ECE4634 - Digital Communications
November 1, 2020
Nima Mohammadi
nimamo@vt.edu

Modulation

Design Project 2
Technical Memorandum

I. Introduction

In this technical report we will investigate the impact modulation on transmission of voice signals. Different modulation schemes are examined to show the trade-off between bandwidth efficiency and energy efficiency. Again, we are working with the 5.86 second voice clip as our input signal.

We start by converting our input "analog" signal to a digital one through sampling and quantization. Our sampling rate of $f_s = 8192Hz$ was shown in the last report to be adequate for the voice recording based on Nyquist sampling rate and the fact that a bandwidth of 3.685kHz holds 99.9% energy of this signal. We are also using $\mu-$law to quantize the signal value into 64 levels represented by 6 bits. We have shown previously that this nonlinear quantization scheme outperform a linear method and both qualitatively and quantitatively confirms its efficiency with the number of required bits compared to a naïve linear approach. Upon this process we transform the signal to a bitstream of 0s and 1s.

In the next sections, we examine how PSK and QAM can be utilized for modulation of this signal and present our observations on possible designs.

II. Modulation

We convert the baseband signal comprised of a binary string to a bandpass signal using sinusoidal carriers. In the report, we investigate binary and M-ary Phase-shift keying (PSK) which convey the data by modulating the phase of the constant frequency carrier wave. BPSK is the binary form of this family of phase modulation schemes, followed by QPSK which can encode two bits per symbol. Gray coding can be utilized to minimize the bit error rate (BER). The impact of multi-level modulation can be stated based on this two modulation schemes: In theory, QPSK can halve the bandwidth requirement of BPSK while maintaining the data rate (or doubling the data rate while maintaining the bandwidth). PSK can be regarded as a special case of Quadrature amplitude modulation (QAM) where the amplitude of the transmitted signal is a constant, but the phase varies.

II.I Bug Found in Implementation of Modulation Schemes

Inspecting the code provided with the assignment I found a nasty bug in implementation of the modulation schemes. I have contacted Prof. Liu and told him about the bug. Assume that I want to perform modulation on all 4 possible symbols for 2 bit per symbol. The code snippet below should carry out this:

```
>> PhaseMod([0 0 0 1 1 0 1 1], 2, 0)
ans =
   0.0000 + 1.0000i
   1.0000 + 0.0000i
   0.0000 + 1.0000i
  -0.0000 - 1.0000i
```

This, however, to my utter astonishment, maps the bitstream to only 3 constellation points! Upon further inspection, I realized that it is due to the incorrect reshaping of the bit array in the implementation of the modulation scheme. The culprit is this line in your PhaseMode.m:

```
b = reshape(x,a/l,l);
```

Which internally transforms the vector into this matrix:

```
b =
   0    1
   0    0
   0    1
   1    1
```

To fix this behavior, I have changed that line into this:

```
b = reshape(x, l, a/l)';
```

I found out about this since I wanted to show that although Gray coding does not change the placement of the constellation points, it "relabels" them into a different permutation.

III. Communication Channel

The signal goes through a channel characterized by Additive White Gaussian Noise (AWGN). We were instructed to add noise to this channel with an SNR per bit of 7.5dB but we need to convert this value as the provided `AddNoise()` function takes linear `SNRperBit`.

IV. Constellation Diagrams

Using constellation diagrams we can better perceive how the modulated signal is projected to the 2-dimensional complex plane for different modulation schemes. In Fig. 1, the constellation diagrams of these nine modulation schemes (four PSK modulations with and without Gray coding) are depicted. In PSK, the constellation points chosen are usually positioned with uniform angular spacing around a circle, whereas in QAM, the constellation points are usually arranged in a square grid with equal vertical and horizontal spacing. The red dots in Fig. 1 show how the actual constellation points are arranged in the constellation diagram. As it can be seen, Gray coding does not modify the placement of constellation points but actually rearranges them into a new permutation in a fashion that the most probable error from one symbol to the next produces only a single bit-error. This can be seen from the plot for 8PSK shown in Fig. 1, for example, how the symbol 7 is put at maximum distance from the symbol 0 in the Gray coding setting, whereas they are adjacent in the regular coding. Moreover, the position of symbols of the received signal are plotted in blue, showing the impact of the additive noise making them scattered probabilistically around the the actual constellation points following a bivariate Gaussian distribution.

