Cairo University
Faculty of Engineering

Department of Computer Engineering

ELC 325B – Spring 2023

Digital Communications

Assignment #2

Submitted to

Eng. Mohamed Khaled

Submitted by

Name	Sec	BN
Moaz Mohamed Hassan Bayoumi	2	25
Youssef Khaled El Sayed Al Waer	2	37

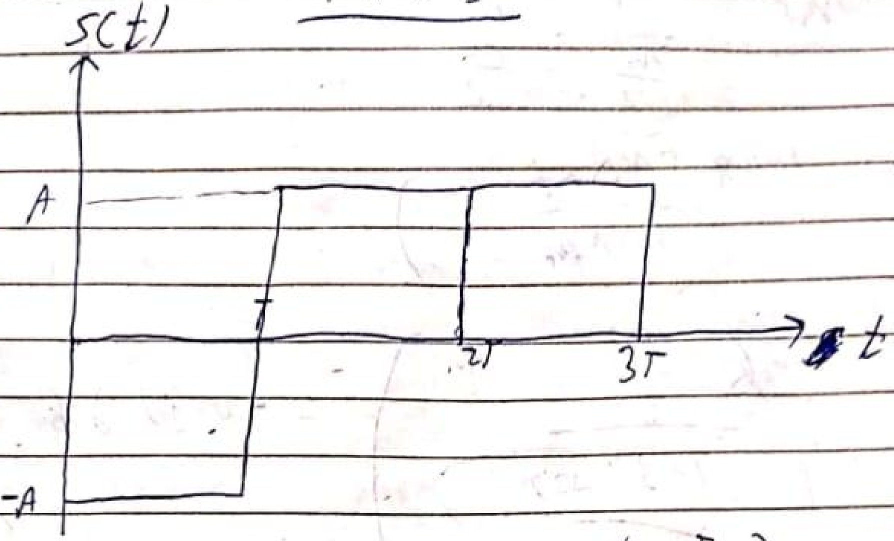
Part I:

الموضوع : .... التاريخ : ....

Assignment 2

Part 1

(a)

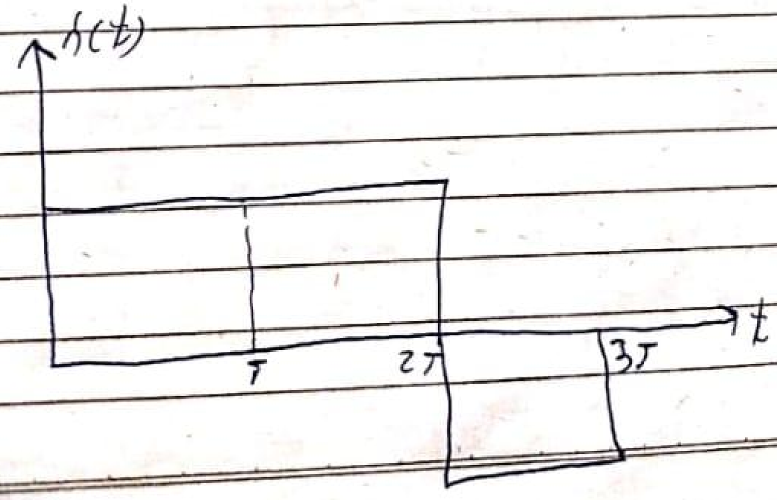


$$s(t) = -A\,\mathrm{rect}\left(\frac{t-\frac{T}{2}}{T}\right) + A\,\mathrm{rect}\left(\frac{t-\frac{3T}{2}}{T}\right) + A\,\mathrm{rect}\left(\frac{t-\frac{5T}{2}}{T}\right)$$

$$h(t) = s(T-t)$$

$$= -A\,\mathrm{rect}\left(\frac{(T-t)-\frac{T}{2}}{T}\right) + A\,\mathrm{rect}\left(\frac{(T-t)-\frac{3T}{2}}{T}\right) + A\,\mathrm{rect}\left(\frac{(T-t)-\frac{5T}{2}}{T}\right)$$

$$= A\left[-\mathrm{rect}\left(\frac{\frac{T}{2}-t}{T}\right) + \mathrm{rect}\left(\frac{-\frac{1}{2}T-t}{T}\right) + \mathrm{rect}\left(\frac{-\frac{3}{2}T-t}{T}\right)\right]$$



الموضوع : .... التاريخ : ....

(c) $t \geq T$, $t < 2T$

$T \leq t < 2T$

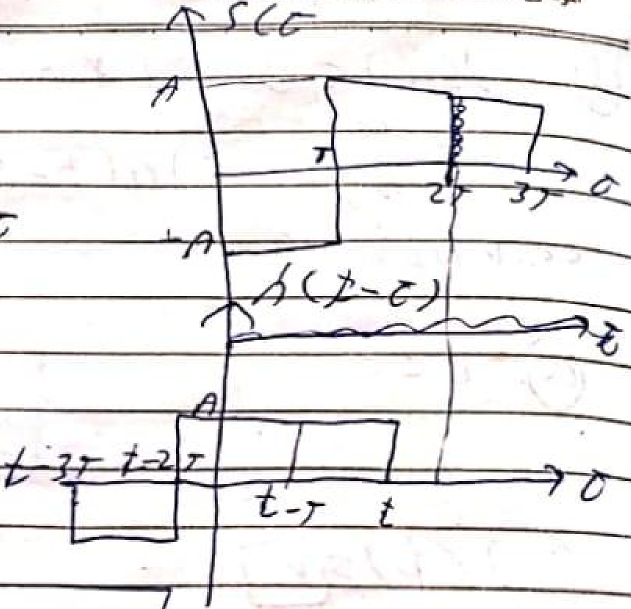
$$y(t) = -\int_0^{t-T} A^2\, d\tau + -\int_{t-T}^{T} A^2\, d\tau + \int_T^{t} A^2\, d\tau$$

