



DNA-Based Audification for Secure Text Encryption

Saravanakumar C, Karthikeyan D, John Fredrick I *, Harish K and Hariharan B

Department of ECE, SRM Valliammai Engineering College Potheri, Tamil Nadu, India.

World Journal of Advanced Engineering Technology and Sciences, 2025, 15(01), 1970-1980

Publication history: Received on 07 March 2025; revised on 20 April 2025; accepted on 23 April 2025

Article DOI: <https://doi.org/10.30574/wjaets.2025.15.1.0337>

Abstract

This project introduces an innovative framework for text encryption through DNA-inspired encoding and subsequent audification, expanding the boundaries of data security and auditory representation. The process initiates by encoding plain text into Unicode (UTF-8), which is then transformed into a binary sequence. This binary data is systematically mapped to DNA nucleotide bases A, T, C, and G creating a DNA-like sequence that emulates the structure of biological encoding. This unique DNA sequence is then audified into non-speech audio signals, where each nucleotide base corresponds to distinct audio properties, allowing the encrypted text to be "heard" as an audio pattern rather than read or seen.

Keywords: Cryptography; Audification; DNA; Encryption; Information Security; Fast Fourier Transforms

1. Introduction

Innovative information security and encoding techniques are essential in today's data-driven environment. Because of its great capacity and longevity, DNA—a basic component of life—can be used for data storage and security. Through the use of Unicode, ASCII binary transformation, DNA nucleotide mapping, and audification, this study investigates the encoding of textual data using DNA sequences and converts plain text into distinctive sound patterns. The ASCII format creates a binary foundation for mapping to DNA nucleotides and reduces the quantity of data. In encrypted DNA sequences, binary pairings like 00, 01, 10, and 11 correspond to the nucleotides A, T, C, and G, respectively. Each nucleotide is then given a distinct sound frequency by audification, which transforms these DNA sequences into non-speech audio representations. With applications in audio watermarking, data compression, and covert communication, this Hardware implemented technique enables safe and efficient data transmission and storage.

2. Methodology

The method involves converting plaintext into Unicode (UTF-8) and then into ASCII format before mapping the binary data to DNA nucleotide sequences (A, T, C, and G). These sequences are then transformed into unique non-speech audio signals, ensuring secure data representation. The implementation is carried out using the Arduino IDE, utilizing hardware components like the Arduino Uno and RF 433 MHz transmitter and receiver module for wireless communication and processing.

2.1. Basic DNA Encryption with Direct Audification

DNA (Deoxyribonucleic Acid), which carries genetic information, is explored as a new medium for data encryption due to its capacity to hold vast amounts of data and provide high security. The process begins with the conversion of plain text into its Unicode (UTF-8) value, creating a numeric representation for each character. Each Unicode value is then converted to an 8-bit binary representation, ensuring standardization to a fixed length for each character.

* Corresponding author: John Fredrick I

Binary to DNA Sequence Mapping: Divide the binary sequence into pairs of bits. Use a predefined mapping of binary pairs to DNA nucleotides:

00 → A, 01 → T, 10 → C, and 11 → G.

Table 1 DNA SEQUENCE

DNA BASE	BINARY VALUE
A	00
G	01
C	10
T	11

2.2. DNA Sequence to Non-Speech Audio (Audification)

After encrypting the data using DNA sequences, the encrypted text is further converted into audio signals, specifically non-speech sounds. This process is known as audification. "Each character in the encrypted text is mapped to a specific frequency in the audio signal, adding an additional layer of security.

2.3. Hardware Testing, and Validation

The system is thoroughly tested and validated using Arduino IDE to ensure precision, data integrity, and reliability. By integrating encryption, audio processing, and bioinformatics, it enables secure and efficient data transmission and storage. Comprehensive testing and comparisons with conventional methods confirm the effectiveness and practicality of this approach.

3. Implementation

The implementation of the project "DNA-Based Audification for Secure Text Encryption" involves converting text data into a secure audio format through several stages. Initially, text is transformed into Unicode (UTF-8), then further into ASCII, and finally into 8-bit binary sequences. These binary sequences are mapped to DNA nucleotides (A, T, C, and G) to create encrypted DNA sequences. These sequences are then audified into non-speech audio signals by assigning specific frequencies to each nucleotide. The entire process is implemented using the Arduino IDE, utilizing hardware components such as the Arduino Uno and RF 433 MHz transmitter and receiver module for wireless communication. This interdisciplinary approach, combining bioinformatics, encryption, and audio processing, offers an innovative method for secure data storage and transmission.

