Lab-7

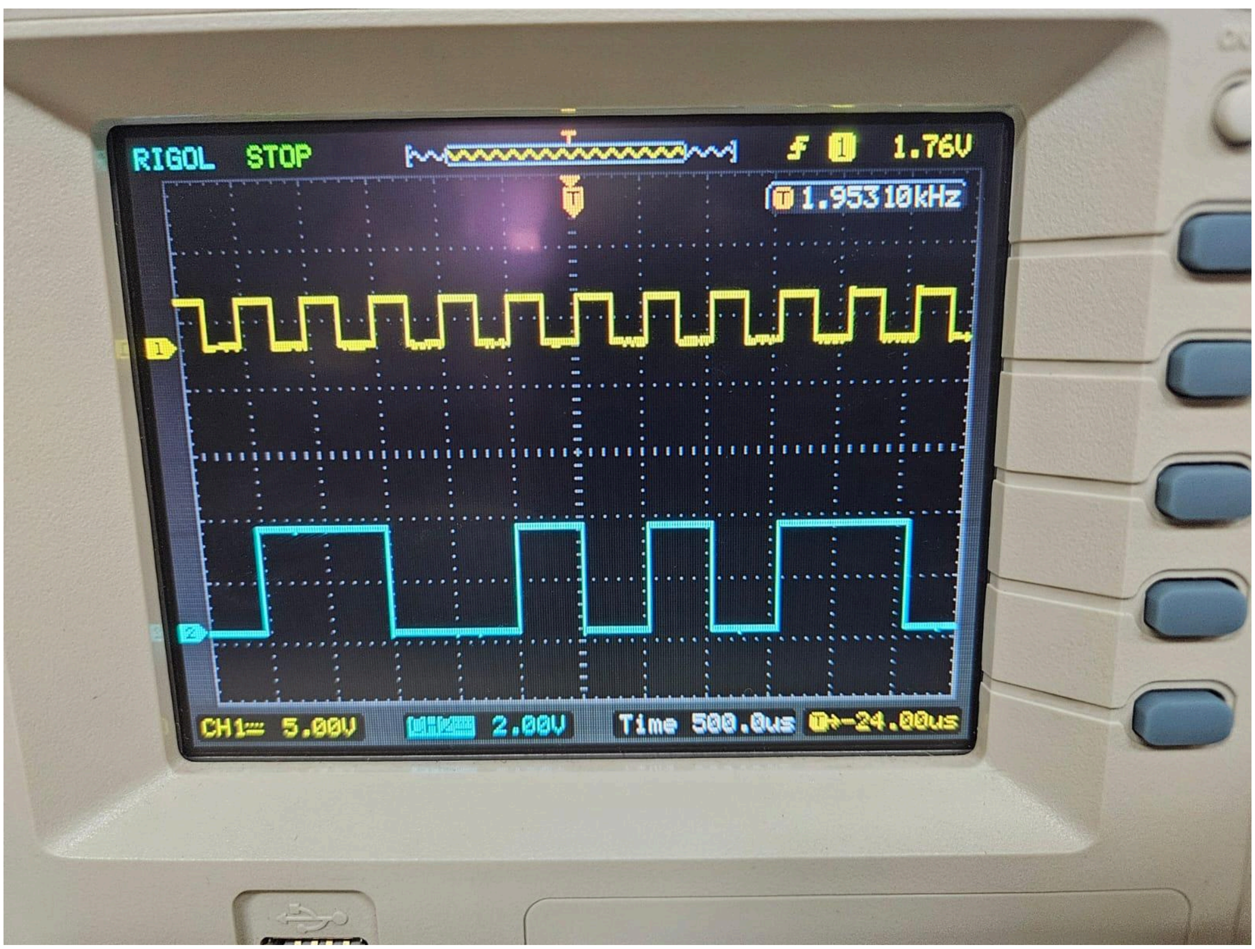
Raj-202201403, Bhoomish-202201414

Manthan-202201416, Rakshit-202201426

Experiment 1

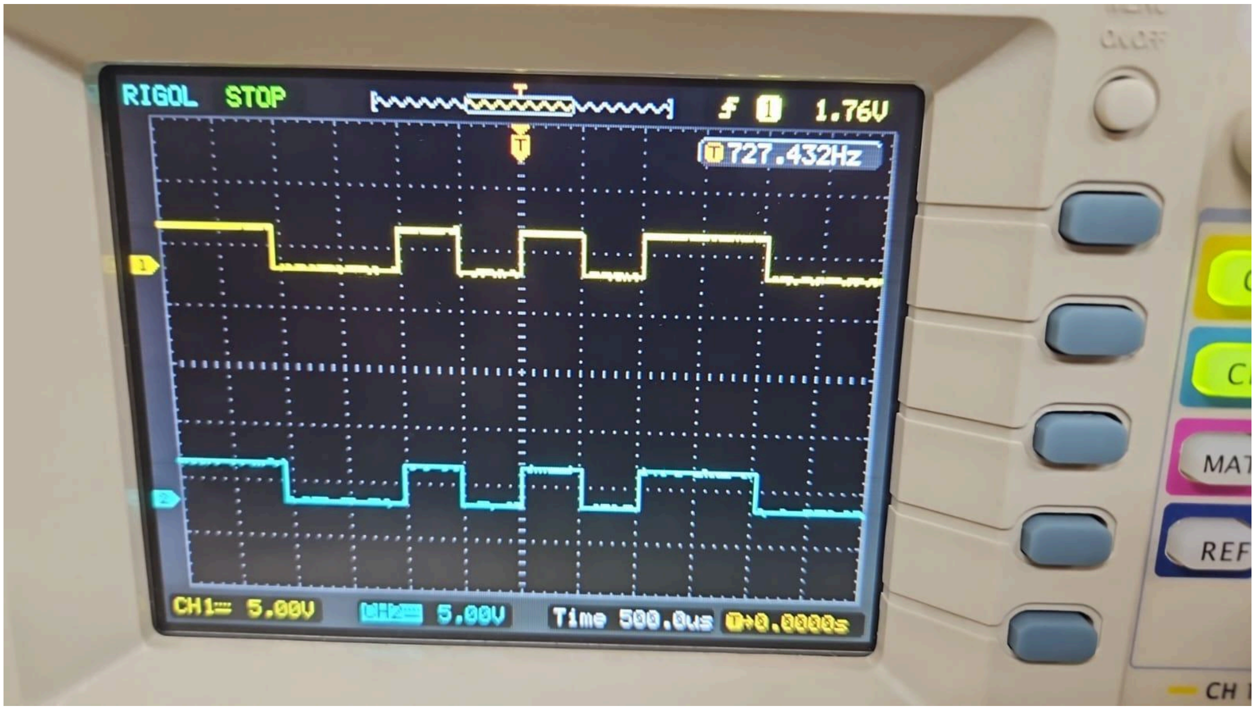
