Intro to Digital Communication

Lab Project

Amr Ayman Elmaghraby ***19016107***

Omar Hany Elsayed Salem ***19016102***

Omar Wael Mohamed Ali ***19016103***

March 14, 2023(Spring).

Part1: Performance of Matched filters and correlators:

MATLAB Code:

```matlab

% Simulation Parameters
num_bits = 1e5;               % number of bits
snr_range = 0:2:30;           % SNR range in dB
num_samples = 20;             % samples per waveform
sampling_instant = 20;        % sampling instant
s1_amp = 1;                   % amplitude of rectangular signal s1(t)
s2_amp = 0;                   % amplitude of zero signal s2(t)
s1_waveform=ones(1,20);
s2_waveform=zeros(1,20);

% Generate random binary data
bits = randi([0 1],1,num_bits);

% Represent each bit with waveform
waveform = reshape(repmat(bits,20,1),1,[]);

%Initialize BER vector of MF, Correlator
ber_MF=zeros(1,length(snr_range));
ber_Corr=zeros(1,length(snr_range));
ber_simple=zeros(1,length(snr_range));

% Apply noise to waveform
for snr_idx = 1:length(snr_range)
    snr = snr_range(snr_idx);
    snr_lin = 10^(snr/10);  % convert dB to linear scale

    % calculate noise power based on SNR
    s1_power = s1_amp^2;
    noise_power = s1_power/snr_lin;

    % add noise to waveform
    noisy_waveform = awgn(waveform,snr,'measured','linear');
    noisy_bits = awgn(bits,snr,'measured','linear');

    % matched filter
    filter=flip(s1_waveform-s2_waveform);
    detected_bits=bits;
    for i=0:length(bits)-1
        frame=( noisy_waveform(i*20+1:(i+1)*20) );
        % Apply convolution process in receiver
        output_samples = conv(frame,filter);
        detected_bits(i+1) = output_samples(round((length(frame)+length(filter)-1)/2 ));
    end
    % Decide whether the Rx_sequence is '1' or '0' by comparing with threshold
    threshold = 0.5;
    detected_bits = detected_bits > threshold;

    % Calculate bit error rate (BER)
    num_errors_MF = nnz(xor(bits,detected_bits));
    ber_MF(snr_idx) = num_errors_MF/num_bits;

```

```
    % correlator
    output_samples=bits.*noisy_bits;
    % Decide whether the Rx_sequence is ‘1’ or ‘0’ by comparing with threshold
    threshold=0.5;

    % Calculate bit error rate (BER)
    num_errors_Corr = nnz(xor(bits,detected_bits));
    ber_Corr(snr_idx) = num_errors_Corr/num_bits;

    %simple detector
    detected_bits=noisy_waveform(10:20:end)>0.5;
    % Calculate bit error rate (BER)
    num_errors_simple = nnz(xor(bits,detected_bits));
    ber_simple(snr_idx) = num_errors_simple/num_bits;
end

% Plot BER vs SNR curve
figure
semilogy(snr_range,ber_MF,'-o');
xlabel('SNR (dB)');
ylabel('Bit Error Rate (BER)');
grid on;
hold on;
semilogy(snr_range,ber_Corr,'-o');
hold off;
legend("Matched reciever","Correlator reciever");
```

Transmitted power:

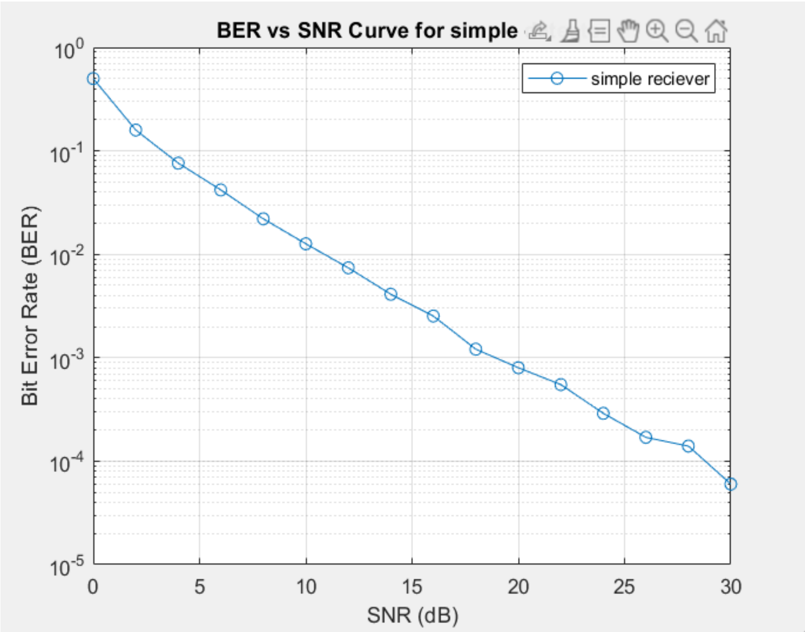
```
    % Calculating Transmitted power
    TR_pwr = sum(abs(waveform).^2)/length(waveform);
    % waveform: is the waveform generated for transmitted bit

```

Command Window

```
Transmitted Power = 0.499 Watts
```

I- <u>Simple Detector:</u>

Simple detector receiver works by detecting the presence of a signal in a communication channel and converting it into a usable output. It typically consists of an antenna to receive the signal, a tuning circuit to select a specific frequency, and a detector circuit to extract the signal from the carrier wave. It is often used in low-cost or low-power applications, such as in walkie-talkies, AM radios, or wireless doorbells. While simple detector receivers have limited bandwidth and sensitivity, they can be highly reliable and cost-effective for certain applications.



