Digital Communication Engineering

Assignment 2

Submitted by:

Roopesh O R
Roll. no: 31
Reg. no: 20323085

Question 1

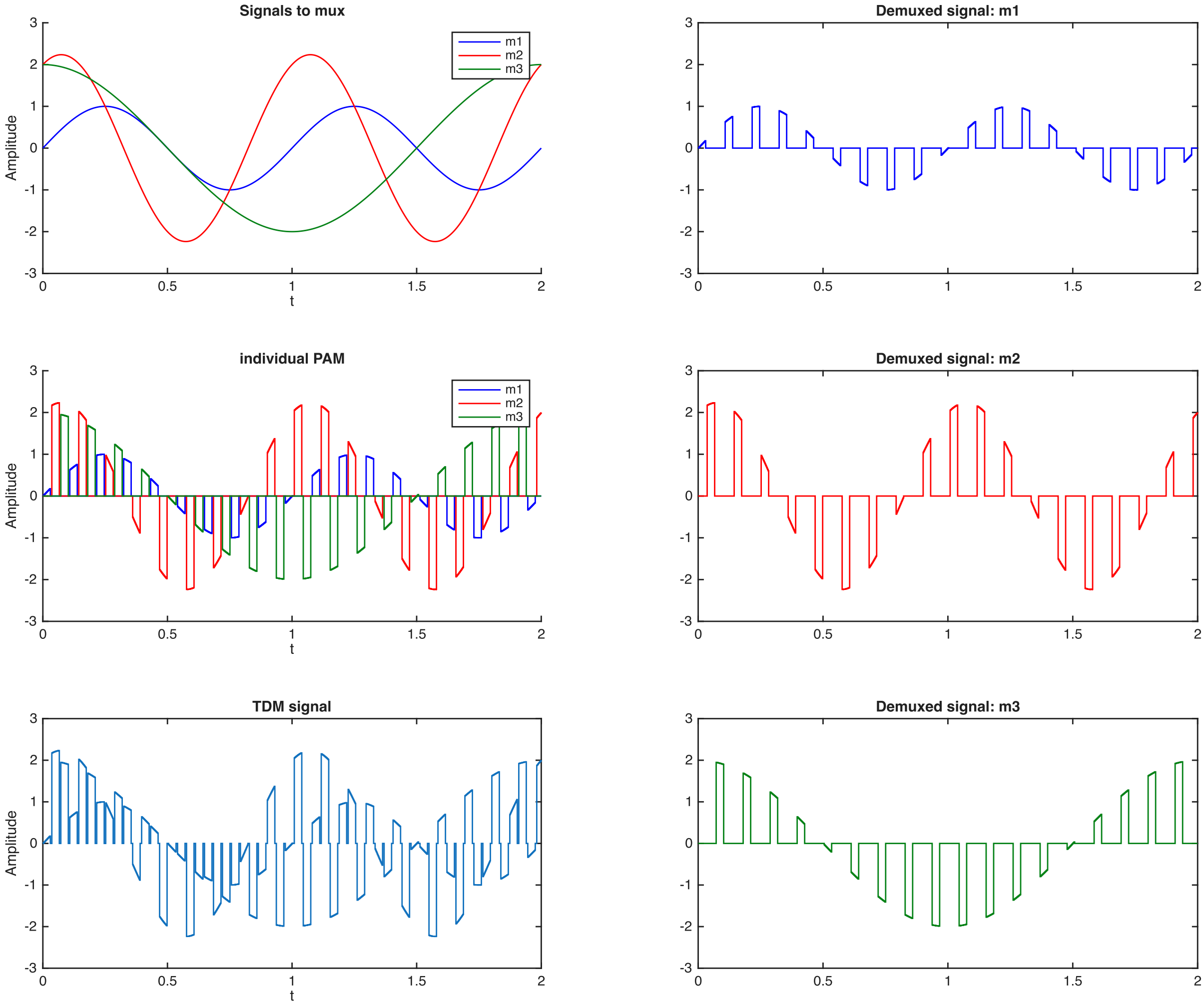
Implement a TDM system using MATLAB code / SIMULINK model.