CH1: Input Data Clock

CH2: Input Data 8-Bit



CH1: Input Data 8-Bit

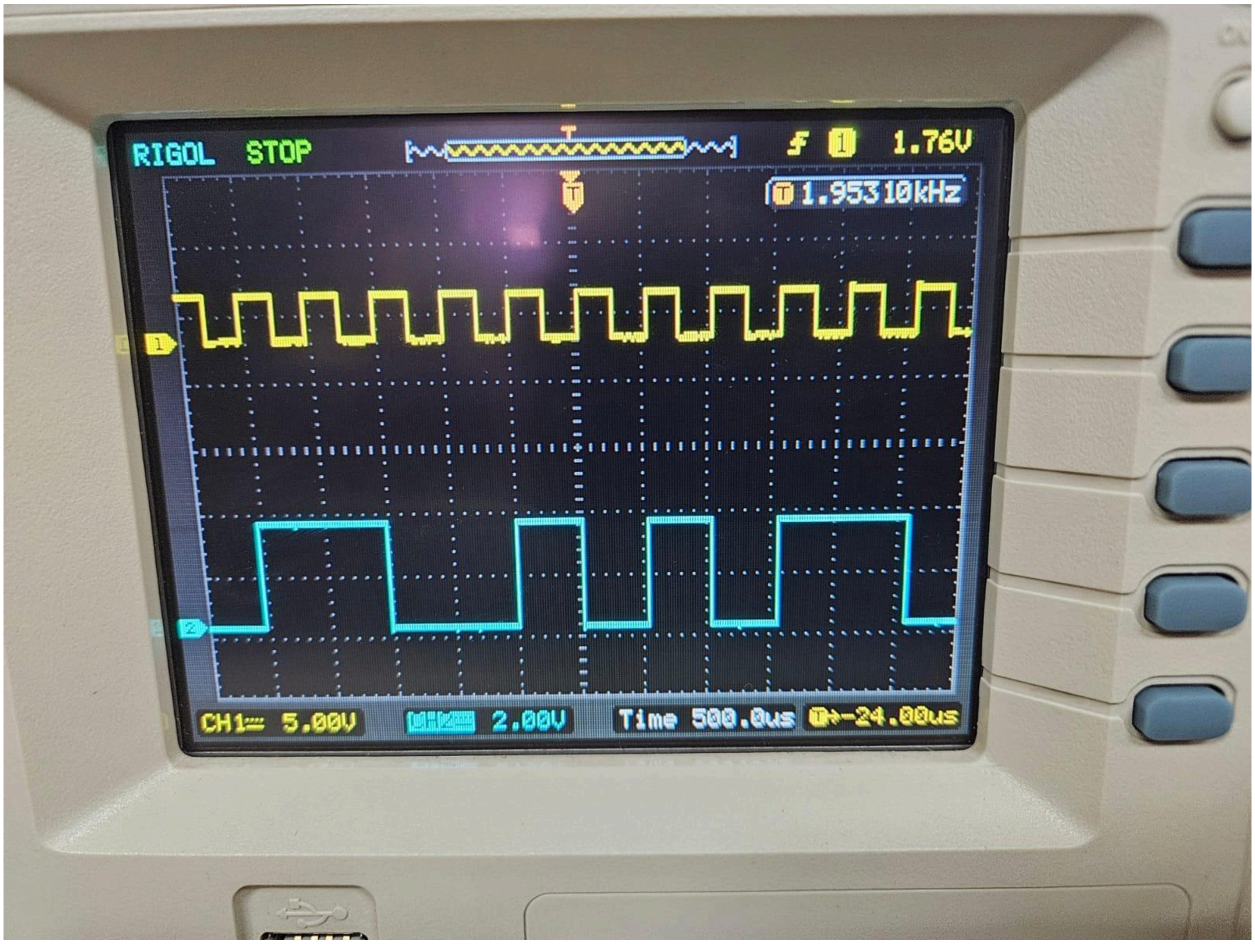
CH2: 1 - bit encoded data



Experiment 2

