ISSN XXXX XXXX © 2017 IJESC

Research Article Volume 7 Issue No.6

FPGA Implementation of Secret Data Sharing through Image by using LWT and LSB Steganography Technique

Harini.V[1], Vijayaraghavan.V. M.E (PhD) [2]
PG Scholar[1], Assistant Professor[2]
Department of Applied Electronics[1], Department of ECE[2]
Adithya Institute of Technology, Coimbatore, Tamil Nadu, India

Abstract:
Security of information is very important in terms of communication and/or the secrecy of how to decode it. The enhancement of security system for secret data communication through encrypted data embedding in colour images is proposed. Initially the cover image is converted to any one plane process and encrypted by using Chaos encryption. Adaptive LSB replacement algorithm is used for hiding the secret message bits into the encrypted image. In the secret data extraction module, the secret data will be extracted by utilizing significant key for choosing the image pixels to extract the data. This technique is particularly helpful in applications such as medical and military imaging. The proposed methodology provides better performance in terms of number of slices, number of IOBs. It is implemented in FPGA (field programmable gate array). The design architecture when implemented on FPGA Spartan III offers high processing speed, which might give an impulse for the researchers to a very fast, programmable & cost effective hardware solution in the area of Secure Communication.

Index terms: Data hiding, chaos encryption, Adaptive LSB replacement, Lifting Wavelet Transform, FPGA Spartan 3 EDK.

I. INTRODUCTION:

The last decade has witnessed the rapid development in information technologies and the wide availability of digital consumer device such as digital cameras, scanners etc .But at the same time this leads to the hacking vulnerability and duplicity of the original information. The most modern solution technique to this problem is digital steganography scheme .Digital steganography algorithms could be considered as digital communication scheme where auxiliary message is embedded in digital multimedia signal and are available where ever the later signals move. Steganography is the process that embeds data called a watermark, tag or label into a multimedia object such that watermark can be detected or extracted later to make an assertion about the object [1]. Watermarks of varying degree of visibility are added to presentation media as a guarantee of authenticity, quality, ownership and source. In general, any steganography scheme consists of three parts, such as the watermark, the encoder (insertion algorithm) and the decoder and comparator (verification or extraction or detection algorithm). The insertion algorithm incorporates the watermark into the object, whereas the verification algorithm authenticates the object, determining both the owner and the integrity of the object. The watermarks can be applied either in spatial or in frequency domain (FFT, DCT or wavelet) [2]. Even though spatial domain steganography is less robust, the spatial domain schemes have less computational overhead compared to frequency domain schemes. According to the human perception, the digital watermarks can be dividing into four different types, such as visible, invisible robust, invisible-fragile and dual. Each of the above steganography schemes is equally important due to its unique applications. In this work, we focus on VLSI implementation of an invisible-robust and an invisible-fragile spatial domain steganography algorithm. The VLSI chip can insert any one or both the watermarks depending on the requirements of the user. The proposed steganography chip can be easily incorporated as a module in any existing JPEG encoder and a secured JPEG encoder can be developed. We provide an outline of such a secure JPEG encoder. It may be noted that the corresponding watermark extraction module has to be inbuilt in a secure JPEG decoder. The secure JPEG codec can be a part of a scanner or a digital camera so that the digitized images are steganography right at the origin. In most of the algorithms designed based on the principle of data hiding, requires the sending original cover image along with the encoded cover image to the receiver. This approach makes the designed algorithm weaker as it conveys some idea of data hiding to the sender. But our method only the encrypted image will be sent to the receiver. The design of this technique is based on extensive analysis of the data-hiding process.

Digital Steganography Technique:

The Block diagram of Invisible steganography method using lifting method is shown in the figure. First create a header file for cover Image by using the matlab software. Cover Image header file is inputted to the Least Significant Bit adaptive LSB technique to embed the data in that cover image, After getting the steganography image apply the Lifting based DWT and chaos encryption to compress the image [3]. The above process will repeat in reverse process means getting decompression and adaptive LSB reverse method to get original image

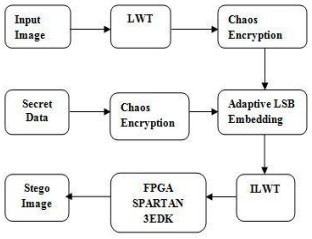

Figure.1. Block Diagram of Digital Steganography

Creation of Header file:
By using Matlab software to create the header file for Cover image and secret software.

