**A Project Report**

**On**

“Enhancing Audio Security Through LSB-Based Steganography”

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

**Electronics and Communication Engineering**

by

GODESI VENKATA SAI TRINADH RAJU – RO200532

GUNDAVARAPU DURGA SIVANI – RO200364

SINGAMSETTI SWETHA – RO200071

AYINAPARTHI SAHITHI SUPRIYA – RO200888

KSHATHRI BHANU SAMYUKTHA – RO200290

**Under the Guidance of**

**Ms. G. Vijaya Kumari**

Assistant Professor

Department of Electronics and Communication engineering



**Approval Sheet**

This report entitled by “Enhancing Audio Security Through LSB-Based Steganography” by Godesi Venkata Sai Trinadh Raju-(RO200532), Gundavarapu Durga Sivani-(RO200364), Singamsetti Swetha-(RO200071), Ayinaparthi Sahithi Supriya -(RO200888) and Kshahtri Bhanu Samyuktha-(RO200290) is approved for the degree of Bachelor of Technology Electronics and Communication Engineering.

**Examiner(s):**

**Name of Examiner -1 : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Designation of Examiner -1 : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Signature of Examiner -1 : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Name of Examiner -2 : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Designation of Examiner -2 : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Signature of Examiner -2 : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Supervisor:**

**Name of Supervisor : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Designation of Supervisor : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Signature of Supervisor : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Date of Submission: \_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Place of Submission: \_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Stamp of the Department Mrs. N. Padmavathi**

Head of Department

Department of

Electronics and Communication

Engineering

**Candidate’s Declaration**

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included. I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission had not been taken when needed.

Name of the Candidate with ID Number Signature of the Candidate

Godesi Venkata Sai Trinadh Raju (RO200532) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Gundavarapu Durga Sivani (RO200364) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Singamsetti Swetha(RO200071) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ayinaparthi Sahithi Supriya(RO200888) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Kshathri Bhanu Samyuktha (RO200290) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Department of Electronics and Communication Engineering**

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

ANDHRA PRADESH, INDIA

ONGOLE CAMPUS

Kurnool Road, Ongole, Prakasam District, Andhra Pradesh – 523225

(AY 2024 – 2025)



**CERTIFICATE**

This is to certify that the mini project report entitled “Enhancing Audio Security Through LSB-Based Steganography” submitted by G. Venkata Sai Trinadh Raju, G. Durga Sivani, S. Swetha, A. Sahithi Supriya, K. Bhanu Samyuktha bearing ID numbers RO200532, RO200364, RO200071, RO200888 and RO200290 respectively in partial fulfilment of the requirements for the award of Bachelor Of Technology in Electronics And Communication Engineering is a bonafied work carried by them under my supervision and guidance.

Ms. G. Vijaya Kumari Mrs. N. Padmavathi

Assistant Professor Head Of Department

Department Of Department of

Electronics and Communication Electronics and Communication

Engineering Engineering

**Acknowledgement**

It is our privilege to express a profound sense of respect, gratitude and indebtedness to our guide **Ms. G. Vijaya Kumari, Assistant Professor,** Department. of Electronics and Communication Engineering, Rajiv Gandhi University of Knowledge Technologies – Ongole Campus for her guidance, technical and moral support and for her efforts in successful completion of our project.

We express our sincere gratitude to **Mrs. N. Padmavathi, Assistant Professor and Head of Department (i/c) Electronics and Communication Engineering**, Rajiv Gandhi University of Knowledge Technologies – Ongole Campus, for her suggestions and co-operation for the successful completion of the work.

We extend our sincere thanks to **Mr. M. RupasKumar, DeanAcademics**, Research and Development, Rajiv Gandhi University of Knowledge Technologies – Ongole Campus, for his constant help and support in our academic activities.

We extend our sincere thanks **Prof. Dr. Bhaskar Patel** , **Director**, Rajiv Gandhi University of Knowledge Technologies – Ongole Campus for his overall vision and guidance.

With Sincere Regards,

G. Venkata Sai Trinadh Raju (RO200532),

G. Durga Sivani (RO200364),

S. Swetha (RO200071),

A. Sahithi Supriya (RO200888),

K. Bhanu Samyuktha (RO200290).

**A Project Report on**

“Enhancing Audio Security Through LSB-Based Steganography”

**ABSTRACT**

This mini-project explores the use of **audio steganography** to securely hide and extract the secret messages within audio files. The project focuses on the implementing the **Least Significant Bit (LSB)** technique, a simple yet effective method where the least significant bits of audio samples are altered to embed secret data. This approach allows for the hiding of information without significantly altering the quality of audio file.

In the age of digital communication, data security and privacy have become crucial concerns. This project explores audio steganography as a means of securing sensitive information, focusing specifically on the **Least Significant Bit (LSB)** method. Audio steganography involves embedding data within audio files in a way that is imperceptible to the human ear. The LSB method modifies the least significant bits of audio samples to encode binary data, such as text or images, without significantly altering the audio quality. This approach provides a balance between security and simplicity, making it a suitable choice for hiding confidential data within commonly used audio formats. The project includes the development of an algorithm that reads an audio file, embeds the binary representation of the secret data into the audio samples, and reconstructs the modified audio. It also supports the extraction of hidden data from the modified audio file. This method can be applied in various fields, including secure communication and digital watermarking, offering a lightweight and effective solution for data concealment.

In which the least significant bit of every byte is altered by the bit- string representing the embedded file. This encryption and decryption of the images will be done using MATLAB code.

Keywords: Steganography, LSB Algorithm, Information hiding, MATLAB.

**TABLE OF CONTENTS**

Title Page i

Approval Sheet ii

Declaration Form iii

Certificate iv

Acknowledgment v

Abstract vi

1. Introduction 1
   1. Intro
   2. Main Objective
   3. Key Points
2. Literature Survey 3
3. Cryptography 4
   1. Introduction to Cryptography
   2. Terminology
   3. History
   4. Classic Cryptography
   5. Modern Crptography
   6. Symmetric Key Crptography
4. Steganography 8
   1. Introduction to Steganography
   2. Steganography Domains
   3. Steganography Carriers
   4. Information Hiding Requirements
      1. Transparency
      2. Capacity
      3. Robustness and security
      4. Complexity
   5. Information Hiding in Audio
      1. Temporal Domain
      2. Compressed Domain
   6. Interference
5. Audio Steganography In MATLAB 14
   1. Introduction
   2. LSB Encoding
   3. Standard LSB
   4. Modified LSB
   5. Proposed LSB Technique With Increased Capacity
      1. Technique And Algorithm
6. Experimental Details 20
   1. Description
   2. Read, Playback and Visualize and Audio Signal
   3. Bits Manipulation
7. Code Implementation 22
   1. MATLAB Code For Encryption
   2. MATLAB Code For Decryption
8. Results And Analysis 25
9. References. 26

**CHAPTER 1**

**INTRODUCTION**

**1.1 INTRO**

Information is shared universally through the Web, in computerized shape. There are issues and difficulties in regards to the security of information in travel from senders to collectors. The real issue is the assurance of computerized information against any type of interruption, entrance, and robbery. The significant test is building up an answer for ensure information and guarantee their security amid transmission. Three parts of information security are privacy, trustworthiness, and accessible. Privacy guarantees that information is kept mystery from any unapproved get to. This should be possible through information hiding systems, to be specific cryptography and steganography. Cryptography includes the demonstration of encryption and decryption of a computerized information. The significant shortcomings of such methods are that despite the fact that the message has been scrambled, regardless it exists. Steganography harps on disguising any computerized information in a harmless advanced transporter; the word steganography is gotten from an old Greek word which implies secured composing.

Steganography is worried about techniques for guaranteeing that mystery message is embedded (which can be serial number or a clandestine communication or a copyright check) in a cover message (like a sound chronicle or a video film, or significantly PC code). Parameterization of the embedding is finished by a key; without the learning of presence of this key. It is hard for an outsider to expel or identify the embedded material, when cover protest has material embedded in it, this is called stego question. For example, it may implant a content in a cover image or a stamp in a cover content to give or giving a stego-image or stego content, and soon.