3.1. Encoding procedure (Text → Audio)

- Step1: Input: Plain text
- Step 2: Convert Unicode to Binary.
 - Represent each Unicode value as a binary string.
 - Example: Unicode: [72, 69, 76, 76]
 - Binary (8-bit): ['01001000','01000101','01001100','01001100']
- Step 3: Convert Binary to DNA sequence - For each Unicode value group 2-bit binary and map each 2-bit binary to a nucleotide:
 - '00' → 'A'
 - '01' → 'T'
 - '10' → 'G'
 - '11' → 'C'

Concatenate the nucleotide mappings into a DNA sequence.

- Step 4: Convert DNA sequence to audio signal - Assign a specific frequency to each nucleotide:
 - 'A' → 440 Hz 'T' → 660 Hz 'C' → 600 Hz 'G' → 780 Hz

- For each nucleotide n in D , generate a sine wave signal corresponding to its frequency and concatenate all generated signals into an audio waveform
- Output: Audio signal S representing the encoded text.

3.2. Decoding Procedure (Audio → Text):

- Step1: Input: Audio signal non speech format.
- Step 2: Convert audio signal to DNA sequence.
 - Split the audio signal S into chunks corresponding to the duration of each nucleotide. For each chunk c , identify the dominant frequency
 - Map the frequency to its corresponding nucleotide: 440 Hz → 'A'
 - 660 Hz → 'T'
 - 600 Hz → 'C'
 - 780 Hz → 'G'

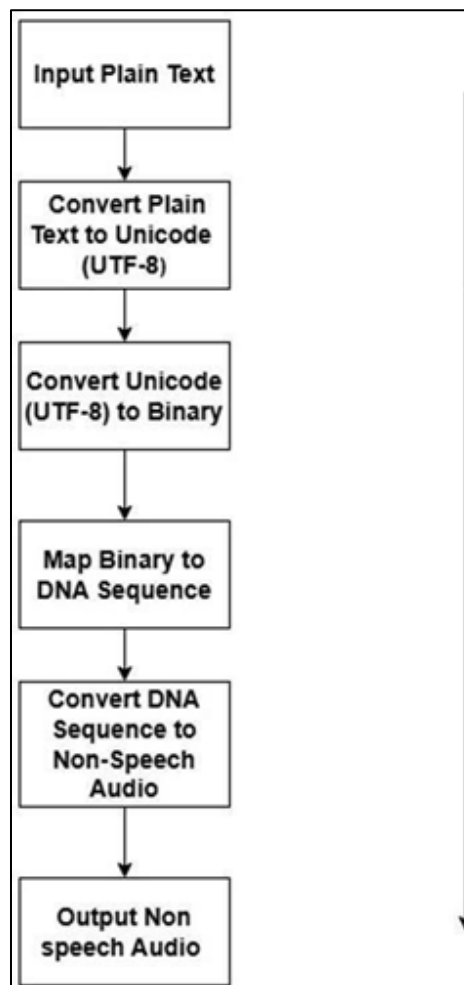


Figure 1 Encoding Part

- Step 3: Convert DNA sequence to Unicode.
 - For each nucleotide n in DNA sequence, map back to the corresponding 2-bit Binary form:
 - 'A' → '00'
 - 'T' → '01'
 - 'G' → '10'
 - 'C' → '11'
 - Concatenate the 2-bit chunks into a binary string
- Step 5: Convert Unicode to text.
 - Convert the Unicode value to its corresponding character and combine characters to get the original text.
 - Output: Decoded text which matches the original input text.

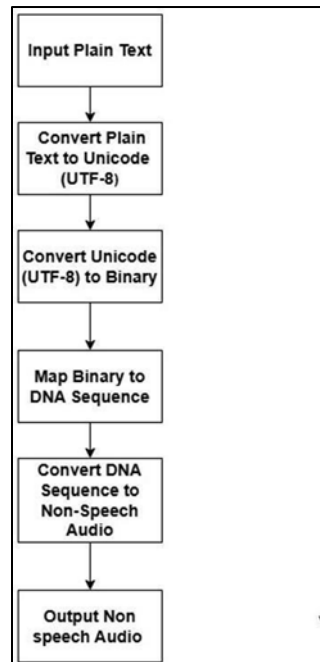


Figure 2 Decoding part

3.3. Software requirements

3.3.1. ARDUNO IDE (1.8.19)

ARDUNIO IDE brings new features and improvements that are particularly beneficial for projects involving complex data transformations, like our project “audification of Text encrypted with DNA nucleotide sequences”. This process, which includes encoding text in Unicode (UTF-8), converting it to binary, mapping it to DNA nucleotide sequences (such as ATCG), and finally transforming it into non-speech audio, greatly benefits from ARDUNIO IDE enhanced computational and visualization capabilities