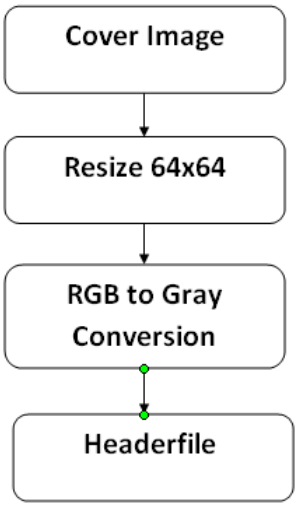

Figure.2. Header file Creation

Adaptive LSB Hiding Technique
Fig. shows the 1-bit adaptive LSB. In Fig. 1, the pixel value of the cover image is 141(10001101)2 and the secret data is 0. It applies to adaptive LSB-1 that the changed pixel value of the cover is 140(10001100)2. Adaptive LSB can store 1-bit in each pixel. If the cover image size is 64 x 64 pixel image, it can thus store a total amount of bytes of embedded data.

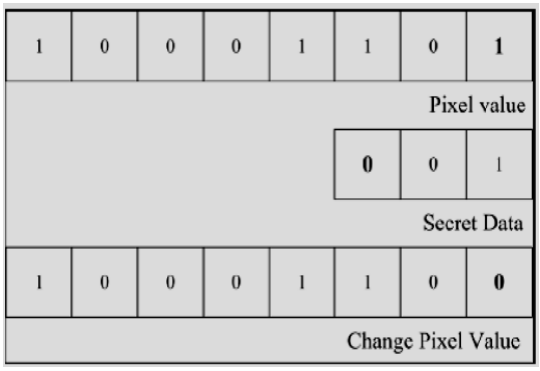

Figure.3.Adaptive LSB technique

Proposed method based on adaptive LSB technique, we propose a new steganography algorithm [4]. Most of researchers have proposed the first adaptive LSB and the third and forth adaptive LSB for hiding the data but our proposed steganography algorithm is using the third and fourth adaptive LSB for hiding the data. And using the RGB watermark image embedding in blue component of original image because of less sensitivity. This is because of the security reason. So, no one will expect that the hidden data in the third and the forth adaptive LSB. Fig. 2 shows the framework of the proposed method. First, we select the image which is a colour image and we will transfer the data to binary value after typing it. Then, we hide the data in the image using the proposed algorithm. Fig. 3 shows the embedding algorithm in VLSI.

Proposed Lifting Technique
Fig. I shows the classical implementation and the lifting based implementation of DWT. Classical implementation is realized by the convolution of the input signals with the low pass filter (h) and the high pass filter (hi). The lifting scheme is a new method to construct wavelet basis, which was first introduced by Sweden's.

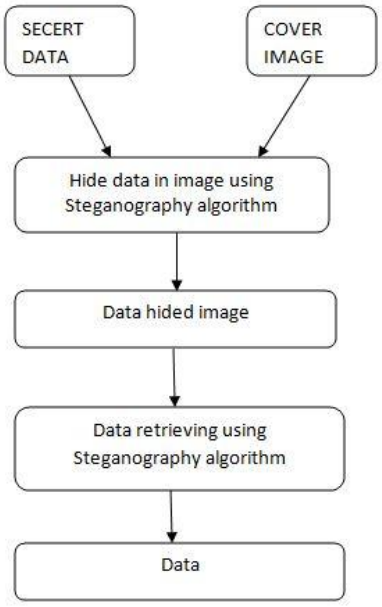

Figure.4. Steganography algorithm

The lifting scheme entirely relies on the spatial domain, has many advantages compared to filter bank structure, such as lower area, power consumption and computational complexity. The lifting scheme can be easily implemented by hardware due to its significantly reduced computations. Lifting has other advantages, such as "in-place" computation of the DWT [5]; integer-to-integer wavelet transforms which are useful for lossless coding. The lifting scheme has been developed as a flexible tool suitable for constructing the second generation wavelets. It is composed of three basic operation stages: split, predict and update. Fig.3. shows the lifting scheme of the wavelet filter computing one dimension signal. The three basic steps in Lifting based DWT are:

Split: where the signal is split into even and odd points, because the maximum correlation between adjacent pixels can be utilized for the next predict step. For each pair of given input samples x(n) split into even x(2n) and odd coefficients x(2n+1).