**1.2 MAIN OBJECTIVE**

The main objective of an **audio steganography** project is to develop a technique for securely embedding secret information within an audio file in such a way that the presence of hidden data is imperceptible to the human ear. Audio steganography aims to provide a method for covert communication by encoding messages (such as text, images, or files) into an audio signal without altering its perceptible sound quality. This enables secure transmission of information, ensuring that only authorized recipients can extract and understand the hidden message, while remaining undetectable to unintended listeners. The challenge lies in embedding data in a way that maintains the audio’s natural properties, such as clarity, fidelity, and sound integrity, so that even sophisticated analysis does not reveal the presence of hidden data.

**1.3 KEY POINTS**

1. **Data Embedding**: The primary goal of audio steganography is to embed hidden information (text, images, or other files) within an audio file in a way that is undetectable to the human ear.
2. **Imperceptibility**: The embedded data must not alter the perceptible quality of the audio. It should remain inaudible or indistinguishable from the original audio to prevent detection.
3. **Robustness**: The steganographic method should be resilient to common audio transformations like compression, noise addition, or format changes. This ensures the hidden data remains intact even after manipulation.
4. **Security**: The hidden data should be protected from unauthorized access. Encryption techniques can be used to secure the message, ensuring only authorized parties can decode and retrieve the secret content.
5. **Payload Capacity**: The method should allow for a sufficient amount of data to be embedded without degrading audio quality. There is often a trade-off between the size of the payload and the quality of the audio.
6. **Extraction Process**: There should be a reliable and accurate method for extracting the hidden data from the audio file. This process must not distort the original audio and should only be executable by the intended recipient.
7. **Techniques Used: (**Least Significant Bit Encoding) Modifying the least significant bits of the audio signal to encode data

**CHAPTER 2**

**LITERATURE SURVEY**

Information is shared universally through the Web, in computerized shape. There are issues and difficulties in regards to the security of information in travel from senders to collectors. The real issue is the assurance of computerized information against any type of interruption, entrance, and robbery. The significant test is building up an answer for ensure information and guarantee their security amid transmission. Three parts of information security are privacy, trustworthiness, and accessible. Privacy guarantees that information is kept mystery from any unapproved get to. This should be possible through information hiding systems, to be specific cryptography and steganography. Cryptography includes the demonstration of encryption and decryption of a computerized information. The significant shortcomings of such methods are that despite the fact that the message has been scrambled, regardless it exists. Steganography harps on disguising any computerized information in a harmless advanced transporter; the word steganography is gotten from an old Greek word which implies secured composing.

Steganography is worried about techniques for guaranteeing that mystery message is embedded (which can be serial number or a clandestine communication or a copyright check) in a cover message (like a sound chronicle or a video film, or significantly PC code). Parameterization of the embedding is finished by a key; without the learning of presence of this key. It is hard for an outsider to expel or identify the embedded material, when cover protest has material embedded in it, this is called stego question. For example, it may implant a content in a cover image or a stamp in a cover content to give or giving a stego-image or stego content, and soon. In a stego framework that is impeccable, the stego image isn't being discernable from the first cover. A cover can without much of a stretch distinguish and after that perhaps remake the message. So as to maintain a strategic distance from coincidental reuse, both receiver and sender ought to wreck all spreads they as of now have utilized for exchange of information.

Electronic communication is liable to capture attempt and mediation, particularly amid the Information Age. With regards to issues of security and protection, a great many people's first idea would turn towards encryption. By and large, just the planned beneficiary would have the capacity to decode the message. The thought is that regardless of whether somebody caught a scrambled message, the message would be absolutely garbled. The field of cryptography is an all around created field sponsored by an orderly scientific establishment. Then again, sending an encoded message is a barefaced demonstrate that the message was intended to just be shared between particular gatherings. Steganography exhibits an approach to secretively exchange a message between planned gatherings with nobody else thinking about it. Steganography will enable somebody to shroud messages in harmless protests with a specific end goal to maintain a strategic distance from location. The capacity to conceal messages can be exceptionally significant in territories where an encoded message may draw undesirable consideration. Just as cryptanalysis is utilized to check cryptography, so too is steganalysis used to neutralize steganography. So as to build up any great security conspire, one must invest energy and exertion endeavoring to break said plot.

**CHAPTER 3**

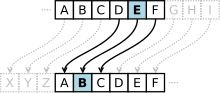
**CYPTOGRAPHY**

**3.1 INTRODUCTION TO CRYPTOGRAPHY**

Cryptography(from [Ancient Greek](https://en.wikipedia.org/wiki/Ancient_Greek_language): [romanized](https://en.wikipedia.org/wiki/Romanization_of_Ancient_Greek" \o "Romanization of Ancient Greek): kryptos "hidden, secret"; and graphein, "to write", or "study", respectively), is the practice and study of techniques for [secure communication](https://en.wikipedia.org/wiki/Secure_communication) in the presence of [adversarial](https://en.wikipedia.org/wiki/Adversary_(cryptography)) behavior. More generally, cryptography is about constructing and analyzing [protocols](https://en.wikipedia.org/wiki/Communication_protocol) that prevent third parties or the public from reading private messages. Modern cryptography exists at the intersection of the disciplines of mathematics, [computer science](https://en.wikipedia.org/wiki/Computer_science), [information security](https://en.wikipedia.org/wiki/Information_security), [electrical engineering](https://en.wikipedia.org/wiki/Electrical_engineering), [digital signal processing](https://en.wikipedia.org/wiki/Digital_signal_processing), physics, and others. Core concepts related to [information security](https://en.wikipedia.org/wiki/Information_security) ([data confidentiality](https://en.wikipedia.org/wiki/Confidentiality), [data integrity](https://en.wikipedia.org/wiki/Data_integrity), [authentication](https://en.wikipedia.org/wiki/Authentication), and [non-repudiation](https://en.wikipedia.org/wiki/Non-repudiation)) are also central to cryptography. Practical applications of cryptography include [electronic commerce](https://en.wikipedia.org/wiki/Electronic_commerce), [chip-based payment cards](https://en.wikipedia.org/wiki/Smart_card#EMV), [digital currencies](https://en.wikipedia.org/wiki/Digital_currencies), [computer passwords](https://en.wikipedia.org/wiki/Password), and [military communications](https://en.wikipedia.org/wiki/Military_communications).

The growth of cryptographic technology has raised [a number of legal issues](https://en.wikipedia.org/wiki/Cryptography_law) in the [Information Age](https://en.wikipedia.org/wiki/Information_Age). Cryptography's potential for use as a tool for espionage and [sedition](https://en.wikipedia.org/wiki/Sedition) has led many governments to classify it as a weapon and to limit or even prohibit its use and export. In some jurisdictions where the use of cryptography is legal, laws permit investigators to [compel the disclosure](https://en.wikipedia.org/wiki/Key_disclosure_law) of [encryption keys](https://en.wikipedia.org/wiki/Key_(cryptography)) for documents relevant to an investigation.

**3.2 Terminology**

[](https://en.wikipedia.org/wiki/File:Caesar_cipher_left_shift_of_3.svg)**Fig.1**

Until modern times, cryptography referred almost exclusively to "encryption", which is the process of converting ordinary information (called [plaintext](https://en.wikipedia.org/wiki/Plaintext)) into an unintelligible form (called [ciphertext](https://en.wikipedia.org/wiki/Ciphertext)).[[13]](https://en.wikipedia.org/wiki/Cryptography#cite_note-Kahn-1967-13) Decryption is the reverse, in other words, moving from the unintelligible ciphertext back to plaintext. A [cipher](https://en.wikipedia.org/wiki/Cipher) (or cypher) is a pair of algorithms that carry out the encryption and the reversing decryption. The detailed operation of a cipher is controlled both by the algorithm and, in each instance, by a "key". The key is a secret (ideally known only to the communicants), usually a string of characters (ideally short so it can be remembered by the user), which is needed to decrypt the ciphertext. In formal mathematical terms, a "[cryptosystem](https://en.wikipedia.org/wiki/Cryptosystem)" is the ordered list of elements of finite possible plaintexts, finite possible cyphertexts, finite possible keys, and the encryption and decryption algorithms that correspond to each key. Keys are important both formally and in actual practice, as ciphers without variable keys can be trivially broken with only the knowledge of the cipher used and are therefore useless (or even counter-productive) for most purposes. Historically, ciphers were often used directly for encryption or decryption without additional procedures such as [authentication](https://en.wikipedia.org/wiki/Authentication) or integrity checks.

