

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA VIỄN THÔNG I



BÁO CÁO BÀI TẬP LỚN
MÔ PHỎNG HỆ THỐNG TRUYỀN THÔNG

Giảng viên	:Ngô Thị Thu Trang
Nhóm lớp học	:Nhóm 5
Họ và tên	:Nguyễn Di Đan
Mã sinh viên	:B20DCVT086
Lớp	:D20CQVT06-B

Hà Nội – 2023

MỤC LỤC

Nhiệm vụ 1	3
1. Mô tả nhiệm vụ.	3
2. Yêu cầu.....	3
a) Mô tả phương pháp thực hiện chuyển đổi file nhạc yêu cầu thành chuỗi nhị phân và ngược lại.	3
b) Viết chương trình MATLAB.....	4
c) Xác định các thông số về file nhạc cần chuyển đổi.....	6
Nhiệm vụ 2	7
1. Mô tả nhiệm vụ:	7
2. Yêu cầu:	7
a) Mô tả hệ thống mô phỏng bằng sơ đồ khối và xác định các tham số của hệ thống.....	7
b) Viết chương trình MATLAB.....	8
c) Biểu diễn biểu đồ chòm sao, dạng sóng tín hiệu, mẫu mắt và phổ của tín hiệu.	9
d) So sánh file nhạc được khôi phục sau khi truyền qua hệ thống mô phỏng tại các mức SNR yêu cầu.....	9

NHIỆM VỤ 1

1. Mô tả nhiệm vụ.

Đọc và xử lý nguồn tin là một file nhạc AssignmentD20.wav. File này có thể lấy về theo địa chỉ dưới đây:

https://drive.google.com/file/d/1fown3-91kGSUhlZA6L8eIjS1Oh_JCjyW/view?usp=sharing

Thực hiện đọc và chuyển đổi file nhạc yêu cầu thành chuỗi bit nhị phân làm nguồn tin đầu vào cho nhiệm vụ 2 và thực hiện ngược lại chuyển đổi chuỗi bit nhị phân thu được thành file nhạc.

2. Yêu cầu

a) Mô tả phương pháp thực hiện chuyển đổi file nhạc yêu cầu thành chuỗi nhị phân và ngược lại.

- Chuyển đổi file nhạc yêu cầu thành chuỗi bit nhị phân:
 - Xác định các thông số của file âm thanh cần chuyển đổi bằng hàm *audioinfo* trong thư viện Audio Processing Toolbox của MATLAB.
 - Đọc file âm thanh vào 1 mảng dữ liệu số bằng việc dùng hàm *audioread* trong thư viện Audio Processing Toolbox của MATLAB.
 - Chuyển đổi mảng dữ liệu âm thanh thu được thành mảng dữ liệu số và có kiểu dữ liệu giống với file âm thanh ban đầu (unit16).
 - Chuyển đổi mảng dữ liệu số thành chuỗi bit nhị phân bằng việc sử dụng hàm *dec2bin* trong MATLAB.
- Chuyển đổi chuỗi nhị phân thành file âm thanh.
 - Chuyển đổi chuỗi bit nhị phân thành các mảng số nguyên có các giá trị nhị phân [0,1].
 - Chuyển đổi mảng nhị phân này thành các mảng dữ liệu có kiểu (unit16) bằng cách sử dụng hàm *bi2de*.
 - Tiếp tục chuyển đổi mảng dữ liệu số này sang mảng dữ liệu âm thanh.
 - Ghi mảng dữ liệu âm thanh ra cùng với tần số cùng với tần số của file gốc để tạo thành 1 file âm thanh mới bằng cách sử dụng hàm *audiowrite* trong thư viện Audio Processing Toolbox của MATLAB.
 - Kiểm tra các thông số của file âm thanh mới có giống với file âm thanh gốc không bằng hàm *audioinfo* trong thư viện Audio Processing Toolbox của MATLAB.

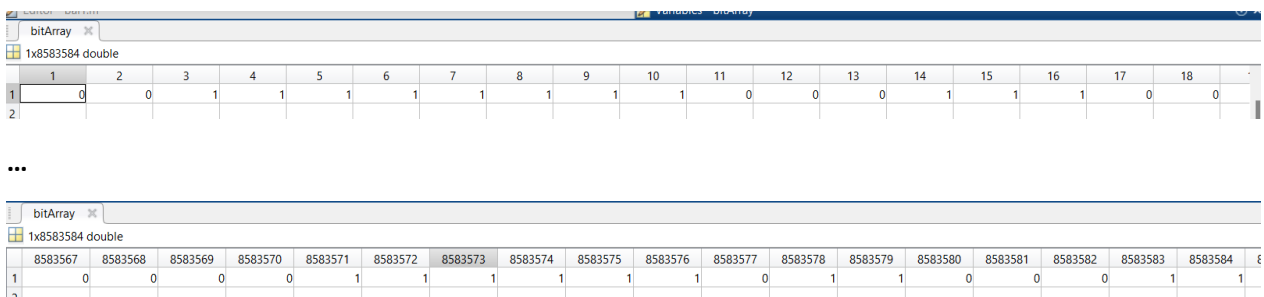
b) Viết chương trình MATLAB.