II- Matched Filter:

MF is a digital signal processing technique that maximizes the signal-to-noise ratio of a received signal by correlating it with a known reference signal. It is commonly used in communication systems to detect signals that are buried in noise or interference. The matched filter compares the received signal with a reference signal that has been designed to match the signal characteristics, such as pulse shape or frequency spectrum. By correlating the two signals, the matched filter amplifies the desired signal and suppresses the noise and interference. The output of the matched filter is then compared to a threshold to determine if a signal is present or not. Matched filters are highly effective in detecting weak signals in noisy environments and can be used in a variety of applications, such as radar, sonar, and wireless communication systems.

Adv of matched filter over simple detector:

1- Better sensitivity: MF can extract signals from a much lower signal-to-noise ratio (SNR) compared to a simple detector. This is because matched filter is designed to maximize the SNR by correlating the received signal with a known reference signal.

2- Improved timing accuracy: The use of a matched filter can improve the accuracy of the timing of the signal, as the correlation function output by the filter provides a precise estimate of the signal arrival time. This is particularly important in communication systems that require accurate timing, such as in wireless communication or radar systems.

3- Better noise rejection: A matched filter can effectively suppress noise and interference by using a reference signal that is matched to the desired signal. This means that the matched filter can reduce the impact of noise and interference on the detection of the signal, leading to fewer false alarms and missed detections.

4- Improved spectral efficiency: Matched filter receivers can be used to improve the spectral efficiency of a communication system by reducing the amount of bandwidth required to transmit a given amount of data. This can be achieved by using more complex modulation schemes, which are better suited to matched filter detection.

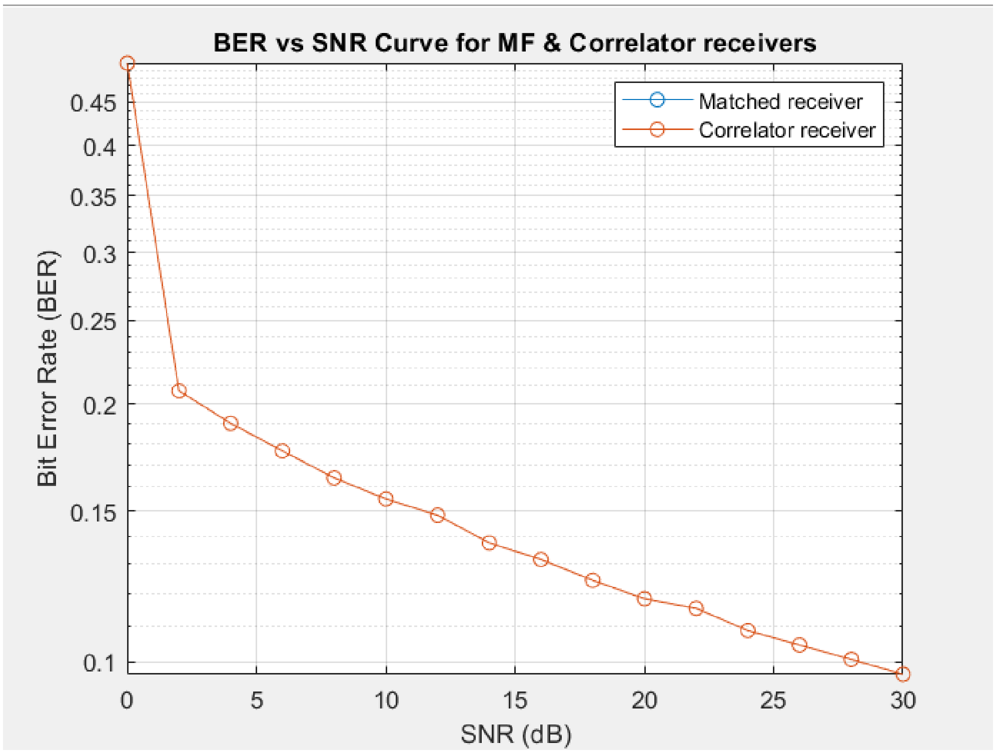
III- Correlator:

Correlator receiver is a type of receiver that uses a correlator to detect signals in a communication channel. It is a digital signal processing technique that compares the received signal with a stored reference signal to determine the degree of similarity between the two signals. The output of the correlator is a correlation function that indicates the presence and timing of the signal. Correlator receivers are commonly used in satellite communication systems and global positioning systems (GPS) to detect and track weak signals in noisy environments. Correlator receivers are also used in spread-spectrum communication systems, where the received signal is spread over a wide bandwidth using a pseudorandom code. The correlator receiver can despread the signal by correlating it with the same pseudorandom code, allowing the receiver to extract the original signal. Correlator receivers offer high sensitivity and can detect weak signals even in the presence of noise and interference.