[Cryptanalysis](https://en.wikipedia.org/wiki/Cryptanalysis) is the term used for the study of methods for obtaining the meaning of encrypted information without access to the key normally required to do so; i.e., it is the study of how to "crack" encryption algorithms or their implementations.

Some use the terms "cryptography" and "cryptology" interchangeably in English,while others (including US military practice generally) use "cryptography" to refer specifically to the use and practice of cryptographic techniques and "cryptology" to refer to the combined study of cryptography and cryptanalysis. English is more flexible than several other languages in which "cryptology" (done by cryptologists) is always used in the second sense above advises that [steganography](https://en.wikipedia.org/wiki/Steganography) is sometimes included in cryptology.[[19]](https://en.wikipedia.org/wiki/Cryptography#cite_note-rfc2828-19)

The study of characteristics of languages that have some application in cryptography or cryptology (e.g. frequency data, letter combinations, universal patterns, etc.) is called [cryptolinguistics](https://en.wikipedia.org/w/index.php?title=Cryptolinguistics&action=edit&redlink=1" \o "Cryptolinguistics (page does not exist)). Cryptolingusitics is especially used in military intelligence applications for deciphering foreign communications.

**3.3 History**

Before the modern era, cryptography focused on message confidentiality (i.e., encryption)—conversion of [messages](https://en.wikipedia.org/wiki/Information) from a comprehensible form into an incomprehensible one and back again at the other end, rendering it unreadable by interceptors or eavesdroppers without secret knowledge (namely the key needed for decryption of that message). Encryption attempted to ensure [secrecy](https://en.wikipedia.org/wiki/Secrecy) in communications, such as those of [spies](https://en.wikipedia.org/wiki/Spy), military leaders, and diplomats. In recent decades, the field has expanded beyond confidentiality concerns to include techniques for message integrity checking, sender/receiver identity authentication, [digital signatures](https://en.wikipedia.org/wiki/Digital_signature), [interactive proofs](https://en.wikipedia.org/wiki/Interactive_proof_system) and [secure computation](https://en.wikipedia.org/wiki/Secure_multiparty_computation), among others.

**3.4 Classic cryptography**

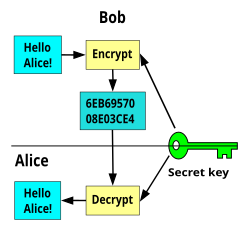
The main classical cipher types are [transposition ciphers](https://en.wikipedia.org/wiki/Transposition_cipher), which rearrange the order of letters in a message (e.g., 'hello world' becomes 'ehlol owrdl' in a trivially simple rearrangement scheme), and [substitution ciphers](https://en.wikipedia.org/wiki/Substitution_cipher), which systematically replace letters or groups of letters with other letters or groups of letters (e.g., 'fly at once' becomes 'gmz bu podf' by replacing each letter with the one following it in the [Latin alphabet](https://en.wikipedia.org/wiki/Latin_alphabet)).[[22]](https://en.wikipedia.org/wiki/Cryptography#cite_note-22) Simple versions of either have never offered much confidentiality from enterprising opponents. An early substitution cipher was the [Caesar cipher](https://en.wikipedia.org/wiki/Caesar_cipher), in which each letter in the plaintext was replaced by a letter three positions further down the alphabet.[[23]](https://en.wikipedia.org/wiki/Cryptography#cite_note-23) [Suetonius](https://en.wikipedia.org/wiki/Suetonius) reports that [Julius Caesar](https://en.wikipedia.org/wiki/Julius_Caesar) used it with a shift of three to communicate with his generals. [Atbash](https://en.wikipedia.org/wiki/Atbash) is an example of an early Hebrew cipher. The earliest known use of cryptography is some carved ciphertext on stone in [Egypt](https://en.wikipedia.org/wiki/Egypt) (c. 1900 BCE), but this may have been done for the amusement of literate observers rather than as a way of concealing information

**3.5 Modern cryptography**

Just as the development of digital computers and electronics helped in cryptanalysis, it made possible much more complex ciphers. Furthermore, computers allowed for the encryption of any kind of data representable in any binary format, unlike classical ciphers which only encrypted written language texts; this was new and significant. Computer use has thus supplanted linguistic cryptography, both for cipher design and cryptanalysis.

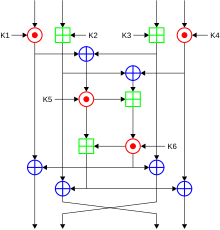
Many computer ciphers can be characterized by their operation on [binary](https://en.wikipedia.org/wiki/Binary_numeral_system) [bit](https://en.wikipedia.org/wiki/Bit) sequences (sometimes in groups or blocks), unlike classical and mechanical schemes, which generally manipulate traditional characters (i.e., letters and digits) directly. However, computers have also assisted cryptanalysis, which has compensated to some extent for increased cipher complexity. Nonetheless, good modern ciphers have stayed ahead of cryptanalysis; it is typically the case that use of a quality cipher is very efficient (i.e., fast and requiring few resources, such as memory or CPU capability), while breaking it requires an effort many orders of magnitude larger, and vastly larger than that required for any classical cipher, making cryptanalysis so inefficient and impractical as to be effectively impossible.

**3.6 Symmetric-key cryptography**

[](https://en.wikipedia.org/wiki/File:Symmetric_key_encryption.svg)

**Fig.2: Symmetric-key cryptography**

Symmetric-key cryptography refers to encryption methods in which both the sender and receiver share the same key (or, less commonly, in which their keys are different, but related in an easily computable way). This was the only kind of encryption publicly known until June 1976.

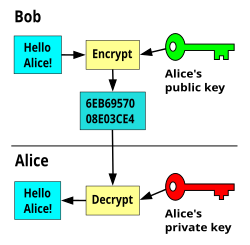
[](https://en.wikipedia.org/wiki/File:International_Data_Encryption_Algorithm_InfoBox_Diagram.svg)

**Fig.3: Time-efficient encryption of messages**

Symmetric key ciphers are implemented as either [block ciphers](https://en.wikipedia.org/wiki/Block_ciphers) or [stream ciphers](https://en.wikipedia.org/wiki/Stream_ciphers). A block cipher enciphers input in blocks of plaintext as opposed to individual characters, the input form used by a stream cipher.

The [Data Encryption Standard](https://en.wikipedia.org/wiki/Data_Encryption_Standard) (DES) and the [Advanced Encryption Standard](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard) (AES) are block cipher designs that have been designated [cryptography standards](https://en.wikipedia.org/wiki/Cryptography_standards) by the US government (though DES's designation was finally withdrawn after the AES was adopted). Despite its deprecation as an official standard, DES (especially its still-approved and much more secure [triple-DES](https://en.wikipedia.org/wiki/Triple-DES) variant) remains quite popular; it is used across a wide range of applications, from ATM encryption[[45]](https://en.wikipedia.org/wiki/Cryptography#cite_note-45) to [e-mail privacy](https://en.wikipedia.org/wiki/E-mail_privacy)and [secure remote access](https://en.wikipedia.org/wiki/Secure_Shell). Many other block ciphers have been designed and released, with considerable variation in quality. Many, even some designed by capable practitioners, have been thoroughly broken, such as [FEAL](https://en.wikipedia.org/wiki/FEAL).

**3.7 Asymmetric cryptography**

[](https://en.wikipedia.org/wiki/File:Public_key_encryption.svg)

**Fig.4: Public-key cryptography, where different keys are used for encryption and decryption.**

Symmetric-key cryptosystems use the same key for encryption and decryption of a message, although a message or group of messages can have a different key than others. A significant disadvantage of symmetric ciphers is the [key management](https://en.wikipedia.org/wiki/Key_management) necessary to use them securely. Each distinct pair of communicating parties must, ideally, share a different key, and perhaps for each ciphertext exchanged as well. The number of keys required increases as the [square](https://en.wikipedia.org/wiki/Square_(algebra)) of the number of network members, which very quickly requires complex key management schemes to keep them all consistent and secret.