3.4. Hardware requirements

3.4.1. Arduino UNO

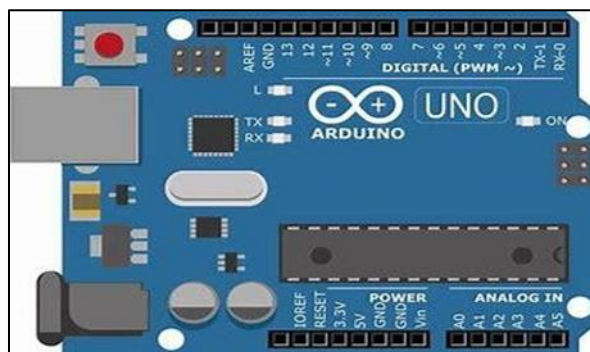


Figure 3 Arduino UNO

Arduino Uno is a popular microcontroller board based on the ATmega328P chip, widely used for embedded systems and IoT projects. It features 14 digital I/O pins, 6 analog inputs, a 16 MHz clock, and supports communication via USB. It is programmed using the Arduino IDE and supports various sensors, modules, and actuators. Its simplicity, open-source nature, and strong community support make it ideal for beginners and advanced users alike.

3.4.2. RF 433 MHz Transmitter Module



Figure 4 RF 433 MHz Transmitter Module

The RF 433 MHz transmitter is a budget-friendly wireless module used for short-range communication in applications like IoT, remote controls, and home automation.

It functions at a 433 MHz frequency with ASK (Amplitude Shift Keying) modulation and can cover 100–200 meters in open areas. The module operates with VCC (5V), GND, and DATA input from a microcontroller like Arduino Uno. In your DNA-based audification project, it plays a key role in wirelessly transmitting sound-encoded encrypted data, which the receiver deciphers back into text.

3.4.3. RF 433 MHz Receiver Module

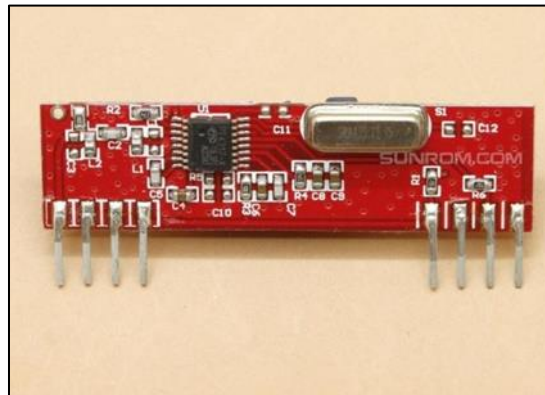


Figure 5 RF 433MHz Receiver Module

The RF 433MHz receiver is a wireless module designed to receive signals from a 433MHz transmitter using ASK (Amplitude Shift Keying) modulation. It operates at 433 MHz frequency and features VCC, GND, and DATA output pins, which can be connected to a microcontroller like Arduino Uno for signal processing. With a range of 100–200 meters in open areas, its performance may vary indoors due to interference. In the DNA-based audification project, this receiver captures transmitted frequency-encoded data, deciphers it into DNA sequences, and reconstructs the original text after decoding.

3.4.4. Buzzer



Figure 6 Buzzer

The active piezo buzzer in this project is responsible for generating sound based on encoded frequencies. It connects to the Arduino Uno, producing specific tones that represent DNA sequences (A = 440 Hz, T = 660 Hz, C = 780 Hz, G = 600 Hz). These audio signals serve as an alternative method for secure text transmission.

On the receiving end, the buzzer reproduces the received frequencies, which are then converted back into DNA sequences and decoded into the original text. This approach enhances security by using non-speech sound for data communication.

Apart from the Arduino Uno, RF 433MHz modules, and buzzer, several other components are essential for building the DNA-based audification for secure text encryption system. A breadboard is used to create temporary circuits without soldering, allowing easy modifications and testing. Jumper wires act as connectors, establishing electrical links between components such as the Arduino, buzzer, RF transmitter, and receiver. A USB connection provides the necessary power supply. These components work together to ensure a functional and adaptable setup for secure data transmission.

