Design And Implementation Of Audio Steganography On Fpga

Akash Mecwan[1], *Vijay Savani*[1*], *Viren Gajjar*[2]
[1]Assistant Professor, Institute of Technology, Nirma University
[2]MTech. Student at (VLSI Design), Nirma University

ABSTRACT

As the emergence of latest hacking programs and attention of hackers in hacking more and more data, the need arises to make an integrated circuit that can help hiding the message in such a form that no one can hack it easily. The audio Steganographer is a device or an integrated circuit that facilitates the user to hide a massage in the ongoing digital audio stream. It also ensures the purity of the audio upon playback. The integrated circuit consists of various blocks like Encoder/ Decoder, Encryption/Decryption, Pseudo Random Number generator, Register Control Unit etc. The Report talks on the design aspects of all the listed blocks in detail. The report also discusses the various protocol developed during the design phase. It is obvious that the digital audio stream will definitely be distorted by stegenography. The care has been taken that the audio stream is not damaged to an extent that it cannot be recovered back. The last part of the report discusses the verification of the design. The development of the test bench and the simulation results are also indicated.

Keywords: *Steganography, PRN, Register Control Logic*

***Author for correspondence** Email: vijay.savani@nirmauni.ac.in

INTRODUCTION

Steganography is an art of hiding the message. There are many ancient techniques available for the same. For the last many years the Steganography is carried out in the form of software only. Lots of algorithms like RSA or AES or DES are available for hiding the data or to encrypt the data. These kinds of algorithms are time consuming and the separate process of decryption is necessary to get the data back once after the data is received.

In the modern era hardware is taking the action back from the software. With the evident of VLSI technology it is easy to build the hardware which can replace the existing software. All the algorithms like RSA, AES or DES can be built using the VLSI circuits. The audio stegenograher presented in this paper is hardware encryption techniques. The whole new algorithm is developed to hide the message data in the outgoing audio stream.

Such kind of hardware is extremely fast and the output is original data in the first go itself. No need to decrypt the data once it is received. The basic requirement is to put the chip on both the transmitter side and receiver side. The single chip works as both. Many such chips with different algorithms can be developed in the similar way presented here. The following chapters of this paper throw light on the design aspect of the Audio Steganography.

System Block Diagram

The Audio Steganographer is basically design to take 8 Bit digital audio stream. This input is an 8–Bit parallel input to the system. The output of the system is also an 8 – Bit parallel output digital audio stream. As the basic aim is steganography, the message that to be embedding in the audio stream can be taken to the system using 8–Bit parallel message inputs. Figure 2.1.1 indicates the block diagram of audio steganographer. All the blocks are discussed in detail here.

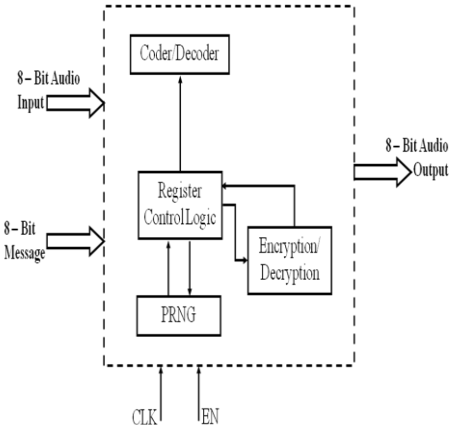

Fig. 1 Block Diagram

Coder/Decoder: The system can work as both transmitter as well as receiver. To facilitate the system as transmitter, this block will work as coder and to facilitate it as receiver it works as decoder. The coder/decoder block takes the 8 bit digital audio data serially from the register control unit. The coder, at the transmitter end, encodes the audio stream. The encoding operation is done using the LSB coding. The message to be transmitted is added in the LSB of the audio samples. The message bits are also received serially from the register control unit.

The encoder basically builds a frame using 7 bits of audio stream and 1 bit of message. The 1 message bit has to be placed in the LSB of the audio stream. The position of this LSB is decided by the register control unit that will be discussed in coming sections.

The coder/decoder unit works as decoder at the receiving end. The message data, that is transmitted in the LSB bits, is taken out using the exact the reverse process that of the transmitter. The content received is not at all changed but only the message bit is copied in the temporary storage and transmitted to the register control unit. The audio sample is also supplied serially to the register control unit. As the LSB of original audio is changed the audio will be disturbed, but as the LSB contains only finer details it is not observable to human ears upon playback.

The operation selection of coding or decoding can be done by the selection pin provided in the system. This selection is user friendly and user can select it as per the requirement.

Encryption/Decryption: The encryption/ decryption block provides an extra security to the message data. The basic security is already provided to the message as we transmit it with audio stream. The extra security is in terms of encryption is also provided by the steganographer.

Many encryption algorithms are available with various complexities. In this system, as the encryption is the added security feature, the system provides very easy encryption algorithm. The exact format of algorithm is discussed in the next section. The message data is supplied serially to the encryption block. At a time 8 bit data is processed. The data is encrypted using the key provided. This encrypted message is again sent back to the register control unit in serial fashion. This data is then provided to encoder to encode in the LSB of the audio stream.