Code

```
clearvars; close all;
t = linspace(0, 2, 1000);
signals = {
    sin(2*pi * t);
    2*cos(t * 2*pi) + sin(t * 2*pi);
    2*cos(t * pi);
};
signalNames = {"m1", "m2", "m3"};
signalCols = {"blue", "red", "#058217"};
N = length(signals); % no of signals to mux
% contains all PAM modulated signals
pamMods = zeros(N, length(signals{1}));
% no of data points each each signal can tx at a time (pulse width)
Tsignal = 15;
Tspacing = 3; % spacing between each signal
% create N shifted pulse trains each with pulse width Tsignal
MuxPulses = kron(eye(N), [ones(1, Tsignal), zeros(1, Tspacing)]);
for i=1:N
    pamMods(i, :) = pamMod_Demod(signals{i}, MuxPulses(i, :));
end
tdm = sum(pamMods); % sum all PAMs along column axis

demuxed = zeros(N, length(tdm));
for i=1:N
    demuxed(i, :) = pamMod_Demod(tdm, MuxPulses(i, :));
end

subplot(3, 2, 1); hold on
for i = 1:N
    plot(t, signals{i}, "Color", signalCols{i});
end
legend(signalNames);
title("Signals to mux");
xlabel("t"); ylabel("Amplitude");

subplot(3, 2, 3); hold on
for i=1:N
    stairs(t, pamMods(i, :), "Color", signalCols{i});
end
title("individual PAM");
xlabel("t"); ylabel("Amplitude");
legend(signalNames);
```

```
subplot(3, 2, 5); stairs(t, tdm);
title("TDM signal");
xlabel("t"); ylabel("Amplitude");
for i=1:N
    subplot(3, 2, 2*i);
    stairs(t, demuxed(i, :), "Color", signalCols{i});
    title("Demuxed signal: " + signalNames{i});
end
ylim(findobj(gcf, 'Type', 'Axes'), [-3 3]);

function y = pamMod_Demod(x, seq)
    L = ceil(length(x) / length(seq));
    sq = kron(ones(1, L), seq);
    sq = sq(1:length(x)); % make length same
    y = x .* sq;
end
```

Output waveforms



Question 2

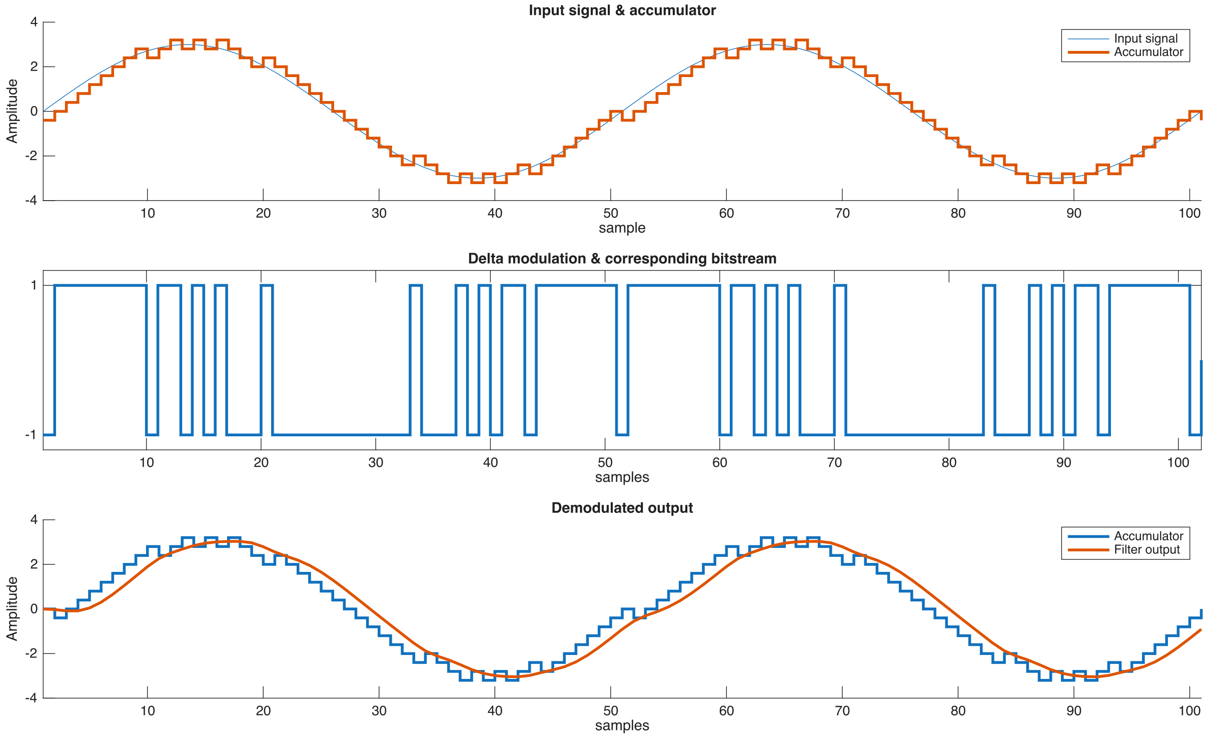
Create a MATLAB code to simulate Delta modulation and demodulation system.