$$= -\int_0^{T} A^2\, d\tau + \int_T^{t} A^2\, d\tau$$

$$= -A^2 T + A^2 [\tau]_T^t$$

$$= -A^2 T + A^2 (t - T)$$

$$= -2A^2 T + A^2 t = \boxed{A^2(-2T + t)}$$



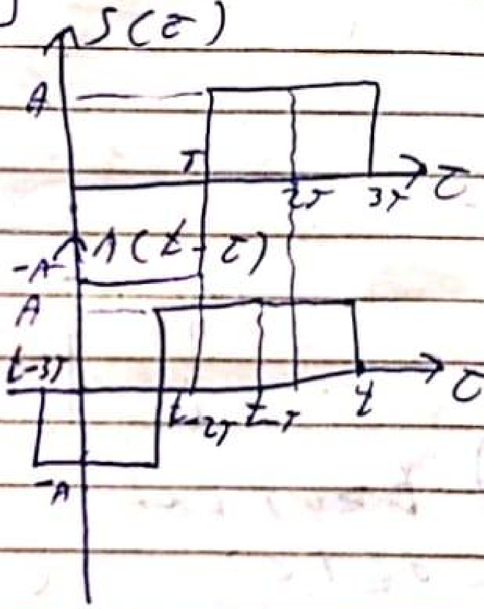
(d) $t \geq 2T$, $t < 3T$

$2T \leq t < 3T$

$$y(t) = \int_0^{t-2T} A^2\, d\tau + -\int_{t-2T}^{T} A^2\, d\tau + \int_T^{t-T} A^2\, d\tau + \int_{t-T}^{2T} A^2\, d\tau + \int_{2T}^{t} A^2\, d\tau$$

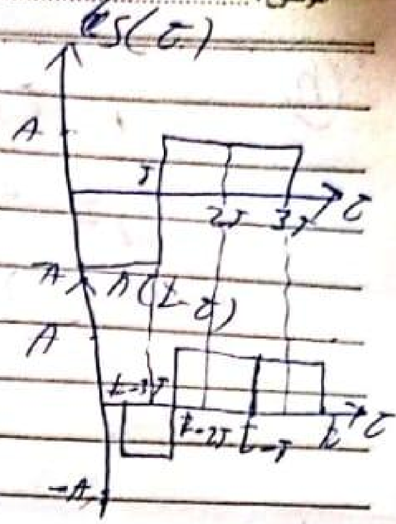
$$= A^2 \left[(t - 2T) - (T - t + 2T) + (T) + (t - 2T) \right]$$

$$= \boxed{A^2 (3t - 6T)}$$



(e) $t \ge 3T$, $t-T < 3T$

$\therefore 3T \le t < 4T$



$$y(t) = \int_{t-3T}^{T} A^2 \, d\tau + -\int_{T}^{t-2T} A^2 \, d\tau + \int_{t-2T}^{2T} A^2 \, d\tau + \int_{2T}^{t-T} A^2 \, d\tau + \int_{t-T}^{3T} A^2 \, d\tau$$

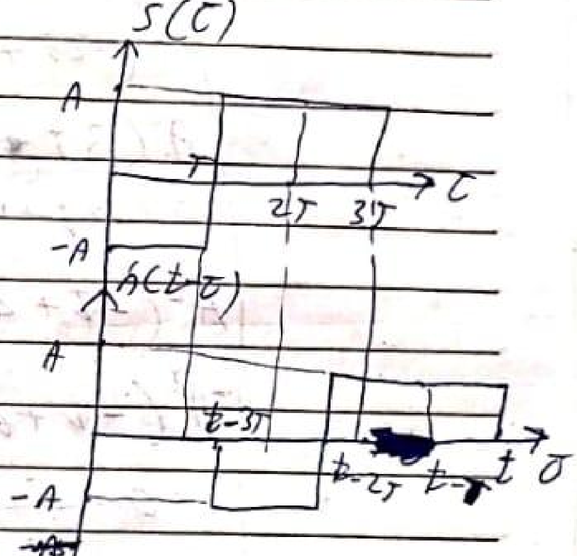
$$= A^2\left[(T-t+3T) - (t-3T) + (2T-t+2T) + (T)\right]$$

$$= \boxed{A^2[-3t+12T]}$$

(f) $t-T \ge 3T \therefore t \ge 4T$

, $t-2T < 3T \therefore t < 5T$

$4T \le t < 5T$



$$y(t) = -\int_{t-3T}^{2T} A^2 \, d\tau + -\int_{2T}^{t-2T} A^2 \, d\tau + \int_{t-2T}^{3T} A^2 \, d\tau$$

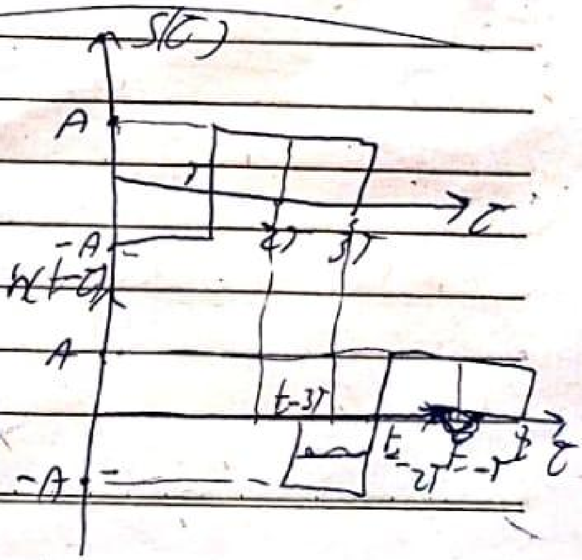
$$= A^2\left[-(2T-t+3T) - (t-4T) + (3T-t+2T)\right]$$

$$= \boxed{A^2[-t+4T]}$$

(g) $t-2T \ge 3T \therefore t \ge 5T$

, $t-3T < 3T \Rightarrow t < 6T$

$\Rightarrow 5T \le t < 6T$



$$y(t) = -\int_{t-3T}^{3T} A^2 \, d\tau = \boxed{-A^2[-t+6T]}$$

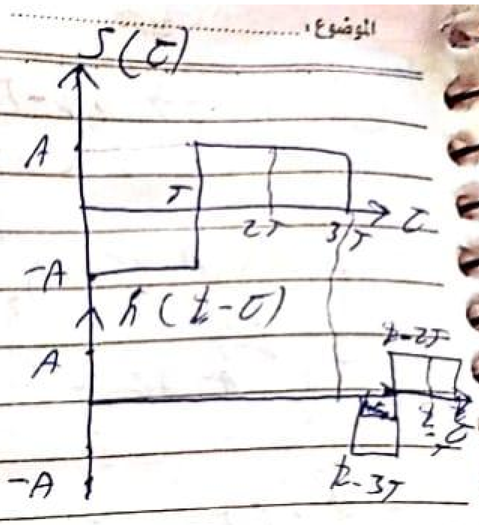
الموضوع :

