American International University- Bangladesh (AIUB) Faculty of Engineering (FE)

|  |  |  |  |
| --- | --- | --- | --- |
| **Course Name:** | DATA COMMUNICATION | **Course Code:** | COE3103 |
| **Semester:** | Summer 2024-25 | **Section:** | D |
| **Faculty:** | MOHAMMAD ASADUZZAMAN KHAN | **Group:** | 05 |

|  |  |
| --- | --- |
| **Experiment No:** | 02 |
| **Experiment Name:** | Study of signal frequency, spectrum, bandwidth, and quantization using MATLAB |

|  |  |  |  |
| --- | --- | --- | --- |
| **Group Members** | | **Name** | **ID** |
|  | 1. | Basudeb Kundu | 23-50856-1 |
| 2. | Prohlad Chandra Das | 23-50922-1 |
| 3. | Debashis Kumar Das | 23-50953-1 |
| 4. | Indronill Dutta Nill | 23-50974-1 |
| 5. | Nafiur Rahman Nirob | 23-50991-1 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Performance Date:** | 20-7-25 | **Due Date:** | 27-7-25 |

**Marking Rubrics (to be filled by Lab Instructor)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Category | Proficient [6] | Good [4] | Acceptable [2] | Unacceptable [1] | Secured Marks |
| **Theoretical Background, Methods & procedures sections** | All information, measures and variables are provided and  explained. | All Information provided is sufficient, but more explanation is  needed. | Most information is correct, but some information may be  missing or inaccurate. | Much information is missing and/or inaccurate. |  |
| **Results** | All of the criteria are met; results are described clearly and accurately; | Most criteria are met, but there may be some lack of clarity and/or incorrect information. | Experimental results don’t match exactly with the theoretical values and/or analysis  is unclear. | Experimental results are missing or incorrect; |  |
| **Discussion** | Demonstrates thorough and sophisticated understanding.  Conclusions drawn are appropriate for  analyses; | Hypotheses are clearly stated, but some concluding statements not supported by data or data not well  integrated. | Some hypotheses missing or misstated; conclusions not supported by data. | Conclusions don’t match hypotheses, not supported by data; no integration of data from different sources. |  |
| **General formatting** | Title page, placement of figures and figure captions, and other  formatting issues all correct. | Minor errors in formatting. | Major errors and/or missing information. | Not proper style in text. |  |
| **Writing & organization** | Writing is strong and easy to understand; ideas are fully elaborated and connected; effective transitions between sentences; no  typographic, spelling, or grammatical errors. | Writing is clear and easy to understand; ideas are connected; effective transitions between sentences; minor typographic, spelling, or grammatical errors. | Most of the required criteria are met, but some lack of clarity, typographic, spelling, or grammatical errors are present. | Very unclear, many errors. |  |
| Comments: |  | | | Total Marks (Out of **30**): |  |

**Title**

Study of signal frequency, spectrum, bandwidth, and quantization using MATLAB

**Abstract**

This experiment is designed to-

1.To understand the use of MATLAB for solving communication engineering problems.

2.To develop understanding of MATLAB environment, commands, and syntax.

**Introduction**

Digital communication is fundamental to modern information exchange, ensuring efficient and reliable data transmission across various applications. This experiment focuses on understanding and implementing **Line Coding Techniques**, which are essential for converting digital data into transmittable signals.

Key concepts in digital communication include **frequency, spectrum, bandwidth, and quantization**, which define the characteristics and efficiency of signal transmission. **Frequency** refers to the rate at which a signal oscillates, measured in Hertz (Hz). The **spectrum** represents the range of frequencies a signal occupies, while **bandwidth** denotes the portion of the spectrum utilized for data transmission, directly influencing the data rate and system performance. **Quantization**, a crucial process in digital communication, involves converting continuous analog signals into discrete digital values, reducing information loss and optimizing signal processing.

In this experiment, different **line coding techniques** such as **Unipolar, Polar, Bipolar, and Manchester encoding** are analyzed to examine their impact on signal transmission efficiency. By studying these methods, we gain insights into their bandwidth requirements, spectral characteristics, and practical applications in real-world communication systems.