Code

```
clearvars; close all;

fm = 2; % highest msg freq
fs = 100; % sampling freq
t = 0:1/fs:1;

msg = 3 * sin(2*pi*t*fm);
dy = .4; % delta size

%%%% Modulation
accumulator = 0;
len = length(t);
deltaMod = repelem(-1, len);
accumulated = zeros(1, len);
for i = 1:len
    if (msg(i) > accumulator)
        deltaMod(i) = 1;
        accumulator = accumulator + dy;
    else
        accumulator = accumulator - dy;
    end
    accumulated(i) = accumulator;
end

subplot(3, 1, 1);
hold on; plot(msg);
stairs([accumulated, 0], "LineWidth",2);
xlim([1, len]);
xlabel("sample");
ylabel("Amplitude");
legend(["Input signal", "Accumulator"]);
title("Input signal & accumulator");

% modulated
subplot(3, 1, 2);
stairs([deltaMod, 0], "LineWidth",2);
title("Delta modulation & corresponding bitstream");
xlabel("samples");

xlim([1, len+1]);
ylim([-1.2, 1.2]); yticks([-1, 1])

%%%% Demodulation
```

```
N = length(deltaMod);
accum = zeros(1, N);
for i = 2:N
accum(i) = accum(i-1) + dy * deltaMod(i-1);
end

% apply low pass
[b, a] = butter(2, fm / fs * 10);
y = filter(b, a, accum);

subplot(3, 1, 3);
hold on;
stairs(accum, "LineWidth",2);
plot(y, "LineWidth",2);
xlim([1, N]);
title("Demodulated output");
legend(["Accumulator", "Filter output"])
xlabel("samples");
ylabel("Amplitude");
```