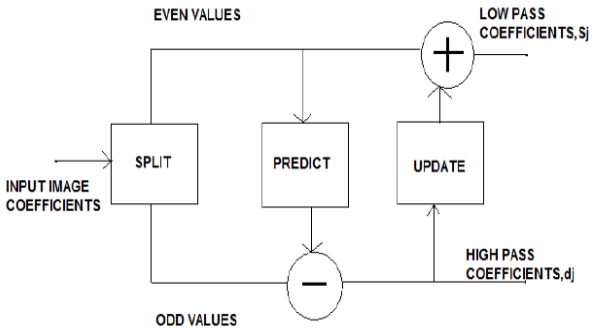

Figure.5. Block Diagram of Lifting technique

Predict: The even samples are multiplied by the predict factor and then the results are added to the odd samples to generate the detailed coefficients. Detailed coefficients results in high pass filtering.

Update: The detailed coefficients computed by the predict step are multiplied by the update factors and then the results are

added to the even samples to get the coarse coefficients .The coarser coefficients gives low pass filtered output. The inverse transform could easily be found by exchanging the sign of the predict step and the update step and apply all operations in reverse order as shown in Fig.4.The implementation of lifting based inverse transform (IDWT) is simple and it involves order of operations in DWT to be reversed. Hence the same resources can be reused to define a general programmable architecture for forward and inverse DWT.

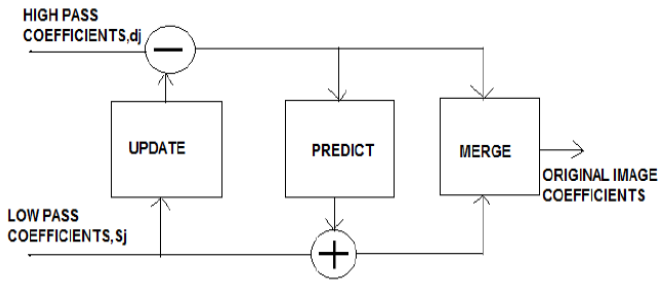

Figure.6. Block diagram of Inverse Lifting technique

Xilinx Platform Studio

The Xilinx Platform Studio (XPS) is the development environment or GUI used for designing the hardware portion of your embedded processor system. B. Embedded Development Kit Xilinx Embedded Development Kit (EDK) is an integrated software tool suite for developing embedded systems with Xilinx MicroBlaze and PowerPC CPUs. EDK includes a variety of tools and applications to assist the designer to develop an embedded system right from the hardware creation to final implementation of the system on an FPGA. System design consists of the creation of the hardware and software components of the embedded processor system and the creation of a verification component is optional. A typical embedded system design project involves: hardware platform creation, hardware platform verification (simulation), software platform creation, software application creation, and software verification. Base System Builder is the wizard that is used to automatically generate a hardware platform according to the user specifications that is defamed by the MHS (Microprocessor Hardware Specification) file. The MHS file defines the system architecture, peripherals and embedded processors].