Having the signal passed through the AWGN channel and demodulating the received signal allows us to once again use the constellation plots to identify the misidentified symbols. The result is depicted in Fig. 2 where the misclassified received symbols are shown in red. The plot shows how, as expected, these misidentified symbols are mostly located between the actual constellation points. These regions can be grasped as parts of the problem space where the demodulation phase has higher uncertainty. An interesting observation is how the number of misidentified symbols and BER are vastly different between the schemes 16PSK and 16QAM (approximately an order of magnitude), even though the have they map the input signal into an identical number of symbols. This is expected as QAM achieves a higher average distance between adjacent points as it can be vividly seen in the plot.

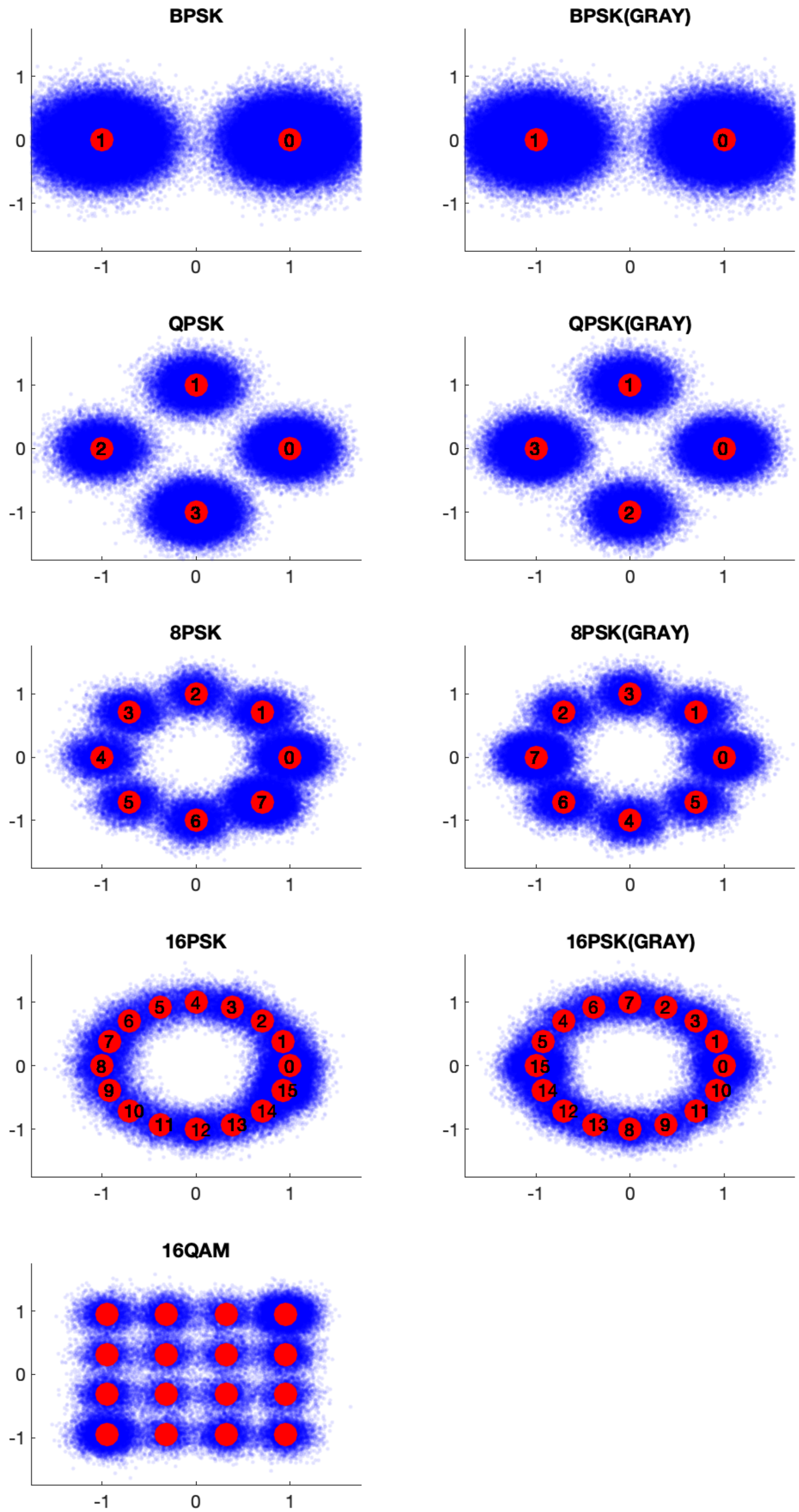

Fig. 1: The constellation diagram for different modulation schemes. The red circles with overlaying labels indicate the actual constellation points. The blue dots correspond to symbols received after signal goes through an AWGN channel