Output waveforms



Question 3

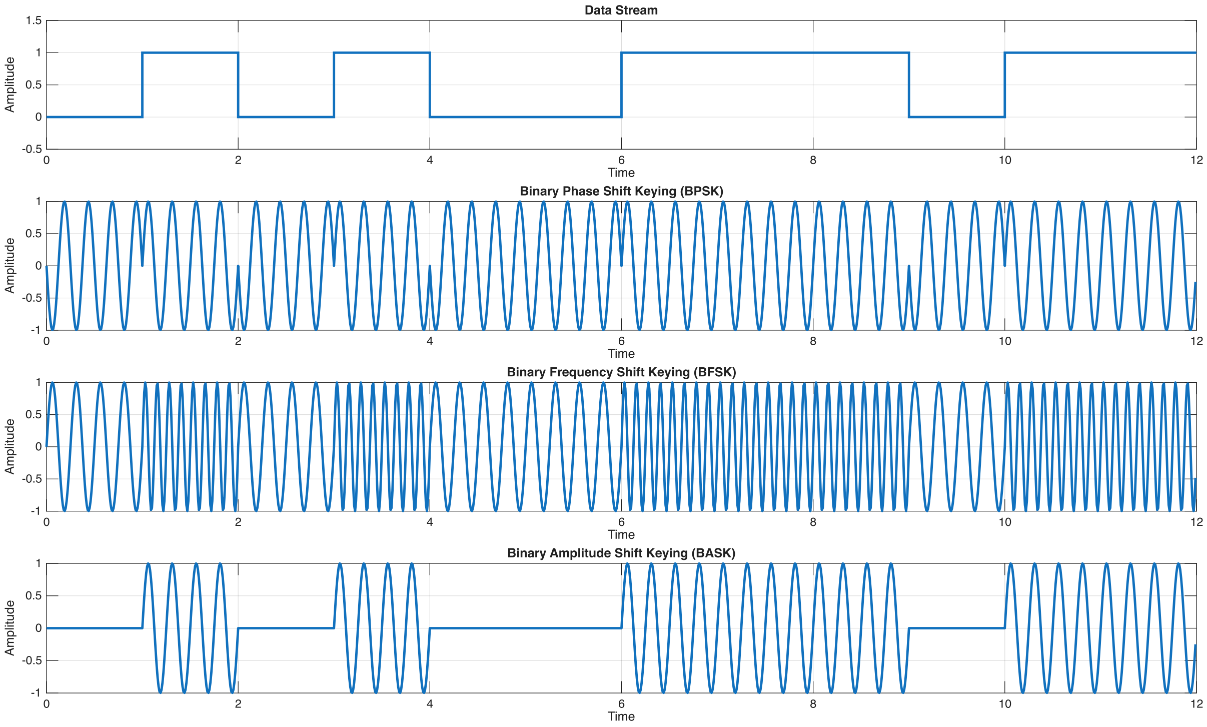
Implement BPSK, BASK and BFSK modulation using MATLAB code/ SIMULINK model.

Code

```
x = [0 1 0 1 0 0 1 1 1 0 1 1];
len = length(x);
T_b = 100; % samples per bit
t = 0 : 1/T_b : len; % time vector
%% Parameters
f_c = 4; % Carrier frequency for BPSK and BASK
f1 = 4; % BFSK frequency for bit 0
f2 = 8; % BFSK frequency for bit 1
%% Initialize signals
bpsk = [];
bfsk = [];
bask = [];
%% Generate modulated signals
for i = 1:len
% BPSK
if x(i) == 1
bpsk = [bpsk sin(2*pi*f_c*(0:1/T_b:1 - 1/T_b))];
else
bpsk = [bpsk -sin(2*pi*f_c*(0:1/T_b:1 - 1/T_b))];
end
% BFSK
if x(i) == 1
bfsk = [bfsk sin(2*pi*f2*(0:1/T_b:1 - 1/T_b))];
else
bfsk = [bfsk sin(2*pi*f1*(0:1/T_b:1 - 1/T_b))];
end
% BASK
if x(i) == 1
bask = [bask sin(2*pi*f_c*(0:1/T_b:1 - 1/T_b))];
else
bask = [bask zeros(1, T_b)];
end
end
%% Plot data stream
figure('Name','Digital Modulation Signals','NumberTitle','off');
subplot(4,1,1);
stairs(0:len, [x x(end)], 'LineWidth', 2);
ylim([-0.5 1.5]);
title('Data Stream');
xlabel('Time');
ylabel('Amplitude');
grid on;
```

```
%% Plot BPSK
subplot(4,1,2);
plot(t(1:end-1), bpsk, 'LineWidth', 2);
title('Binary Phase Shift Keying (BPSK)');
xlabel('Time');
ylabel('Amplitude');
grid on;
%% Plot BFSK
subplot(4,1,3);
plot(t(1:end-1), bfsk, 'LineWidth', 2);
title('Binary Frequency Shift Keying (BFSK)');
xlabel('Time');
ylabel('Amplitude');
grid on;
%% Plot BASK
subplot(4,1,4);
plot(t(1:end-1), bask, 'LineWidth', 2);
title('Binary Amplitude Shift Keying (BASK)');
xlabel('Time');
ylabel('Amplitude');
grid on;
```

Output waveforms


Data Stream
Amplitude
Time
Binary Phase Shift Keying (BPSK)
Amplitude
Time
Binary Frequency Shift Keying (BFSK)
Amplitude
Time
Binary Amplitude Shift Keying (BASK)
Amplitude
Time