3.5. Algorithm

- Input plain text.
- Convert plain text to a binary array using Unicode encoding, then flatten into a row matrix.
- Encode the binary array into nucleotides by mapping binary pairs to nucleotide bases (A, T, G, C).
- Generate music from nucleotides by assigning specific frequencies to each nucleotide and creating corresponding sine waves, then combine these into the final audio signal.
- Store the audio signal as a .wav file.
- Read the saved .wav file to retrieve the audio signal.
- Convert the audio signal back to data by detecting peaks and mapping frequencies to nucleotide bases.
- Decode the nucleotide sequence back into a binary array and reshape it into the original dimensions.
- Convert the binary array back to the original plain text using Unicode.

3.6. Applications

- Audio watermarking: To guarantee the integrity and authenticity of audio content, secret data can be embedded within sound files using the encrypted audio signals.
- Covert Communications: Sensitive information can be securely transmitted in a manner that is challenging to decode without the right decoding tools by converting text into non-speech sounds.
- Data Storage: By using the data density and stability of DNA, text may be encoded as DNA sequences and then further encoded as audio, allowing for the safe and compact storage of vast volumes of data.
- Cybersecurity: Adding DNA and audio encoding to conventional encryption techniques can strengthen them against cyberattacks and unwanted access.
- IoT Security: Implementing this technique in IoT devices can add an extra layer of encryption, preventing data breaches in connected systems.
- Steganography in Embedded Systems: This method can be integrated into microcontroller-based systems for hidden data transmission, useful in anti-surveillance operations.
- Medical Data Encryption: Patient records can be securely stored and transmitted using DNA-based encoding, ensuring privacy in healthcare applications.
- Authentication Systems: The audified DNA sequence can serve as a unique biometric key for hardware-based security authentication, replacing traditional passwords or PINs.
- Secure RFID Communication: Implementing this method in RFID-based access control systems can enhance security in restricted areas such as research labs and data centers.

