synthesis for this implementation was carried out on 45nm design library with Synopsys Design Compiler. The main steps in synthesis are as follows. Tcl scripts were written to carry out each step.

1 Read in the design, check for problems, specify target library for synthesis.
2 Specify timing related Constraints
3 Specify goals
4 Optimize the design and check for timing violations
5 Write out the netlist

Table 3 contains the area consumption in square micrometer and maximum possible clock frequency for a single 3×3 block embedding circuit.

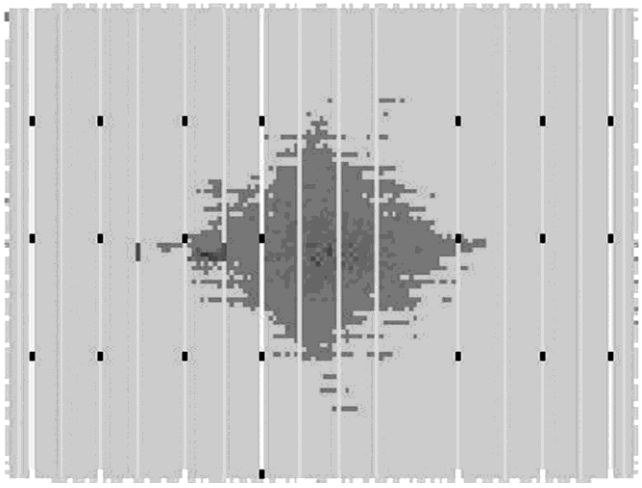

Fig.1—Chip planner results for data embedding on 90×90 grayscale image with proposed algorithm with k = 4

Table 3—ASIC implementation results

Clock Period (ns)	Area (sq. µm)	Maximum Clock Frequency(MHz)
10	2661	100.0
9.5	2661	105.2
9	2661	111.1
8.5	2660	117.6
8	2673	125.0
7.5	2694	133.3
7	2724	142.8

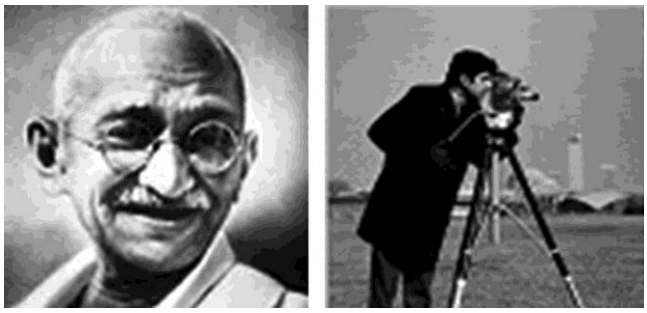

Fig.2—Cover Images (a) Mahathma Gandhiji (b) Cameraman

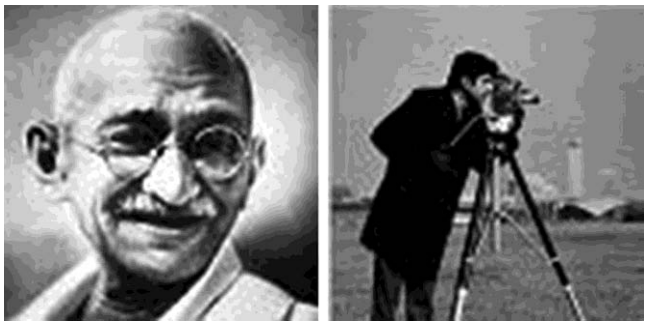

Fig.3—Stego images (a) Mahathma Gandhiji (k = 2) (b) Cameraman (k =1)

Cover & Stego Images:

Fig. 2(a) and 2(b) show the 90 × 90 grayscale cover images which were used for implementing the layer router algorithm and fig. 3(a) and 3(b) show the corresponding stego images.

Conclusion

A hardware steganographic system implemented on ASIC as well as FPGA platforms has been discussed in this paper. Our proposed algorithm is a pixel value dependent approach based hardware stego system on FPGA and ASIC when compared to other approaches for hardware stego systems proposed earlier on FPGA. The important advantage of this approach is that it is an adaptive steganographic system where the embedding process will be decided by the shortest route among the routes connecting every pixel of a 3×3 block. As an extension of this work, the MSE of the stego image can be greatly improved by selecting the traversal path for each 3×3 block in an image which results in good MSE.

Acknowledgement

The authors wish to express their sincere thanks to DRDO, New Delhi for their financial support (ERIP/ER/1003836/M/01/1230). They also wish to acknowledge SASTRA University, Thanjavur for extending infrastructural support to carry out the study.

References

1 Petitcolas F, Anderson R J & Kuhn M G, Information hiding – a survey, *Proc IEEE: Spl Issue Ident and Protect Mult Cont*, **87 (**1999) 1062–1078.

2 Cheddad A, Condell J, Curran K & Kevitt P M, Digital image steganography: Survey and analysis of current methods, *Signal Process,* **90** (2010) 727–752.

3 Chin-Feng Lee & Hsing-Ling Chen, A novel data hiding scheme based on modulus function, *J Syst Softw*, **83** (2010) 832–843.

4 Rajagopalan S, Amirtharajan R ,Upadhyay H N & Rayappan J B B, Survey and analysis of hardware cryptographic and steganographic systems on FPGA, *J Appl Sci*, **12** (2012) 201-210.

5 Leung H Y, Cheng L M, Cheng L L & Chi-Kwong Chan, Hardware Realization of Steganographic Techniques, *Third Int Conf Intl Inf Hiding and Mult Sig Process (IIHMSP 2007)*, **1** (2007) 279 – 282.

6 Farouk H & Saeb M, Design and implementation of a secret key steganographic micro-architecture employing FPGA, *Des Aut and Test Eur Conf and Exh* 2004, **3** (2004) 212- 217.

7 Mahmood A F, Kanai N A & Mohmmad S S, An FPGA Implementation of Secured Steganography Communication System, *Tikrit J Eng Sci*, **21 (1)** (2014) 1-9.

8 Gómez-Hernández E , Feregrino-Uribe C & Cumplido R, FPGA hardware architecture of the steganographic ConText technique, *Proc 18th Int Conf Elec Comm and Comp (CONIELECOMP 2008)*,(2008) 123-128.

9 Farouk H A & Saeb M, An Improved FPGA implementation of the Modified Hybrid Hiding Encryption Algorithm (MHHEA) For Data Communication Security, *Proc Des Aut and Test Eur Conf and Exh (DATE'05)*, (2005).

10 Jamil Mohd B, Abed S, Al-Hayajneh T & Alouneh S, FPGA Hardware of the LSB Steganography Method, *Int Conf Comp Inf and Telecomm Sys (CITS)*, (2012) 1- 4.

11 Balakrishnan R, Amirtharajan R & Rayappan J B B, Stego on FPGA: An IWT Approach, *Sci World J*, **2014** (2014).

12 Amirtharajan R & Rayappan J B B, An intelligent chaotic embedding approach to enhance stego-image quality, *Inform Sci*, **193** (2012) 115-124.

13 Janakiraman S, Amirtharajan R, Thenmozhi K & Rayappan J B B, Firmware for data security: A review, *Res J Inform Technol*, **4** (2012) 61-72.