Advantages of correlator receivers:

1. Higher processing gain: Correlator receivers offer higher processing gain than matched filter or simple detector receivers. This is because they can use a longer reference sequence for correlation, which results in a higher gain and better performance in low SNR environments.
2. Improved multipath mitigation: Correlator receivers can effectively mitigate the effects of multipath propagation, which can cause signal distortion and interference. This is achieved by using a reference signal that is designed to match the propagation characteristics of the channel, which helps to compensate for the effects of multipath.
3. Higher accuracy: Correlator receivers offer higher accuracy in signal detection and timing synchronization compared to other types of receivers. This is because they provide a precise estimate of the time of arrival of the signal, which allows for more accurate demodulation and decoding of the signal.
4. Better frequency selectivity: Correlator receivers can be designed to provide better frequency selectivity than other types of receivers. This is achieved by using a reference signal that is tailored to the frequency characteristics of the signal, which allows for better rejection of out-of-band interference.
5. Lower interference: Correlator receivers can provide lower interference compared to matched filter or simple detector receivers. This is because they can be designed to reject interference from other signals that are not correlated with the reference signal. This allows for more reliable and accurate signal detection in high traffic communication channels.

<u>BER vs SNR for MF and Correlator receivers:</u>



Part2: Line codes:

MATLAB Code:

Generating random bits, Defining parameters and pulse shape

```
% Parameters
nBits = 5;   % number of bits
Ts = 1e-3;      % symbol period
T = nBits*Ts;    % total time
t = 0:Ts:T-Ts;   % time vector

% Generate random bits
bits = randi([0 1],1, nBits);
figure;
stem(bits);
ylim([-2 2]);
figure;
% Define pulse shape
pulse_width = 10;  % Width of square pulse in samples
pulse_shape = ones(1, pulse_width);  % Square pulse

```

Define non-return zero, return zero, Alternative mark inversion, Manchester and Multi-level transmission-3 line Codes.

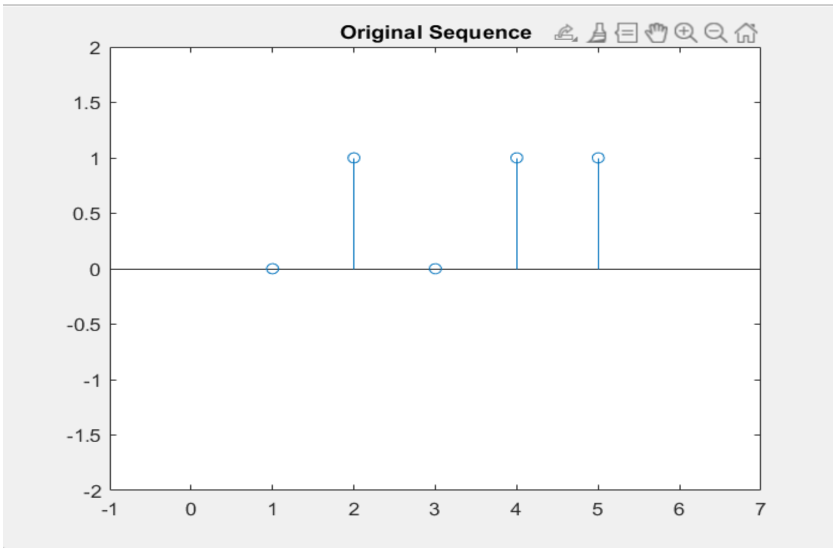
```
% Modulate using different line codes
%%%%%% non-return to zero
nrtz = bits.*2-1;

%%%%%%% NRTZ
nrzi = zeros(1, length(bits)*2);

%%%%%%% RTZ
rtz = zeros(1, length(bits)*2);

%%%%%%% AMI
pulse = -1;
ami = zeros(1, 2*length(bits)); % Initialize the AMI output vector with zeros
j = 1; % Index variable for the AMI output vector

%%%%%%% MAN
MAN = zeros(1, length(bits)*2);
% Set initial signal level to positive
signal_level = -1;

%%%%% MLT3
% define the three signal levels
levels = [-1 0 1 0];

% initialize the MLT-3 encoded signal
mlt = zeros(1, 2*length(bits));

% initialize the current signal level of MLT3
current_level = 2;

for i = 1:length(bits)

    % Loop through binary data and encode using NRZI
    if bits(i) == 0
        % No transition, keep same signal level
        nrzi(j:j+1) = signal_level;
    else
        % Transition, invert signal level
        signal_level = -signal_level;
        nrzi(j:j+1) = signal_level;
    end

    % Modulate using RTZ line code
    if bits(i) == 1
        rtz(2*i-1) = 1;
    else
        rtz(2*i-1) = 0;
    end

