Green University of Bangladesh

Department of Computer Science and Engineering (CSE)
Semester: (Fall, Year: 2023), B.Sc. in CSE (Day)

Data Transmission Simulator

Course Title: Data Communication Lab
Course Code: CSE - 308
Section: 213 D3

Students Details

Name	ID
MD SAIFUL ISLAM RIMON	213002039
NAZMUN NAHAR	191002251

Submission Date: 09-01-24
Course Teacher's Name: MS. RUSMITA HALIM CHAITY

[For teachers use only: Don't write anything inside this box]

Lab Project Status	
Marks:	**Signature:**
Comments:	**Date:**

Contents

Chapter 1

Introduction

1.1 Overview

The goal of the **"Data Transmission Simulator"** project is to create a thorough modeling tool for different data transfer methods. A wide range of techniques will be covered by this simulator, such as bit and character stuffing and de-stuffing, encoding and decoding schemes such as NRZ-I, Manchester, AMI, and pseudo ternary, error detection and correction using Hamming Code, and Cyclic Redundancy Check (CRC) using Parity Checker. It will also have an IPv4 implementation for converting Decimal to Binary and vice versa.

1.2 Motivation

Efficient and error-free data transfer is critical in the digital era. It is essential for academics, professionals, and students studying computer science and telecommunications to comprehend and replicate these mechanisms. Nevertheless, interactive tools covering a wide variety of data transmission methods are hard to come across. Our study seeks to close this knowledge gap by offering a practical learning aid for simulating and comprehending these important methods.

1.3 Problem Definition

1.3.1 Problem Statement

It is now quite difficult to fully comprehend and illustrate the different facets of data transmission in a research or instructional context. A comprehensive simulation tool is required, one that not only illustrates the theoretical elements but also enables users to interactively investigate and comprehend the real-world ramifications of various data transmission strategies.

1.3.2 Complex Engineering Problem

Table 1.1: Summary of the attributes touched by the mentioned projects

Name of the P Attributess	Explain how to address
P1: Depth of knowledge required	A solid understanding of digital communications theory, including error correction techniques and data encoding schemes, is required for this project. It's essential to be proficient in programming. Accurate simulation of data transmission processes requires a thorough understanding of networking protocols, stuffing and de-stuffing, encoding and decoding, IPv4, etc.
P2: Range of conflicting requirements	—-
P3: Depth of analysis required	Conducting a concise performance analysis of data transmission techniques is essential, focusing on the effectiveness of error detection and correction algorithms like CRC and Hamming Code and then the analysis of stuffing and de-stuffing. It's important to assess user interaction and the simulation is understandable to the end user.
P4: Familiarity of issues	—-
P5: Extent of applicable codes	This project includes lots of algorithms and techniques integrated within a single project which is helpful for educational purposes, research and so on. So, we can say that, this indicates a professional and standard project. This project includes lots of creative practice here.
P6: Extent of stakeholder involvement and conflicting requirements	—-
P7: Interdependence	—-

1.4 Design Goals/Objectives

The project's design goals and objectives are as follows:

1. **Comprehensive Coverage:** To include a wide range of data transmission techniques from basic stuffing and de-stuffing to complex encoding/decoding schemes.

2. **Interactive Simulation:** Provide a user-friendly interface for simulating and visualizing different data transmission scenarios.

3. **Accuracy and Reliability:** Ensure the simulations are accurate and reflect real-world scenarios.

4. **Educational Utility:** Design the simulator to be an effective educational tool for students and professionals alike.

5. **User-Friendly Interface:** Create an intuitive and user-friendly interface

6. **Extensibility:** Build the simulator in a way that allows future expansion and inclusion of more data transmission techniques.

1.5 Application

Data Transmission Simulator is designed for use in various facilities, including:

- **Educational Tool:** Ideal for educational institutions for teaching data communication and networking concepts.
- **Research:** Researchers can use the simulator to test and visualize different data transmission scenarios.
- **Professional Training:** Professionals in telecommunications and IT can use the simulator for training and development purposes.
- **Network Planning and Analysis:** Helps in planning and analyzing network data flow and error handling mechanisms.