At the receiver side the Encryption/Decryption block works for decryption of the received message. The received message data, after decoding from the decoder, is supplied to the register control unit. The RCU supplies it to the decryption block. As the cryptography process used is symmetric key algorithm, the exact reverse process that of the transmitter can provide the decrypted message back.

Pseudo Random Number Generator: The PRNG block is very important block of the system. As discussed earlier the message data has to be coded in the LSB of the audio stream. The system works on the 8 bit samples so last two bits of audio samples can be replaced by the encrypted message bits. The system replaces any one bit of the last two bits in a sample. The register control unit decides which bit is to be replaced. Either it replaces the last bit (7th bit) or it may replace the second last bit (6th bit) or it may not replace any of the bits and the audio sample goes as it is without any massage bit. This LSB selection is done with the help of PRNG block. PRNG block generates a pseudo random number which is fed to the RCU. Depending on the number fed by PRNG, the RCU decides the selection of LSB to encode the message data.

Register Control Unit: The RCU is the heart of the device. As discussed earlier every block is anyhow connected with RCU. The RCU is connected serially with PRNG, Encryption/decryption block and the Encoder/Decoder block also. The RCU consists of three 8 bit registers each of them operates in First in First Out mode for every bit.

Initially when the system starts the 8 bit audio stream is taken to RCU in parallel fashion. With this message data is also taken to the RCU in the similar manner. The audio stream is supplied to the encoder block at the transmitter end. Till then the audio is stored in the first 8 bit register. By the time the message stored is in the second 8 bit register of the RCU. RCU supplies it serially to encrypter. It also supplies the key to the encrypter. After encryption the data is taken back to the RCU and supplied to the encoder as the audio stream upgrades. With 8 samples of audio, 8 bit of message is encoded. So in every 64 bits of data one can send 8 bits of hidden message. The position of encoding is decided by the feed sequence provided by the PRNG block. The encoded audio stream is taken back to the RCU and stored in the third 8 bit register of the RCU. This stream is the data output of the system from transmitter end.

At the receiver, the encoded data comes to RCU in parallel fashion. The audio is stored in the first 8 bit register. Then the audio stream is supplied to the decoder block. The position of decoding is decided by the feed sequence provided by the PRNG block. Depending on this position the decoding of the data takes place. The decoded audio stream is taken back to the RCU and stored in the second 8 bit register of the RCU. Then it is supplied to the decryption block. After decryption the data is taken back to the RCU. From RCU 8 bit audio stream can be taken out. The decoded message is stored in the extra memory provided in the RCU upto 8 bits. To store the key for encryption and decryption extra memory is also provided in the RCU.

Test Bench

In the Audio Steganographer, the test bench is design in a very simple way. All the discussed parts are not checked individually rather the final input and output are checked directly. One top module file is designed for the test bench. This file contains the transmitter and the receiver program as a component in it. The main file also provides the input excitation to the transmitter component. These inputs are nothing but a predetermined data stored in a byte file. This file is read by the main module and the bytes are fetched from it. These bytes are sent to the transmitter block as inputs. The transmitter performs the steganography operation of the audio. The message data is also sent in the same manner. The output of the transmitter block is directly provided to the receiver module. The receiver is provided with only the synchronous clocks. Rest of the work is done by the receiver block it self. The outputs results are directly compared with the input byte file. This is the same data that was transmitted. If both of them match means the design is working perfectly. If they don’t match that means there is a bug inside.

RESULTS

This chapter is mainly concern with various results obtained in this system design. The first results are of the synthesis of the system. The system is synthesized FPGA. Then the simulation results are shown. These results are generated using the third party tool Modelsim. Test bench and all other codes are written using the Xilinx ISE 9.2.