```

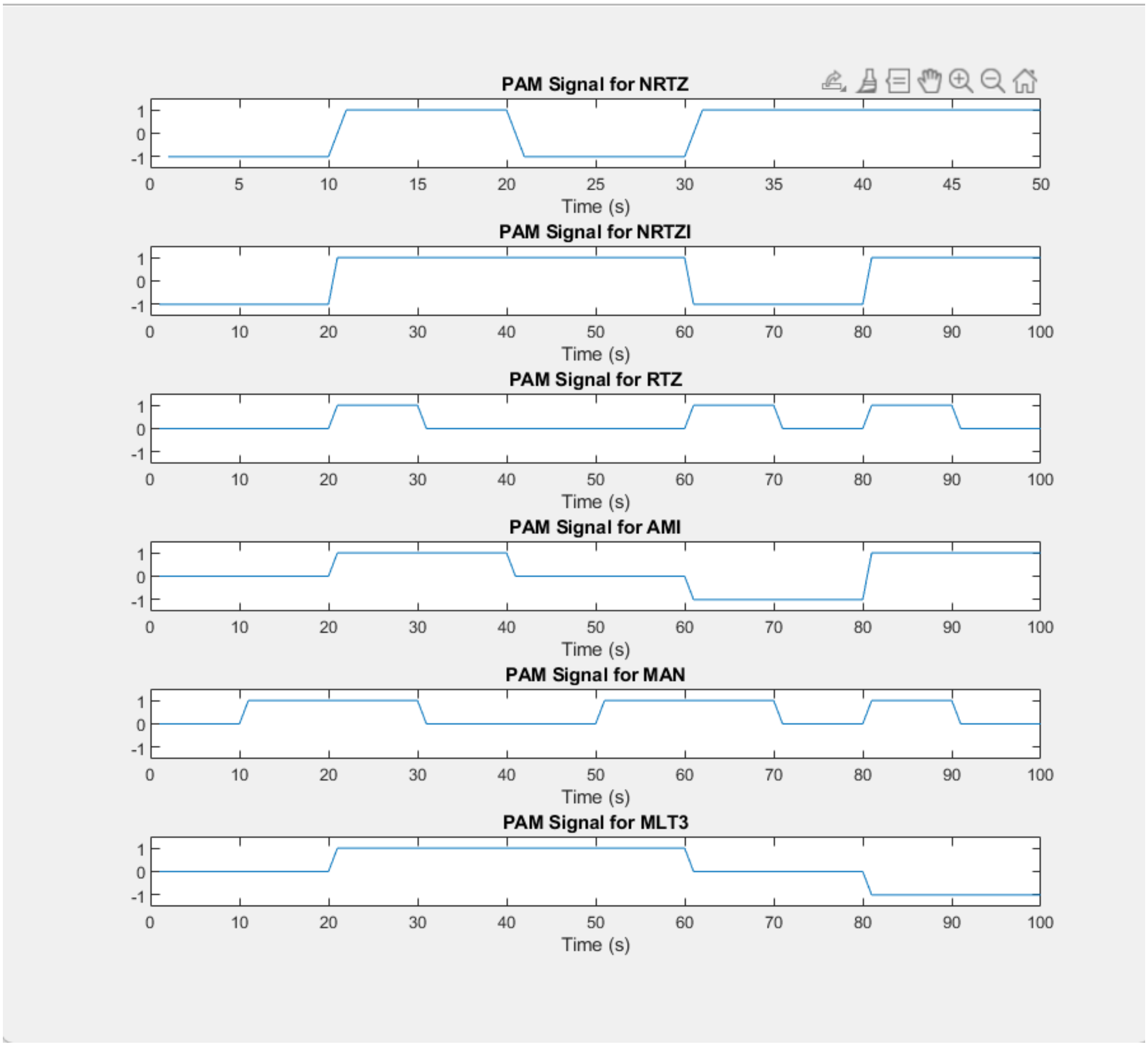
```
    % Modulate using AMI (Alternate Mark Inversion) line code
    if bits(i) == 1
        if pulse == -1
            ami(j:j+1) = 1;
        else
            ami(j:j+1) = -1;
        end
        pulse = -pulse; % Toggle the pulse
    end
    % Modulate using Manchester line code
    if bits(i) == 1
        MAN(2*i-1) = 1;
    else
        MAN(2*i) = 1;
    end
    % Modulate using MLT3 line code
    % if the current bit is 0, the signal level remains the same
    if bits(i) == 0
        mlt(j:j+1) = levels(current_level);
    % if the current bit is 1, the signal level changes to the next available level
    else
        % change the signal level
        current_level = mod(current_level, 4) + 1;
        % set the new signal level for the current bit
        mlt(j:j+1) = levels(current_level);
    end
     j = j + 2; % Increment the index variable by 2 for each bit
end
```