CH1: Input Data Clock

CH2: Input Data

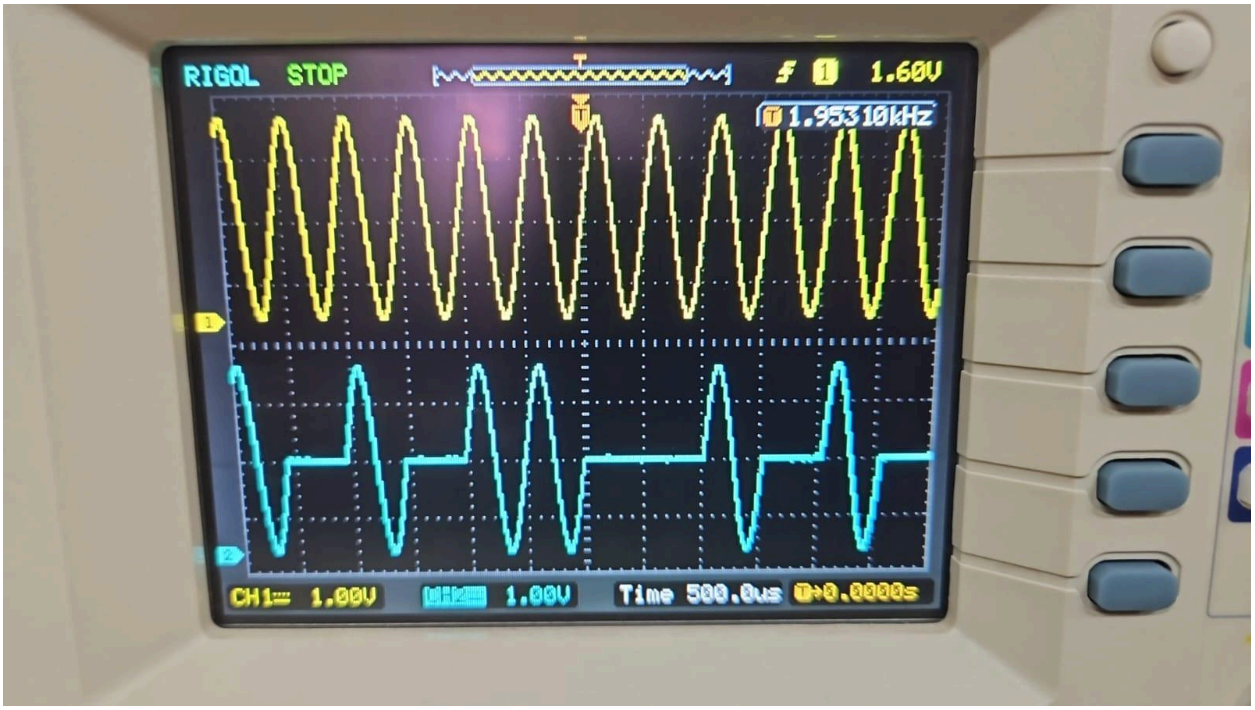


CH1: Input Data

CH2: Carrier Signal

CH1: Carrier Signal

CH2: ASK Output



