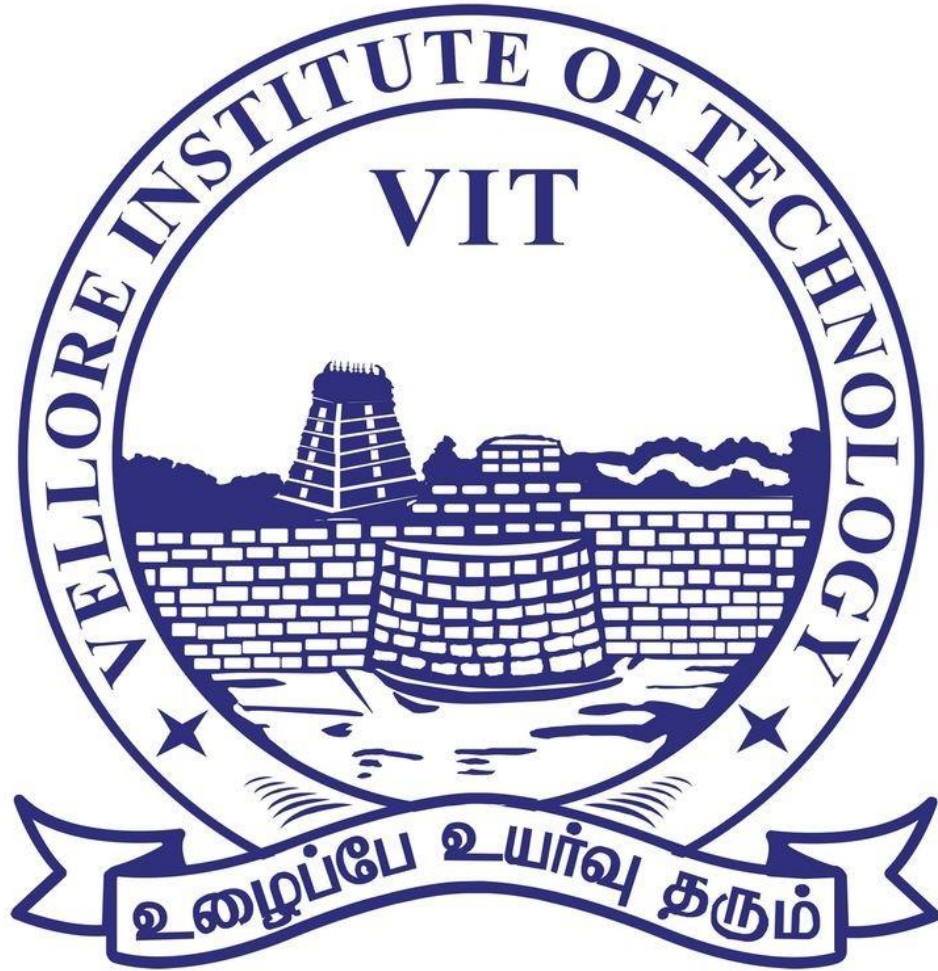


ECE2006 - Digital Signal Processing

Final Report



ECE2006 Digital Signal Processing
Review – I

TOPIC:
Encryption of Speech Signals Using
El-Gamal's method and elliptical curve method

Team Members:

Leela Praneeth - 20BEC0771

Ankit Raj - 20BEC0766

Nagaraju S M - 20BEC0278

Under the guidance of:

Dr. Kalaivani. S

Abstract:

Information Security is an important aspect of data sharing. A large portion of Corporate, Government and legal documents are being primarily shared over the internet. Prime among these is audio signals which are easy to manipulate and doctor, which can cause a lot of strife and wasted time or sometimes physical risk to the parties sharing the information. We propose different encryption techniques.

El-Gamal Algorithm is a very secure and efficient way to transmit speech signals wirelessly without worry of privacy breach or high noise. We will also look at elliptical curve cryptography. ECC here is a complex encryption technique and is harder to breach using brute force or stolen key.

Project flow:

- We first look at the Research paper [https://doi.org/10.1016/j.procs.2020.03.402.\(https://www.sciencedirect.com/science/article/pii/S1877050920308681\)](https://doi.org/10.1016/j.procs.2020.03.402.(https://www.sciencedirect.com/science/article/pii/S1877050920308681)) and analyze the methods they use to go about the simulation and the theory they use in the encryption algorithm.
- <https://doi.org/10.1016/j.compeleceng.2022.108022> [.\(https://www.sciencedirect.com/science/article/pii/S0045790622002877\)](https://www.sciencedirect.com/science/article/pii/S0045790622002877) we look at this as a reference for elliptical cryptography.
- Our goal is to find an efficient and easy to implement security layer in audio based communication over the internet.
- After the analysis, we move forward to starting a MATLAB code to import a speech signal in the form of a WAV file and use the methods to encrypt and then compare in terms of efficiency and security.
- The analysis of Noise is the next step. We can verify the feasibility of these techniques.
- All of the data and simulation results are stored and recorded in a report.

Literature review:

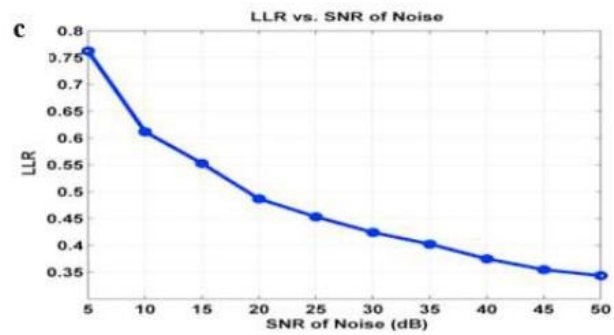
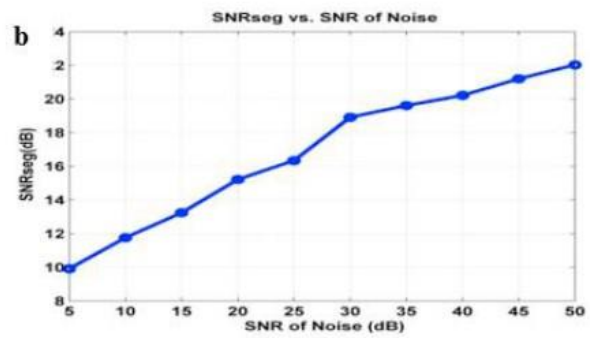
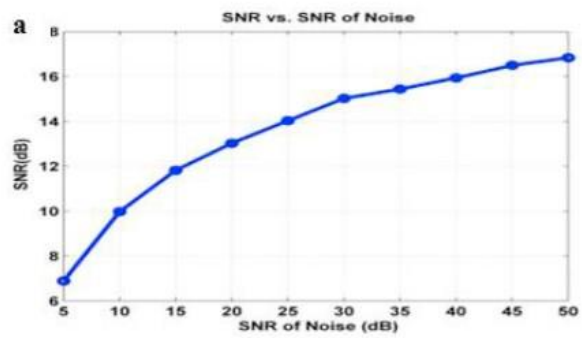
- [<https://doi.org/10.3390/app112110190>] : In this paper we can see that the authors used the advanced mathematical concepts of Chaos theory to create an encryption method that is hard to crack. An Ikeda map is used for visualization. A Pseudo Random byte generator is used.
- [<https://doi.org/10.3390/sym14010017>] : In this paper, the authors have looked at different methods such as Fast Walsh–Hadamard Transform (FWHT), FFT, DCT are consolidated into one Chen memristor chaotic system.
- [<https://doi.org/10.1016/j.procs.2016.03.038>] : In this paper the encryption history is analyzed and a better method over asymmetric keys is proposed, a Hybrid cryptography ECC-elliptic curve method.
- [[Elliptic curve cryptosystem and its applications | IEEE Conference Publication](#)] This research paper talks about ECC. Elliptic curve cryptography (ECC) fits well for an efficient and secure encryption scheme. It is more efficient than the ubiquitous RSA based schemes because ECC utilizes smaller key sizes for equivalent security. A comparative study of ECC with RSA is made in terms of key size, computational power, size of data files and encrypted files.
- [[Performance Analysis of Elliptic Curves for Real-Time Video Encryption | IEEE Conference Publication](#)] This paper discusses Video encryption. Video encryption can be done with symmetric key as well as asymmetric key encryption. Among different asymmetric key encryption techniques, ECC performs better than other algorithms like RSA in terms of smaller key size and faster encryption and decryption operation. In this work, they have analyzed the performance of 18 different ECC curves and suggested some suitable curves for real-time video encryption.
- [[Performance Analysis of Secure Real-time Transport Protocol Using Elliptic Curves | IEEE Conference Publication](#)] In this work, they have implemented an ECC based encryption technique in a softphone to encrypt real-time voice calls. They have used 15 elliptic curves to measure the performance of audio calls, and based on the result, have proposed some suitable elliptic curves for real-time audio encryption.
- [[Implementation of text based cryptosystem using Elliptic Curve Cryptography | IEEE Conference](#)