(h)

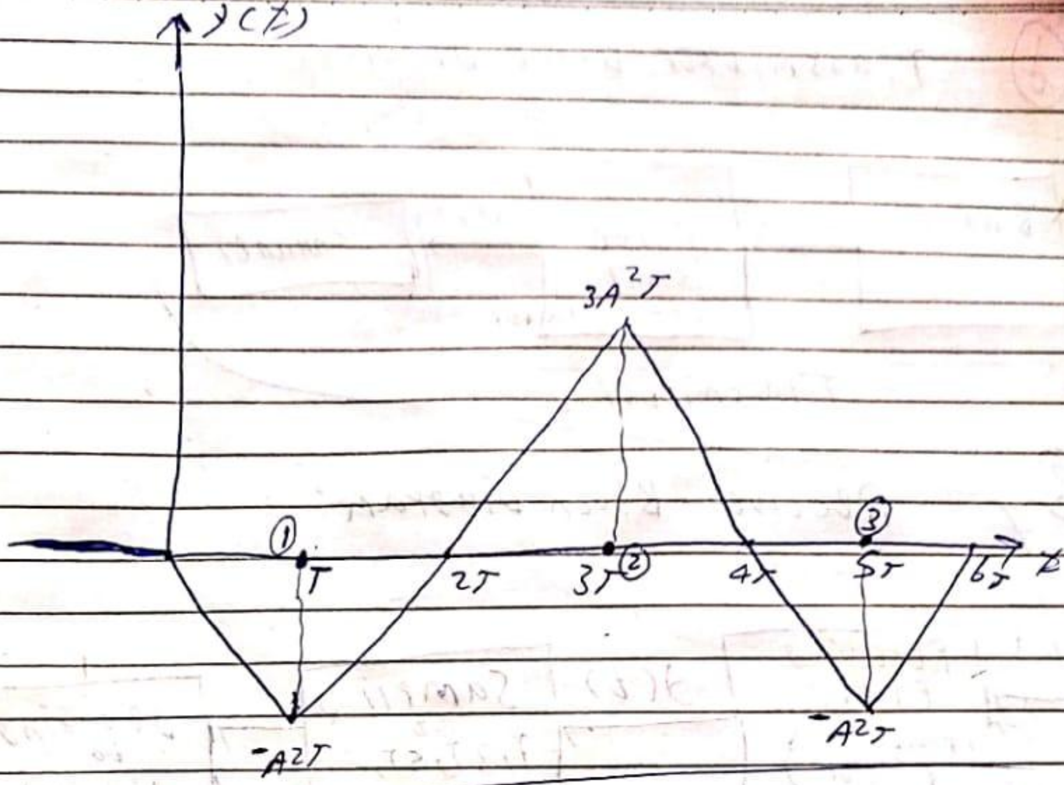
$t-3T \geq 3T$

$6T \leq t < \infty$

$y(t) = \boxed{0}$



$$\therefore y(t) = \begin{cases} 0 & , t < 0 \\ -A^2 t & , 0 \leq t < T \\ A^2(-2T+t) & , T \leq t < 2T \\ A^2(3t-6T) & , 2T \leq t < 3T \\ A^2(-3t+12T) & , 3T \leq t < 4T \\ A^2(-t+4T) & , 4T \leq t < 5T \\ -A^2(-t+6T) & , 5T \leq t < 6T \\ 0 & , 6T \leq t < \infty \end{cases}$$

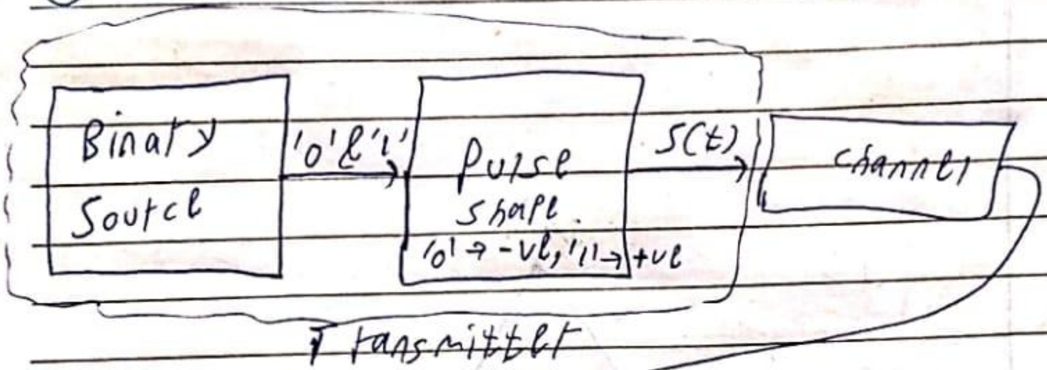


(C) ~~As marked in the previous figure, 3 samples will be at the 3 peaks at (1), (2), (3) at times~~