Thực hiện chuyển đổi file nhạc *AssignmentD20.wav* thành chuỗi tín hiệu nhị phân và ngược lại. Có thể xây dựng dưới dạng hàm chuyển đổi để sử dụng trong các chương trình khác.

- Code:

```
function [bitArray, bitString, originInfo, newInfo] = bail(fileName)
% Đọc và lưu giữ liệu âm thanh vào mảng
[samples, fs] = audioread(fileName);
% Đọc các tham số của file âm thanh gốc
originInfo = audiointo(fileName);
title = originInfo.Title;
artist = originInfo.Artist;
% Lưu kích thước ma trận samples vừa đọc được từ âm thanh gốc
[M, N] = size(samples);
% Chuyển đổi dữ liệu âm thanh sang dữ liệu số rồi chuyển về chuỗi bit nhị phân có độ dài 16 bit
bitString = dec2bin(round((samples + 1) * (2^15-1)/2), 16)';
% Vì samples có giá trị [-1,1] => samples + 1 dùng để đưa các giá trị về các số dương
bitArray = bitString(:)' - '0'; % Chuyển chuỗi bit nhị phân sang vector chứa các số 0 1
% Chuyển đổi chuỗi bit nhị phân thành dãy số nguyên kiểu unit 16
samples = bi2de(reshape(bitArray, 16, []), 'left-msb');
% Chuyển đổi ngược lại dãy số nguyên thành giá trị âm thanh có dạng như mảng dữ liệu ban đầu khi đọc file âm thanh
samples = (samples/(2^15-1))*2-1;
% Trả về kích thước ban đầu của mảng dữ liệu samples
samples = reshape(samples, [], N);
% Ghi dữ liệu âm thanh vào file mới với các thông số từ file gốc
audiowrite('newfile.wav', samples, fs, 'Title', title, 'Artist', artist);
% Kiểm tra thông số của file âm thanh mới đã giống với file gốc chưa
newInfo = audiointo('newfile.wav');
```

- Kết quả sau khi thực hiện hàm *bail*:



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	0	0	1	1	1	1	1	1	1	0	0	0	1	1	1	0	0
2																	

...

8583567	8583568	8583569	8583570	8583571	8583572	8583573	8583574	8583575	8583576	8583577	8583578	8583579	8583580	8583581	8583582	8583583	8583584
1	0	0	0	0	1	1	1	1	1	0	1	1	0	0	0	1	1
2																	

c) **Xác định các thông số về file nhạc cần chuyển đổi.**

Một số hàm công cụ tiện ích trong thư viện Audio Processing Toolbox có thể cần sử dụng để thực hiện nhiệm vụ (để biết chi tiết hơn về từng câu lệnh, sử dụng help):

audioread – đọc dữ liệu từ file nhạc.

audiowrite – ghi file nhạc.

audioinfo – thông tin về file nhạc.

unit16 – chuyển đổi kiểu dữ liệu về unit16.

Sử dụng lệnh *originInfo = audioinfo('AssignmentD20.wav');* ta được các thông số của file nhạc cần chuyển đổi:

```
originInfo =
```

```
struct with fields:
```

```
Filename: 'D:\User\Downloads\Documents\KI6\MoPhongHeThongTruyenThong\Assignment\AssignmentD20.wav'
CompressionMethod: 'Uncompressed'
NumChannels: 2
SampleRate: 8000
TotalSamples: 268237
Duration: 33.5296
Title: 'Impact Moderato'
Comment: []
Artist: 'Kevin MacLeod'
BitsPerSample: 16
```

- *Filename*: Tên và đường dẫn của file mới.
- *CompressionMethod*: Phương pháp nén file, ở đây là "Uncompressed" tức không nén.
- *NumChannels*: Số kênh của file, ở đây là 2 kênh (đã được ghi rõ trong mã code MATLAB).
- *SampleRate*: Tốc độ lấy mẫu của file, ở đây là 8000 Hz (đã được ghi rõ trong mã code MATLAB).
- *TotalSamples*: Tổng số mẫu trong file, ở đây là 268237 mẫu.
- *Duration*: Thời lượng của file, ở đây là 33.5296 giây.
- *Title*: Tiêu đề của file, ở đây là "Impact Moderato".
- *Comment*: Chú thích (nếu có) của file.
- *Artist*: Tác giả của file, ở đây là "Kevin MacLeod".
- *BitsPerSample*: Số bit trên mẫu âm thanh của file, ở đây là 16 bit.