[Publication](#) In this paper, a text based Elliptic Curve Cryptosystem is implemented. Each character in the message is represented by its ASCII value. Each of these ASCII value is transformed into an affine point on the EC, by using a starting point called Pm. Transformation of the plaintext ASCII value by using an affine point is one of the contributions of this work.

- <https://www.ijser.org/paper/Elgamals-Algorithm-in-C-ryptography.html> The work discussed in this paper is related to the basic understanding of the El gamel encryption algorithm including the security parameters.
- <https://iopscience.iop.org/article/10.1088/1742-6596/1235/1/012054/pdf> In this paper the modification of chipertext Elgamal algorithm using split merge is discussed.
- http://ipco-co.com/PET_Journal/Papers%20CEIT'14/025.pdf In this paper ElGamal public-key encryption is discussed which is one of the cryptographic types.
- <https://reader.elsevier.com/reader/sd/pii/S1877050920308681?token=086CB3C4D8972497BA82A8DFA09FDFEBBC211EDEEC83F47EB093C4C8EB3D377D698EBACD9682E37BD08EBCC380DB1ABC&originRegion=eu-west-1&originCreation=20220912161552> In this research paper Implementation of El-Gamal algorithm for speech signals encryption and decryption is explained

Expected Outcome:

- The first output we get here is the signal of the input speech signal, the encrypted signal and the decrypted speech signal. We need an encryption algorithm that is capable of masking the signal unintelligible in the encrypted form, where it is not possible for malicious actors to interpret the signal. But the decryption using key must be good enough to reproduce the original signal without noise or loss of data.
- The second output is actually a quality check of the decrypted signal. We can look at the different values of the Signal Noise Ratio(SNR) and Log-LikeliHood Ratio(LLR), to see the amount of noise produced and determine whether it is manageable.



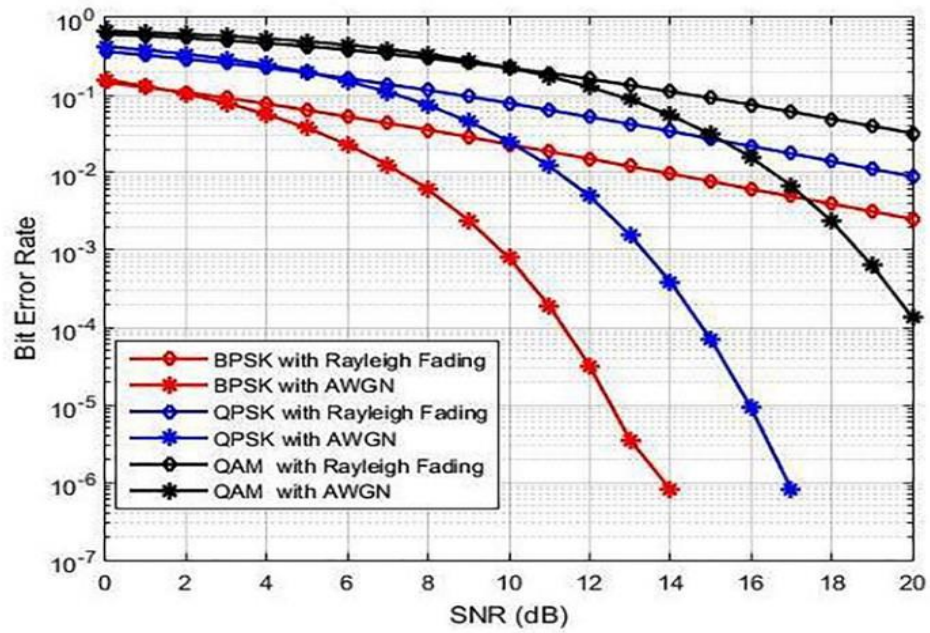
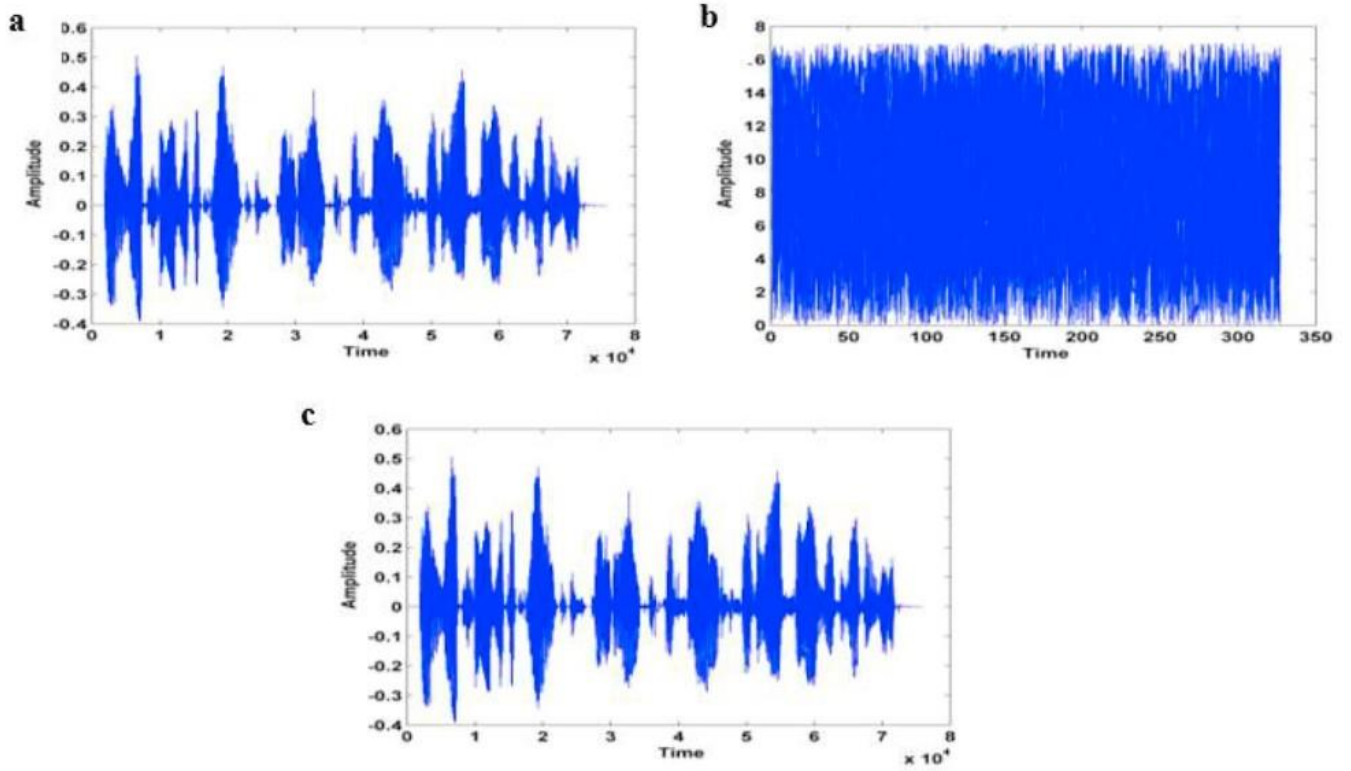