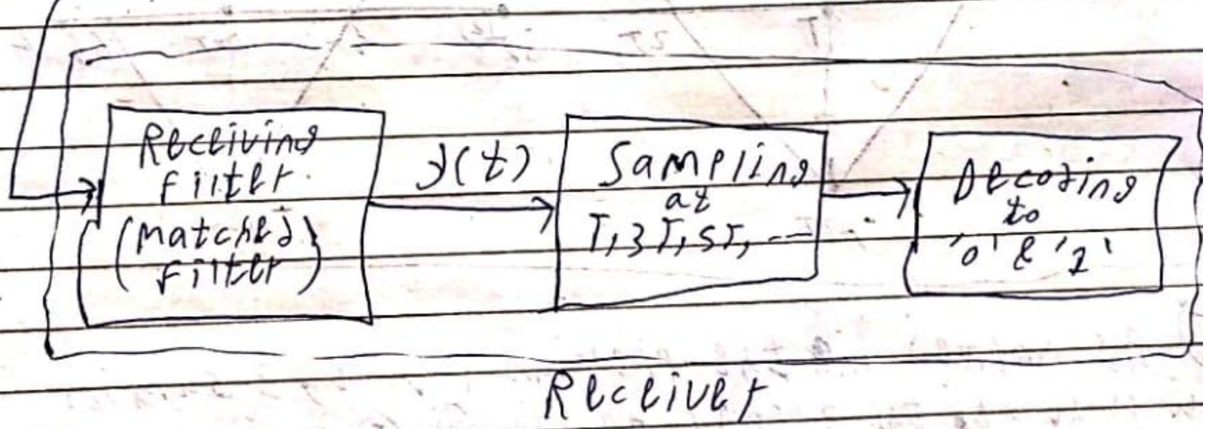
$\boxed{t=T,\ t=3T,\ t=5T}$,

~~Notice: there is an error at b_2 detection it will be detected as 0 instead of 2~~

(d) Transmitter Block Diagram:



(e) Receiver Block Diagram:



Part II: Simulation

1) **Derivation of probability of error the three cases:**

a)

Part 2

$$g(t) = \begin{cases} -A & 0<t<T \\ A & \end{cases}$$

(a) ~~The receiver filter h(t) is a matched filter with~~ unit energy.

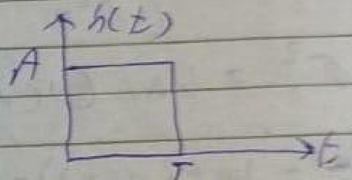
$g(t) = A\,\text{rect}\left(\frac{t-\frac{T}{2}}{T}\right)$ $\qquad r(t) \rightarrow \boxed{h(t)} \rightarrow y(t)$

$h(t) = g(T-t) = A\,\text{rect}\left(\frac{T-t-\frac{T}{2}}{T}\right) = A\,\text{rect}\left(\frac{\frac{T}{2}-t}{T}\right)$

$r(t) = g(t) + w(t)$, $\quad y(t) = r(t) * h(t)$

$\therefore y(t) = g(t) * h(t) + w(t) * h(t)$

$= g_0(t) + n(t)$



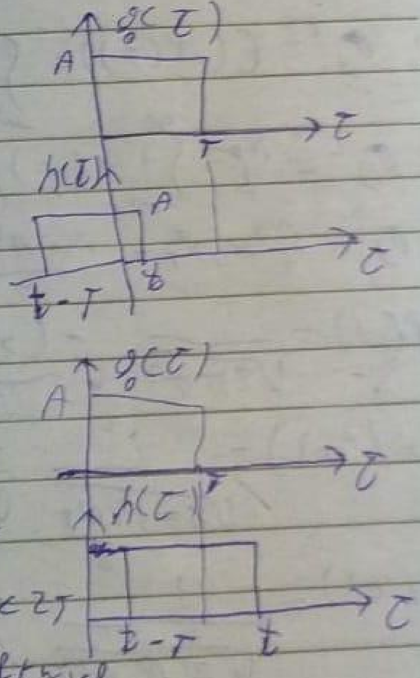
To calculate $g_0(t)$:

$0 \le t < T$

$\therefore g_0(t) = \int_0^t A^2\, d\tau = A^2 t$

$T \le t < 2T$

$g_0(t) = \int_{t-T}^{T} A^2\, d\tau = A^2(2T-t)$



$$\therefore g_0(t) = \begin{cases} A^2 t & 0 \le t < T \\ A^2(2T-t) & T \le t < 2T \\ 0 & \text{otherwise} \end{cases}$$

$$\therefore y(T) = \begin{cases} -A^2 T + n(T) & \text{'0'} \\ A^2 T + n(T) & \text{'1'} \end{cases}$$

$$\mu_y = E(y(T)) = E(s_o(T)) + E(n(T))$$

$$\mu_y = E(\pm A^2T + n(T)) = \pm A^2T + E(n(T))$$

$$E(n(T)) = E\left(\int w(\tau)h(T-\tau)\,d\tau\right) = E\left(\int w(\tau)s(\tau)\,d\tau\right)$$

$$= E\left(\int_0^T \pm A\, w(\tau)\,d\tau\right) = \int_0^T \pm A\, E(w(\tau))\,d\tau = 0$$

$$\therefore \boxed{\mu_y = \pm A^2T}$$

$$\therefore \mu_y = \begin{cases} -A^2T & '0' \\ A^2T & '1' \end{cases}$$

$$\sigma_y^2 = Var(y(T)) = E(s_o(T) + n(T))$$

$$\sigma_y^2 = Var(n(T)) = E(n^2(T)) - E(\underbrace{n(T)}_{0})^2$$

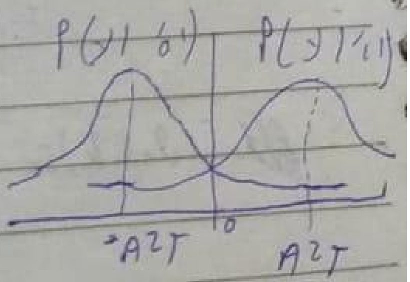
$$\sigma_y^2 = E(n^2(T)) = \int_{-\infty}^{\infty} S_n(f)\,df$$

$$\sigma_y^2 = \frac{N_o}{2}\int_{-\infty}^{\infty} |H(f)|^2\,df = \frac{N_o}{2}\int_0^T |h(t)|^2\,dt$$

$$\sigma_y^2 = \frac{N_o}{2} * A^2T = \frac{N_o A^2T}{2}$$

$$P(y) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{\frac{-(y-\mu_y)^2}{2\sigma^2}}$$

$$P(y|'0') = \frac{1}{\sqrt{2\pi \frac{N_o A^2T}{2}}}\, e^{\frac{-(y+A^2T)^2}{2\frac{N_o A^2T}{2}}}$$



$$P(y|'0') = \frac{1}{\sqrt{\pi N_o A^2T}}\, e^{\frac{-(y+A^2T)^2}{N_o A^2T}}$$

$$P(y|'1') = \frac{1}{\sqrt{\pi N_o A^2T}}\, e^{\frac{-(y-A^2T)^2}{N_o A^2T}}$$

الموضوع: .... التاريخ: ....