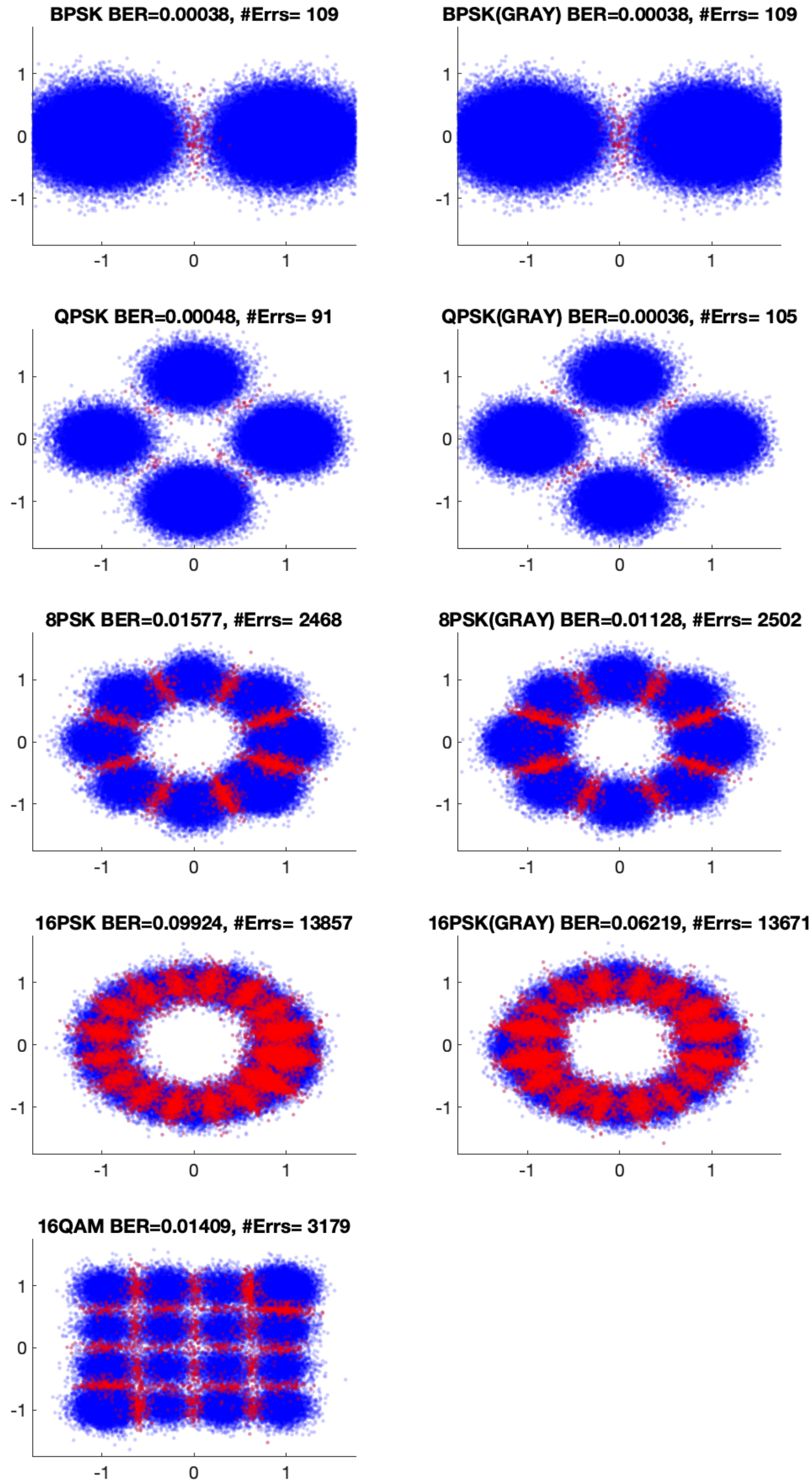

Fig. 2: The constellation diagram showing the misidentified symbols at the receiving side. The plot shows the regions the modulation scheme have higher uncertainty. The captions show the corresponding BER and number of misidentified symbols.

V. Signal-to-Noise Ratio and Bit-Error Rate

The bit error ratio (BER) can be considered as an approximate estimate of the bit error probability. The bit error ratio is the number of bit errors divided by the total number of transferred bits. This ratio is closely linked to the Signal-to-Noise-Ratio (SNR) which is measured in decibels (dB). SNR is defined as the ratio of the desired signal power to noise power. A high SNR is required for a low BER. A low SNR will have an increased BER. These two measures the reliability of the link between the transmitted and receiver and to a large extent depend on the type of modulation scheme used to transmit the data. As we discussed in the class, for a digital communication system, BER is the most meaningful criterion for performance evaluation.

We present our empirical results in Table 1. We can see that as the SNR decreases, the estimated BER increases. As it was suggested in the Fig. 1, BPSK(Gray) is redundant as it results in the same constellation points as BPSK. We can see in the table that incrementing the modulation levels decreases SNR and increases the BER. We also observe that using Gray coding (except for BPSK) results in a better SNR and BER, a phenomenon that is more pronounced with higher levels. The perceived sound qualities which were found to be consistent with the quantitative criteria are stated in the last column of Table 1. 16QAM is an interesting case here. Although 16QAM has the same bits per symbol as 16PSK, we can vividly observe that it achieved a much better performance based on both the quantitative measure and the perceived subjective voice quality.