Fig. 4. Bit error rate vs. SNR.

Timeline:

12-08-2022:

Review 1 with project finalization, Research paper sources and literature survey. Group contributions and Timeline finalization.

7-10-2022:

Review 2 with a portion of the code and GUI designed and the Workflow of the project presented.

5-11-2022:

Final review with culmination of the project properly documented in an accepted format. Group presentation.

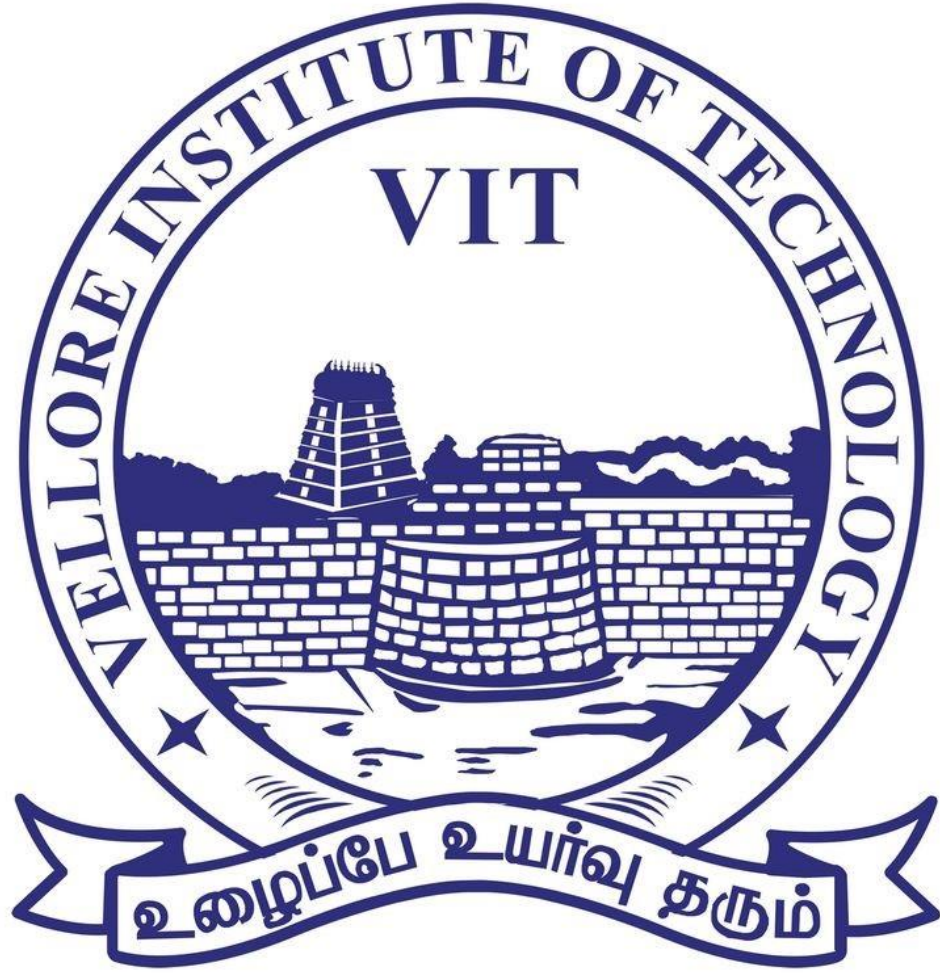
Contribution of Team-members:

- Leela Praneeth: Literature Survey, Document Outlining, Abstract
- Ankit Raj: Literature Survey, Expected Outcomes
- Nagaraju SM: Literature Survey, Workflow, timeline

Sources

- 1) Stoyanov, B.; Ivanova, T. Novel Implementation of Audio Encryption Using Pseudo Random Byte Generator. Appl. Sci. 2021, 11,10190.
<https://doi.org/10.3390/app112110190>
- 2) Dai, W.; Xu, X.; Song, X.; Li, G. Audio Encryption Algorithm Based on Chen Memristor Chaotic System. Symmetry 2022, 14, 17.
<https://doi.org/10.3390/sym14010017>
- 3) Sridhar C. Iyera, , R.R. Sedamkarb, , Shiwani Guptac. A Novel Idea on Multimedia Encryption using Hybrid Crypto Approach.
<https://doi.org/10.1016/j.procs.2016.03.038>
- 4) S. M. C. Vigila and K. Muneeswaran, "Implementation of text based cryptosystem using Elliptic Curve Cryptography," doi: 10.1109/ICADVC.2009.5378025.
[\[Implementation of text based cryptosystem using Elliptic Curve Cryptography | IEEE Conference Publication\]](#)

- 5) N. Sen, R. Dantu and M. Thompson, "Performance Analysis of Secure Real-time Transport Protocol Using Elliptic Curves," [\[Performance Analysis of Secure Real-time Transport Protocol Using Elliptic Curves | IEEE Conference Publication\]](#)
- 6) G. V. S. Raju and R. Akbani, "Elliptic curve cryptosystem and its applications," [\[Elliptic curve cryptosystem and its applications | IEEE Conference Publication\]](#)
- 7) S. M. C. Vigila and K. Muneeswaran, "Implementation of text based cryptosystem using Elliptic Curve Cryptography," [\[Implementation of text based cryptosystem using Elliptic Curve Cryptography | IEEE Conference Publication\]](#)
- 8) Omar A. Imran, Sura F. Yousifa, Isam Salah Hameeda, Wisam Najm Al-Din Abeda, Ali Thaeer Hammi,
<https://reader.elsevier.com/reader/sd/pii/S1877050920308681?token=65AD8E45AC6A4CDEDD082895FAC96E00D40D5CFD3AE0C72EA76DC16376A53D231C696F5C0A6D01A30A50495EE55D9ADF&originRegion=eu-west-1&originCreation=20220912163425>
- 9) Amer Daeri, Amer R. Zerek, Mohamed A. Abuinjam, http://ipco-co.com/PET_Journal/Papers%20CEIT'14/025.pdf
- 10) Rashmi Singh, Shiv Kumar, <https://www.ijser.org/paper/Elgamals-Algorithm-in-Cryptography.html>



**ECE2006 - Digital Signal Processing
Review – II**

TOPIC:

Encryption of Speech Signals Using El-Gamal's method and elliptical curve method

Team Members:

Leela Praneeth - 20BEC0771

Ankit Raj - 20BEC0766

Nagaraju S M - 20BEC0278

Under the guidance of:

Dr. Kalaivani. S

El Gamal Algorithm

Introduction:

Data communication and data protection depend heavily on security. It helps prevent unwanted access to sensitive data that could lead to data loss or alteration by unidentified parties, making data transmission unsafe. El Gamal encryption uses public key cryptography. It encrypts the message and uses asymmetric key encryption for two-party communication. This cryptosystem is based on the fact that it is quite challenging to calculate g^ak even if we know g^k and g^a , the discrete logarithm in the cyclic group. If we want to understand the whole picture, we have to go step by step by actually encrypting and decrypting the messages. We will use the example of two peers who are willing to use the El Gamal algorithm to communicate data in a secure manner.