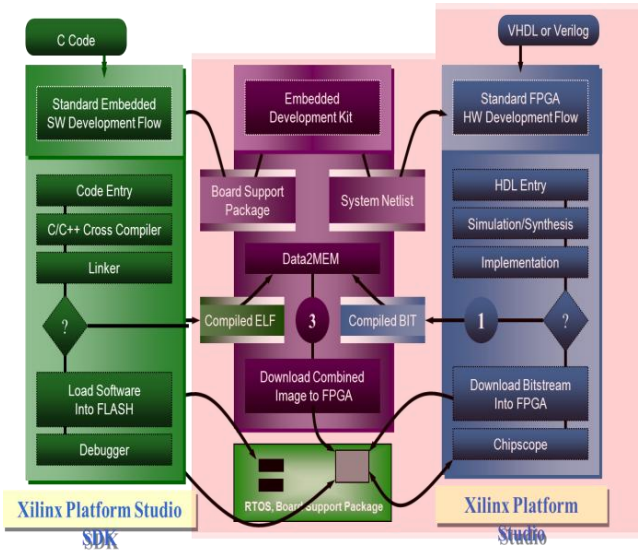

Figure.7. Embedded Development Kit Design Flow

The Platform Generation tool creates the hardware platform using the MHS file as input. The software platform is defamed by MSS (Microprocessor Software Specification) file which defines driver and library customization parameters for peripherals, processor customization parameters, standard 110 devices, interrupt handler routines, and other software related routines. The MSS file is an input to the Library Generator tool for customization of drivers, libraries and interrupts handlers. The creation of the verification platform is optional and is based on the hardware platform. The MHS file is taken as an input by the Simgen tool to create simulation files for a specific simulator. Three types of simulation models can be generated by the Simgen tool: behavioural, structural and timing models. Some other useful tools available in EDK are Platform Studio which provides the GUI for creating the MHS and MSS files. Create / Import IP Wizard which allows the creation of the designer's own peripheral and import them into EDK projects. Platform Generator customizes and generates the processor system in the form of hardware net lists. Library Generator tool configures libraries, device drivers, file systems and interrupt handlers for embedded processor system. Bit stream Initializer tool initializes the instruction memory of processors on the FPGA shown in figure2. GNU Compiler tools are used for compiling and linking application executables for each processor in the system [6]. There are two options available for debugging the application created using EDK namely: Xilinx Microprocessor Debug (XMD) for debugging the application software using a Microprocessor Debug Module (MDM) in the embedded processor system, and Software Debugger that invokes the software debugger corresponding to the compiler being used for the processor. C. Software Development Kit Xilinx Platform Studio Software Development Kit (SDK) is an integrated development environment, complimentary to XPS, that is used for C/C++ embedded software application creation and verification. SDK is built on the Eclipse open source framework. Soft Development Kit (SDK) is a suite of tools that enables you to design a software application for selected Soft IP Cores in the Xilinx Embedded Development Kit (EDK).The software application can be written in a "C or C++" then the complete embedded processor system for user application will be completed, else debug & download the bit file into FPGA. Then FPGA behaves like processor implemented on it in a Xilinx Field Programmable Gate Array (FPGA) device.