This lab provides hands-on experience in digital communication, enabling students to observe waveform representations, compare encoding methods, and assess their effects on data integrity. The report presents the experimental setup, methodology, observations, and conclusions derived from the study of line coding and its role in efficient data transmission.

|  |  |
| --- | --- |
| Fig1: A composite periodic signal | Fig2: Decomposition of a composite periodic signal in the time and frequency domains |

|  |  |  |
| --- | --- | --- |
| Fig3: Bandwidth of a periodic & nonperiodic signal | Fig4: An example analog signal | Fig5: A quantized signal |

**Results and Discussion**

**(a)** Show time domain and frequency domain representations of signal\_x in a single figure window using subplot. Use axis, or xlim, or ylim to appropriately represent the signal.

|  |
| --- |
| **Code:** |
| clc;      clear;        % //Define given values      E = 8;      F = 5;      G = 6;      % //Compute amplitudes and frequencies      a1 = G + 1;  % 7      a2 = F + 2;  % 7      a3 = E + 3;  % 11      f1 = E + 1;  % 9 Hz      f2 = F + 2;  % 7 Hz      f3 = G + 3;  % 9 Hz      % //Time vector      fs = 100;      t = 0:1/fs:1;      % //Generate signals      x1 = a1 \* cos(2 \* pi \* f1 \* t);      x2 = a2 \* sin(2 \* pi \* f2 \* t);      x3 = a3 \* cos(2 \* pi \* f3 \* t);      % //Composite signal      signal\_x = x1 + x2 + x3;      % //Compute FFT      N = length(signal\_x);      Y = fft(signal\_x);      Y\_mag = abs(Y/N);    % Normalize magnitude      frequencies = (0:N-1) \* (fs/N);  % Frequency axis      half\_N = floor(N/2);      % //Plot time-domain representation      figure;      subplot(2,1,1);      plot(t, signal\_x, 'b', 'LineWidth', 1.5);      xlabel('Time (s)');      ylabel('Amplitude');      title('Time-Domain Representation of Composite Signal');      grid on;      xlim([0 1]);      % //Plot frequency-domain representation      subplot(2,1,2);      stem(frequencies(1:half\_N), Y\_mag(1:half\_N), 'r', 'LineWidth', 1.5);      xlabel('Frequency (Hz)');      ylabel('Magnitude');      title('Frequency-Domain Representation (FFT)');      grid on;      xlim([0 20]); % Show relevant frequency range |

|  |
| --- |
| **Result and Discussion:** |
| **Fig6: Frequency-domain & Time-domain representation**  **representation** |
|  The time-domain representation of the composite signal effectively illustrates the summation of individual sinusoidal components.   The frequency-domain representation, obtained using the Fast Fourier Transform (FFT), highlights the spectral content of the signal, showing peaks at the expected frequencies.   The observed spectral components validate the theoretical predictions, demonstrating how different frequencies contribute to the overall signal composition. |

**(b)** Quantize signal\_x in 4 equally distributed levels and provide image for one cycle of the original signal and quantized signal. Use axis, or xlim, or ylim to appropriately represent the signal. [Use quantiz() function]

|  |
| --- |
| **Code:** |
| clc;  clear;  % Define given values  E = 8;  F = 5;  G = 6;  % Compute amplitudes and frequencies  a1 = G + 1; % 7  a2 = F + 2; % 7  a3 = E + 3; % 11  f1 = E + 1; % 9 Hz  f2 = F + 2; % 7 Hz  f3 = G + 3; % 9 Hz  % Time vector  fs = 100;  t = 0:1/fs:1;  % Generate signals  x1 = a1 \* cos(2 \* pi \* f1 \* t);  x2 = a2 \* sin(2 \* pi \* f2 \* t);  x3 = a3 \* cos(2 \* pi \* f3 \* t);  % Composite signal  signal\_x = x1 + x2 + x3;  % Quantization  num\_levels = 4;  min\_val = min(signal\_x);  max\_val = max(signal\_x);  % Define step size and levels  step\_size = (max\_val - min\_val) / (num\_levels - 1);  codebook = min\_val:step\_size:max\_val;  % Manual quantization  quantized\_signal = zeros(size(signal\_x));  for i = 1:length(signal\_x)  [~, idx] = min(abs(signal\_x(i) - codebook));  quantized\_signal(i) = codebook(idx);  end  % Extract one cycle of the original and quantized signal  T = 1 / f1;  idx = t <= T;  % Plot the signals  figure;  plot(t(idx), signal\_x(idx), 'b', 'LineWidth', 1.5); hold on;  stairs(t(idx), quantized\_signal(idx), 'r', 'LineWidth', 1.5);  xlabel('Time (s)');  ylabel('Amplitude');  title('Original and Quantized Signal (One Cycle)');  legend('Original Signal', 'Quantized Signal');  grid on;  xlim([0 T]);  ylim([min\_val - 1, max\_val + 1]); |
| **Result and Discussion:** |
| **Fig7: Original and Quantized Signal for 4 equally distributed levels** |
|  The quantization process reduces the continuous signal into discrete levels, introducing quantization error.   The graphical representation shows a clear distinction between the original and quantized signal.   Lower quantization levels result in higher distortion, reducing signal accuracy. |