**Cryptography use cases:**

* Secure communication
* Data encryption
* Data integrity
* Authentication
* Non-repudiation
* Key exchange
* Securing API communication

**CHAPTER 4**

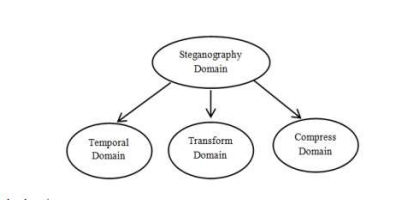
**STEGANOGRAPHY**

**4.1 INTRODUCTION TO STEGANOGRAPHY:**

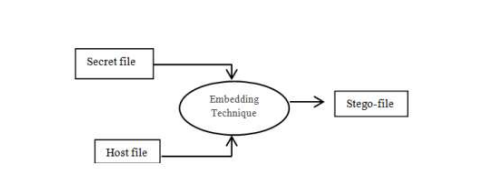
Steganography is the schema of concealing confidential data in a cover file like image, audio or text so that no one other than sender and intended receiver is able to notice that secret data is hidden inside. Steganography is a word derived from the ancient Greek words steganos, which means covered and graphia, which in turn means writing. The earliest technique used for hiding important data was cryptography which has a similar protocol to steganography in protecting the data but there is a difference between them. The first scrambles the data so that anyone who gets the file can expect that there is something abnormal while the second hides the data in a way that can't be observed or even sense its presence. In steganography techniques, the sender hides the secret message into host file. This produced a stegofile then delivers it to the receiver that will process dehide the stego file to retrieve the secret message. The secret data and the host can be any of various file type like text, audio, image and video file. If the host file is an audio file then the method is called audio steganography. Embedding secret data in hostaudio file is more challenging than using images since Human Auditory System (HAS) is more sensitive in comparison to Human Visual System (HVS). On the other hand, in comparison to other types of files, audio file is larger in size than other carrier’s file, high level of redundancy and high data transmission rate, the facts that make it more suitable as host file. Thus, this study focuses on audio Steganography and gives a wide view oftrends in this field.

**4.2 STEGANOGRAPHY DOMAINS:**

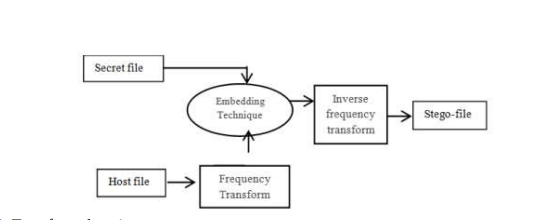
Hiding data can be classified into three domains according to which the steganography technique has been applied: Temporal, Transform and Compressed domain as shown in Fig. 1. In temporal domain, the secret data is hidden directly into host file in which the steganography techniques are simple and easy to implement. However this domain suffers from low robustness and security. The earliest algorithm employed in such domain is LSB which is used in embedding process. This method hides the bits of secret data directly into the least significant bits of the cover file. Although this method has high embedding capacity and easily to implement, it has low robustness and the attacker can easily recover the secret message by collecting the entire LSB bits. Many techniques try to combine temporal steganography with other methods to enhance the robustness. However, they have some drawbacks like less security and sensitivity to compression. The other domain used in data hiding is transform domain. In this domain, the cover file is transformed first , and then the secret data is embedded into the transform coefficients. This enables steganography system to embed the data into perceptual significant components and makes it difficult to recover the embedded data. This will offer high level of security and robustness against signal manipulation like amplification and filtering. On the other hand, the hidden data suffer from data compression so the retrieved secret data may not be accurate. The most common transforms used in steganography are Wavelet transform (WT), Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT). In compressed domain, varieties of techniques are developed. In this domain cover data or secret data is compressed using different compression techniques to develop steganography techniques and produce high capacity and compression ratio. Vector Quantization (VQ) is one technique that used to hide secret data in compressed cover file. Fractal Compression (FC) is another technique that compresses the secret data before hidden in cover file. Those are the most common compression techniques that are used**.**



**Fig.5: Steganography domain**



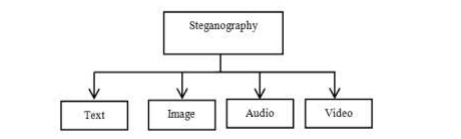
**Fig. 6: Temporal domain**



**Fig. 7: Transform domain**

**4.3 STEGANOGRAPHY CARRIERS**

Steganography is used to hide fidelity information in unsecure channels so it has to transmit and receive safely to the authority part using a medium that is called carrier or cover file. In this era, many types of digital files are used to protect the secret data however multimedia files were frequently used because diffusion around the internet and thousands of them is shared daily between users. In addition, data can also be hidden or embedded using network protocols (Bandyopadhyay et al., 2008). This section exhibits a review of the carriers that are used as a cover file for embedding secret data in various steganography techniques which is generally classified into four categories (text, image, video and audio).



**Fig.8: Steganography Carriers**

**Text:** The process of encoding secret message into text file can be applied in different ways using the properties of the sentences like altering format text or depending on the number of words. It is the earliest method that is used in steganography. This type of steganography is difficult to implement with secret data of huge size due to low redundancy of the information in the text file and thereby low hiding capacity.

**Image:** The most common technique that is used in steganography is image steganography. Image files spread through the Internet between users makes it perfect to use those files for hiding secret information in addition to the low sensitivity toward Human Visibility System (HVS) and the redundancy of information inside the image file. The hidden process is done by slight alteration of the visible properties of the image file in order to reduce the awareness of the presence of the secret data. The earliest methods that were used in embedding involved using the least significant bit or LSB, echo hiding and spread spectrum. These techniques can be achieved with disparate degrees of success on different kinds of image files.

**Video:** Video is generally a combination of audios and images. Therefore, most of image steganography techniques can be applied to video files. A large amount of information can be concealed into the video file and changes might not be observed because of continuous flow of information. This is the main advantage of using video steganography.

**Audio:** Audio steganography approach uses the audio file as a host (cover) file in embedding which is not simple and can be considered as a challenge due to the sensitivity of Human Auditory System (HAS). HAS senses the variation of audio file over a range of power greater than one billion to one and range of frequencies greater than one thousand to one. However, it has holes and a large dynamic range can contribute to data hiding. Audio files make convenient host file for hiding because of its high level of redundancy and high data transmission rate in addition to the large size in comparison with other multimedia files. Several techniques are discussed later.

**4.4 INFORMATION HIDING REQUIREMENTS**

Various information hiding algorithms were proposed for different purposes and it is necessary to measure the efficiency of them. The efficiency depends on several standards which is called requirements. There is a set of certain and essential requirements have to be fulfilled in most of algorithms. These are transparency, capacity, robustness and complexity of algorithm. The hiding capacity is the most significant characteristic in the steganography followed by the transparency and security while robustness has more important role in watermarking. These are related to each other. For instance, increasing capacity will diminish the transparency and vice versa. A good steganographic system has to trade-off between these requirements and which is hard to achieve in one algorithm.

**4.4.1 Transparency:** Most of the data hiding techniques have to insert data as much as possible without affecting the perceptual degradation in quality of the host file. It is one of the most important factors in designing algorithms for hiding data. The fidelity of the steganography algorithm is usually known as a perceptual resemblance between the cover file and stego-cover. However, the differences should be with minimal levels. The evaluation of imperceptibility is usually based on an objective measure of quality or subjective test. Some steganography techniques can be categorized as methods that have high transparency such as that proposed that produce high quality stego object.

**4.4.2 Capacity:** The amount of information that information hiding scheme can successfully hide without introducing any perceptual distortion is the capacity. It represents the number of hidden bits according to the size of host cover. The difficulty lies in the way how to embed secret data as much as possible while preserving the quality of the host cover. It is measured in bits per pixel for images steganography and bits per second for audio steganography. Many algorithms were developed to improve the capacity of cover file.

**4.4.3 Robustness and security:** Robustness is defined as how hidden message should not be prone to elimination or modification while Security prevents unauthorized person from extracting the hidden data. There are two kind of attacks that may have effect on the stego-cover: unintentional attack that try to modify or destroy the stego-cover (such as compression, rotation, blurring, noising and other filtering techniques) and intentional attack that try to reveal the stego-cover and extract the hidden information. Usually there is a trade-off between robustness and capacity that can hardly be fulfilled together in the same steganographic system. The robustness is an important factor for copyright protection and watermarking applications, while imperceptibility and high hiding capacity is more significant for steganography applications because the goal is to hide as large amount of data with preserving the quality of the cover file.

**4.4.4 Complexity:** Hiding algorithms consumes time in performing hide/de-hide process and this depends on the complexities of the algorithm. Secret data should be rapidly embedded/extracted into/from the host file, so that streaming data hiding real time can be delivered over the network.