RIGOL STOP
1.60V
1.95310kHz
CH1 1.00V
1.00V
Time 500.0us
0.0000s

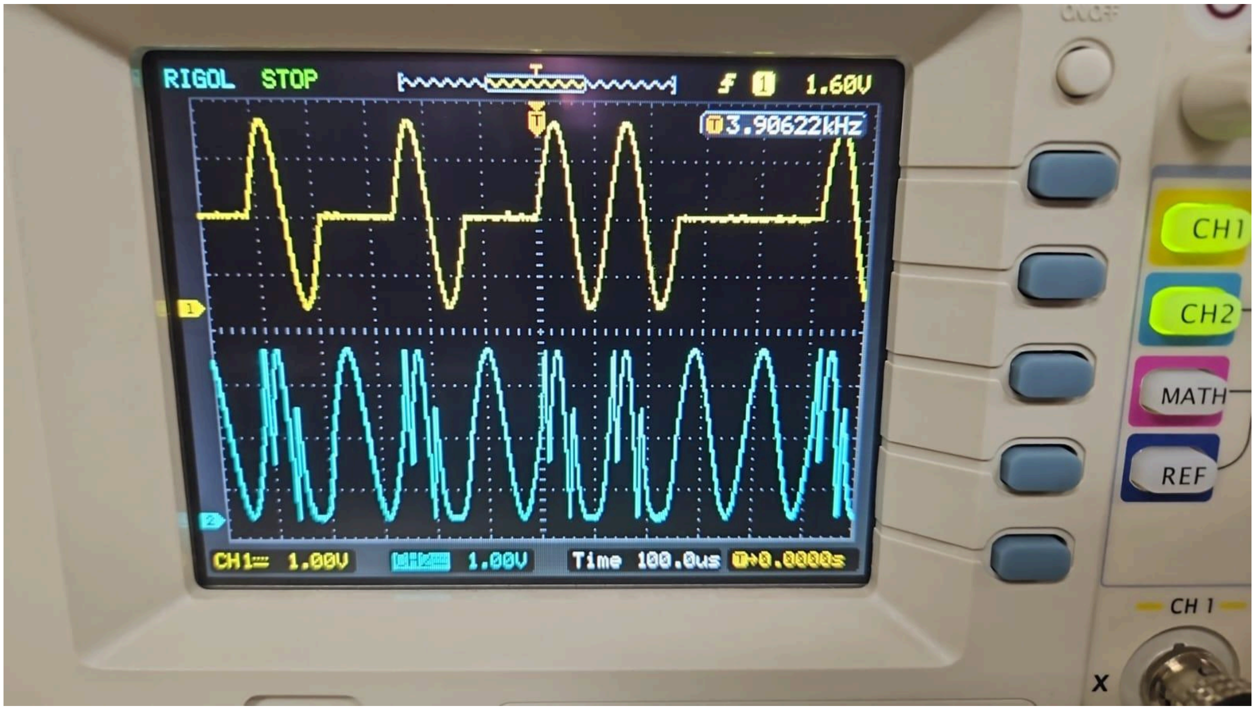
Experiment 3

CH1: ASK Output

CH2: Multiplier Output (2KHz)