$$P(e|'0') = \int_0^{\infty} \frac{1}{\sqrt{\pi N_0 A^2 T}} e^{-\frac{(y+A^2T)^2}{N_0 A^2 T}} dy$$

let $z = \frac{y + A^2T}{\sqrt{N_0 A^2 T}}$

$\Rightarrow dz = \frac{dy}{\sqrt{N_0 A^2 T}}$

$$P(e|'0') = \int_{\frac{A^2T}{\sqrt{N_0 A^2 T}}}^{\infty} \frac{1}{\sqrt{\pi}\,\cancel{\sqrt{N_0 A^2 T}}} \times e^{-z^2} \times \cancel{\sqrt{N_0 A^2 T}}\, dz$$

$$P(e|'0') = \int_{\frac{A^2T}{\sqrt{N_0 A^2 T}}}^{\infty} \frac{1}{\sqrt{\pi}} e^{-z^2} dz = \frac{1}{2} erfc\left(\frac{A^2T}{\sqrt{N_0 A^2 T}}\right)$$

$P(e) = P(e|'1')P('1') + P(e|'0')P('0')$

$P('1') = P('0') = 0.5$

$\therefore P(e) = 0.5\left(P(e|'1') + P(e|'0')\right) = P(e|'0')$

$\therefore P(e) = \frac{1}{2} erfc\left(\frac{A^2T}{\sqrt{N_0 A^2 T}}\right)$

$A = 1,\ T = 1$

$\Rightarrow P(e) = \frac{1}{2} erfc\left(\frac{1}{\sqrt{N_0}}\right)$

b)

(b) The receiver filter $h(t)$ is not existent ($h(t) = \delta(t)$)

$$y(t) = r(t) * h(t) = r(t) * \delta(t) = r(t)$$

$$y(T) = s(T) + w(T)$$

$$= \pm A + w(T)$$

$$E(y(T)) = E(\pm A + w(T)) = \pm A + \underbrace{E(w(T))}_{\to 0}$$

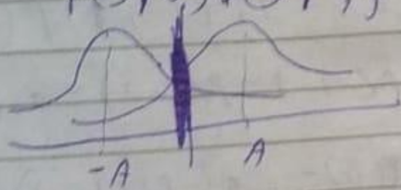
$$\mu_y = \begin{cases} -A & '0' \\ A & '1' \end{cases}$$

$$\sigma_y^2 = Var(y(T)) = Var(\pm A + w(T)) = Var(w(T))$$

$$\sigma_y^2 = Var(w(T)) = \frac{N_0}{2}$$

$$p(y|'0') = \frac{1}{\sqrt{N_0 \pi}} e^{-\frac{(y+A)^2}{N_0}}$$

$$p(y|'1') = \frac{1}{\sqrt{N_0 \pi}} e^{-\frac{(y-A)^2}{N_0}}$$



$$P(e|'0') = \int_0^\infty p(y|'0')\, dy$$

$$= \int_0^\infty \frac{1}{\sqrt{N_0 \pi}} e^{-\frac{(y+A)^2}{N_0}}\, dy$$

let $z = \frac{y+A}{\sqrt{N_0}}$, $dz = \frac{dy}{\sqrt{N_0}}$

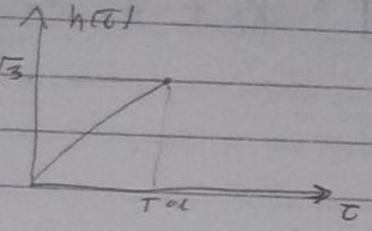
$$P(e|'0') = \int_{\frac{A}{\sqrt{N_0}}}^{\infty} \frac{1}{\sqrt{\pi}\cancel{\sqrt{N_0}}} \times e^{-z^2} \times \cancel{\sqrt{N_0}}\, dz = \frac{1}{2} erfc\left(\frac{A}{\sqrt{N_0}}\right)$$

$$P(e) = P(e|'0')P('0') + P(e|'1')P('1') = \frac{1}{2}\left(P(e|'0') + P(e|'1')\right) = P(e|'0')$$

$$= \frac{1}{2} erfc\left(\frac{A}{\sqrt{N_0}}\right) = \frac{1}{2} erfc\left(\frac{1}{\sqrt{N_0}}\right)$$

c)

The recieive filter $h(t)$ has the following implose response

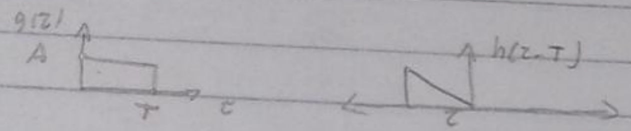


$y(t) = r(t) * h(t) = g_o(t) + h(t)$

$g_o(t) = g(t) * h(t)$

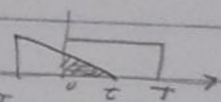
$g_o(t) = \int_{-\infty}^{\infty} g(\tau)\, h(t-\tau)\, \partial\tau$

→ For $\tau < 0$



$g_o(\tau) = 0$

→ For $0 < \tau \le T$



$g_o(\tau) = \int_0^{\tau} \sqrt{3}\, A \tau\, \partial\tau = \sqrt{3}\, A \left[\frac{\tau^2}{2}\right]_0^{\tau}$

$g_o(\tau) = \sqrt{3}\, A \tau$

$y(\tau) = \begin{cases} -\frac{\sqrt{3}}{2} A T^2 + n(\tau) & (0) \\ \frac{\sqrt{3}}{2} A T^2 + n(\tau) & (1) \end{cases}$