Consider that users 1 and 2 wish 3 to transmit information covertly; in which case the steps below will be performed.

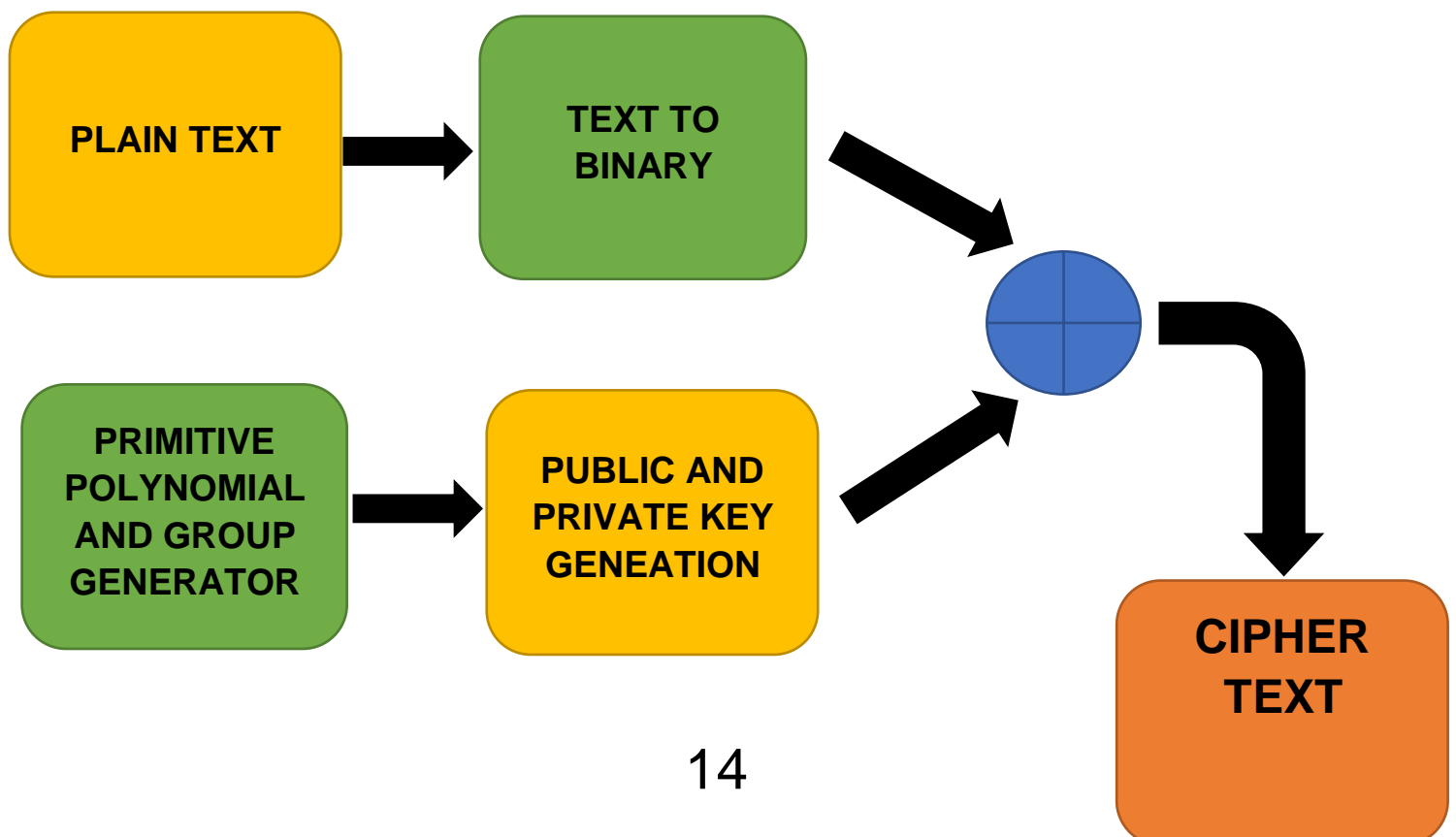
Step 1: Public and Private Key Generation. User 1 tries to select a very long or large integer x while also selecting a cyclic group. It will also select the next element c and the component b from this cyclic group. The values will be chosen so that when a particular function is used, the result is equal to 1. After the value selection phase, the value will be calculated and used to generate the private key. The value will be determined using the formula $fm=bc$. User1 in the current case chooses F as his public key, $fm = bc$, a and b . The values of a will be kept as a private key, which will then be used as a private key.

Step 2: User2 encrypts the data using user1's public key. There are specific settings that user2 must select in order to begin encrypting a message. In addition, user2 will have to choose one of the p values of the cyclic group. As with user1, a cyclic group will be used. The value should be chosen in such a way that Inc passes through and results in a 1 in the specified

function. The message will be encrypted with the public key using some additional values generated by user2. $P_m = b_p$ is the value that will be created. The second revaluation will make b_c equal to $b_a p$. To get closer to the encryption method, the result of this calculation will be multiplied by the second variable Z . At some point, the value will be transferred using the results of the calculations to $b_p, Z * b_a p$.

Step 3: Decrypting the message at the end of user 1. User 1 then determines the correct number that will be used to decrypt the encrypted message by computing the values chosen in the first and second stages. To get the decrypted value, user1 processes $b_a p$ and divides the result by Z . The value that was decrypted was something that was encrypted in the second step. In the above case, user 1 started the procedure by calculating the private and public keys, which are the heart and soul of the algorithm. User2 continues to use the key in the next step to encrypt the method. The message is encrypted so it can be decrypted using the value calculated in the initial phase. In the third stage, it was seen that after dividing the full value by the number generated in the third step, the message was completely decrypted, allowing the end user to read it. The same procedure is used every time the impulse arises to send a message securely

Block diagram:



Code:

```
clc;
close all;
clear all;

m = 15;
q = 2^m;

polynomial = primpoly(m,'nodisplay');

primeFactors = unique(factor(2^m-1));
rng(123456);
while 1
    g = gf(randi([1,q-1]),m,polynomial);
    Primitive = true;
    for i = 1:length(primeFactors)
        if g^((q-1)/primeFactors(i)) == gf(1,m,polynomial)
            Primitive = false;
            break;
        end
    end
    if Primitive
        break;
    end
end

privateKey = 12;
publicKey = {g,g^privateKey,polynomial};
disp(publicKey)
disp(polynomial)
disp('Enter the text to be encrypted:')
```

```

text = input('-->','s');
disp(' ')
disp('Entered text:')
disp(text);
disp(' ')

bitsPerChar = 7;
binaryMsg = int2bit(int8(text'),bitsPerChar);
PaddedBits = m - mod(numel(binaryMsg),m);
if PaddedBits == m
    PaddedBits = 0;
end
binaryMsg = [binaryMsg; zeros(PaddedBits,1)];
textToEncrypt = bit2int(binaryMsg,m);

cipherText = gf(zeros(length(textToEncrypt),2),m,polynomial);

for i = 1:length(textToEncrypt)
    k = randi([1 2^m-2]);
    cipherText(i,:) =
[publicKey{1}^k,gf(textToEncrypt(i),m,polynomial)*publicKey{2}^k];
end

temp = cipherText.x;
disp('Encrypted text:');
disp(dec2char(temp(:,2),bitsPerChar,m));
disp(' ')

decipher = gf(zeros(size(cipherText,1),1),m,polynomial);
for i = 1:size(cipherText,1)
    decipher(i) = cipherText(i,2) * cipherText(i,1)^(-privateKey);
end

disp('Decrypted text:')
disp(dec2char(decipher.x,bitsPerChar,m));

function text = dec2char(msg,bitsPerChar,m)
binDecipherText = int2bit(msg,m);
text = char(bit2int(binDecipherText(1:end-
mod(numel(binDecipherText),bitsPerChar)),bitsPerChar));
end

```


Procedure of the code:

1. Find a group generator and primitive polynomial. Create a repeating outcome by changing the seed of the random number generator.
2. Then create the public and private keys. Then the original text is entered and displayed.
3. The message should be converted to binary, then grouped every m bits. ASCII characters are used in the message. Seven bits per character are adequate because there are 128 characters in the ASCII table.
4. Using the binary data encrypt the data text.
5. The encrypted text is displayed.
6. Decrypt the encrypted text using the supporting function `de2char` which is used to convert binary bits data and then convert to char.
7. The decrypted data which is the original text is displayed.

Results and discussion:

The screenshot displays the MATLAB R2022a - academic use environment. The Editor window shows a script named `DSP_JCOMP.m` with the following code:

```
1 clc;
2 close all;
3 clear all;
4
5 m = 15;
6 q = 2^m;
7
8 polynomial = primpoly(m,'nodisplay');
9
10 primeFactors = unique(factor(2^m-1));
11 rng(123456);
12 while 1
```

The Command Window shows the output of the script:

```
{1x1 gf} {1x1 gf} {[32771]}

32771

Enter the text to be encrypted:
-->Digital signal processing

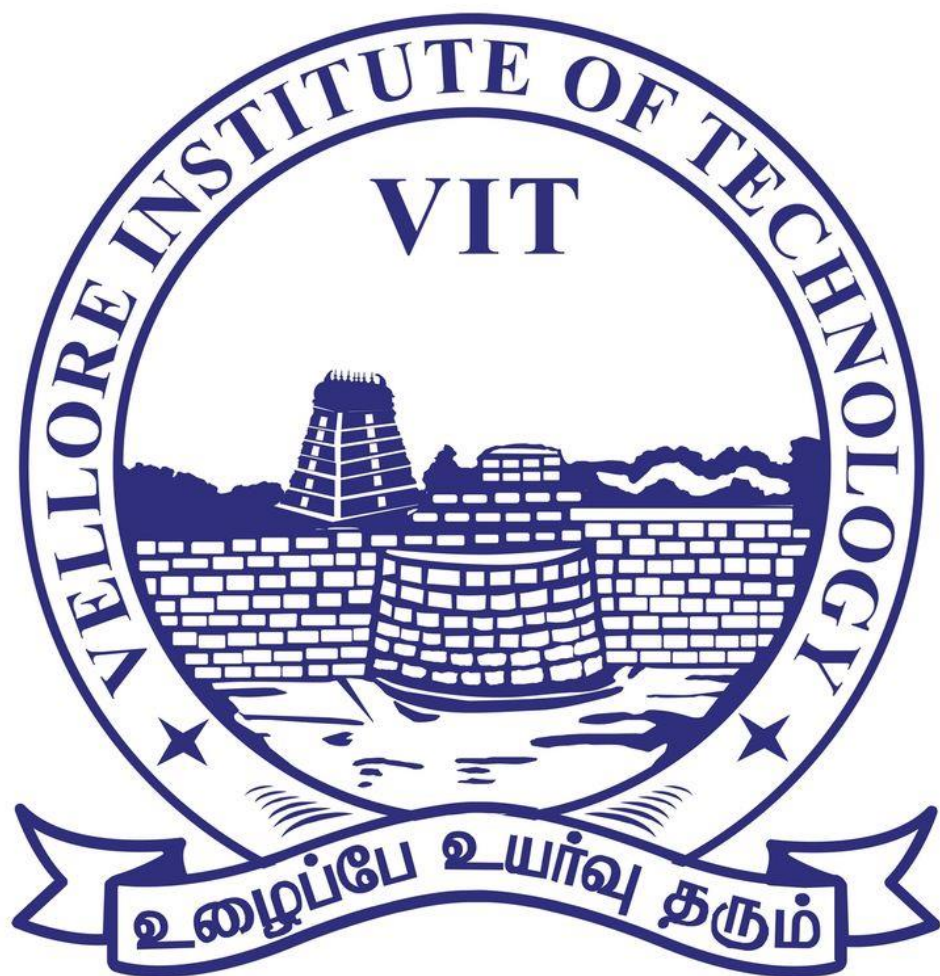
Entered text:
Digital signal processing

Encrypted text:
V_oQ,#[ - BhQl
0[_ydfp

Decrypted text:
Digital signal processing
fx >>
```

A single plaintext can be encrypted to a large number of different ciphertexts thanks to the probabilistic nature of El Gamal encryption. As a result, a general El Gamal encryption results

in a 1:2 expansion of the size of the ciphertext over the plaintext. Two exponentiations are necessary for El Gamal encryption, but they are independent of the message and can be computed beforehand if necessary. One exponentiation and one computation of a group inverse are needed for decryption, although they can both be readily combined into a single exponentiation.



ECE2006 Digital Signal Processing Review - III

TOPIC:

Encryption of Speech Signals Using El-Gamal's method and elliptical curve method

Team Members:

Leela Praneeth - 20BEC0771

Ankit Raj - 20BEC0766

Nagaraju S M - 20BEC0278

Under the guidance of:

Dr. Kalaivani. S

Elliptical curve cryptography:

The elliptical curve despite its name is different from an ellipse. In ECC we use an elliptical curve over a finite field, where it is a cubic function and the solutions (x,y). Victor Miller and Neal Koblitz first put it forward as an alternative to existing algorithms like RSA, but with a smaller key size for the same level of security. Like any cryptosystem, ECC is a one-way function and it is easy to perform the prime multiplication but hard to get the prime factors of the resulting number, this makes brute force hacking and guessing of the keys really improbable.

We use Diffie–Hellman key exchange to generate a shared key from the private keys of the sender and recipient along with their public keys and in this case the general information available such as the parameters a, b, and the prime number p of the elliptical curve over a finite field.

The above cryptographic techniques work in tandem to provide an easy-to-use and secure way to transmit data over the internet which is able to fend off cyber attacks and is up to modern standards. ECC using Diffie-Hellman is a perfect example of harmonious integration of age-old proven methodologies with new innovations that help the technology exceed the sum of its parts.