Chapter 2

Design/Development/Implementation of the Project

2.1 Introduction

This project is a comprehensive Matlab language program designed for a **Data Transmission Simulator**. The code is written in Matlab language and is intended to run on the Matlab App Designer, specifically using the Matlab software.
This project is a robust example of a transmission simulator application written in Matlab language, showcasing efficient use of Matlab packages, Matlab code & tools, Matlab App Designer program, tools in App Designer Program & how to use those tools to embed all the functionalities of application development.

2.2 Project Details

Key Features:

1. **User Interface:** The code starts with a menu-driven interface which we are calling "Home page", offering various options like
   (a) Line Coding
   (b) Hamming Encoding & Decoding
   (c) Analog to Digital Signal Conversion
   (d) Character Stuffing & Destuffing
   (e) IPv4 Conversion
   (f) Bit Stuffing & Destuffing
   (g) CRC Error Detection
2. **Easy to use:** The program consists of input boxes, output boxes, a graphical demonstration system for signals and buttons for different operations.

Technical Overview:

- **Matlab Language:** The program is written in Matlab language, which provides high-level control over hardware and is ideal for user-level programming including GUI (Graphical User Interface).
- **Functions for Operations:** The code is organized into multiple functions, each handling a specific task. In the whole program, every single task is done in function including lite operations like: buttons, input box, output box etc.
- **Classes for Operations:** Classes are available here for different purposes like Public or Private declarations and others. Some of the functions, variables or other package are in public access and some are in private access based on the operation.
- **Callbacks for Operations:** Callbacks are available here for calling a specific task in the background of that particular object like: buttons, input boxes, output boxes, signal generators.

Figure 2.1: Features are embeded in this project

2.3 Implementation

The implementation of the Data Transmission Simulator in Matlab language encompasses various elements detailed in multiple subsections. This section will cover the main aspects, including the workflow and the tools and libraries used.

The workflow

Start Screen and Home Page

- Upon launching the app, display a start screen that transitions to a home page.
- The home page should present a menu with the options listed (Line Coding, Hamming Encoding & Decoding, etc.).

Menu Selection

- Implement a menu-driven interface where users can select any of the options (a-g) to perform a specific task.

Task Modules

- For each option selected from the menu, open the corresponding module:
  - Line Coding: A module that converts data into a line code format, suitable for transmission.
  - Hamming Encoding & Decoding: A module for error-detecting and error-correcting code operations.
  - Analog to Digital Signal Conversion: A module that samples analog signals and converts them into digital.
  - Character Stuffing & Destuffing: A module that adds or removes special characters to/from data to provide a clear start and end signal.
  - IPv4 Conversion: A module that deals with the conversion and manipulation of IPv4 addresses.
  - Bit Stuffing & Destuffing: A module for inserting or removing bits from data to prevent misinterpretation.
  - CRC Error Detection: A module that calculates and checks CRC values for error detection in data transmission.

Input and Output

- Each module should provide input boxes for user input and display output boxes for results.
- Where applicable, provide graphical representation of the data or signals.

Operations and Interactivity

- Include buttons and interactive elements for users to execute operations like 'encode', 'decode', 'convert', etc.
- Provide clear instructions or tooltips for users to understand what each button does.

Validation and Feedback

- Include data validation to ensure users input data in the correct format.
- Provide immediate feedback on operations, such as success messages or error prompts.

Documentation

- Offer a help section or documentation within the app to assist users with complex tasks or to provide information about the algorithms used.

Exit Workflow

- Allow users to easily navigate back to the home page or exit the app from any module.

Implementation details (with screenshots and programming codes)