CH1: ASK Output

CH2: Multiplier Output (8KHz)


RIGOL STOP
1.60V
3.90622kHz
CH1 1.00V
1.00V
Time 100.0us
0.0000s
CH1
CH2
MATH
REF
CH 1

Experiment 4

CH1: ASK Output

CH2: Integrator Output (2KHz)

CH1: ASK Output

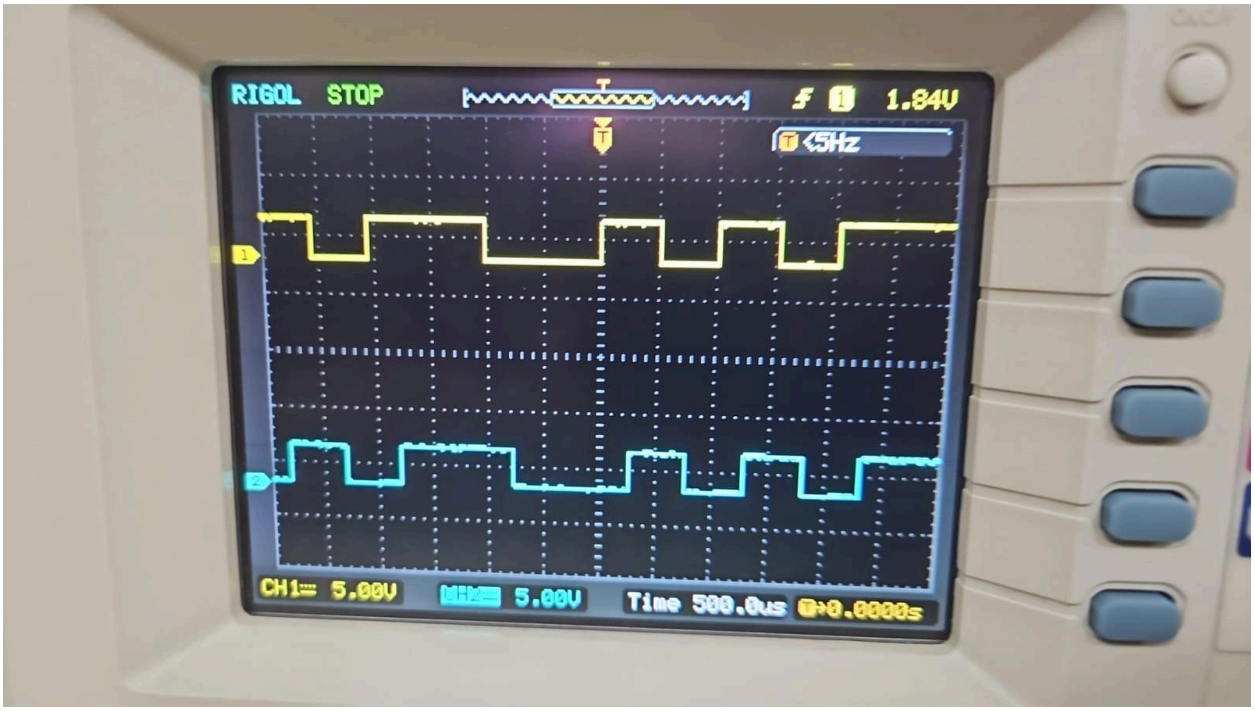
CH2: Integrator Output (16KHz)