3.7. Flowchart

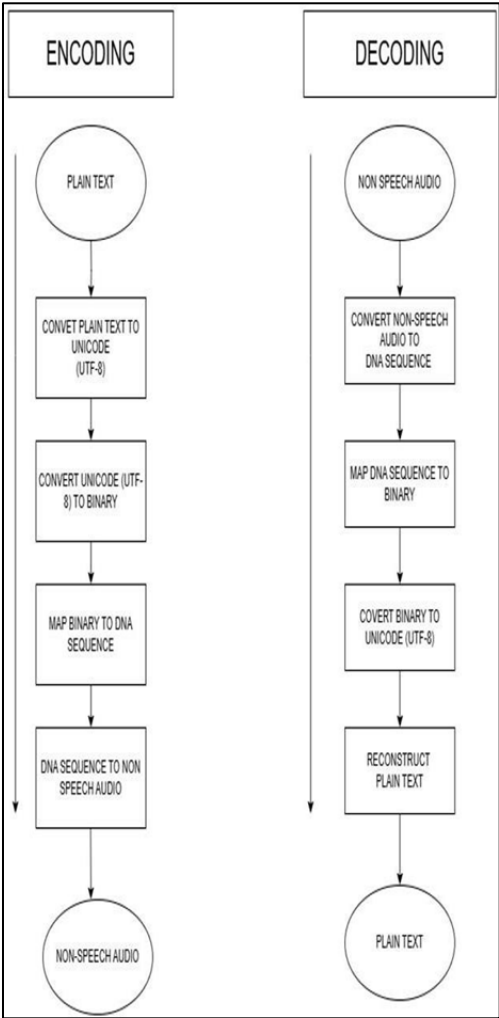


Figure 7 Flow chart

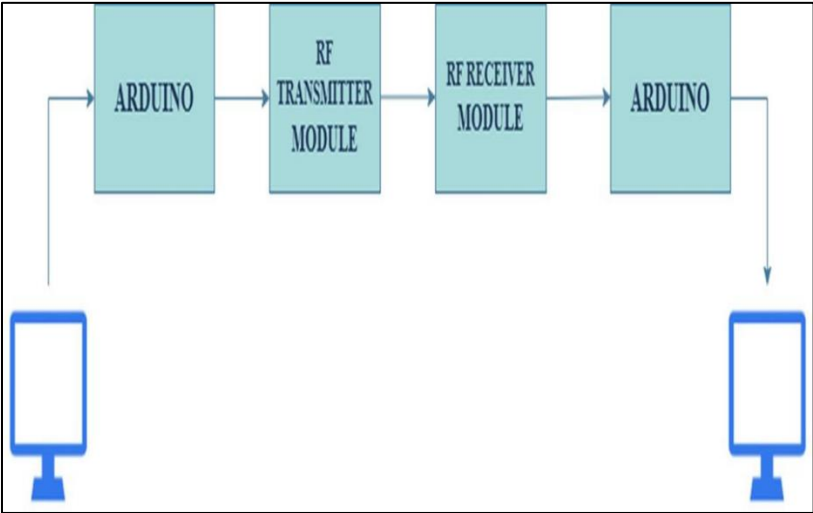


Figure 8 Proposed Block Diagram

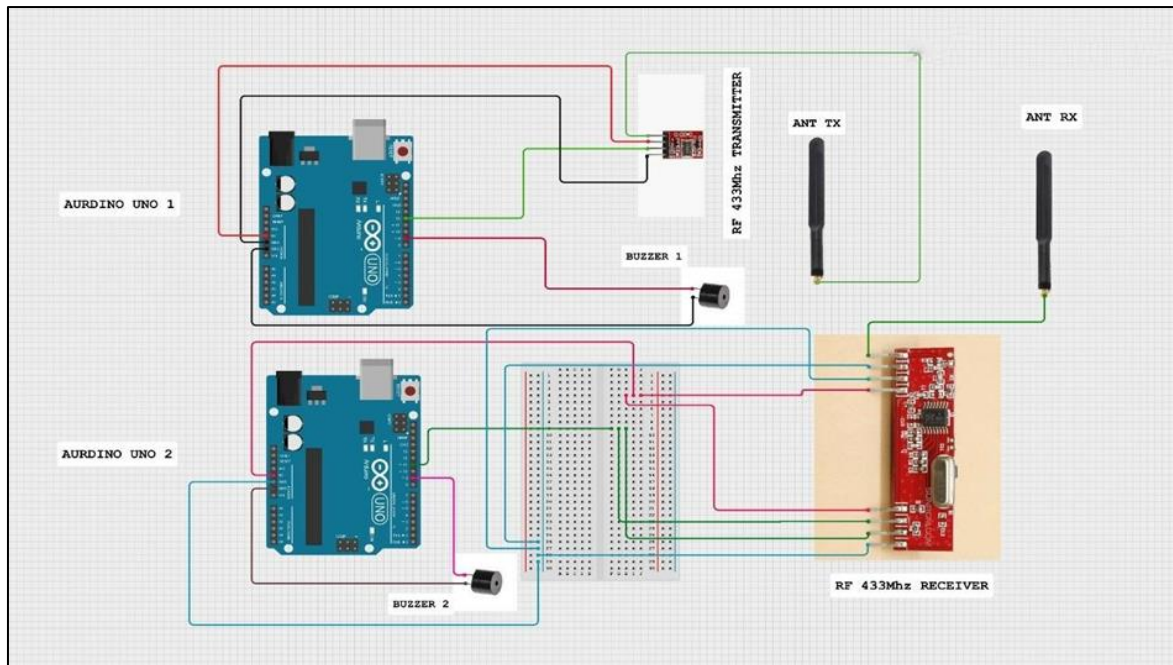


Figure 9 Circuit Diagram

4. Results and discussions

4.1. Advantages

- **Increased Security:** A layer of security is added by DNA-based encryption, which makes it challenging to decode without specialized knowledge of the mapping and audification technique.
- **Condensed Data Display:** DNA sequences compress enormous volumes of text into short sequences, allowing for efficient data storage.
- **Distinct Audio Signatures:** To help with data differentiation and verification, each DNA sequence generates a unique non-speech audio pattern.
- **Accessibility for Users with Visual Impairments:** Users can hear recognize and distinguish encrypted documents thanks to non-speech audio signals.
- **Checks for Data Integrity and Verification:** Through mismatched audio waveforms, audification enables the rapid identification of data damage or tampering.
- **Real-Time Secure Communication:** Using RF 433 MHz modules, encrypted messages can be transmitted securely in real time without reliance on external networks
- **Low Power Consumption:** Arduino-based implementation consumes minimal power, making it suitable for portable and energy-efficient applications.
- **Offline Encryption & Transmission:** Since the encryption and transmission occur on standalone hardware modules, there is no need for an internet connection, reducing hacking risks.
- **Scalability for IoT Integration:** The system can be integrated into IoT devices, enhancing secure data transmission in smart applications.
- **Tamper-Resistant Hardware:** Unlike software- based encryption, a hardware-implemented system is more resistant to cyber threats such as hacking or malware attacks.
- **Compact and Cost-Effective:** Using components like Arduino Uno and RF modules, the system remains compact, affordable, and easy to deploy in various real-world applications.
- **Versatile Hardware Compatibility:** The technique can be adapted to work with different microcontrollers and wireless modules, making it flexible for diverse use cases.

