Chapter 7
Design of Reconfigurable Architectures for Steganography System

Sathish Shet
JSS Academy of Technological Education, India

A. R. Aswath
Dayananda Sagar College of Engineering, India

M. C. Hanumantharaju
BMS Institute of Technology and Management, India

Xiao-Zhi Gao
Aalto University School of Electrical Engineering, Finland

ABSTRACT
The most crucial task in real-time processing of image or video steganography algorithms is to reduce the computational delay and increase the throughput of a steganography embedding and extraction system. This problem is effectively addressed by implementing steganography hiding and extraction methods in reconfigurable hardware. This chapter presents a new high-speed reconfigurable architectures that have been designed for Least Significant Bit (LSB) and multi-bit based image steganography algorithm that suits Field Programmable Gate Arrays (FPGAs) or Application Specific Integrated Circuits (ASIC) implementation. Typical architectures of LSB steganography comprises secret message length finder, message hider, extractor, etc. The architectures may be realized either by using traditional hardware description languages (HDL) such as VHDL or Verilog. The designed architectures are synthesizable in FPGAs since the modules are RTL compliant. Before the FPGA/ASIC implementation, it is convenient to validate the steganography system in software to verify the concepts intended to implement.

INTRODUCTION
Steganography is the art as well as the skill that deals with smacking secret data in an image, video, audio, etc. Steganography may be stated as the method of smacking a furtive information inside a larger one such that someone may not know the presence of concealed information (Kipper, 2003). The main difference between steganography and cryptography is that the steganography scheme hides the secret
message inside the cover medium whereas the cryptography encrypts the secret message before transmitting it to the destination. Also, the breaking of steganography is known as steganalysis and breaking of cryptography is labeled as cryptanalysis. In image steganography, secret information is hidden inside the lower bits of image pixels, such that Human Visual System (HVS) is unable to visualize the presence of it. The Steganography system deals with the implementation of embedding and extraction hardware that hides and extracts secret information from the cover image. The original archives is denoted as cover media. The secret message encapsulated in cover media is referred to as stego image. Hiding or extraction process is protected by secret code to protect it from hackers.

Hardware implementation of steganography algorithms can be classified into several categories based on the architectural approach used; these are General Purpose Processor (GPP), dedicated Digital Signal Processors (DSPs), ASICs and FPGAs implementations. The choice of a particular approach is based upon the flexibility, memory requirement, throughput and cost etc. The image steganography algorithms implementation on GPP possess a great deal of flexibility compared with other implementations. In the recent years, processing speed of GPP has been increased significantly. However, the technology has approached the upper limit. The GPP are limited in performance since their instruction sets do not seem to suit for fast processing of high resolution of image enhancement algorithms. In addition, the instructions in GPP are executed in sequence, eventually, the throughput of the system decreases. DSPs (Hines, 2004) have been employed for enhancement of images provides some improvement over the GPPs. Only marginal improvement has been achieved since parallelism and pipelining incorporated in the design are inadequate. The hardware implementation of image enhancement techniques has been strongly influenced by the evolution of Very Large Scale Integration (VLSI) technology. The complex computational tasks in image processing such as 2D convolution, filtering, smoothing and contrast stretching operations encountered in image enhancement techniques are well suited for VLSI implementations. In addition, VLSI based hardware implementation exploits pipelining and massive parallel processing operations, resulting in increased throughput of the image enhancement system.

The image steganography methods implemented in ASIC increases the performance compared with other realizations. However, ASIC implementations require a large time to market and initial investments are high. The ASIC designs are best suited for relatively high volume productions. In the recent years, FPGAs has become an attractive choice of solution for hardware implementation of image steganography algorithms, especially when high throughputs are the needs of the hour. In an image steganography application, for rapid prototyping of new algorithms developed and to dynamically reconfigure, FPGAs are the right choice. FPGAs provides an optimal blend of flexibility of a GPPs and high speed processing that can be achieved using ASICs. An image enhancement architecture design for FPGA/ASIC technology can exploit fully the data and I/O parallelism for image enhancement application. Further, FPGA implementation of image enhancement techniques is cost effective for low volume productions. Therefore, in this chapter, LSB based image steganography algorithms are developed and implemented in FPGAs.

Steganography algorithms are highly essential to secure the data from hackers. Embedding secret message in image or video requires comparatively less delay to ensure real-time results. However, the computation complexity and hence the delay increases with increase in the payload. Therefore, the hardware implementation of steganography algorithms is the best choice that is preferred for this type of application. Although numerous researchers have been proposed software based steganography solutions, throughput achieved from these schemes are inadequate. The top design module invokes several other sub modules developed in the form of tree structure. The test bench has been developed to pass the stimulus for the top design and analyze the output results. A Matlab program was written in order