

Lab 05
Simulation of Line Encoding Schemes in
Matlab (Part II)

OBJECTIVES OF THE LAB

In this lab, we will cover the following topics:

- Implement Manchester encoding scheme
-

5.1 IMPLEMENTING MANCHESTER ENCODING IN MATLAB

Manchester is a Biphase encoding technique. It uses transition at the middle of each bit period. The mid-bit transition serves as a clocking mechanism and also a data: **low-to-high** transition represents **binary-1**, and **high-to-low** transition represents **binary-0**.

Matlab Code for Manchester Encoding:

```
function [signal,tim] = manchester(message)

% original message
message = [0 1 0 0 1];
% message with redundant information
data = zeros(1,4*length(message));
data(1:4:end) = message;

% index representing original message in 'data' vector
i = 1:length(message);
n = 1.49:length(message)+0.49;

% index representing redundant information in 'data' vector
l = 1.50:length(message)+0.50;
j = 1.99:length(message)+0.99;

% generating 'time' vector by concatenating indices i &
% j to represent 'data' vector
tim = [];
for(k = 1:length(message))
    tim = [tim i(k) n(k) l(k) j(k)];
end

% generating digital signal
signal = [];
N = length(data);
for(t = 1:4:N)
    if(data(t)==1)
        signal(t:t+1) = -1;
        signal(t+2:t+3) = 1;
    else %if(data(t)==0)
        signal(t:t+1) = 1;
        signal(t+2:t+3) = -1;
    end
end

% displaying digital signal & message
figure(1);
```

```

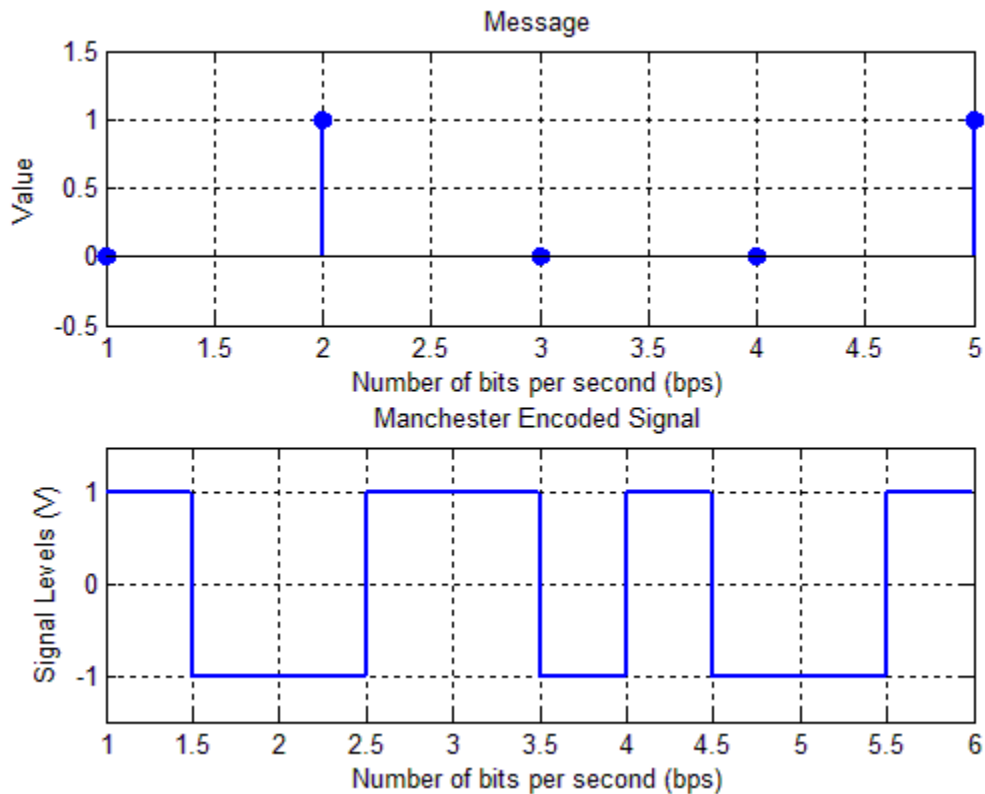
subplot(211);
stem(message, 'filled', 'linewidth', 2);
title('Message');
xlabel('Number of bits per second (bps)');
ylabel('Value');
axis([1 length(message) -0.5 1.5]);
grid on;

```

```

subplot(212) ;
plot(tim,signal,'linewidth',2);
title('Manchester Encoded Signal');
xlabel('Number of bits per second (bps)');
ylabel('Signal Levels (V)');
axis tight;
axis([1 length(message)+1 -1.5 1.5]);
grid on;

```



-----TASK 01-----

Write matlab code that converts the following message bits into Manchester signal:

Message = [0 1 0 0 1 1 0 0 0 1 1]