**4.5 INFORMATION HIDING IN AUDIO**

As mentioned in the introduction, many steganography techniques that used different digital files were proposed that used different digital file. This part will present several methods that are used in hiding secret data in audio cover file which fall under temporal domain, frequency and compressed domain that can be briefed as follow:

**4.5.1 Temporal domain:** LSB is the simplest and earliest approach in temporal domain that is used in hiding the secret information into the Least Significant Bit of the audio samples. However, it has demerits like low robustness and easy to recover. Therefore many researches have proposed algorithms to enhance the LSB technique to overcome these demerits. They proposed two methods Parity and XOR for hiding secret data (image, audio, text) in host audio file with encryption to add level of security. The perceptual quality of the stego signal is high while it should increase the security by using multiple LSB’s.

New schema for audio steganography is presented by Pathak which is based on diffusing the secret message over host audio. Selection position of secret bit from 0th to 7th LSB in host sample depends on the decimal value of 3 MSB’s . There is also a proposed technique that hides the secret data in dual bits in the 4thand 1th LSB and makes modification to the other bits on the same sample. Moreover a technique that used two bits (3th and 4th) for embedding and intelligent algorithm for altering the bits (2th and 5th) to minimize the difference between stego sample and original audio file.

Encryption also is used with LSB to enhance the security of the stego file, Khan and Asad used AES (Advanced Encryption Standard) approach to encrypt the secret message. However, there is limitation in this algorithm in selecting host file because the sample of host message should be eight times larger than the number of secret bits to ensure fully embedding and the weakness in this algorithm quality of sound depends on the size of selected audio and length of the message respectively.

Steganographic approach for hiding secret data into acoustic file based on analog modulation was presented by Shiu. It is based on translating secret data into digital form, which is transformed into a high frequency signal. This signal is above the threshold of human audibility. After that, it is integrated with public music signal. This approach has accomplishe high security during transmission and it is tested with many attack approaches like high-pass-filter (HPF), direct current (dc), re-quantization (8-16 bits), echo injection and random noise.

**4.5.2 Compressed domain:** In the above two domains, embedding secret data requires large size cover file relatively to the size of secret data and high bandwidth to transmit the cover file, therefore many compression techniques for data hiding have been proposed. The main objective for adopting compression algorithms is to increase the amount of the secret information that is transmitted to specific receiver. Generally, the data hiding adopts compression method in two ways: first hiding data in compress-cover file using VQ (Vector Quantization) and second hidden data into cover file after compress it using FC (Fractal Coding).

VQ is one of the popular compressed methods which were widely used with images. It generates a codebook to classify the secret data and use this codebook in reconstruction of the secret data without using cover file in the receiver side. However, shared codebook is needed in both sender and receiver sides.

FC is another method was used to compress multimedia files. The similarity in different parts of audio is exploited so that each block of the audio file can be represented by set of IFS coding and these sets are used to reconstruct the secret data. These sets are the affine transformation coefficients that are produced from mapping function between cover and secret blocks. These will be hidden in cover and used in reconstruction process in the receive side. It is widely used with images steganography. The main weakness of classical fractal coding scheme is the exhaustive mapping search which is time consuming so many developments were found to reduce the fractal time and increase compression.

In addition block indexing and moment descriptor in selection strategy instead of tradition search and filtering the cover blocks was another method proposed for reduction in the encoding time good quality of the reconstructed audio file.

**4.6 INFERENCE**

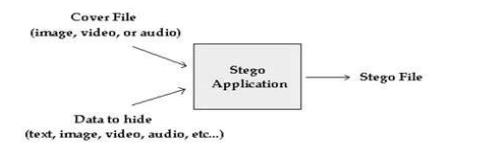
Audio steganography plays a vital role in transmitting secret and important information from the sender to the authorized receiver. The secret information is embedded slightly into host audio through changing the host file directly or transform it into another form in order to achieve optimum capacity and robustness. Several approaches are presented for hiding audio data signals and each of them has achieved a degree of success from a point view of steganography requirements. The techniques that were proposed in temporal domain provide easy and simple way to hide, although they are unable to tolerate the noise, less robust and few techniques have been developed at present. Frequency domain in embedding algorithm has given better result in signal processing, robustness and security. Techniques that adopt two different domains to utilize from the poses of these domains are continuously developing and they accomplish perfect steganography techniques. On the other hand, compressed techniques are also adopted by several researchers to increase the data to be hidden and develop algorithm that can be utilized in real time data hiding. Finally, steganography requirements are the dependable factors of decision making in selecting technique and the domain.

**CHAPTER 5**

**AUDIO STEGANOGRAPHY IN MATLAB**

**5.1 INTRODUCTION**

The scientific study in the open literature began in 1983 when Simmons stated the problem in terms of communication in a prison. In his formulation, two inmates Alice and Bob are trying to hatch an escape plan. The only way they can communicate with each other is through a public channel, which is carefully monitored by warden of the prison ward. If ward detects any encrypted messages or code, he will throw both Alice and Bob into solitary confinement. The problem of steganography is introduced then; how can Alice and Bob cook up an escape plan by communicating over the public channel in such a way that Ward doesn’t suspect “anything unusual” is going on. Notice, how the goal of steganography is different from classical cryptography, which is about hiding the content of secret message: steganography is about hiding the very existence of the secret message. Steganographic protocols have a long and intriguing history that goes back to antiquity. There are stories of secret messages written in invisible ink or hidden in love letters (the first character of each sentence can be used to spell a secret, for instance). More recently, steganography was used by prisoners and soldiers during World War II because all mails in Europe was carefully inspected at the time. Postal censors crossed out anything that looked like sensitive information (e.g. long strings of digits), and they prosecuted individuals whose mail seemed suspicious. In many cases, censors even randomly deleted innocent-looking sentences or entire paragraphs in order to prevent secret messages from going through. Over the last few years, steganography has been studied in the framework of computer science, and several algorithms have been developed to hide secret messages in innocent looking data.



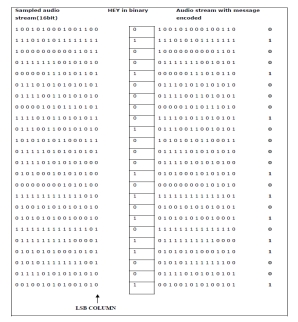
**Fig.9**

**5.2. LSB ENCODING:**

Least significant bit (LSB) coding is the simplest way to embed information in a digital audio file. By substituting the least significant bit of each sampling point with a binary message, LSB coding allows for a large amount of data to be encoded. The following diagram illustrates how the message 'HEY' is encoded in a 16-bit CD quality sample using the LSB method: In LSB coding, the ideal data transmission rate is 1 kbps per 1 kHz. In some implementations of LSB coding, however, the two least significant bits of a sample are replaced with two message bits. This increases the amount of data that can be encoded but also increases the amount of resulting noise in the audio file as well. Thus, one should consider the signal content before deciding on the LSB operation to use. For example, a sound file that was recorded in a bustling subway station would mask low-bit encoding noise. On the other hand, the same noise would be audible in a sound file containing a piano solo. To extract a secret message from an LSB encoded sound file, the receiver needs access to the sequence of sample indices used in the embedding process. Normally, the length of the secret message to be encoded is smaller than the total number of samples in a sound file. One must decide then on how to choose the subset of samples that will contain the secret message and communicate that decision to the receiver. One trivial technique is to start at the beginning of the sound file and perform LSB coding until the message has been completely embedded, leaving the remaining samples unchanged. This creates a security problem, however in that the first part of the sound file will have different statistical properties than the second part of the sound file that was not modified. One solution to this problem is to pad the secret message with random bits so that the length of the message is equal to the total number of samples. Yet now the embedding process ends up changing far more samples than the transmission of the secret required. This increases the probability that a would be attacker will suspect secret communication. A more sophisticated approach is to use a pseudorandom number generator to spread the message over the sound file in a random manner. One popular approach is to use the random interval method, in which a secret key possessed by the sender is used as a seed in a pseudorandom number generator to create a random sequence of sample indices. The receiver also has access to the secret key and knowledge of the pseudorandom number generator, allowing the random sequence of sample indices to be reconstructed. Checks must be put in place, however, to prevent the pseudorandom number generator from generating the same sample index twice. If this happened, a collision would occur where a sample already modified with part of the message is modified again. The problem of collisions can be overcome by keeping track of all the samples that have already been used. Another approach is to calculate the subset of samples via a pseudorandom permutation of the entire set through the use of a secure hash function. This technique insures that the same index is never generated more than once.