4.2. Disadvantages

- **Complicated Execution:** Demands specific expertise in signal processing, cryptography, and bioinformatics.
- **Low Density of Information:** Large or complicated data may be difficult for non- speech audio to accurately represent.

- Problems with Standardization: Inconsistencies result from the lack of a common standard for translating sequences to audio frequencies.
- Overlap of Audio Patterns: There may be overlap or ambiguity since similar DNA sequences might make similar sounds

4.3. Future scope

- Developments in Data Security: New benchmarks for safe data transport and storage may be established using DNA encryption combined with audification.
- Integration of AI and Machine Learning: AI is capable of automatically classifying data and analyzing and interpreting acoustic patterns from DNA-encoded sequences
- Accessibility for People with Visual Impairments: Through sound, improved audio signatures can improve data accessibility for people with visual impairments.
- Data Storage and Bioinformatics; Compact, safe storage of biological data can result from DNA audification and encryption.
- Uniformity: System compatibility would be ensured by standardizing the mapping of binary data to audio and DNA frequencies.
- Monitoring in Real Time: Real-time audio feedback in biological monitoring and secure communications for data integrity and error detection.
- Extension of Education: Beneficial for promoting multidisciplinary research and teaching bioinformatics, cryptography, and sound engineering concepts.
- Innovative Interfaces: Creating audio-based data interfaces that use sound patterns to analyse and store encrypted data