$N_y = E(y(t)) = E(g_o(t)) + E(n(t))$

$N_y = E(\pm\frac{\sqrt{3}}{2} A T^2 + n(t)) = \pm\frac{\sqrt{3}}{2} A T^2 + E(n(t))$

$E(n(t)) = E(\int w(t)\, g(t)\, \partial t) = E(\int_0^T \pm A\, w(t)\, \partial t)$

$= \int_0^T \pm A\, E(w(t))\, \partial t = 0$

$N_y = \begin{cases} -\frac{\sqrt{3}}{2} A T^2 & (0) \\ \frac{\sqrt{3}}{2} A T^2 & (1) \end{cases}$

$$\sigma_y^2 = Var(y(t)) = E(g_0(t) + n(t)) = E\left(\pm\frac{\sqrt{3}}{2}AT^2 + n(t)\right)$$

$$\sigma_y^2 = Var(n(t)) = E(n^2(t)) + E(n(t))^2 = E(n^2(t)) = \int_{-\infty}^{\infty} S_n(f)\, df$$

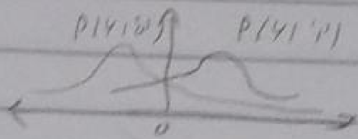
$$\sigma_y^2 = \frac{N_0}{2}\int_{-\infty}^{\infty}|H(f)|^2\, df = \frac{N_0}{2}\int_0^T |h(t)|^2\, dt = \frac{N_0}{2}\int_0^T 3t^2\, dt$$

$$\sigma_y^2 = \frac{N_0}{2}\left[\frac{3t^3}{3}\right]_0^T = \frac{N_0}{2}T^3 \quad //$$

$$P(y) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{\frac{-(y-\mu)^2}{2\sigma^2}}$$

$$P(y|'0') = \frac{1}{\sqrt{2\pi \times \frac{N_0}{2}T^3}} \times e^{\frac{-(y+\frac{\sqrt{3}}{2}AT^2)}{N_0T^3}} = \frac{1}{\sqrt{N_0\pi T^3}} e^{\frac{-(y+\frac{\sqrt{3}}{2}AT^2)}{N_0T^3}}$$

$$P(y|'1') = \frac{1}{\sqrt{\pi N_0 T^3}} e^{\frac{-(y-\frac{\sqrt{3}}{2}AT^2)}{N_0T^3}}$$



$$P(e|'0') = \int_0^{\infty} \frac{1}{\sqrt{\pi N_0 T^3}} e^{\frac{-(y+\frac{\sqrt{3}}{2}AT^2)}{N_0T^3}}$$

$$\text{let } z = \frac{y+\frac{\sqrt{3}}{2}AT^2}{\sqrt{N_0T^3}}$$

$$dz = \frac{dy}{\sqrt{N_0T^3}}$$

$$P(e|'0') = \int_{\frac{\frac{\sqrt{3}}{2}AT^2}{\sqrt{N_0T^3}}} \frac{1}{\sqrt{\pi N_0 T^3}} e^{-z} \cancel{\sqrt{N_0T^3}}\, dz$$

$$= \frac{1}{2}\, erfc\left(\frac{-\sqrt{3}}{2}\frac{AT^2}{\sqrt{N_0T^3}}\right)$$