**(c)** Quantize signal\_x in 8 equally distributed levels and provide image for one cycle of the original signal and quantized signal. Use axis, or xlim, or ylim to appropriately represent the signal. [Do not use quantiz() function]

|  |
| --- |
| **Code:** |
| clc;      clear;  % //Define given values      E = 8;      F = 5;      G = 6;      % //Compute amplitudes and frequencies      a1 = G + 1;  % 7      a2 = F + 2;  % 7      a3 = E + 3;  % 11      f1 = E + 1;  % 9 Hz      f2 = F + 2;  % 7 Hz      f3 = G + 3;  % 9 Hz      % //Time vector      fs = 100;      t = 0:1/fs:1;   % Time from 0 to 1 sec      % //Generate signals      x1 = a1 \* cos(2 \* pi \* f1 \* t);      x2 = a2 \* sin(2 \* pi \* f2 \* t);      x3 = a3 \* cos(2 \* pi \* f3 \* t);      % //Composite signal      signal\_x = x1 + x2 + x3;      % //Quantization Process (8 levels)      num\_levels = 8;      min\_val = min(signal\_x);      max\_val = max(signal\_x);        % //Define quantization levels and step size      step\_size = (max\_val - min\_val) / (num\_levels - 1);      quant\_levels = min\_val:step\_size:max\_val;  % 8 levels      % //Manual Quantization: Find the nearest level for each sample      quantized\_signal = zeros(size(signal\_x));      for i = 1:length(signal\_x)          % Find the closest quantization level          [~, idx] = min(abs(quant\_levels - signal\_x(i)));          quantized\_signal(i) = quant\_levels(idx);      end      % //Extract one cycle of the original and quantized signal      T = 1 / f1;      idx = t <= T;      figure;      plot(t(idx), signal\_x(idx), 'b', 'LineWidth', 1.5); hold on;      stairs(t(idx), quantized\_signal(idx), 'r', 'LineWidth', 1.5);      xlabel('Time (s)');      ylabel('Amplitude');      title('Original and Quantized Signal (One Cycle)');      legend('Original Signal', 'Quantized Signal');      grid on;      xlim([0 T]); % Display only one cycle      ylim([min\_val-1, max\_val+1]); % Adjust Y-axis limits |
| **Result and Discussion:** |
| **Fig8: Original and Quantized Signal for 8 equally distributed levels** |
| * Increasing the number of quantization levels improves signal accuracy by reducing quantization error. * The quantized signal closely follows the original waveform compared to the 4-level case. * The trade-off in quantization is between data resolution and storage/bandwidth requirements. |

**Conclusion**

This experiment provided valuable insights into the fundamental concepts of signal processing, including frequency, spectrum, bandwidth, and quantization. By analyzing the time and frequency domain representations of signals, we observed how different frequency components contribute to the overall signal structure. The use of MATLAB for signal visualization and analysis proved to be an effective approach for understanding these concepts. Additionally, the quantization process demonstrated the trade-off between signal accuracy and data compression, where increasing the number of quantization levels reduced quantization errors but required more data storage. Overall, this study reinforced the significance of these principles in digital communication and signal processing applications, highlighting their practical importance in real-world scenarios.