Nhiệm vụ 2

1. Mô tả nhiệm vụ:

Mô phỏng hệ thống truyền dẫn số tại tốc độ dữ liệu N Mb/s, với N là số cuối cùng của mã số sinh viên (nếu số đó là 0 thì sẽ lựa chọn số liền kề bên cạnh). Nguồn tin của hệ thống được lấy từ file nhạc thực hiện trong nhiệm vụ 1, trong trường hợp không thực hiện lấy nguồn tin từ nhiệm vụ 1 được hãy thay thế bằng một chuỗi tín hiệu nhị phân ngẫu nhiên tương đương. Mỗi sinh viên sẽ lựa chọn một trong các kỹ thuật điều chế sau cho hệ thống mô phỏng của mình:

- Điều chế M-QAM nếu số cuối cùng trong mã sinh viên là lẻ, với $M = 16$ nếu số liền kề là lẻ và $M = 4$ nếu số liền kề là chẵn.
 - Điều chế M-DPSK nếu số cuối cùng trong mã sinh viên là chẵn, với $M = 4$ nếu số liền kề là lẻ và $M = 2$ nếu số liền kề là chẵn.
- Sử dụng mô hình mô phỏng tương đương băng gốc, tín hiệu phát có thể được biểu diễn như sau:

$$s(t) = \left[\sum_{k=-\infty}^{\infty} d_k p(t - kT_{sym}) \right] e^{j\Phi_0}$$

trong đó d_k là các ký hiệu (symbol) phức được xác định từ chuỗi bản tin đầu vào và kỹ thuật điều chế; T_{sym} là chu kỳ của symbol; Φ_0 là pha của tín hiệu phát và $p(t)$ xác định dạng xung được phát, với:

$$p(t) = \sqrt{\frac{2E_s}{T_{sym}}} \left[1 - \cos\left(\frac{2\pi t}{T_{sym}}\right) \right] \quad \text{cho tín hiệu M-DPSK}$$

$$p(t) = \sqrt{\frac{2E_s}{T_{sym}}} \left[1 - \sin\left(\frac{2\pi t}{T_{sym}}\right) \right] \quad \text{cho tín hiệu M-QAM}$$

với E_s là năng lượng mỗi symbol.

2. Yêu cầu.

- a) Mô tả hệ thống mô phỏng bằng sơ đồ khối và xác định các tham số của hệ thống, có thể bao gồm cả các bộ lọc sử dụng để có được bộ thu tối ưu.

- Sơ đồ khối hệ thống 2-DPSK:



Sử dụng bộ điều chế 2-DPSK. Các chuỗi bit đầu vào sẽ được tạo thành các xung, sau đó các xung này sẽ được đưa vào bộ điều chế 2-DPSK và được đưa lên kênh AWGN để tới được máy thu. Tại máy thu các tín hiệu sẽ được giải điều chế và đưa qua bộ lọc để khôi phục chuỗi bit ban đầu.

- Xác định tham số của hệ thống 2-DPSK:

- Tốc độ dữ liệu $R_b = 6 \text{ Mb/s}$.
- Tốc độ symbol $R_s = R_b/2 = 3 \text{ Mb/s}$.
- Chu kỳ symbol $T_{\text{sym}} = 1/R_s = 333.33 \text{ ns}$.
- Năng lượng symbol $E_s = 1 \text{ J}$.
- Tần số lấy mẫu $F_s = 12 \text{ MHz}$.
- Tần số sóng mang $F_c = 4 \text{ MHz}$.
- $\text{SNR} = 5 \text{ dB}$.
- BER tại các mức SNR $(E_s/N_0) = [2 \ 5 \ 10] \text{ dB}$.
- Pha ban đầu của tín hiệu $\phi_0 = 0$.

b) Bằng việc sử dụng MATLAB, viết chương trình mô phỏng hệ thống truyền dẫn số sử dụng kỹ thuật điều chế đã lựa chọn trên kênh AWGN với nguồn tín hiệu là tín hiệu thu được ở nhiệm vụ 1. Ước tính xác suất lỗi tại các mức tỉ số tín hiệu trên nhiễu SNR lần lượt bằng 2, 5 và 10 dB theo phương pháp Monte Carlo.

c) Biểu diễn biểu đồ chòm sao, dạng sóng tín hiệu, mẫu mắt và phổ của tín hiệu tại các điểm sau trên hệ thống: đầu ra bộ điều chế, sau khi truyền qua kênh AWGN tại $\text{SNR} = 5\text{dB}$, sau khi được xử lý và khôi phục tại bộ thu.

d) So sánh file nhạc được khôi phục sau khi truyền qua hệ thống mô phỏng tại các mức SNR yêu cầu.