**5.3 Standard LSB:**

Data hiding in the least significant bits (LSBs) of audio samples in the time domain is one of the simplest algorithms with very high data rate of additional information [2]. The LSB watermark encoder usually selects a subset of all available host audio samples chosen by a secret key. The substitution operation on the LSBs is performed on this subset, where the bits to be hidden substitute the original bit values. Extraction process simply retrieves the watermark by reading the value of these bits from the audio stego object. Therefore, the decoder needs all the samples of the stego audio that were used during the embedding process. The random selection of the samples used for embedding introduces low power additive white Gaussian noise (AWGN). It is well known from the psychoacoustics literature that the human auditory system (HAS) is highly sensitive to AWGN. That fact limits the number of LSBs that can be imperceptibly modified during watermark embedding. The main advantage of the LSB coding method is a very high watermark channel bit rate; use of only one LSB of the host audio sample gives capacity of 44.1 kbps (sampling rate 44 kHz, all samples used for data hiding) and a low computational complexity. The obvious disadvantage is considerably low robustness, due to fact that simple random changes of the LSBs destroy the coded watermark. As the number of used LSBs during LSB coding increases or, equivalently, depth of the modified LSB layer becomes larger, probability of making the embedded message statistically detectable increases and perceptual transparency of stego objects is decreased. Therefore, there is a limit for the depth of the used LSB layer in each sample of host audio that can be used for data hiding. Subjective listening test showed that, in average, the maximum LSB depth that can be used for LSB based watermarking without causing noticeable perceptual distortion is the fourth LSB layer when 16 bits per sample audio sequences are used. The tests were performed with a large collection of audio samples and individuals with different background and musical experience. None of the tested audio sequences had perceptual artifacts when the fourth LSB has been used for data hiding, although in certain music styles, the limit is even higher than the fourth LSB layer. Robustness of the watermark, embedded using the LSB coding method, increases with increase of the LSB depth used for data hiding. Therefore, improvement of watermark robustness obtained by increase of depth of the used LSB layer is limited by perceptual transparency bound, which is the fourth LSB layer for the standard LSB coding algorithm.



**Fig.10: LSB Encoding Technique**

**5.4 Modified LSB:**

This method is able to shift the limit for transparent data hiding in audio from the fourth LSB layer to the sixth LSB layer, using a two- step approach [4]. In the first step, a watermark bit is embedded into the ith LSB layer of the host audio using a LSB coding method. In the second step, the impulse noise caused by watermark embedding is shaped in order to change its white noise properties. The standard LSB coding method simply replaces the original host audio bit in the ith layer (i=1,...,16) with the bit from the watermark bit stream. In the case when the original and watermark bit are different and ith LSB layer is used for embedding the error caused by watermarking is 2i[1] quantization steps (QS)(amplitude range is [-32768, 32767]). The embedding error is positive if the original bit was 0 and watermark bit is 1 and vice versa. The key idea of the proposed LSB algorithm is watermark bit embedding that causes minimal embedding distortion of the host audio. It is clear that, if only one of 16 bits in a sample is fixed and equal to the watermark bit, the other bits can be flipped in order to minimize the embedding error. For example, if the original sample value was (0...01000)2=(8)10, and the watermark bit is zero is to be embedded into 4th LSB layer, instead of value (0...00000)2=(0)10, that would the standard algorithm produce, the proposed algorithm produces sample that has value (0...00111)2=(7)10, which is far more closer to the original one. However, the extraction algorithm remains the same; it simply retrieves the watermark bit by reading the bit value from the predefined LSB layer in the watermarked audio sample. In the embedding algorithm, the (i+1)th LSB layer (bit ai) is first modified by insertion of the present message bit. Then, the algorithm given below is run. In case that the bit ai need not be modified at all due to being already at a correct value, no action is taken with that signal sample[16]. To hide a message in wave sample grab one carrier unit, put one bit of the message into the lowest 4th bit of the carrier unit, flip the rest one and write the changed unit to the destination stream.

LSB coding is explained in the following procedure:

1. Read one sample from the wave stream.

2. Get the next bit from the current message byte.

3. Place it in the current 4th bit of the sample.

4. Flip the rest 3 bits accordingly.

5. Copy the rest of the wave without changes.

**5.5 Proposed LSB Technique With Increased Capacity**

**5.5.1 Technique and Algorithm:**

In the modified LSB encoding technique we have seen that the 4th bit is set according to the secret message. If the sample bit is not equal to the secret message bit we then simply flip the rest of the bits of that given sample. In our proposed model we take consecutive two bits from the secret message and instead of changing a single bit in a sample we change two bits (4th and 3rd position) of the sample. If there is change in this two bits we flip rest of the LSB otherwise there is no change. For example, if the original sample value was (0...01000)2=(8)10, and the watermark bits 01 are to be embedded into 4th and 3rd LSB layer, the standard algorithm will produce the value (0...00000)2=(0)10 to embed the 1st watermark bit only and for the 2nd bit we need another sample, the modified algorithm produces sample that has value (0...00111)2=(7)10, which is far more closer to the original one but here also we need another sample to embed the 2nd watermark bit. Our proposed algorithm will produce (0...00111)2 = (7)10 which is equal to the value produced by modified LSB technique but this sample contains two watermark bit (here 0 and 1) instead of one bit. So we can say that with the same bit error we increased the capacity of the sample to hide more secret message.

Our proposed LSB technique is explained in the following procedure:

1. Read one sample from the wave stream.

2. Get the next two bits from the current message byte.

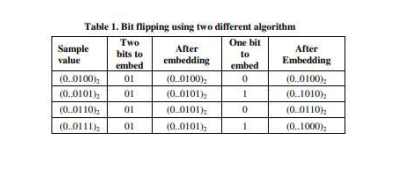
3. Place it in the current 4th and 3rd bit of the sample.

4. Flip the rest 2 bits accordingly.

5. Copy the rest of the wave without

For example, suppose we have 4 samples to hide a secret message 01010101

It is clear that the proposed method introduces smaller error and higher capacity during water- mark embedding. The secret message is completely embedded when we are trying two embed two bit values but when we are embedding 1 bit value then last 4 bits are not getting any sample to be inserted. If the 4th LSB layer is used, the absolute error value ranges from 1 to 4 QS, while the standard method in the same conditions causes constant absolute error of 8 QS. The average power of introduced noise is therefore 9.31 dB smaller if the proposed LSB coding method is used. In addition to decreasing objective quality measure, expressed as signal to noise ratio (SNR) value, proposed method introduces, in the second step of embedding, noise shaping in order to increase perceptual transparency of the method. A similar concept, called error diffusion method is commonly used in conversion of true color images to palette based color images. In our algorithm, embedding error is spread to the four consecutive samples, as samples that are predecessors of the current sample cannot be altered because information bits have already been embedded into their LSBs. Let e(n) denote the embedding error of the sample a(n). For the case of embedding into the 4th LSB layer, the next four consecutive samples of the host audio are modified according to these expressions.



a(n+1)=a(n+1)+be(n)c a(n+3)=a(n+3)+be(n)c/3

a(n+2)=a(n+2)+be(n)c/2 a(n+4)=a(n+4)+be(n)c/4

where bAc denotes floor operation that rounds A to the nearest integer less than or equal to A. Error diffusion shapes input impulse noise, introduced by LSB embedding, by smearing it and changing its distribution to a perceptually better-tuned one. Effect is most emphasized during silent periods of audio signal and in fragments with low dynamics e.g. broad minimums or maximums. The both embedding steps jointly increase the subjective quality of audio stego object. Therefore, we expect that, using the proposed two-step algorithm, we can increase the depth of watermark embedding further than the 4th LSB layer and accordingly increase algorithm's robustness towards noise addition.

**CHAPTER 6**

**EXPERIMENTAL DETAILS**

**6.1 DESCRIPTION**

Matlab supports multi-channel wave format with up to 16 bits per sample. To load a wave file, you can use “[Y, Fs, Nbits] = wavread (wave\_filename)”, where wave\_filename is the file name of the wave file, Y the sampled data with dimension number of samples number of channels, Fs (in Hz) the sampling rate, and Nbits the number of bits per sample used to encode the data in the file. Amplitude values in Y vector are normalized to the range [-1, +1] according to the formula, Y = X /[2(Nbits-1)] – 1, where X is the original unsigned Nbits integer expression. For instance, while X is 128 with Nbits = 8, Y is 0. To generate a wave file and store it in the hard disk,you can use “wavwrite(Y, Fs, Nbits, wave\_filename)”. To playbackthe signal in vector Y, use “sound(Y, Fs, Nbits)”.