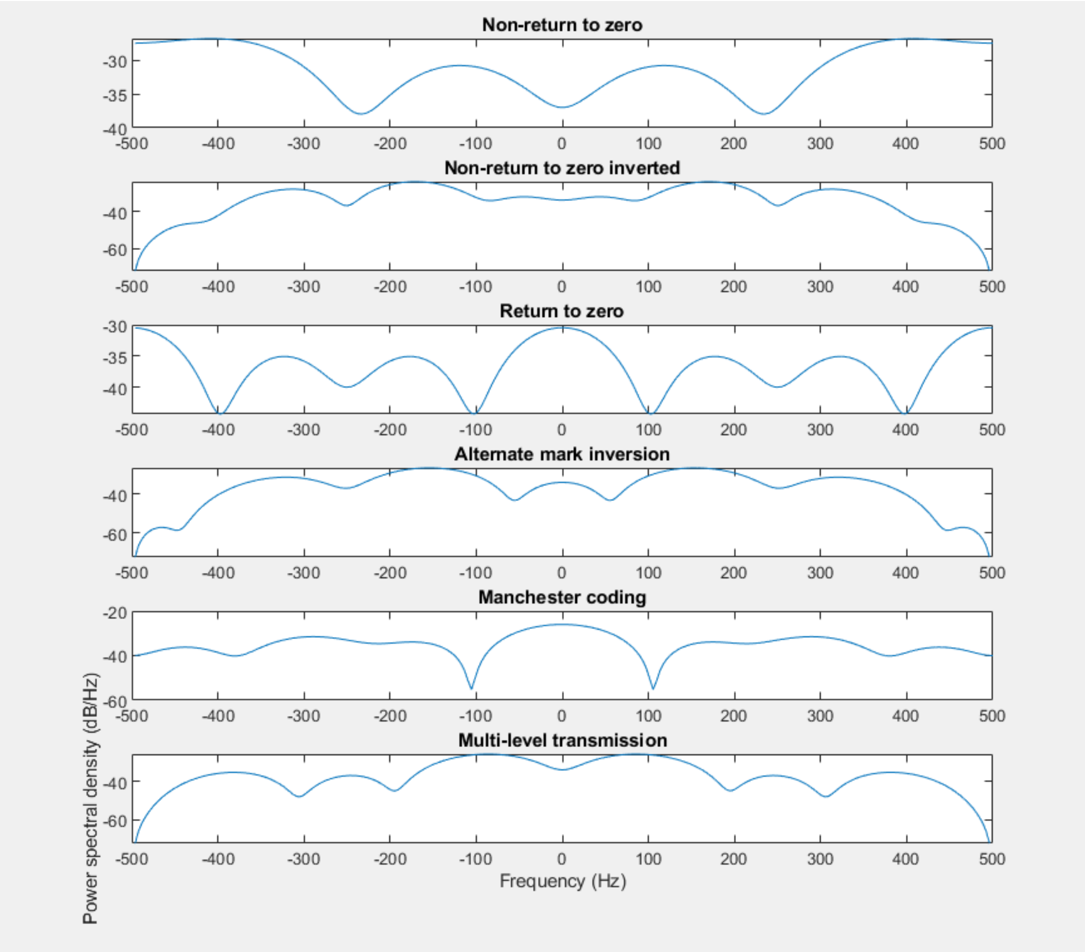
```
% Signals to PAM signals To make plotting like Square waves
signals = {nrtz, nrzi, rtz, ami, MAN, mlt};
signals_names = {'NRTZ', 'NRTZI', 'RTZ', 'AMI', 'MAN', 'MLT3'};
pam_signals = cell(size(signals));

for k = 1:numel(signals)
    signal = signals{k};
    pam_signal = zeros(1, length(signal*pulse_width));
    for i = 1:length(signal)
        start_idx = (i-1)*pulse_width+1;
        end_idx = start_idx+pulse_width-1;
        pam_signal(start_idx:end_idx) = signal(i) * pulse_shape;
    end
    pam_signals{k} = pam_signal;
end

% plotting signals
for k = 1:numel(pam_signals)
    subplot(numel(pam_signals), 1, k);
    plot(pam_signals{k});
    axis([0 length(pam_signals{k}) -1.5 1.5]); ylim([-1.5 1.5]);
    title(sprintf('PAM Signal for %s', upper(signals_names{k})));
    xlabel('Time (s)');
end
```

```
% Plotting PSD
figure;
for k = 1:numel(pam_signals)

N = length(pam_signals{k}); % number of samples
Fs = 1/Ts; % sampling frequency
f = linspace(-Fs/2, Fs/2, N);% frequency vector

subplot(numel(pam_signals), 1, k);
plot(f, 10*log10(abs(fftshift(fft(pam_signals{k})) .^ 2)));
xlim([-Fs/2, Fs/2]); grid on;
title(sprintf('PSD for %s', upper(signals_names{k})));
xlabel('Frequency (Hz)');
ylabel('Power/Frequency (dB/Hz)');
end
```

Figures:


Original Sequence


PAM Signal for NRTZ
Time (s)
PAM Signal for NRTZI
Time (s)
PAM Signal for RTZ
Time (s)
PAM Signal for AMI
Time (s)
PAM Signal for MAN
Time (s)
PAM Signal for MLT3
Time (s)



I- Non-return to zero:

NRZ (Non-Return-to-Zero) line code is a simple and widely used digital encoding scheme. It is easy to implement and has a higher data rate compared to other line codes like Manchester and RZ. However, it is vulnerable to synchronization errors, as there is no mechanism to recover the clock signal from the data stream. Also, it suffers from baseline wander, which is the gradual shift of the signal level over time due to the accumulation of DC components. Signal stays at a constant voltage level for the entire duration of each bit, representing either a 1 or a 0. It does not use a return to zero level, which is where the signal returns to zero between each bit.