II. RESULTS:

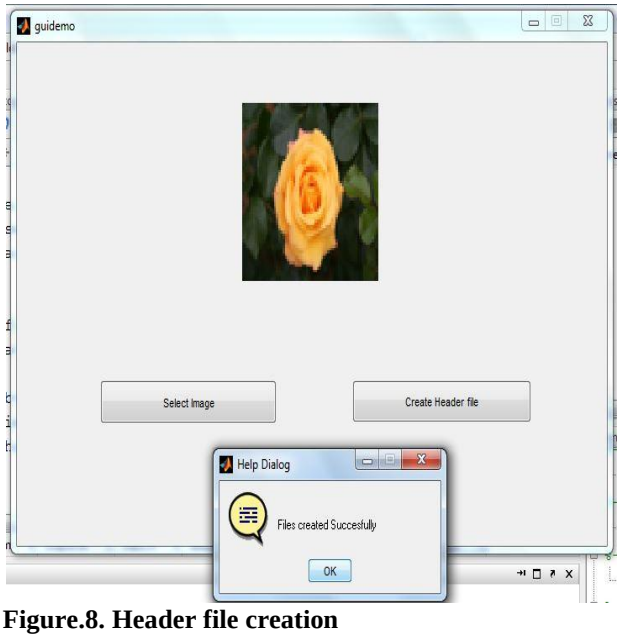

Figure.8. Header file creation

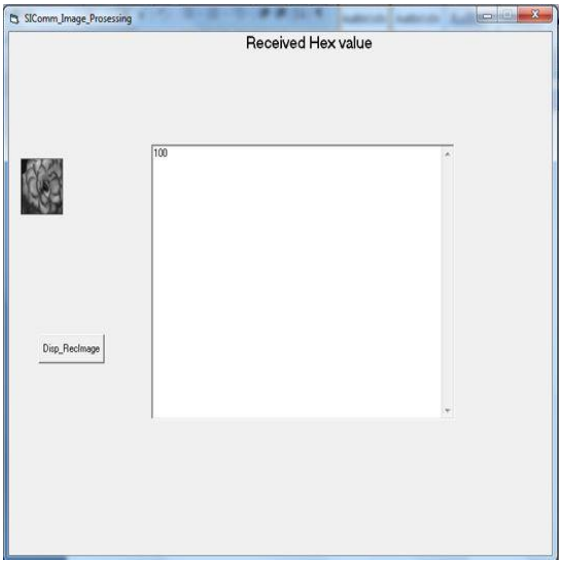

Figure.9. Cover Image

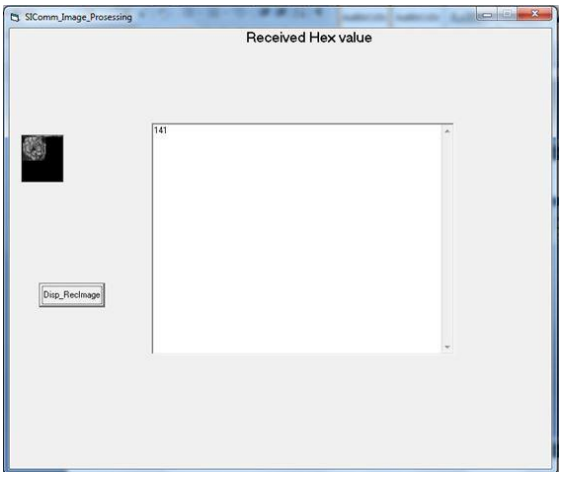

Figure.10. Lifting Image

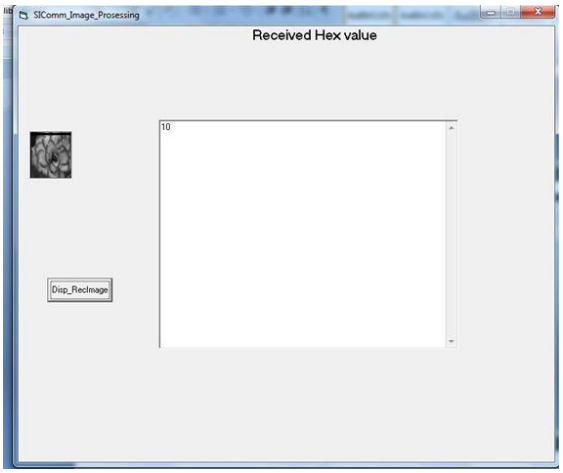

Figure.11. Decompressed Cover Image

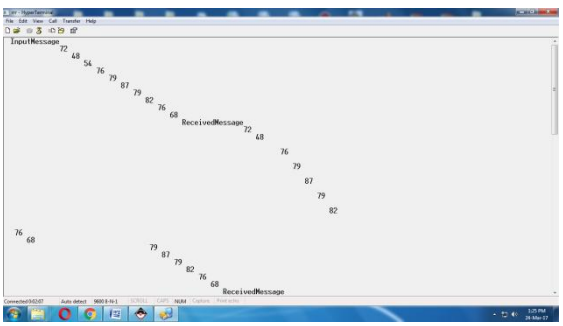

Figure.12. Secret Image Data

```
Selected Device : 3s500efg320-4

 Number of Slices:                     2649  out of   4656    56%
 Number of Slice Flip Flops:           3343  out of   9312    35%
 Number of 4 input LUTs:               3794  out of   9312    40%
    Number used as logic:              3118
    Number used as Shift registers:     356
    Number used as RAMs:                320
 Number of IOs:                          83
 Number of bonded IOBs:                  40  out of    232    17%
    IOB Flip Flops:                      55
 Number of BRAMs:                         7  out of     20    35%
 Number of MULT18X18SIOs:                 3  out of     20    15%
 Number of GCLKs:                         7  out of     24    29%
 Number of DCMs:                          2  out of      4    50%

Timing Summary:
---------------
Speed Grade: -4

   Minimum period: 12.384ns (Maximum Frequency: 80.749MHz)
   Minimum input arrival time before clock: 41.553ns
   Maximum output required time after clock: 13.840ns
   Maximum combinational path delay: 3.344ns
```

Figure.13. Synthesis report

III. CONCLUSION:

In this Paper the high speed low power invisible steganography technique is implemented by using discrete wavelet transform technique to the secret image to get the better results. The hardware implementation of this digital steganography could be significant in the copyright networks for the processing of the secret network based images. The related work for this implementation could be recognized over the processing of this adaptive LSB technique for the tamper proofing also.

IV. FUTURE SCOPE:

In future we can implement this steganography technique on Video Streaming Smart devices

V. REFERENCES:

[1]. S. Bhattacharyya,. "A survey of steganography and steganalysis technique in image, text, audio and video as cover carrier." Journal of global research in computer science 2, no. 4 (2011).

[2]. S. Saejung, A. Boondee, J. Preechasuk, and C. Chantrapornchai, "On the comparison of digital image steganography algorithm based on DCT and wavelet," in Computer Science and Engineering Conference (ICSEC), 2013 International, 2013, pp. 328–333.

[3]. M. Tayel, H. Shawky and A. E. S. Hafez, "A New Chaos Steganography Algorithm for Hiding Multimedia Data," 14th International Conference on Advanced Communication Technology, pp. 208 – 212, 2012.

[4]. N. Sathisha, G. N. Madhusudan, S. Bharathesh, K. B. Suresh, K. B. Raja and K. R. Venugopal, "Chaos based Spatial Domain Steganography using MSB", International Conference on Industrial and Information Systems(ICIIS), pp. 177-182, 2010.

[5]. N. Raftari and A.-M. E. Moghadam, "Digital Image Steganography Based on Assignment Algorithm and Combination of DCT-IWT," in 2012 Fourth International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN), 2012, pp. 295–300.