DIGITAL OSCILLOSCOPE
100MHz 1GSa/s
RIGOL
STOP
CH1 1.00V
2.00V
Time 50.00us

Experiment 5

CH1: Input Data

CH2: Comparator Out



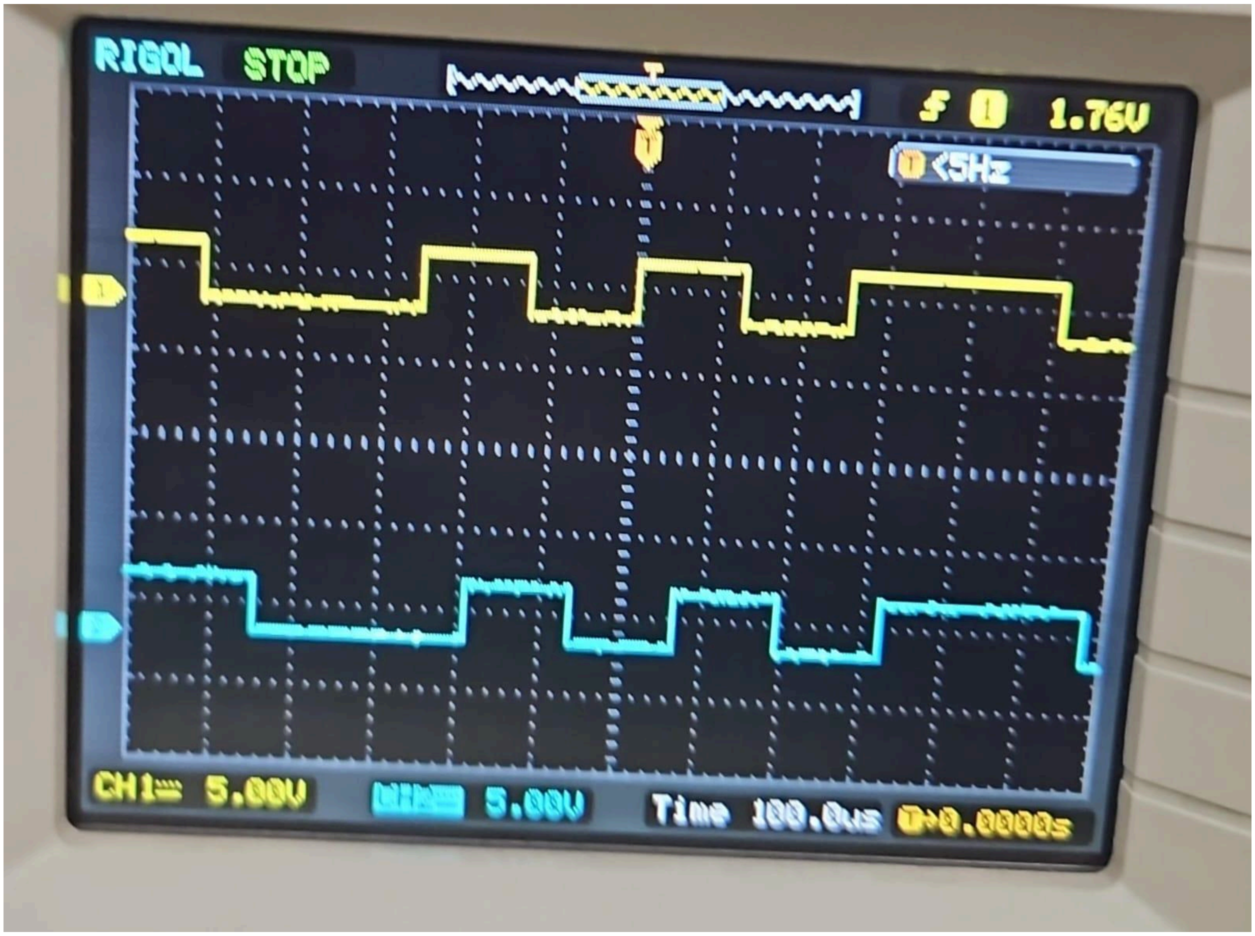
Experiment 6

CH1: Input Data

CH2: Demodulator Out

CH1: Input Data

CH2: Demodulator Out



Matlab Exercises

202201403 - Raj Shah

202201414 - Bhoomish Patel

202201416 - Manthan Parmar

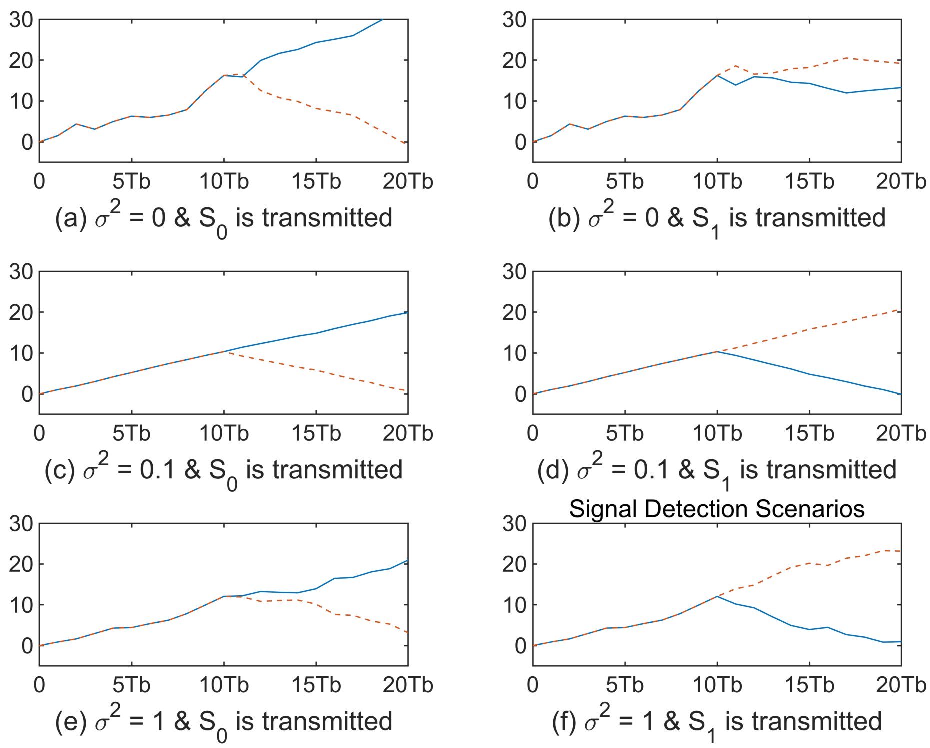
202201426 - Rakshit Pandhi

Illustrative Problem 5.2

```
% MATLAB script for Illustrative Problem 5.2
% Initialization:
K = 20; % Number of samples
A = 1;  % Signal amplitude

% Defining signal waveforms:
s_0 = A * ones(1, K);                    % Constant signal
s_1 = [A * ones(1, K/2), -A * ones(1, K/2)];  % Bipolar signal

% Initialize output signals:
r_0 = zeros(1, K);
r_1 = zeros(1, K);

% Case 1: noise ~ N(0, 0)
noise = randn(1, K);

% Sub-case s = s_0:
s = s_0;
r = s + noise;  % Received signal
for n = 1:K
    r_0(n) = sum(r(1:n) .* s_0(1:n));
    r_1(n) = sum(r(1:n) .* s_1(1:n));
end

% Plotting the results:
subplot(3, 2, 1)
plot(0:K, [0, r_0], '-', 0:K, [0, r_1], '--')
set(gca, 'XTickLabel', {'0', '5Tb', '10Tb', '15Tb', '20Tb'})
axis([0, 20, -5, 30])
xlabel('(a) \sigma^2 = 0 & S_0 is transmitted', 'fontsize', 10)

% Sub-case s = s_1:
s = s_1;
r = s + noise;
for n = 1:K
    r_0(n) = sum(r(1:n) .* s_0(1:n));
    r_1(n) = sum(r(1:n) .* s_1(1:n));
end
```

```
subplot(3, 2, 2)
plot(0:K, [0, r_0], '-', 0:K, [0, r_1], '--')
set(gca, 'XTickLabel', {'0', '5Tb', '10Tb', '15Tb', '20Tb'})
axis([0, 20, -5, 30])
xlabel('(b) \sigma^2 = 0 & S_1 is transmitted', 'fontsize', 10)