CODE [Home Page]:

```
classdef home < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                        matlab.ui.Figure
        DeveloperButton                 matlab.ui.control.Button
        FeedbackButton                  matlab.ui.control.Button
        DocumentationButton             matlab.ui.control.Button
        EXITButton                      matlab.ui.control.Button
        CRCButton                       matlab.ui.control.Button
        BITSTUFFINGDESTUFFINGButton     matlab.ui.control.Button
        CHARACTERSTUFFINGDESTUFFINGButton  matlab.ui.control.Button
        HAMMINGENCODINGDECODINGButton   matlab.ui.control.Button
        IPv4CONVERSIONButton_2          matlab.ui.control.Button
        ANALOGTODIGITALSIGNALCONVERSIONButton  matlab.ui.control.Button
        LINECODINGButton                matlab.ui.control.Button
        Label                           matlab.ui.control.Label
        DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label  matlab.
            ↪ ui.control.Label
        Image2                          matlab.ui.control.Image
        Image                           matlab.ui.control.Image
        DATATRANSMISSIONSIMULATORLabel  matlab.ui.control.Label
        Image3                          matlab.ui.control.Image
    end

    % Callbacks that handle component events
    methods (Access = private)

        % Button pushed function: LINECODINGButton
        function LINECODINGButtonPushed(app, event)
            app.callingapp = app1;
        end

        % Button pushed function: IPv4CONVERSIONButton_2
        function IPv4CONVERSIONButton_2Pushed(app, event)
            app.callingapp = ipv4;
        end

        % Button pushed function: HAMMINGENCODINGDECODINGButton
        function HAMMINGENCODINGDECODINGButtonPushed(app, event)
            app.callingapp = hammingcode;
        end

        % Button pushed function: ANALOGTODIGITALSIGNALCONVERSIONButton
        function ANALOGTODIGITALSIGNALCONVERSIONButtonPushed(app, event)
             app.callingapp = ADC;
        end

        % Button pushed function: CHARACTERSTUFFINGDESTUFFINGButton
        function CHARACTERSTUFFINGDESTUFFINGButtonPushed(app, event)
            app.callingapp = CharacterStuffingDestuffing;
        end

        % Button pushed function: BITSTUFFINGDESTUFFINGButton
        function BITSTUFFINGDESTUFFINGButtonPushed(app, event)

        end

        % Button pushed function: CRCButton
        function CRCButtonPushed(app, event)
            app.callingapp = crcapp;
        end

        % Image clicked function: Image3
        function Image3Clicked(app, event)

        end

        % Button pushed function: EXITButton
        function EXITButtonPushed(app, event)
            % Close all figure windows
```

```
        figHandles = findall(0, 'Type', 'figure');
        delete(figHandles);
        end

        % Button pushed function: DocumentationButton
        function DocumentationButtonPushed(app, event)
            % Specify the URL you want to open
            url = 'https://drive.google.com/drive/folders/1BMGfCQGcDsBmbeMB_
                ↪ Ep5IJiPZcHgZz15?usp=sharing';

        % Open the link in the default web browser
            web(url, '-browser');
        end

        % Button pushed function: FeedbackButton
        function FeedbackButtonPushed(app, event)
           email = 'saifulislamrimon2014@gmail.com';

        % Create a "mailto" link
        mailtoLink = ['mailto:' email];

        % Open the link in the default email client
        web(mailtoLink, '-browser');
        end

        % Button pushed function: DeveloperButton
        function DeveloperButtonPushed(app, event)
                        % Specify the URL you want to open
            url = 'sites.google.com/view/mdsaifulislamrimon';

        % Open the link in the default web browser
            web(url, '-browser');
        end
    end

    % Component initialization
    methods (Access = private)

        % Create UIFigure and components
        function createComponents(app)

            % Get the file path for locating images
            pathToMLAPP = fileparts(mfilename('fullpath'));

            % Create UIFigure and hide until all components are created
            app.UIFigure = uifigure('Visible', 'off');
            app.UIFigure.Position = [100 100 640 480];
            app.UIFigure.Name = 'MATLAB App';

            % Create Image3
            app.Image3 = uiimage(app.UIFigure);
            app.Image3.ImageClickedFcn = createCallbackFcn(app, @Image3Clicked,
                ↪ true);