Modulation Scheme	MSE	SNR	BER	Voice Quality
w/o transmission	0.000051	22.79	-	Very High
BPSK	0.000176	17.378442	0.000378	High
BPSK(GRAY)	0.000176	17.378442	0.000378	High
QPSK	0.000233	16.172070	0.000479	High
QPSK(GRAY)	0.000151	18.047235	0.000365	High
8PSK	0.005448	2.477835	0.015767	Medium
8PSK(GRAY)	0.003787	4.056955	0.011281	Medium
16PSK	0.030713	-5.033208	0.099240	Low
16PSK(GRAY)	0.017442	-2.575988	0.062187	Low
16QAM	0.004081	3.732619	0.014090	Medium

Table 1: Empirical performance of different modulation schemes comprised of SNR, BER and the perceived voice quality.

Notice that with only quantization and sampling of the analog signal (without transmission) we could reach an SNR of 22.79dB. To compare the amount of distortion added by converting the "analog" signal to a digital one with the error/distortion that is caused by passing through a noisy channel for each of these modulation schemes, the mean squared errors of the resulting "analog" signals are also reported in Table 1.

VI. More Performance Analyses

As we discussed in the class, the bandwidth efficiency of our modulation scheme can be improved using M-ary modulation; that is in expense of degraded energy efficiency. To calculate the theoretical bandwidth of MPSK/QAM with raised cosine pulses with a roll-off factor of $r = 0.25$, the formula below is used:

$$\begin{aligned} BW &= (1+r)R_s \\ &= (1+r)R_b/l \\ &= (1+r)n \cdot f_s/l \end{aligned}$$

The calculated values are reported in Table 2 where higher bits per symbol results in less bandwidth. Notice that Gray coding does not impact the bandwidth requirement of phase modulations.

Modulation Scheme	Theoretical BW
BPSK	61.44 kHz
QPSK	30.72 kHz
8PSK	20.48 kHz
16PSK	15.36 kHz
16QAM	15.36 kHz

Table 2: Theoretical BW requirement of each modulation scheme.