**6.2 Read, Playback and Visualize and Audio Signal**

We use a few examples related to audio and image processing as warm-up exercises before we get into the design lab section. Let’s start Matlab and type “Edit” in the command window. You will see a “M-file Editor”. We will use this editor3 to write M-files[6]. Read, Playback and Visualize an Audio Signal (1) Download the “symphonic.wav” audio file from the course web site. Make sure it is in your working directory. (2)You can read an audio file into an Y matrix using the function wavread:

->[Music,Fs, Nbits] = wavread('symphonic.wav');

(3) To obtain the dimension of this image, type

-> [MusicLength, NumChannel ] = size(Music);

The function size will return the number of samples and number of channels of this audio file into MusicLength and NumChannel.

(4) To playback this audio vector, type

->sound(Music, Fs, Nbits);

Make sure your speaker or earphone is on.

(5) We can visualize the waveform by typing:

->Display\_start = 1;

-Display\_end = MusicLength;

-> subplot(2,1,1);

plot(Music(Display\_start: Display\_end,1));

-> title(‘First channel’);

subplot(2,1,2);

plot(Music(Display\_start: Display\_end,2));

->title('Second channel');

You can adjust the display range by changing Display\_start and Display\_end.

**6.3 Bits Manipulation**

(1) Convert the double value format of Music to an unsigned integer expression with Nbits ->IntMusic = (Music+1)\*power(2,Nbits-1);

(2) Many music files use 16 bits to represent an audio sample. In this case, NBits =16. We can extract the lower byte from the first channel:

-> LowNbits = Nbits/2;

->LowIntMusicCh1 = zeros(MusicLength,1);

->FirstChannel = IntMusic(:,1); % extract the first channel

>for ibit = 1:1:LowNbits

->LowIntMusicCh1 = LowIntMusicCh1+ bitget(FirstChannel, ibit)\*power(2,ibit-1);

->end (3) Convert unsigned integer to the normalized expression and listen to it.

-> LowRecMusicCh1 = LowIntMusicCh1/power(2,LowNbits-1) - 1;

->sound(LowRecMusicCh1,Fs, LowNbits);

(4) Repeat the procedure for the second channel and store the final result.

**CHAPTER 7**

**CODE IMPLEMENTATION**

**7.1 MATLAB CODE FOR ENCRYPTION**

[audioData, sampleRate] = audioread('Some.wav’);

% Convert the audio data to integer format if needed

audioInt = int16(audioData \* 32767); % Assuming 16-bit PCM audio

% Open the text file and read the contents

fileID = fopen('anna.txt', 'r');

textData = fread(fileID, '\*char')';

fclose(fileID);

% Convert the text data to binary

binaryText = dec2bin(textData, 8)'; % Convert each character to 8-bit binary

binaryText = binaryText(:)'; % Reshape into a single row vector

binaryText = binaryText - '0'; % Convert char '0' and '1' to numerical 0 and 1

% Check if the audio file can hold the entire text

numSamples = length(audioInt);

if length(binaryText) > numSamples

error('Text file is too large to fit in the audio file.');

end

% Hide the text in the least significant bit (LSB) of each audio sample

for i = 1:length(binaryText)

audioInt(i) = bitset(audioInt(i), 1, binaryText(i));

end

% Normalize the audio back to -1 to 1 range

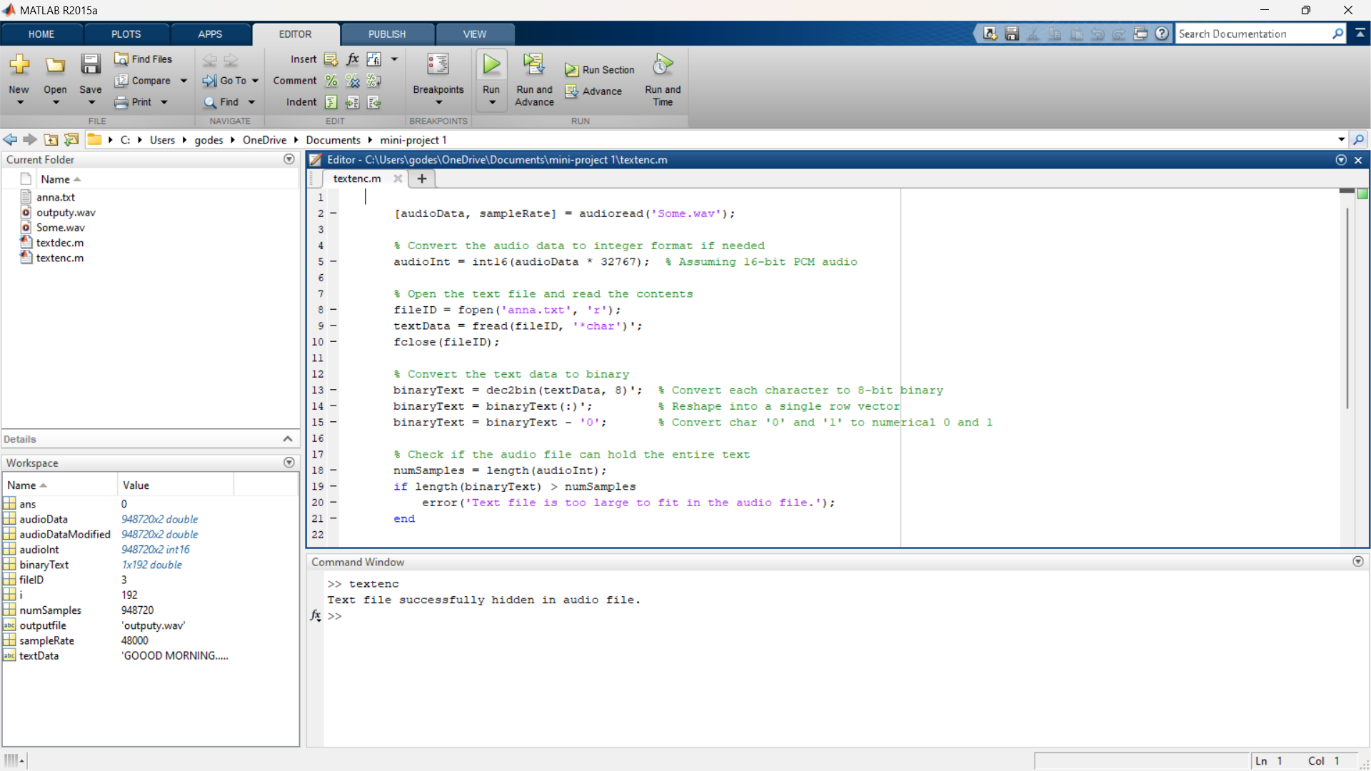
audioDataModified = double(audioInt) / 32767;

outputfile='C:\Users\godes\OneDrive\Documents\MATLAB\output.wav'

% Write the modified audio data to a new file

audiowrite(outputfile , audioDataModified, sampleRate);

disp('Text file successfully hidden in audio file.');



**7.2 MATLAB CODE FOR DECRYPTION**

numChars = 40;

% Read the audio file

[audioData, ~] = audioread('output.wav');

% Convert the audio data to integer format

audioInt = int16(audioData \* 32767);

% Extract the LSBs

extractedBits = bitget(audioInt, 1);

% Convert bits to characters

numBits = numChars \* 8;

extractedBits = extractedBits(1:numBits);

extractedBinary = reshape(extractedBits, 8, [])';