Advantages:

1- Simple to implement: NRZ requires only a simple transmitter and receiver circuitry, making it easy to implement in digital communication systems.
2- Efficient use of bandwidth: NRZ has a high spectral efficiency because it does not use a return-to-zero level, which reduces the bandwidth required for transmission.
3- Low bit error rate: NRZ has a low bit error rate when the signal is transmitted over short distances or in a noise-free environment.
4- Supports high data rates: NRZ can support high data rates because it does not require a clock recovery circuit.

Disadvantages of Non-Return-to-Zero (NRZ) line code are:

1- DC offset: NRZ produces a DC offset, which can lead to signal distortion and affect the performance of the receiver.
2- Lack of synchronization: NRZ does not provide a synchronization mechanism, which makes it difficult to recover the clock signal at the receiver.
3- Vulnerable to noise: NRZ is vulnerable to noise, which can cause bit errors and reduce the overall reliability of the transmission.
4- Long sequences of the same bit: NRZ can produce long sequences of the same bit, which can lead to problems with clock recovery and cause jitter in the received signal.

II- Non-return to zero inverted:

Signal in NRZI changes polarity only when a 1 bit is transmitted and stays at the same polarity when a 0 bit is transmitted. This encoding scheme uses the transitions between consecutive bits to represent the data, which makes it self-clocking and eliminates the need for a separate clock signal. However, NRZI is more susceptible to errors caused by noise and interference and requires a more complex receiver circuitry compared to NRZ.

Advantages:

1- Self-clocking: NRZI is a self-clocking line code, which eliminates the need for a separate clock signal.
2- Efficient use of bandwidth: NRZI has a high spectral efficiency and requires less bandwidth compared to other line codes, including NRZ.
3- Lower DC component: NRZI produces a lower DC component compared to NRZ, which reduces signal distortion and improves the performance of the receiver.
4- Better synchronization: NRZI provides better synchronization compared to NRZ, as it uses the transitions between consecutive bits to transmit data.
5- Higher immunity to baseline wander: NRZI is less susceptible to baseline wander compared to NRZ, as it uses transitions to transmit data rather than relying on a fixed baseline.

Disadvantages of NRZI line code are:

1- Error susceptibility: NRZI is more susceptible to errors caused by noise and interference compared to other line codes, including Manchester and Differential Manchester.
2- Requires a more complex receiver: NRZI requires a more complex receiver circuitry compared to NRZ, which can increase the cost of the system.
3- Long runs of the same bit: NRZI can produce long runs of the same bit, which can cause synchronization problems at the receiver.
4- Bit stuffing required: NRZI requires bit stuffing to ensure that there are enough transitions in the signal to maintain synchronization, which can increase the overhead of the system.

III- <u>Return to zero:</u>

Signal returns to a zero level between each bit, making it easier for the receiver to synchronize with the transmitted signal. RZ provides better error detection and correction compared to non-return-to-zero (NRZ) line codes, but it requires more bandwidth and produces more power. RZ is commonly used in high-speed data transmission applications, such as fiber-optic communication systems and digital audio recording.

Advantages:

1- Better error detection and correction: RZ provides better error detection and correction compared to non-return-to-zero (NRZ) line codes, as it provides additional transitions that help the receiver synchronize with the transmitted signal.
2- Simple receiver circuitry: RZ requires a simple receiver circuitry, which makes it easier and cheaper to implement in digital communication systems.
3- Well-established standard: RZ is a well-established standard for digital communication systems and is widely used in various applications.
4- Improved signal quality: RZ provides improved signal quality compared to other line codes, as it produces less noise and distortion.
5- Compatible with other line codes: RZ is compatible with other line codes, such as Manchester and Differential Manchester.

Disadvantages:

1- Higher bandwidth requirement: RZ requires more bandwidth compared to other line codes, as it produces more power and requires a higher signal rate.
2- Reduced spectral efficiency: RZ has reduced spectral efficiency compared to other line codes, as it requires more bandwidth to transmit the same amount of data.
3- Susceptible to baseline wander: RZ is susceptible to baseline wander, which can cause synchronization problems at the receiver.
4- Limited distance: RZ has a limited transmission distance compared to other line codes, as the signal can become distorted and weakened over longer distances.

IV- <u>Alternative mark inversion:</u>

In AMI the signal alternates between positive, negative, and zero voltage levels to represent the data. The voltage level of each bit depends on its value and the value of the previous bit. The first bit is transmitted with a positive or negative voltage level, and the polarity of the voltage level alternates for every 1 bit. The 0 bits are transmitted with a zero-voltage level, which provides a balanced DC component. This alternating voltage level pattern helps to reduce the number of transitions and the risk of errors caused by noise and interference. However, AMI requires a more complex receiver circuitry compared to other line codes, as it needs to detect the absence of transitions. AMI is commonly used in high-speed digital communication systems, such as ADSL, ISDN, and T1 lines.

Advantages:

1- Balanced DC component: AMI provides a balanced DC component, which reduces signal distortion and improves the performance of the receiver.
2- Lower bandwidth requirement: AMI has a lower bandwidth requirement compared to other line codes, as it produces fewer transitions and requires less spectrum.

3- Improved noise immunity: AMI is less susceptible to noise and interference compared to other line codes, as it uses a balanced DC component and a limited number of transitions.
4- Efficient use of transmission media: AMI can efficiently use the transmission media, which makes it suitable for high-speed communication systems.
5- Compatible with other line codes: AMI is compatible with other line codes, such as Bipolar with eight-zero substitution (B8ZS) and High-Density Bipolar of Order 3 (HDB3).