Some other aspects of these modulation schemes can be observed in Fig. 3. The left column depicts the time-domain representation of demodulated and converted-to-"analog" signal for a short period of time (i.e. ~60ms), clearly showing less reliability as we increase the modulation levels. The column in the middle shows the ordinary-scale ESDs. Notice that the horizontal axis in this column has been purposefully chosen to be identical across different modulation schemes to better depict the frequency components. The right column on the other hand holds the log-scale ESD plots for each modulation scheme where this time I have not chosen identical range for the horizontal axis to get a higher-resolution view of the spectrum.

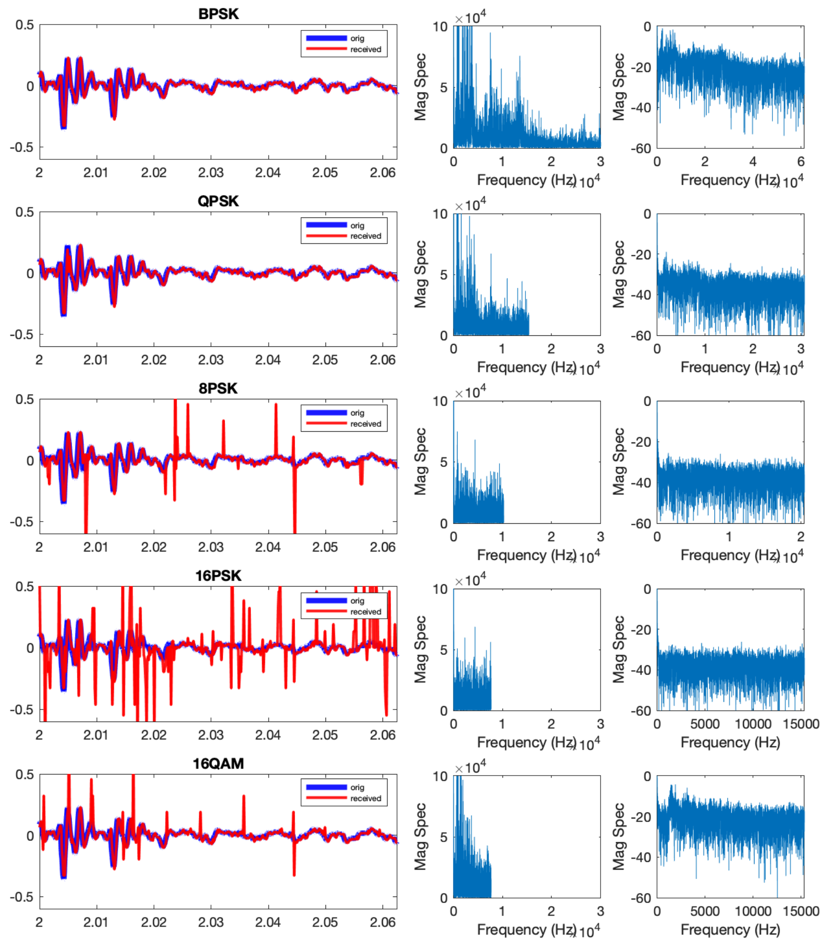

Fig. 3: The time-domain plot (left), ESD ordinary scale (middle) and ESD log scale (right)

VII. An Economical System

Choosing a modulation schemes among the ones we investigated demands a compromise to be made between the bandwidth requirement and our criteria for robustness, namely SNR and BER. As for the modulation scheme, I propose to use **16QAM**. Although 16-PSK has an appealing bandwidth requirement, it is ruled out as corresponding 16-QAM achieves the same bandwidth but with better SNR. I also rule out BPSK and QPSK as the bandwidth requirement are deemed very high. 16QAM clearly outperforms the ordinary 8PSK. Although 8PSK with Gray coding can give us negligibly better sound quality at the receiving side, however the fact that 16QAM requires less bandwidth makes it the winning modulation scheme for us. With a cost of $0.01/minute/kHz, it suggests that our cost for 16QAM would be about 15 cents per minute.