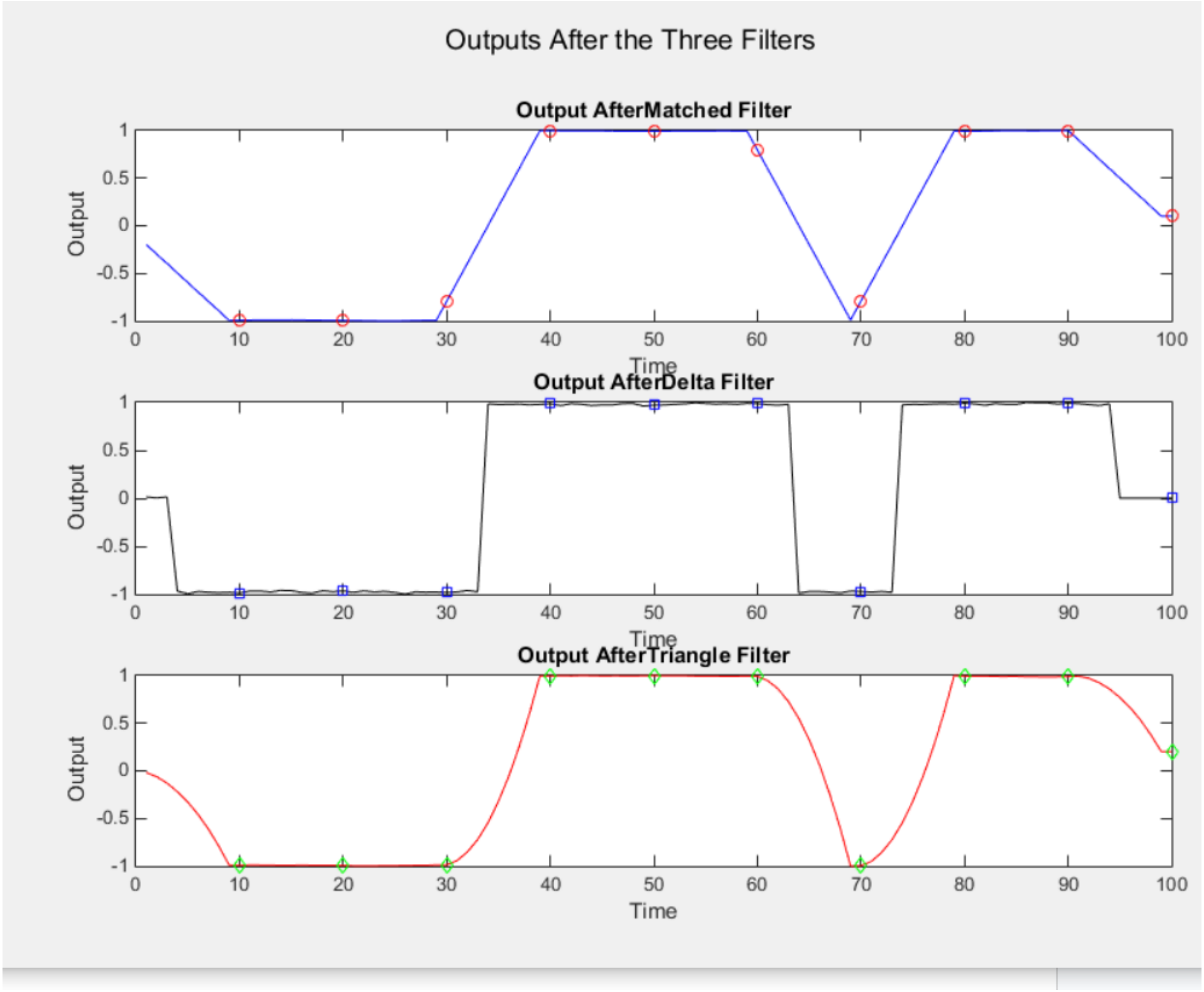
let $P('0') = P('1')$

$$P(e) = P(e|'0') = \frac{1}{2}\, erfc\left(\frac{\sqrt{3}\,AT^2}{2\sqrt{N_0T^3}}\right)$$

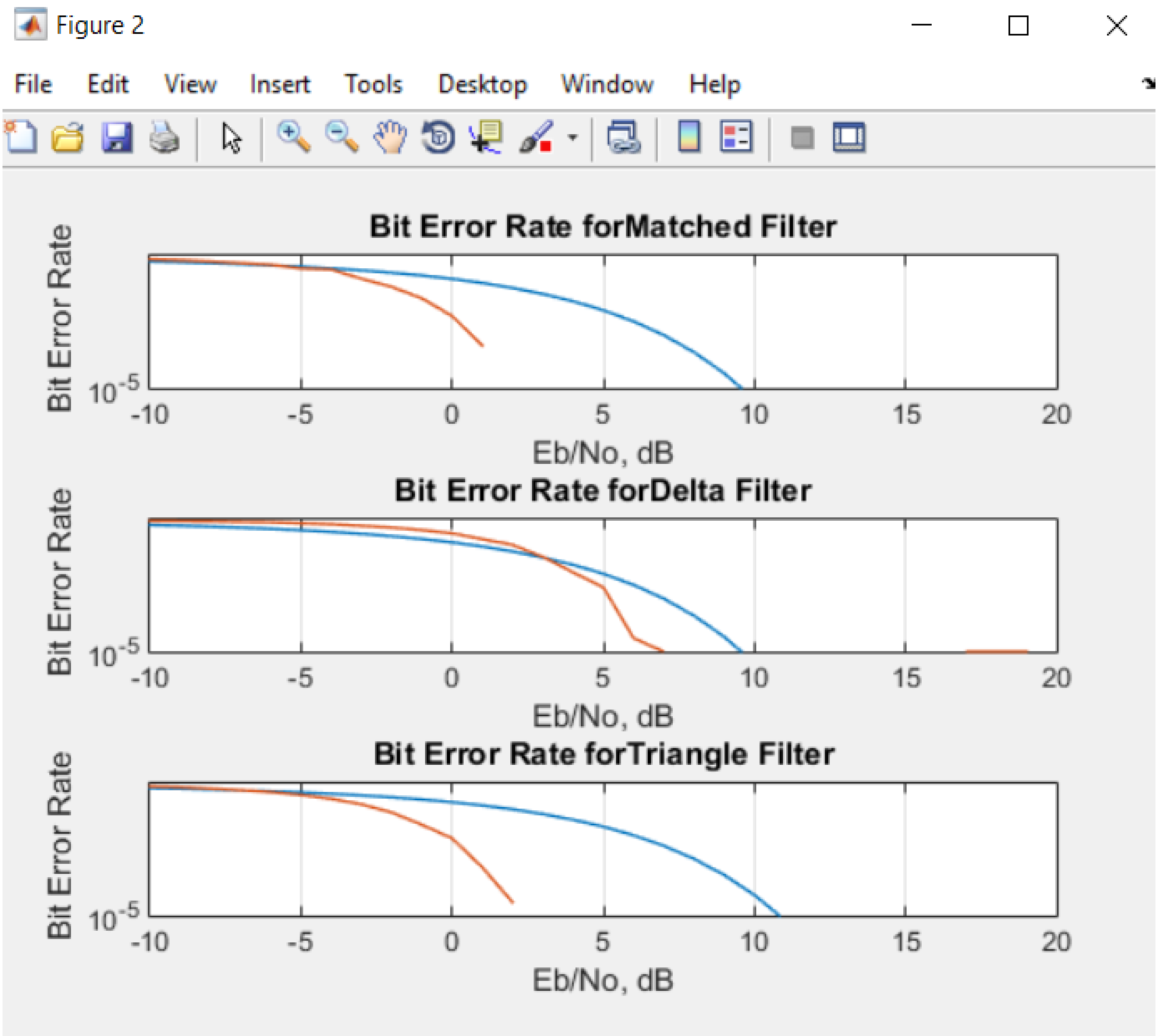
$\because A = 1$ & $T = 1$

$$P(e) = P(e|'0') = \frac{1}{2}\, erfc\left(\frac{\sqrt{3}}{2\sqrt{N_0}}\right)$$

2) Output of the three filters (for 10 bits and 20 for E/N0 (dB))



3) Output of BER simulated and theoretical (for 10^5 bits)



5. Is the BER an increasing or a decreasing function of E/No? Why?

decreasing, because when E/No is increased, Energy of signal increases with respect to energy of noise, so, it is more immune to noise, so, error decreases, and from the formula, BER = 0.5erfc(sqrt(E/No))

6. Which case has the lowest BER? Why?

Matched filter, because it has unit energy and it tries to simulate again the transmitted bits after noise addition.