Algorithm:

- Step 1:1. Input X is taken as an audio file.
- Step 2:2. The message m, i.e. each value of audio file X, can be turned into the coordinate (Xm,Ym) so that it can be used as a point on an elliptical curve.

$$X_m = m * K + J, \quad J = 0, 1, 2, 3$$

$$Y_m = x^3 + ax + b$$

Where; message is m, Random Positive integer is K, Square modulo P is (Xm,Ym), where P is the prime number and $P \nmid K * m$.

- Step 3:3. Encryption/ Decryption system needs a point on G and an elliptic group $E_p(a,b)$. A secret integer s is chosen by the user and is used to compute $Q=s.G$. User B's public key comprises $E_p(a,b)$, and the points G & Q, while private key is s. To encrypt plus transmit message P_m to user B, user A selects a random positive integer k and generates the ciphertext C_m consisting of the pair of points. $C_m=\{kG, P_m+kQ\}$
- Step 4:4. Ciphertext is decrypted using the method $\{P_m + kQ \text{ s.}(kG) = P_m + k(s.G) \text{ s.}(kG)\} = P_m$

Code:

```
clc
close all
clear all

q = 751;
a = 17;
b = 38;
n = 11;

Eq_a_b = generate_elliptic_curve(a,b,q,0,1000);
G_x = Eq_a_b(n,1);
G_y = Eq_a_b(n,2);

% Ankit
n_ankit = input('Ankit_Private_key:');
p_ankit_x = G_x;
p_ankit_y = G_y;

for i = 2:n_ankit
    P = elliptic_curve_add(G_x,G_y,p_ankit_x,p_ankit_y,a,b,q);
    p_ankit_x = P(1);
    p_ankit_y = P(2);
end

%Praneeth
n_praneeth = input('Praneeth_Private_key:');
p_praneeth_x = G_x;
p_praneeth_y = G_y;

% multiply n_praneeth times
for i = 2:n_praneeth
    P2 = elliptic_curve_add(G_x,G_y,p_praneeth_x,p_praneeth_y,a,b,q);
    p_praneeth_x = P2(1);
    p_praneeth_y = P2(2);
end
```

```

end

% key exchange

% Ankit
% K
k1_x = p_praneeth_x;
k1_y = p_praneeth_y;

for i = 2:n_ankit
    P3 = elliptic_curve_add(p_praneeth_x,p_praneeth_y,k1_x,k1_y,a,b,q);
    k1_x = P3(1);
    k1_y = P3(2);
end

% Praneeth
% K
k2_x = p_ankit_x;
k2_y = p_ankit_y;

for i = 2:n_praneeth
    P4 = elliptic_curve_add(p_ankit_x,p_ankit_y,k2_x,k2_y,a,b,q);
    k2_x = P4(1);
    k2_y = P4(2);
end

fprintf('a=%s, b=%s, q=%s, n=%s, n_ankit=%s, n_praneeth=%s\n', ...
    num2str(a),num2str(b),num2str(q),num2str(n),num2str(n_ankit),num2str(n_praneeth));
fprintf('Shared key of Ankit: (%s,%s)\n',num2str(k1_x),num2str(k1_y));
fprintf('Shared key of Praneeth: (%s,%s)\n',num2str(k2_x),num2str(k2_y));

fs=8000;
sample = [1,2.75*fs];
[msg,fs] = audioread('addf8-Alaw-GW.wav',sample);
info = audioinfo('addf8-Alaw-GW.wav')
plot(msg);
title('message signal');
N =length(msg);
r =msg;

for i = 1:N
    if r(i) >= -0.1
        r(i) = 1;
    else

```

```

        r(i) = 0;
    end
end

key1 = char(inputdlg('encrypt key'));
key2 = char(inputdlg('decrypt key'));

if key1==key2
    out=1;
else
    out=msg(1:length(msg)).*msg;
end
[r_encrypt,r_length] = DES_Encrypt(r,key1);
r_decrypt = DES_Decrypt(r_encrypt,key2,r_length);

count = 100;
R = zeros(1,length(r)*count);
R_encrypt = zeros(1,length(r_encrypt)*count);
R_decrypt = zeros(1,length(r_decrypt)*count);
for i = 1:length(r)*count
    R(i) = r(((i-1)-mod((i-1),count))/count+1);
end
for i = 1:length(r_encrypt)*count
    R_encrypt(i) = r_encrypt(((i-1)-mod((i-1),count))/count+1);
end
for i = 1:length(r_decrypt)*count
    R_decrypt(i) = r_decrypt(((i-1)-mod((i-1),count))/count+1);
end
figure;
subplot(3,1,1)
plot(0:1/count:length(r)-1/count,R,'b','LineWidth',2)
% axis([0,500,-1,2])
title('digital')
grid on

subplot(3,1,2)
plot(0:1/count:length(r_encrypt)-1/count,R_encrypt,'b','LineWidth',2)
% axis([0,50,-1,2])
title('encrypted')
subplot(3,1,3)
plot(0:1/count:length(r_decrypt)-1/count,R_decrypt,'b','LineWidth',2)
% axis([0,50,-1,2])
title('decrypted')
grid on

```

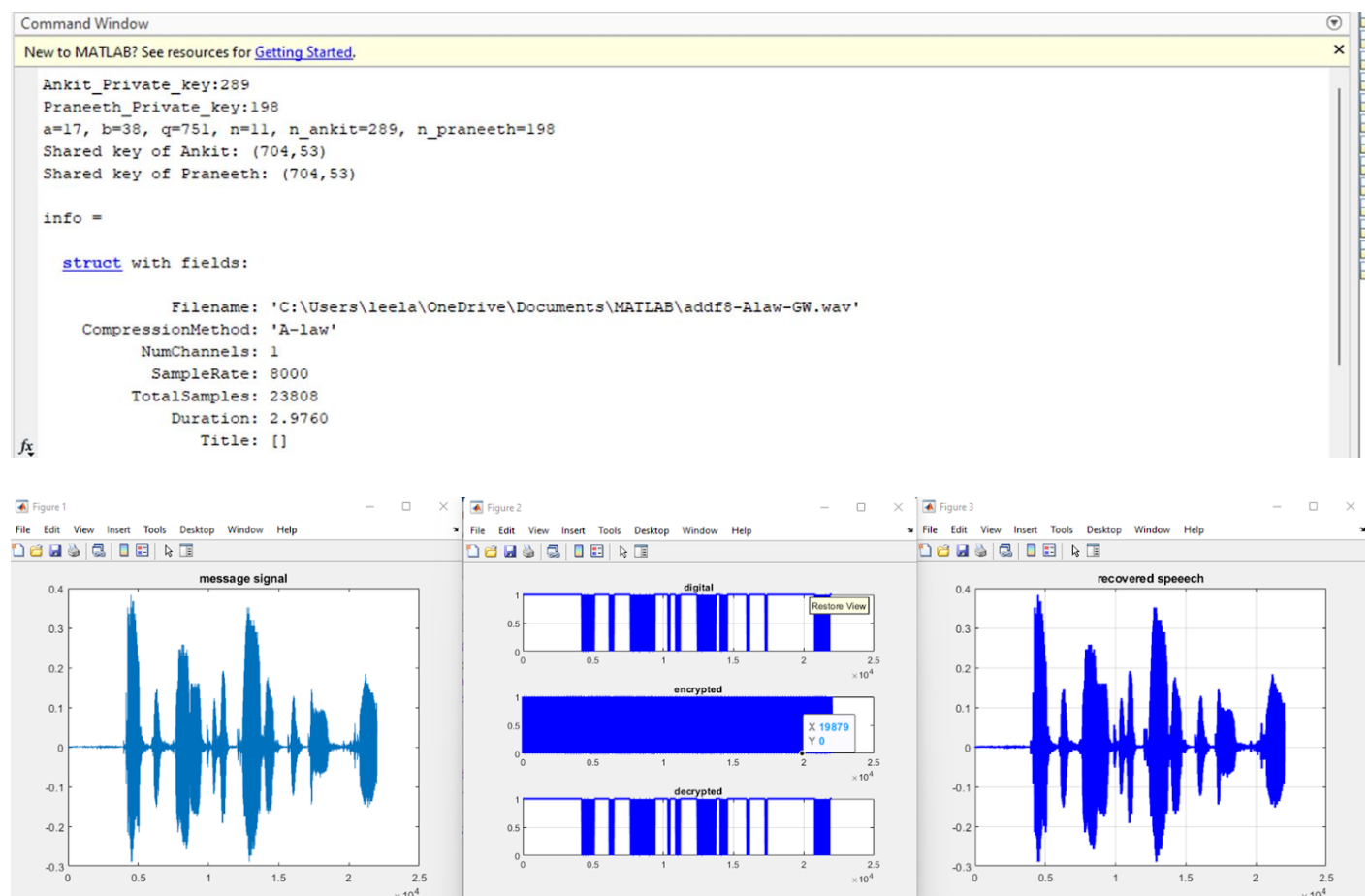


```

figure;
plot(msg.*out,'b','LineWidth',2)
% axis([0,500,-1,2])
title('recovered speech')
grid on