Disadvantages of Alternative Mark Inversion (AMI):

1- Limited data rate: AMI has a limited data rate compared to other line codes, such as Non-Return-to-Zero (NRZ) and Manchester, which limits its use in high-speed communication systems.
2- Requires a more complex receiver: AMI requires a more complex receiver circuitry compared to other line codes, which can increase the cost of the system.
3- Susceptible to synchronization errors: AMI can be susceptible to synchronization errors caused by long runs of 0 bits or changes in the polarity of the signal.
4- Limited distance: AMI has a limited transmission distance compared to other line codes, as the signal can become distorted and weakened over longer distances.

V- Manchester Coding:

It uses transitions to represent data. In Manchester coding, each bit is transmitted as a transition from high to low voltage or from low to high voltage in the middle of the bit period. The first half of the bit period represents the bit value, while the second half represents its complement. This ensures that each bit has a transition in the middle of the bit period, which helps to synchronize the receiver with the transmitted signal. Manchester coding provides a self-clocking mechanism and has a higher bandwidth requirement compared to other line codes. Manchester coding is commonly used in Ethernet, Token Ring, and wireless communication systems.

Advantages:

1- Self-clocking mechanism: Manchester coding provides a self-clocking mechanism that ensures that the receiver is synchronized with the transmitted signal.
2- Better noise immunity: Manchester coding provides better noise immunity compared to other line codes, as it uses transitions to encode data, which reduces the risk of errors caused by noise and interference.
3- Simple decoding: Manchester coding provides simple decoding, as the receiver can extract the clock signal from the transitions in the transmitted signal.
4- Balanced DC component: Manchester coding provides a balanced DC component, which reduces signal distortion and improves the performance of the receiver.

Disadvantages:

1- Higher bandwidth requirement: Manchester coding has a higher bandwidth requirement compared to other line codes, as it uses transitions to encode data.
2- Reduced spectral efficiency: Manchester coding has reduced spectral efficiency compared to other line codes, as it requires more bandwidth to transmit the same amount of data.
3- Increased complexity: Manchester coding is more complex compared to other line codes, as it requires additional circuitry to encode and decode the signal.
4- Limited data rate: Manchester coding has a limited data rate compared to other line codes, such as Non-Return-to-Zero (NRZ), which limits its use in high-speed communication systems.

VI- Multi-level transmission 3:

MLT-3 uses three voltage levels to represent data. In MLT-3, the voltage level of each bit depends on the value of the previous two bits, and the signal alternates between positive, negative, and zero voltage levels. This line code provides a balanced DC component and reduces the number of transitions compared to other line codes, which improves the performance of the receiver and reduces the risk of errors caused by noise and interference. MLT-3 is commonly used in Ethernet and other high-speed communication systems. Uses three voltage levels to represent data. In MLT-3, the voltage level of each bit depends on the value of the previous two bits, and the signal alternates between positive, negative, and zero voltage levels.

Advantages of MLT-3:

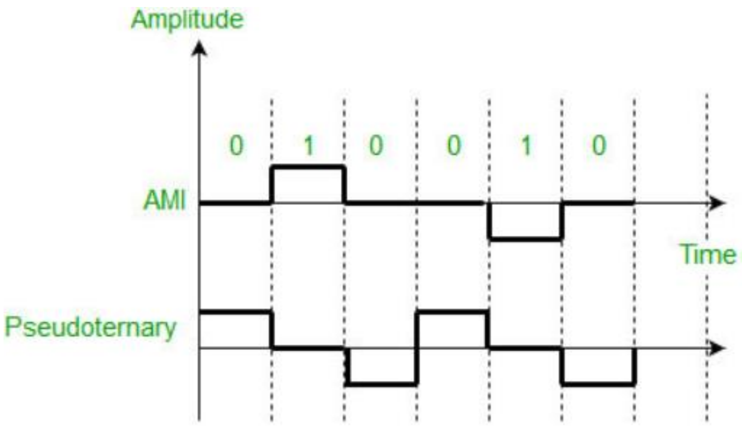
1- Lower electromagnetic interference: MLT-3 generates lower levels of electromagnetic interference (EMI) compared to other line codes, which reduces the risk of interference with other nearby systems.
2- Error detection and correction: MLT-3 can use forward error correction (FEC) to detect and correct errors in the transmitted signal, which improves the reliability of the communication system.
3- Efficient use of transmission media: MLT-3 can efficiently use the transmission media, which makes it suitable for high-speed communication systems.
4- Reduced power consumption: MLT-3 requires less power to transmit data compared to other line codes, which can help to reduce the overall power consumption of the communication system.
5- Robustness: MLT-3 is more robust to signal attenuation and distortion caused by cable length and other transmission impairments, which makes it a reliable option for longer distance transmission.