Matlab Code:

```
function [] = plotSemilogy(eNoDB, theoryBER, simBER, filterName)
   semilogy(eNoDB,theoryBER,'','Linewidth',1);
   hold on
   semilogy(eNoDB,simBER,'','Linewidth',1);
   axis([-10 20 10^-5 0.5])
   grid on
   %legend('Theoretical', 'Simulated');
   xlabel('Eb/No, dB');
   ylabel('Bit Error Rate');
   title(strcat('Bit Error Rate for', ' ', filterName));
end

function [] = subplotOutput(time, output, samplingPoints, filterName,
index, color, shape)
   subplot(3,1,index);
   plot(time, output, color, samplingPoints, output(samplingPoints),
shape);
   xlabel('Time');
   ylabel('Output');
   title(strcat('Output After', ' ', filterName));
end

function [] = plotThreeOutputs(time, outputMatched, outputDelta,
outputTri, samplingPoints)
   figure;
```

```
    subplotOutput(time, outputMatched, samplingPoints, 'Matched Filter',
1, 'b-', 'ro');
    subplotOutput(time, outputDelta, samplingPoints, 'Delta Filter', 2, 'k-',
'bs');
    subplotOutput(time, outputTri, samplingPoints, 'Triangle Filter', 3, 'r-',
'gd');
    suptitle('Outputs After the Three Filters');
end

function [BER, filteredOutput, sampledOutput, gtAfterNoise] =
getBER(bits, gt, filter, bitsSize)
    BER = zeros(1, 31);
    i = 1;

    % Normalize to make the signal with unit energy
    % onePulse = ones(1, 10);
    % gt = gt / norm(onePulse);

    for eNoDB = -10:1:20
        % Add AWGN noise with snr = E / (No/2)
        gtAfterNoise = awgn(gt, 2*eNoDB, 'measured');

        % Apply the filter
        filteredOutput = conv(filter, gtAfterNoise);

        % Repeat last element to make size to be correct
        filteredOutput(end + 1) = filteredOutput(end);

        % Ignore first 10 elements
```

```
        filteredOutput = filteredOutput(11:end);

        % Normalize the output
        outputMin = min(filteredOutput);
        outputMax = max(filteredOutput);
        filteredOutput = (filteredOutput - outputMin) / (outputMax -
outputMin);
        filteredOutput = 2*filteredOutput - 1;

        % Sample the output
        sampledOutput = filteredOutput(10:10:(10*bitsSize));

        % Take a decision
        receivedBits = double((sampledOutput > 0));

        % Counting errors
        BER(i) = sum(bits ~= receivedBits);
        i = i + 1;
    end

    BER = BER / bitsSize;
end

close all

% Set the size of the bits array
% TODO: make this 10^5
bitsSize = 10^5;
```

```
% Generate a random array of zeros and ones
bits = randi([0,1],1,bitsSize);

% Pulse Shaping (make every 0 => -1, 1 => 1)
shaped = 2*bits - 1;

% Array that will hold same data but repeated to simulate holding every bit
gt = zeros(1, 10 * bitsSize);

% Pulse Shaping (repeating every bit 10 times)
for i = 1:1:bitsSize
   gt((10*i):(10*i)+10) = shaped(i);
end

% Cut last 10 elements to make correct size
gt = gt(1:end-10);

tBits = 1:1:bitsSize;
tGt = 1:1:(10 * bitsSize);
eNoDB = -10:1:20;
% plot(tBits, bits),
% xlabel('Time'); ylabel('Bits');
% title('Generation of Random Bits');

% figure;
% plot(tGt, gt),
```

```
% xlabel('Time'); ylabel('g(t)');
% title('After Pulse Shaping');

% 1- Matched Filter
% Get matched filter (rectangular pulse with unit energy)
matchedFilter = ones(1, 10);
[simBERMatched, filteredOutputMatched, sampledOutputMatched,
gtAfterNoiseMatched] = getBER(bits, gt, matchedFilter, bitsSize);

figure;
plot(tGt, filteredOutputMatched),
xlabel('Time'); ylabel('y(t)');
title('After Matched Filter');

disp(simBERMatched);

% Get theoretical BER
theoryBERMatched = 0.5*erfc(sqrt(10.^(eNoDB/10)));

figure
subplot(3,1, 1);
plotSemilogy(eNoDB, theoryBERMatched, simBERMatched, 'Matched
Filter');

% 2- h(t) = delta(t)
% Get matched filter (rectangular pulse with unit energy)
deltaFilter = zeros(1, 10);
deltaFilter(5) = 1;
```

```
[simBERDelta, filteredOutputDelta, sampledOutputDelta,
gtAfterNoiseDelta] = getBER(bits, gt, deltaFilter, bitsSize);

disp(simBERDelta);

% Get theoretical BER
theoryBERDelta = 0.5*erfc(sqrt(10.^(eNoDB/10)));

subplot(3,1, 2);
plotSemilogy(eNoDB, theoryBERDelta, simBERDelta, 'Delta Filter');



% 3- h(t) = right-angled triangle with height = sqrt(5), width = 10
triYValues = linspace(0, sqrt(3), 10);
triYValues = triYValues/ norm(triYValues);
[simBERTri, filteredOutputTri, sampledOutputTri, gtAfterNoiseTri] =
getBER(bits, gt, triYValues, bitsSize);

disp(simBERTri);
% Get theoretical BER
theoryBERTri = 0.5*erfc((sqrt(3) / 2) * sqrt(10.^(eNoDB/10)));

subplot(3,1, 3);
plotSemilogy(eNoDB, theoryBERTri, simBERTri, 'Triangle Filter');

% figure;
% plot(tGt, gtAfterNoise),
% xlabel('Time'); ylabel('g(t)');
```

```
% title('After Adding AWGN');

% figure;
% plot(tGt, filteredOutput),
% xlabel('Time'); ylabel('y(t)');
% title('After Matched Filter');

% tSampled = 1:1:length(sampledOutput);
% figure;
% plot(tSampled, sampledOutput),
% xlabel('Time'); ylabel('ySampled(t)');
% title('After Sampling');

% tTriangle = 1:1:10;
% figure;
% plot(tTriangle, yValues),
% xlabel('Time'); ylabel('tri(t)');
% title('Triangle Shaping');

samplingPoints = 10:10:(10*bitsSize);

plotThreeOutputs(tGt, filteredOutputMatched, filteredOutputDelta,
filteredOutputTri, samplingPoints)
```