```

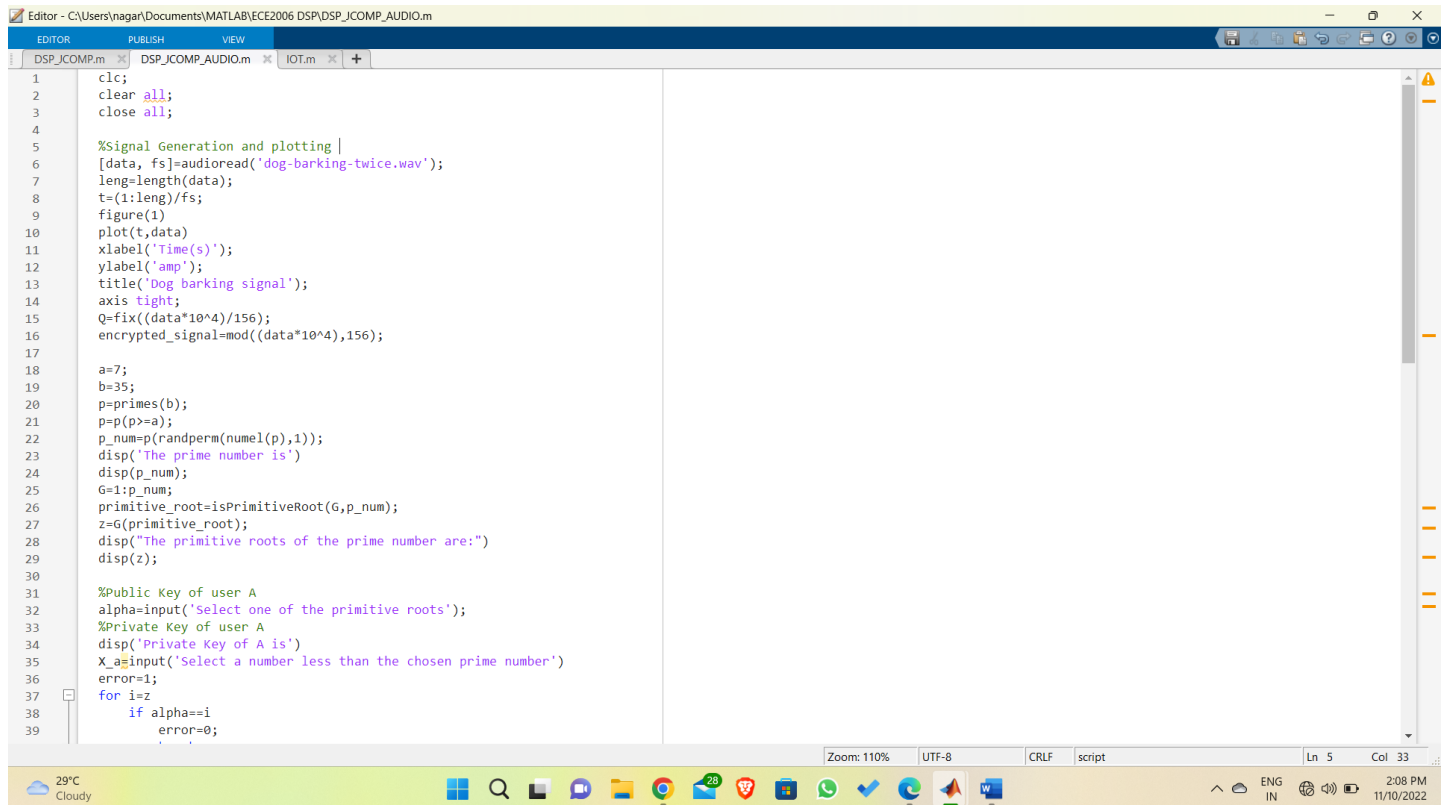
Output:



El Gamal Cryptography:

Data communication and data protection depend heavily on security. It helps prevent unwanted access to sensitive data that could lead to data loss or alteration by unidentified parties, making data transmission unsafe. El Gamal encryption uses public key cryptography. It encrypts the message and uses asymmetric key encryption for two-party communication. This cryptosystem is based on the fact that it is quite challenging to calculate g^ak even if we know g^k and g^a , the discrete logarithm in the cyclic group. If we want to understand the whole picture, we have to go step by step by actually encrypting and decrypting the messages. We will use the example of two peers who are willing to use the El Gamal algorithm to communicate data in a secure manner.

Code:

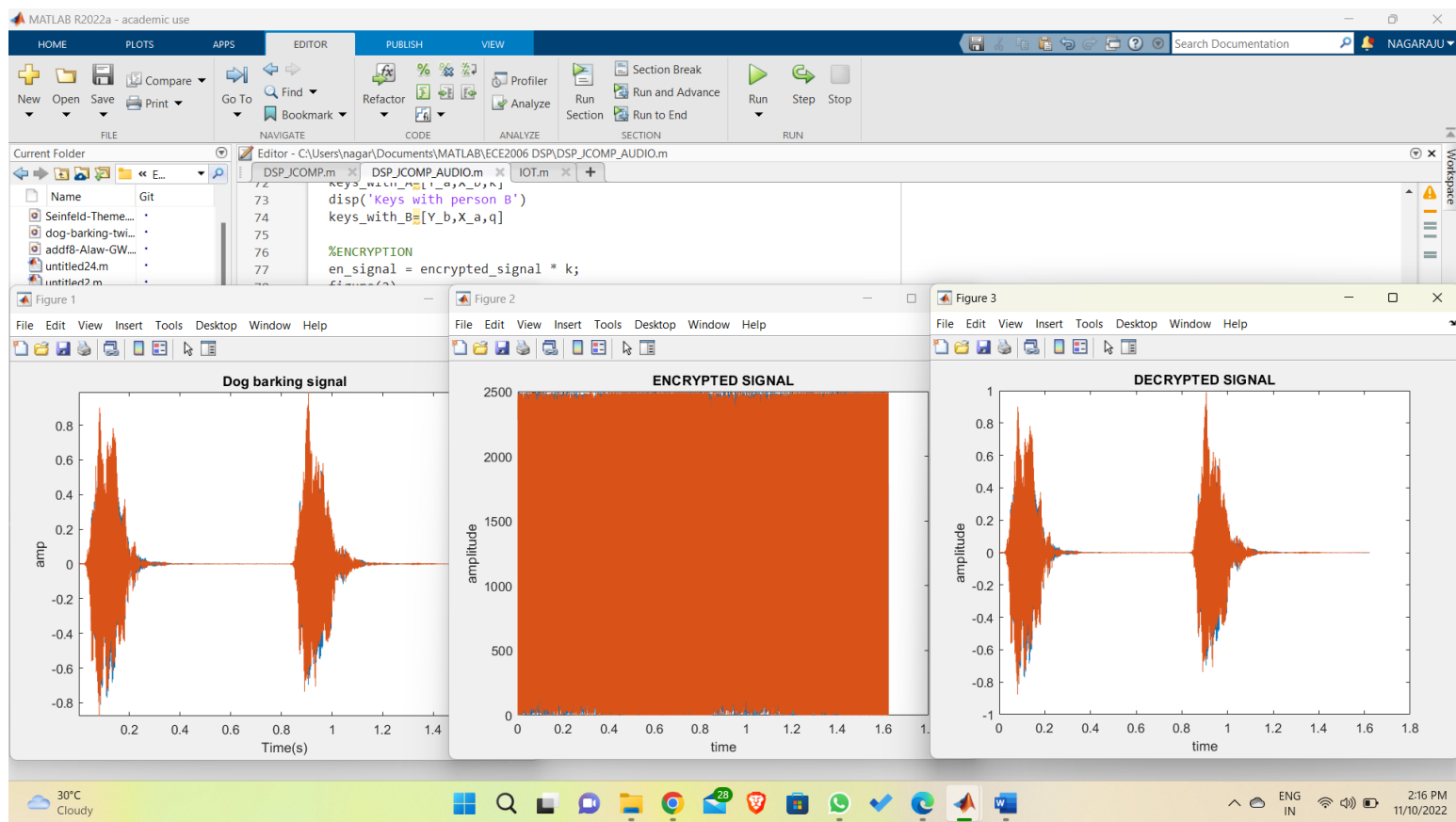


The image shows a MATLAB Editor window with a script titled 'DSP_JCOMP_AUDIO.m'. The script performs the following tasks:

- Initializes the environment with `clc;`, `clear all;`, and `close all;`.
- Generates and plots a signal: `[data, fs]=audioread('dog-barking-twice.wav');`, `leng=length(data);`, `t=(1:leng)/fs;`, `figure(1)`, `plot(t,data)`, `xlabel('Time(s)');`, `ylabel('amp');`, `title('Dog barking signal');`, `axis tight;`.
- Encrypts the signal: `Q=fix((data*10^4)/156);`, `encrypted_signal=mod((data*10^4),156);`.
- Generates a prime number `p` and its primitive root `G`: `a=7;`, `b=35;`, `p=primes(b);`, `p=p(p>a);`, `p_num=p(randperm(numel(p),1));`, `disp('The prime number is')`, `disp(p_num);`, `G=1:p_num;`, `primitive_root=isPrimitiveRoot(G,p_num);`, `z=G(primitive_root);`, `disp('The primitive roots of the prime number are:')`, `disp(z);`.
- Implements a key exchange process: `%Public Key of user A`, `alpha=input('Select one of the primitive roots');`, `%Private Key of user A`, `disp('Private Key of A is')`, `X_a=input('Select a number less than the chosen prime number')`, `error=1;`, `for i=z`, `if alpha==i`, `error=0;`.

The script is displayed in a MATLAB Editor window with a toolbar at the top (EDITOR, PUBLISH, VIEW) and a status bar at the bottom showing zoom (110%), encoding (UTF-8), line endings (CRLF), and script type (script). The Windows taskbar at the bottom shows the date and time as 2:08 PM on 11/10/2022.

Results:



Design Concept:

Cryptography is a really important part of our daily lives due to our constant reliance on the internet and the threat of cyber attacks from malicious third parties. Anything we transmit that is either text, images, or audio.

Audio signals are becoming more and more important with the introduction and popularization of remote work and home automation. There are many purposes for which we need a safe transmission of speech and audio of a conversation or the transcript of a meeting. thus arises the many encryption schemes and algorithms that aid in this aspect.

For our purpose here we will take an audio signal with speech and take down its transcript; these are encrypted with different encryption schemes. We are using ECC for the encryption of the audio signal as it is a viable and efficient option while El-Gamal's method is used for the encryption of the transcript text.

When the encryption of messages in both these forms is safely done, assuming the necessary keys are all present with the senders and recipients they are transmitted over a channel between the concerned parties.

Timeline:

- August 1-15: Work done on researching various fields in Digital signal processing and choosing an appropriate field to base our project and this research on. Scouring through all respected and esteemed journals was done through this time and the recent and emerging trends in this fields were noted.
- August 15-31: When the topic and base papers were selected the work on the project began immediately, we were thinking of the various real-world applications possible with this research as this was already a well-researched field optimization and implementation was our biggest task.
- September 10-20: We started looking for tools and inputs required, we chose MATLAB as it was the most intuitive and the program with which we had the most experience. After this we worked on the algorithm for the flow of the code and started iterating to reduce any errors, make the code flow better and take less time.
- September 20-30: We worked on the El-Gamal's Scheme first. We wanted to first make a transcript-based Cypher text generator which would successfully encrypt any transcribed speech.
- October 1-10: Then We worked on the other encryption scheme, that being the elliptical curve cryptography. We collected all the theory available on this scheme and were able to make a decision on an ECC scheme with Diffie Hellman Key exchange would be both efficient and relatively secure and add a layer of DES for further protection, DES was chosen over AES in demonstration due to the efficiency it provides over a 128-bit key in AES.
- October 20-30: We tweaked the El-Gamal's algorithm to also encrypted any audio file along with the text based transcribed message to be communicated.
- November 1-10: Final Tweaks and the compilation of reports, information and the different modules that each of our team mates worked on was done.

Team Contributions:

- 1) Nagaraju S M - 20BEC0278:
 - a) Worked on the El-Gamal's Text based encryption scheme
 - b) Worked on the algorithmic and flow chart levels to make the scheme work on message points that are hard to extract
 - c) Worked on Digitizing signals to make them easier to transmit
 - d) Worked on the scheme for encrypting audio along with the transcript
- 2) Ankit Raj - 20BEC0766:
 - a) Helped on the El-Gamal's scheme for audio signals
 - b) Worked on the key mechanism for El-Gamal's
 - c) Worked on making the algorithm and code flow for Elliptical curve cryptography
 - d) Helped in the Digitizing of audio signal with speech.
- 3) Leela Praneeth – 20BEC0771:
 - a) Worked on the Diffie Hellman key exchange using the elliptical curves to generate a shared key
 - b) Worked on Generating a point on the elliptical curve from the message signal
 - c) Worked on making the cypher points
 - d) Worked on using DES to further encrypt the digital signal.