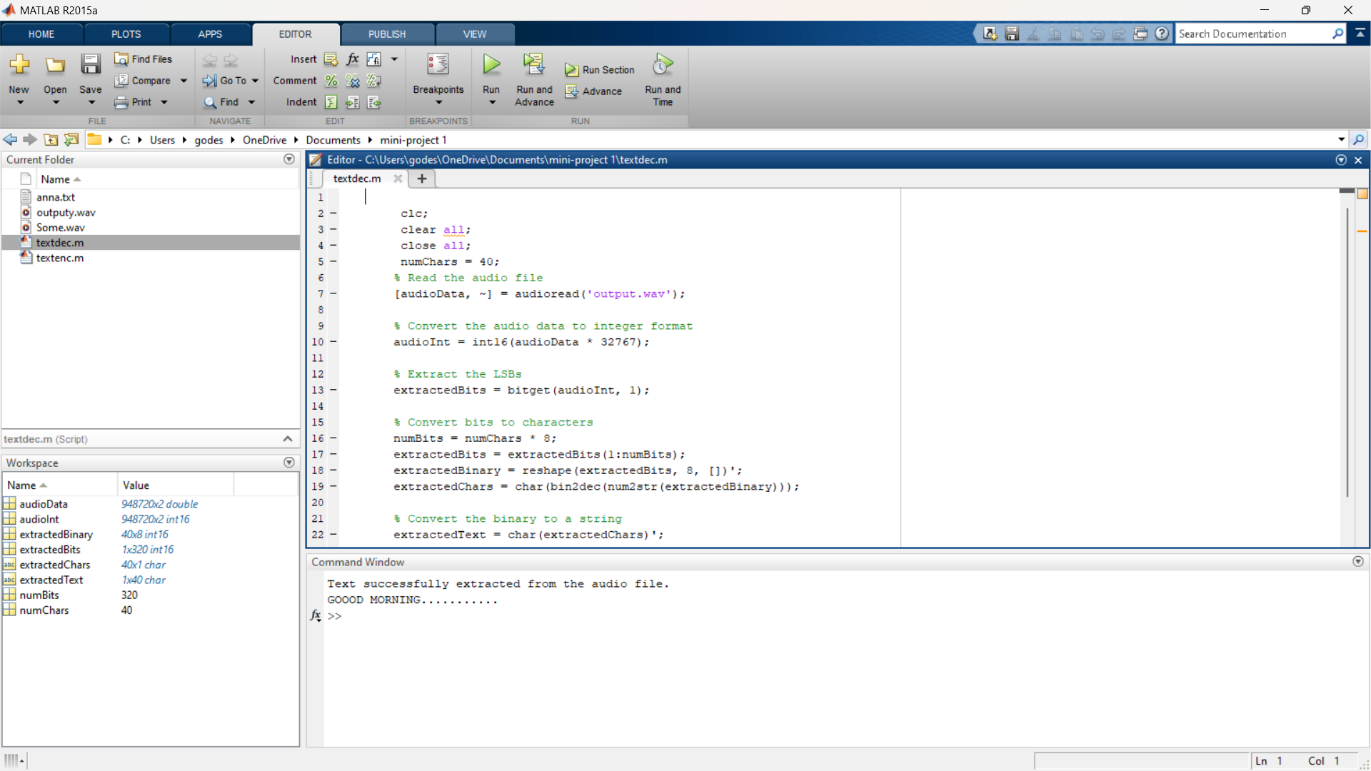
extractedChars = char(bin2dec(num2str(extractedBinary)));

% Convert the binary to a string

extractedText = char(extractedChars)';

disp('Text successfully extracted from the audio file.');

disp(extractedText);



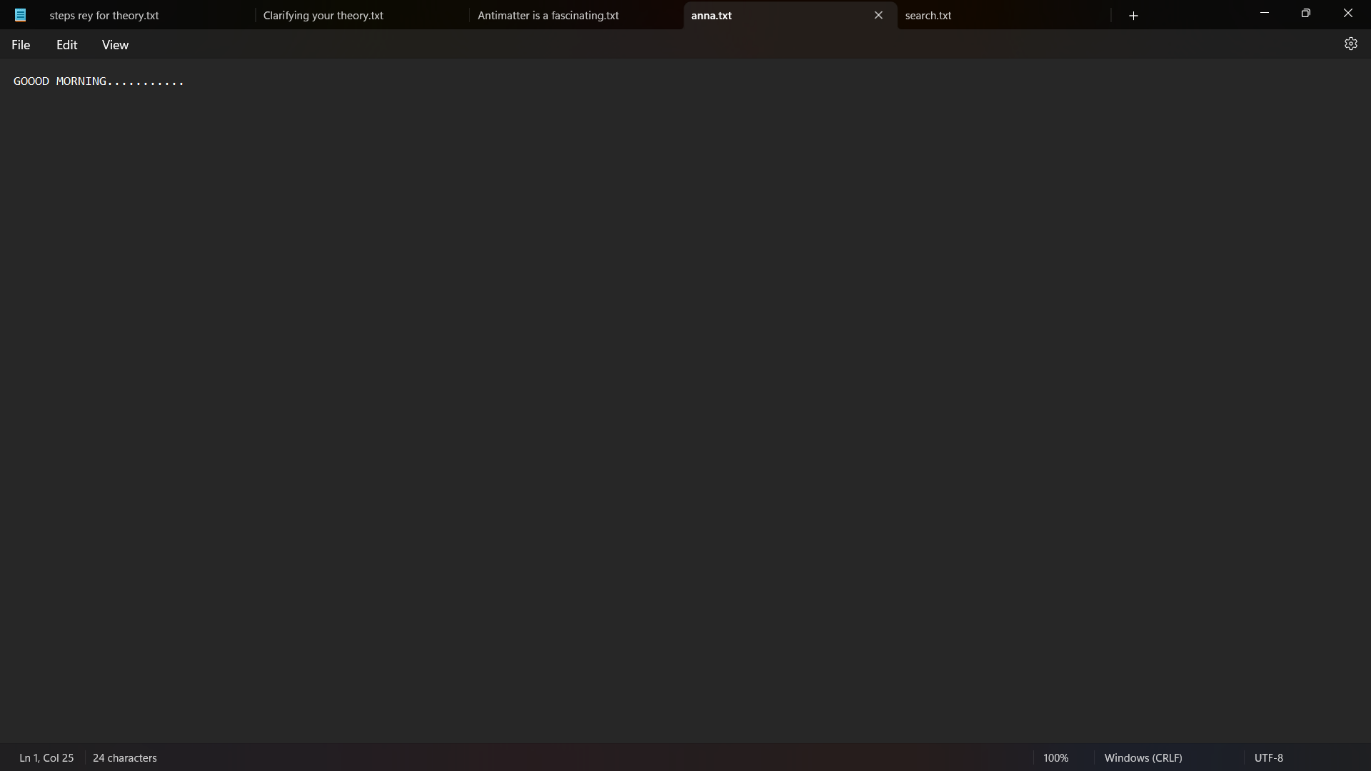
**CHAPTER 8**

**RESULTS AND ANALYSIS**

Modified LSB watermarking algorithm was tested on 11 audio sequences from different music styles (pop, rock, techno, jazz). The audio excerpts were selected so that they represent a broad range of music genres, i.e. audio clips with different dynamic and spectral characteristics. All music pieces have been watermarked using the proposed and modified LSB watermarking algorithm. Clips were 44.1 kHz sampled mono audio files, represented by 16 bits per sample. Duration of the samples ranged from 10 to 15 seconds. As defined above, signal to noise ratio for the embedded watermark is computed as:

SNR = 10 ¢ log10Σ n x2(n)/ Σ n [ x2(n) - y2(n)]

Where x(n) represents a sample of input audio sequence and y(n) stands for a sample of audio with modified LSBs.Results of subjective tests showed that the bit error rate is high when we are using 64 kbps sample audio for proposed LSB. But when we go for the higher kbps sample audio then the bit rate is lesser and it is minimized noticeably.

****

**CHAPTER 9**

**REFERENCES**

1. Anderson, R, Bowman, Petticolas,F. On the limits of Steganography. IEEE Journal selected areas in Communication,16, 4, 474-481

2. Bassia, P., Pitas I., Nikolaidis N. Robust audio watermarking in the time domain, IEEE Transactions on Multimedia 3, 2, 232-241.

3. Brian, J., Yuliya K. and Andrew, L.Fröhlich.2006. audio Steganography Dec 13.

4. Cedric, T., Adi, R., Mcloughlin, I. Data concealment in audio and frequency domain LSB insertion, Proc. IEEE Region 10 International Conference on Electrical and Electronic Technology, Kuala Lumpur, Malaysia, 275-278.

5. Fridrich, J., Goljan, M., Du, R.: 2002 Lossless Data Embedding – New Paradigm in Digital Watermarking, Applied Signal Processing, 2002, 2, 185-196.