% Case 2: noise ~ N(0, 0.1)
noise = 0.1 * randn(1, K);

% Sub-case s = s_0:
s = s_0;
r = s + noise;
for n = 1:K
    r_0(n) = sum(r(1:n) .* s_0(1:n));
    r_1(n) = sum(r(1:n) .* s_1(1:n));
end

subplot(3, 2, 3)
plot(0:K, [0, r_0], '-', 0:K, [0, r_1], '--')
set(gca, 'XTickLabel', {'0', '5Tb', '10Tb', '15Tb', '20Tb'})
axis([0, 20, -5, 30])
xlabel('(c) \sigma^2 = 0.1 & S_0 is transmitted', 'fontsize', 10)

% Sub-case s = s_1:
s = s_1;
r = s + noise;
for n = 1:K
    r_0(n) = sum(r(1:n) .* s_0(1:n));
    r_1(n) = sum(r(1:n) .* s_1(1:n));
end

subplot(3, 2, 4)
plot(0:K, [0, r_0], '-', 0:K, [0, r_1], '--')
set(gca, 'XTickLabel', {'0', '5Tb', '10Tb', '15Tb', '20Tb'})
axis([0, 20, -5, 30])
xlabel('(d) \sigma^2 = 0.1 & S_1 is transmitted', 'fontsize', 10)

% Case 3: noise ~ N(0, 1)
noise = randn(1, K);

% Sub-case s = s_0:
s = s_0;
r = s + noise;
for n = 1:K
    r_0(n) = sum(r(1:n) .* s_0(1:n));
    r_1(n) = sum(r(1:n) .* s_1(1:n));
end

subplot(3, 2, 5)
plot(0:K, [0, r_0], '-', 0:K, [0, r_1], '--')
```

```
set(gca, 'XTickLabel', {'0', '5Tb', '10Tb', '15Tb', '20Tb'})
axis([0, 20, -5, 30])
xlabel('(e) \sigma^2 = 1 & S_0 is transmitted', 'fontsize', 10)

% Sub-case s = s_1:
s = s_1;
r = s + noise;
for n = 1:K
    r_0(n) = sum(r(1:n) .* s_0(1:n));
    r_1(n) = sum(r(1:n) .* s_1(1:n));
end

subplot(3, 2, 6)
plot(0:K, [0, r_0], '-', 0:K, [0, r_1], '--')
set(gca, 'XTickLabel', {'0', '5Tb', '10Tb', '15Tb', '20Tb'})
axis([0, 20, -5, 30])
xlabel('(f) \sigma^2 = 1 & S_1 is transmitted', 'fontsize', 10)

% Display the plots
subtitle('Signal Detection Scenarios');
```