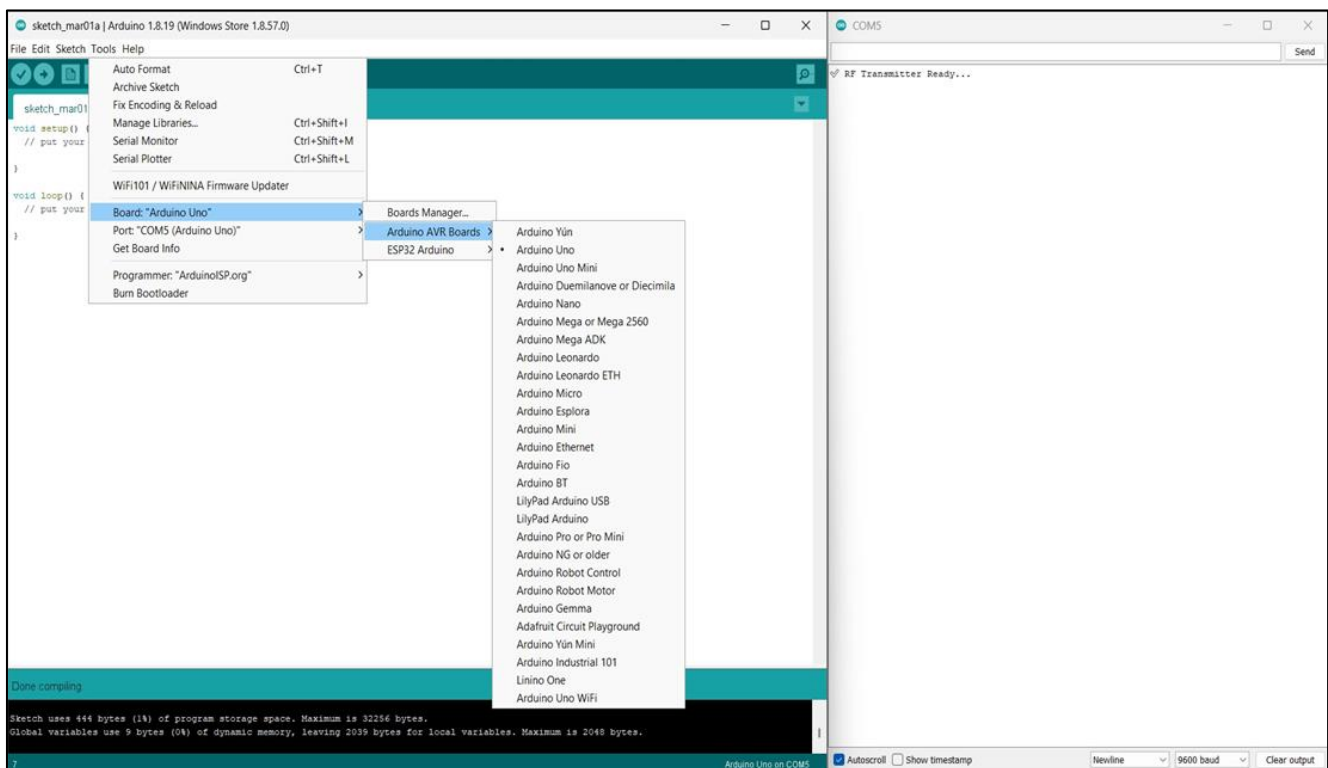
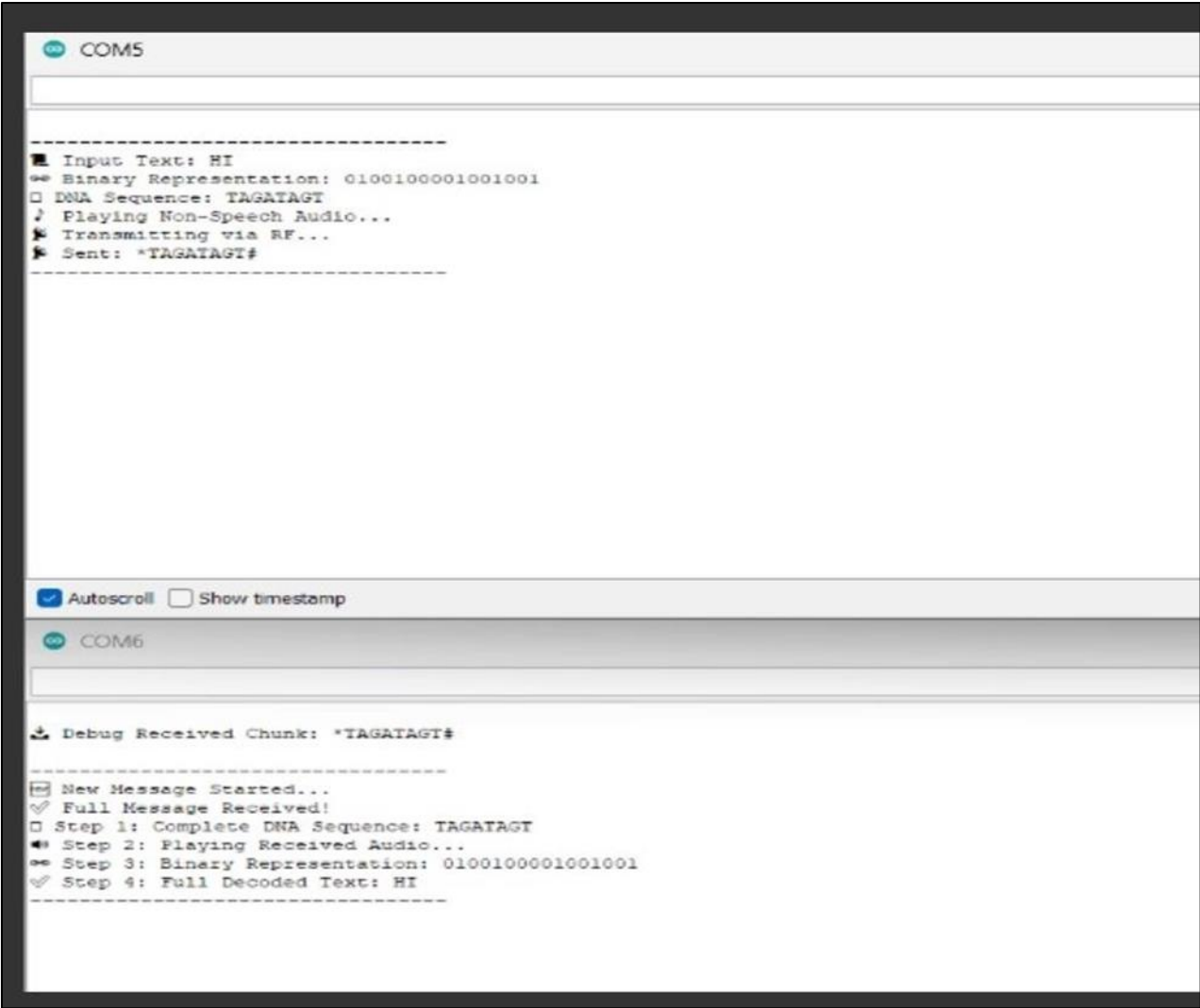