VIII. Final Thoughts

In this technical report, we examined the impacts of various modulation schemes, namely PSK and QAM, on a communication channel built for transmission of voice recordings over an AWGN channel. We based some aspects of our analysis on the constellation diagram, showing the placement of constellation points and how noisy signal scatters around these predetermined points. We also used these plots to identify the regions that have higher uncertainty using the misidentified symbols as a proxy. We empirically calculated two criteria, namely SNR and BER, and showed that they are inversely related and matched our qualitative measure of sound quality assessment. We investigated these modulation schemes more by plot segments of the time-plot and examined the spectrum and bandwidth requirements of these techniques. Overall, we came to the conclusion that a 16QAM modulation could provide us a cost-effective solution as at had appealing bandwidth requirement and shown adequate ability in term of robustness in face of an AWGN channel.

IX. Appendix (MATLAB Code)

```
% Nima Mohammadi
% Design Project 2

% As mentioned in the accompanied report, I have modified PhaseMod.m and
% PhaseDemod.m files to fix a bug in these files

clc
clear

%% Load dataset
load DesignProject1
duration = length(Original) / 65536;

%% Sampling and Quantization ~> bitstream
bits = 6;
fs = 8192;
inpsig = Analog2Digital(Original, fs, bits, 1, 255, 65536);
outsig = Digital2Analog(inpsig, bits, 1, 255);
% sound(outsig, fs);

%% Constellation plots w/ and w/o noise
mods = {'BPSK', 'BPSK(GRAY)', 'QPSK', 'QPSK(GRAY)', '8PSK', '8PSK(GRAY)',...
    '16PSK', '16PSK(GRAY)', '16QAM'};
bps_mods = {1, 1, 2, 2, 3, 3, 4, 4, 4};
tx_sigs = cell(1, 9);
rx_sigs = cell(1, 9);

set(gcf,'PaperSize',[14 15]);
set(gcf,'position',[0 0 550 1000]);

for i = 1:length(mods)
    wo_noise = reshape(de2bi(0:63, 'left-msb')', [], 1);
    labels = string(bi2de(reshape(wo_noise, bps_mods{i}, [])', 'left-msb'));
    if i < 9 % Phase modulations
        % Transmitter modulation
        tx_sigs{i} = PhaseMod(inpsig, bps_mods{i}, mod(i+1, 2));
        wo_noise_mod = PhaseMod(wo_noise, bps_mods{i}, mod(i+1, 2));

    else % 16QAM
        % Transmitter modulation
        tx_sigs{i} = QAM16_mod(inpsig);
```

```
        wo_noise_mod = QAM16_mod(wo_noise);
    end
    % Passing through AWGN channel with SNR per bit of 7.5dB
    rng(1,'philox');
    rx_sigs{i} = AddNoise(tx_sigs{i}, 5.62, bps_mods{i});

    subplot(5, 2, i);
    scatter(real(rx_sigs{i}), imag(rx_sigs{i}), 'b.',...
        'MarkerFaceAlpha', .1,'MarkerEdgeAlpha', .1)
    hold on;
    plot(real(tx_sigs{i}), imag(tx_sigs{i}),'r.',...
        'markers', 40, 'LineWidth', 3)

    plot(real(wo_noise_mod), imag(wo_noise_mod),...
        'r.', 'markers', 45, 'LineWidth', 3)
    if i < 9
        text(real(wo_noise_mod)-.06, imag(wo_noise_mod), labels);
    end

    ylim([-1.75 1.75]);
    xlim([-1.75 1.75]);
    hold off;
    title(mods{i});
end
saveas(gcf, 'plot1_constellations.pdf')

%% Demodulation
demod_sigs = cell(1, 9);
demod_sigs_analog = cell(1, 9);

for i = 1:length(mods)
    if i < 9 %Phase modulations
        demod_sigs{i} = PhaseDemod(rx_sigs{i}, bps_mods{i}, mod(i+1, 2));