Disadvantages:

1- Limited data rate: MLT-3 has a limited data rate compared to other line codes, such as Non-Return-to-Zero (NRZ) and Manchester, which limits its use in high-speed communication systems.
2- Higher bandwidth requirement: MLT-3 has a higher bandwidth requirement compared to other line codes, as it uses multiple voltage levels to represent data.
3- More complex encoding and decoding: MLT-3 requires more complex encoding and decoding circuitry compared to other line codes, which can increase the cost of the system.
4- Limited distance: MLT-3 has a limited transmission distance compared to other line codes, as the signal can become distorted and weakened over longer distances.

VII- Pseudoternary Coding:

Pseudoternary line code is a type of bipolar encoding that uses two voltage levels to represent binary data. In this line code, a logical "1" bit is represented by an alternating positive and negative voltage, while a logical "0" bit is represented by a mid-level voltage. Unlike other bipolar encoding schemes, such as AMI, pseudoternary does not use a zero-voltage level to represent a logical "0" bit, which helps to reduce the DC component of the transmitted signal. Pseudoternary line code is commonly used in T-carrier systems, such as T1 and T3, to provide a balanced signal that is less susceptible to distortion and noise, and to increase the data capacity of the transmission medium.



Advantages:

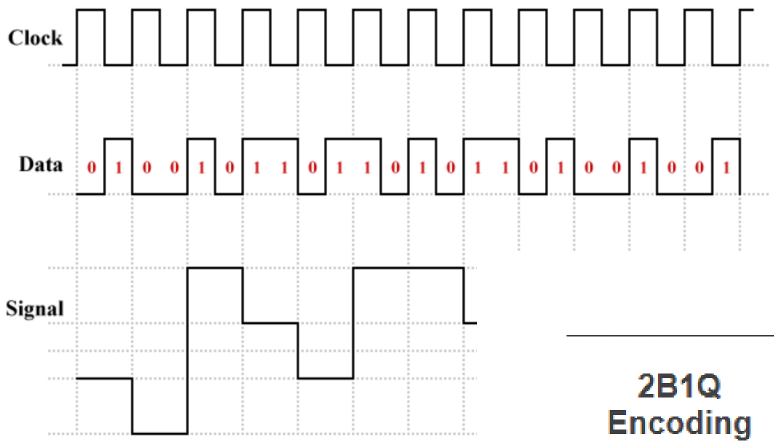
1- Efficient use of spectrum: Pseudoternary line code is efficient in the use of spectrum, which makes it useful for high-speed communication over limited bandwidth.
2- Lower power consumption: The use of only two voltage levels in pseudoternary line code reduces the power consumption required to transmit the data, making it a more energy-efficient option.
3- Compatibility: Pseudoternary line code is compatible with most bipolar transmission systems, making it a useful option for integrating with existing communication infrastructure.
4- Increased data capacity: Pseudoternary line code allows for a higher data rate compared to unipolar encoding schemes.
5- Easy to implement: Pseudoternary line code is easy to implement since it only uses two voltage levels.

Disadvantages:

1- Lack of error detection and correction: This line code does not have built-in error detection and correction mechanisms.
2- Inefficient bandwidth utilization: Pseudoternary line code is less efficient in bandwidth utilization compared to other line codes such as MLT-3.

VIII- <u>2B/1Q:</u>

The 2B1Q (two binary, one quaternary) line coding is a multi-level line code. The idea behind multi-level schemes generally is to encode more than one data bit per signal symbol to maximise bandwidth efficiency. Putting this more formally, the aim is to encode m data elements per signal element using n signal elements. This is often reflected in the name given to a particular line coding scheme. In 2B1Q, for example, each 2-bits are represented by one quaternary symbol, there are four quaternary symbols in 2B1Q (the word quaternary means "consisting of four units or members"). Each quaternary symbol is represented by one of four different signal levels. 2B1Q is thus a four-level pulse amplitude modulation (PAM-4) scheme, so called because the information is encoded in the amplitude of the signal pulses. There are two positive and two negative signal levels, with equal spacing between adjacent levels. The illustration below shows what a typical 2B1Q signal looks like.



2B1Q is a four-level pulse amplitude m

2B1Q Encoding

Data	Output
00	-3
01	-1
10	+3
11	+1

Advantages:

1- Higher data density: 2B/1Q line code can transmit two bits of data per signal element, which results in higher data density and better bandwidth efficiency compared to other unipolar line codes such as NRZ.
2- Simplicity: 2B/1Q line code is a simple and easy-to-implement line code that does not require complicated encoding or decoding circuits, making it a cost-effective option for low-speed applications.
3- Lower power consumption: The use of only two voltage levels in 2B/1Q line code reduces the power consumption required to transmit the data, making it a more energy-efficient option.

4- Low crosstalk: 2B/1Q line code provides good crosstalk rejection, which means that it is less susceptible to interference from adjacent channels or signal lines. This makes it suitable for applications where multiple channels are transmitted over the same transmission medium.

Disadvantages:

1- Higher error rate: The lack of voltage levels to represent each bit in 2B/1Q line code can make it susceptible to noise and distortion, resulting in a higher error rate.
2- Limited distance: 2B/1Q line code is not suitable for long-distance transmission because of its higher error rate and susceptibility to noise and distortion, which can lead to data corruption and loss.
3- Not DC-balanced. Long runs of dibits(2-bits) with the same bit values can introduce a significant DC component into the signal and will also result in the absence of transitions in the signal, potentially causing loss of synchronisation.