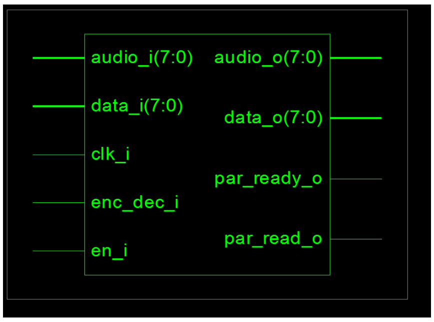

Fig. 2 RTL Schematic Audio Stegenographer

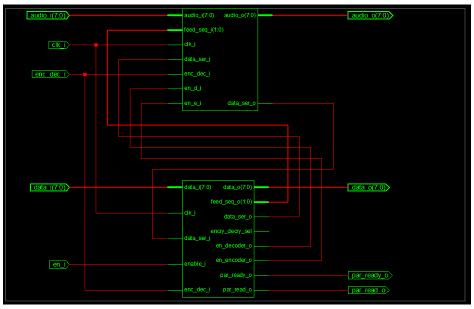

Fig. 3 RTL Schematic Encoder / Decoder

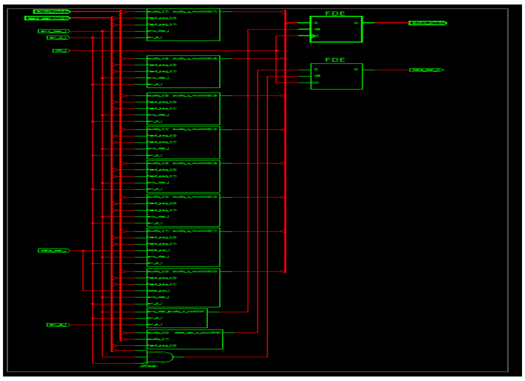

Fig. 4 RTL Schematic RCU

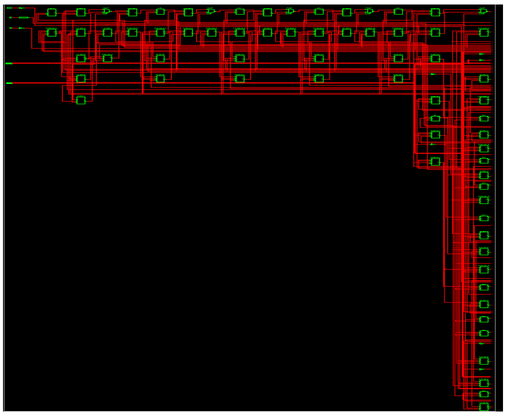

Fig. 5 Technology Schematic Audio Steganographer

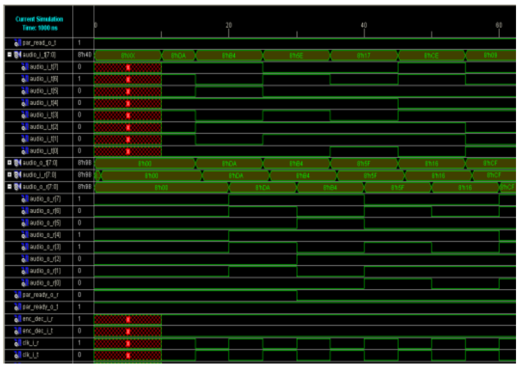

Fig. 6 Simulation Results for Test Bench

APPLICATION

- The basic application of this chip is to hide the data
- This device can be used to identify the copyright of the transmission. This means if a channel is transmitting a signal and wants that no other channel, which doesn't have right to transmit it, transmit after hacking it. So the authorized channel can put their hidden message in the audio using the chip. The unauthorized channel doesn't have the information about the hidden code. If they have the information then also it can not be decoded as the decoding logic and decryption key is not known. So such channel can easily be caught using such techniques.
- Using this system the single available analog channel can be used to transmit both the data as well as the analog information provided the analog information has very fine details in the LSBs.
- The information about the song which was previously not added in the analog signal can be added afterwards so that it can be read when playing song.
- The ability to hide the data of the system can be used to defense as well as other intelligence departments.
- Two signal can be sent in terms of TDM signals using this system, but in this case the speed of one channel will be very less as only one bit in a frame is transmitted.

Many other applications related to signal transmission or prevent data hacking is possible using this system.

FUTURE SCOPE

This system can support lots of modification. The first and the foremost modification to the system are to work with more number of bits. Just now the system is working with only 8 bits of the audio stream at a time. So it can be modified to work with 32 or 64 bits. As the number of bits in audio increase one can go with more LSB bits so instead of only 2 LSB bits 4 or 8 LSB bits can be replaced by the message data. Message data processing size can also be increased to 64 bits which is 8 bits only at present. If 64 bit message can be processed at a time then more robust encryption algorithm can be used. The user defined key can also be incorporated. The basic assumption of same baud rate of message and audio can also be dropped. This can make the system more flexible to work with. ADC and DAC block can be included in the design to facilitate directly analog inputs and outputs.

REFERENCES

1. http://www.jatit.org/volumes/research-papers/Vol5No6/15Vol5No6.pdf
2. http://www.castinc.com/cores/des3/index-.shtml
3. Adli & Nakao *Three Steganography Algorithms for MIDI Files* University of Ryukys, Okinawa, Japan. August 2005.
4. Artz & Donoavan *Digital Steganography: Hiding Data within Data IEEE* Internet Computing Magazine. June. 2001.
5. Mora F. & Torrubia A. *Perceptual Cryptography on MPEG Layer III Bit-Streams* 2000. 48. 4p.
6. Vasiltsov et al. *Development of VHDL-Based Core with Embedded Steganography Function* Lviv-Slasko, Ukraine. Feb 2003.
7. Jamil et al. *An Investigation into the Application of Linear Feedback Shift Registers for Steganography Proceedings IEEE* SoutheastCon. 2002.
8. Keija Z. *VLSI Implementation of High Performance RSA Algorithm Using Vedic Mathematics* Proceedings 5th International Conference on ASIC. 2003.
9. Gudu T. A *New Scalable Hardware Architecture for RSA Algorithm* International Conference on Field Programmable Logic and Applications. 2007.