        demod_sigs_analog{i} = Digital2Analog(demod_sigs{i}, bits, 1, 255);
    else % 16QAM
        demod_sigs{i} = QAM16_demod(rx_sigs{i})';
        demod_sigs_analog{i} = Digital2Analog(demod_sigs{i}, bits, 1, 255);
    end
end

%% Calculate SNRs
mod_SNR = cell(1, 9);
fprintf('%15s \t%s\n', 'Mod', 'SNR');
```

```matlab
for i = 1:length(mods)
    mod_SNR{i} = snr(Original',...
        Original'-interp(demod_sigs_analog{i}, 65536/fs));
    fprintf('%15s \t%f\n', mods{i}, mod_SNR{i});
end
fprintf('----------\n');

%% Calculate BERs
mod_BER = cell(1, 9);

fprintf('%15s \t%s\n', 'Mod', 'BER');
for i = 1:length(mods)
    [num, ratio] = biterr(inpsig, demod_sigs{i});
    mod_BER{i} = ratio;
    fprintf('%15s \t%f\n', mods{i}, mod_BER{i});
end
fprintf('----------\n');

%% Listening to received signals
i = 1; % change i from 1 to 9
sound(demod_sigs_analog{6}, fs)
% sound(outsig, fs);

%% Constellation plots for misidentified points

set(gcf,'PaperSize',[14 15]);
set(gcf,'position',[0 0 550 1000]);
for i = 1:length(mods)
    subplot(5, 2, i);
    errs = bi2de(reshape(demod_sigs{i},...
        bps_mods{i}, [])') ~= bi2de(reshape(inpsig, bps_mods{i}, [])');
    scatter(real(rx_sigs{i}), imag(rx_sigs{i}), 'b.',...
        'MarkerFaceAlpha', .2,'MarkerEdgeAlpha', .2)
    hold on;
    scatter(real(rx_sigs{i}(errs)), imag(rx_sigs{i}(errs)),...
        'r.', 'MarkerFaceAlpha', .3,'MarkerEdgeAlpha', .3)
    title(sprintf("%s BER=%.5f, #Errs= %d", mods{i}, mod_BER{i}, sum(errs)))
    ylim([-1.75 1.75]);
    xlim([-1.75 1.75]);
    hold off;
end
saveas(gcf,'plot2_constellations-misclassified.pdf')
```

```
%% Calculate theoretical BW requirements
bw_theoretical_mods = cell(1, 9);
r = .25;
fprintf('%15s \t%s\n', 'Mod', 'Theoretical BW');
for i = 1:2:length(mods)
    bw_theoretical_mods{i} = (1+r) * bits * fs / bps_mods{i};
    fprintf('%15s \t%d\n', mods{i}, bw_theoretical_mods{i});
end
fprintf('----------\n');

%% Plot time waves and ESD
dd = 16;

begin_orig = 65536*2;
end_orig = begin_orig + 65536/dd-1;

begin_sigs = fs*2;
end_sigs = begin_sigs + fs/dd-1;

begin_orig_t = 2;
end_orig_t = begin_orig_t + (end_orig - begin_orig) / 65536;

slice_orig = Original(begin_orig: end_orig);

t_orig = linspace(begin_orig_t, end_orig_t, 65536/dd);
t_sig = linspace(begin_orig_t, end_orig_t, fs/dd);


set(gcf,'PaperSize',[20 12]);
set(gcf,'position',[0,0,730,1000])

jj = 1;
for i = 1:2:length(mods)
    subplot(5, 4, [jj jj+1]);
    slice_sig = demod_sigs_analog{i}(begin_sigs: end_sigs);
    lh1 = plot(t_orig, slice_orig, 'b', 'LineWidth', 4);
    lh1.Color = [0,0,1,0.9];
    hold on
    lh2 = plot(t_sig, slice_sig, 'r', 'LineWidth', 2);
    lh2.Color = [1,0,0,0.9];
    xlim([begin_orig_t, end_orig_t])
    ylim([-.6 .5])
    title(mods{i});
```

```
    lgd = legend('orig', 'received');
    lgd.FontSize = 6;

    subplot(5, 4, jj+2);
    EnergySpectralDensity(tx_sigs{i}(1:10000),...
        bw_theoretical_mods{i}, [0 30000 0 100000], 0);
    ylabel('Mag Spec');

    subplot(5, 4, jj+3);
    EnergySpectralDensity(tx_sigs{i}(1:10000),...
        2*bw_theoretical_mods{i}, [0 bw_theoretical_mods{i} -60 0], 1);
    ylabel('Mag Spec');
    jj = jj + 4;
end
saveas(gcf,'plot3_esd.pdf')
```