Figure 10 Arduino IDE layout (version 1.8.19)



The screenshot displays the Arduino IDE serial monitor with two active ports, COM5 and COM6. The COM5 window shows the following output:

```
-----  
[x] Input Text: HI  
[x] Binary Representation: 0100100001001001  
[x] DNA Sequence: TAGATAGT  
[x] Playing Non-Speech Audio...  
[x] Transmitting via RF...  
[x] Sent: *TAGATAGT#  
-----
```

Below the COM5 window, there are checkboxes for 'Autoscroll' (checked) and 'Show timestamp' (unchecked). The COM6 window shows the following output:

```
-----  
[x] Debug Received Chunk: *TAGATAGT#  
-----  
[x] New Message Started...  
[x] Full Message Received!  
[x] Step 1: Complete DNA Sequence: TAGATAGT  
[x] Step 2: Playing Received Audio...  
[x] Step 3: Binary Representation: 0100100001001001  
[x] Step 4: Full Decoded Text: HI  
-----
```

Figure 11 Arduino IDE Output

5. Conclusion

DNA-Based Audification for Secure Text Encryption is an innovative approach that combines bioinformatics, cryptography, and digital signal processing. By encoding data in DNA-like sequences and converting them into non-speech audio, this method presents a unique way of securing, storing, and verifying data. It provides robust data security, offers accessibility for visually impaired users, and opens up possibilities for data differentiation through distinct audio patterns.

While there are challenges such as complexity, processing demands, and limited standardization, the potential applications in data security, AI-driven audio analysis, and accessible technology show promising directions for development. As interdisciplinary research in bioinformatics and sound engineering advances, the integration of DNA-based encryption and audification could create new standards in secure data handling and auditory data interfaces.

This approach ultimately highlights a novel fusion of biology-inspired encryption and audio technology, with exciting possibilities for the future of secure, accessible, and efficient data storage and communication.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

This work was carried out by students as part of an academic project and adheres to the ethical standards set by our institution.

References

- [1] Kaur A, Dutta M K. "An optimized high payload audio watermarking algorithm based on LU-factorization". *Multimedia Syst*, vol. 24, no. 3, pp. 341-353, Apl, 2017.
- [2] Liu Z, Huang J, Sun X, Qi C. "A security watermark scheme used for digital speech forensics". *Multimedia Tools Appl*, vol. 76, pp. 9297-9317, Apl, 2017.
- [3] Belazi A, Abd El-Latif A, Belghith S. A novel image encryption scheme based on substitution-permutation network and chaos. *Signal Process*, vol.128, pp. 155- 170, Nov, 2016.
- [4] Souyah A, Faraoun K M. An image encryption scheme combining chaos memory cellular automata and weighted histogram. *Nonlinear Dyn*, vol.86, no.1, pp.639-653, Oct,2016.
- [5] Wang X Y, Gu S X, Zhang Y Q. Novel image encryption algorithm based on cycle shift and chaotic system. *opt Laser Eng*, vol. 68, pp. 126-134, May, 2015
- [6] Yang Y G, Tian J, Sun S J, Xu P. Quantum-assisted encryption for digital
- [7] Aysha Divan and Janice Royds. *Molecular Biology: A Very Short Introduction*. Oxford University Press, 2016.
- [8] Sheu L J. A speech encryption using fractional chaotic systems. *Nonlinear Dyn*, vol. 65, no. 1-2, pp. 103-108, Jul, 2011.
- [9] Wang X Y, Gu S X, Zhang Y Q. Novel image encryption algorithm based on cycle shift and chaotic system. *opt Laser Eng*, vol. 68, pp. 126-134, May, 2015.
- [10] Shipra Jain and Vishal Bhatnagar. Analogy of various dna based security algorithms using cryptography and steganography. In *Issues and Challenges in Intelligent Computing Techniques (ICICT)*, 2014 International Conference on, pages 285–291. IEEE, 2014.
- [11]
- [12]