Problem 5.3

```
% MATLAB script for Illustrative Problem 5.2
% Initialization:
K = 20; % Number of samples
A = 1;  % Signal amplitude

% Defining signal waveforms:
s_0 = A * ones(1, K);                        % Constant signal
s_1 = [A * ones(1, K/2), -A * ones(1, K/2)];  % Bipolar signal

% Initialize output signals:
r_0 = zeros(1, K);
r_1 = zeros(1, K);

% Noise variances:
noise_variances = [0.1, 1, 3];

for var_idx = 1:length(noise_variances)
    % Generate noise with different variances
    noise = sqrt(noise_variances(var_idx)) * randn(1, K);

    % Sub-case s = s_0:
    s = s_0;
    r = s + noise;  % Received signal
    for n = 1:K
        r_0(n) = sum(r(1:n) .* s_0(1:n));
        r_1(n) = sum(r(1:n) .* s_1(1:n));
    end

    % Plotting the results:
    subplot(3, 3, var_idx)
    plot(0:K, [0, r_0], '-', 0:K, [0, r_1], '--')
    set(gca, 'XTickLabel', {'0', '5Tb', '10Tb', '15Tb', '20Tb'})
    axis([0, 20, -5, 30])
    title(['\sigma^2 = ' num2str(noise_variances(var_idx))], 'fontsize', 10)
    if var_idx == 1
        ylabel('Output', 'fontsize', 10)
    end

    % Sub-case s = s_1:
    s = s_1;
    r = s + noise;  % Received signal
    for n = 1:K
        r_0(n) = sum(r(1:n) .* s_0(1:n));
        r_1(n) = sum(r(1:n) .* s_1(1:n));
    end

    % Plotting the results:
    subplot(3, 3, var_idx + 3)
    plot(0:K, [0, r_0], '-', 0:K, [0, r_1], '--')
    set(gca, 'XTickLabel', {'0', '5Tb', '10Tb', '15Tb', '20Tb'})
```

```
    axis([0, 20, -5, 30])
    if var_idx == 1
        ylabel('Output', 'fontsize', 10)
    end
end

% Display the plots
subtitle('Signal Detection Scenarios');
```