            app.Image3.Position = [-147 -17 937 672];
            app.Image3.ImageSource = fullfile(pathToMLAPP, 'bg.jpg');

            % Create DATATRANSMISSIONSIMULATORLabel
            app.DATATRANSMISSIONSIMULATORLabel = uilabel(app.UIFigure);
            app.DATATRANSMISSIONSIMULATORLabel.FontSize = 36;
            app.DATATRANSMISSIONSIMULATORLabel.FontWeight = 'bold';
            app.DATATRANSMISSIONSIMULATORLabel.Position = [17 353 618 47];
            app.DATATRANSMISSIONSIMULATORLabel.Text = 'DATA TRANSMISSION
                ↪ SIMULATOR';

            % Create Image
            app.Image = uiimage(app.UIFigure);
            app.Image.Position = [1 358 189 166];
            app.Image.ImageSource = fullfile(pathToMLAPP, 'Logo-PNG.png');

            % Create Image2
            app.Image2 = uiimage(app.UIFigure);
            app.Image2.Position = [422 339 219 199];
            app.Image2.ImageSource = fullfile(pathToMLAPP, 'cse-dept-logo.png');
```

```
141
142             % Create
                    ↪ DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label
143             app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label =
                    ↪  uilabel(app.UIFigure);
144             app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                    ↪ HorizontalAlignment = 'center';
145             app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                    ↪ FontWeight = 'bold';
146             app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                    ↪ Position = [60 3 521 22];
147             app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                    ↪ Text = 'DEVELOPED BY: MD SAIFUL ISLAM RIMON (213002039) &
                    ↪ NAZMUN NAHAR (191000000)';
148
149             % Create Label
150             app.Label = uilabel(app.UIFigure);
151             app.Label.HorizontalAlignment = 'center';
152             app.Label.FontWeight = 'bold';
153             app.Label.Position = [68 28 499 22];
154             app.Label.Text = 'PROJECT SUPERVISOR: RUSMITA HALIM CHAITY,  LECTURER
                    ↪ , DEPT. OF CSE, GUB';
155
156             % Create LINECODINGButton
157             app.LINECODINGButton = uibutton(app.UIFigure, 'push');
158             app.LINECODINGButton.ButtonPushedFcn = createCallbackFcn(app,
                    ↪ @LINECODINGButtonPushed, true);
159             app.LINECODINGButton.BackgroundColor = [0.4824 0.8 0.2863];
160             app.LINECODINGButton.FontSize = 18;
161             app.LINECODINGButton.FontWeight = 'bold';
162             app.LINECODINGButton.Position = [39 272 178 37];
163             app.LINECODINGButton.Text = 'LINE CODING';
164
165             % Create ANALOGTODIGITALSIGNALCONVERSIONButton
166             app.ANALOGTODIGITALSIGNALCONVERSIONButton = uibutton(app.UIFigure, '
                    ↪ push');
167             app.ANALOGTODIGITALSIGNALCONVERSIONButton.ButtonPushedFcn =
                    ↪ createCallbackFcn(app,
                    ↪ @ANALOGTODIGITALSIGNALCONVERSIONButtonPushed, true);
168             app.ANALOGTODIGITALSIGNALCONVERSIONButton.BackgroundColor = [0.4824
                    ↪ 0.8 0.2863];
169             app.ANALOGTODIGITALSIGNALCONVERSIONButton.FontSize = 14;
170             app.ANALOGTODIGITALSIGNALCONVERSIONButton.FontWeight = 'bold';
171             app.ANALOGTODIGITALSIGNALCONVERSIONButton.Position = [37 209 180 42];
172             app.ANALOGTODIGITALSIGNALCONVERSIONButton.Text = {'ANALOG TO DIGITAL'
                    ↪ ; 'SIGNAL CONVERSION'};
173
174             % Create IPv4CONVERSIONButton_2
175             app.IPv4CONVERSIONButton_2 = uibutton(app.UIFigure, 'push');
176             app.IPv4CONVERSIONButton_2.ButtonPushedFcn = createCallbackFcn(app,
                    ↪ @IPv4CONVERSIONButton_2Pushed, true);
177             app.IPv4CONVERSIONButton_2.BackgroundColor = [0.4784 0.8 0.2902];
178             app.IPv4CONVERSIONButton_2.FontSize = 18;
179             app.IPv4CONVERSIONButton_2.FontWeight = 'bold';
180             app.IPv4CONVERSIONButton_2.Position = [39 154 178 37];
181             app.IPv4CONVERSIONButton_2.Text = 'IPv4 CONVERSION';
182
183             % Create HAMMINGENCODINGDECODINGButton
184             app.HAMMINGENCODINGDECODINGButton = uibutton(app.UIFigure, 'push');
185             app.HAMMINGENCODINGDECODINGButton.ButtonPushedFcn = createCallbackFcn
                    ↪ (app, @HAMMINGENCODINGDECODINGButtonPushed, true);
186             app.HAMMINGENCODINGDECODINGButton.BackgroundColor = [0.4824 0.8
                    ↪ 0.2863];
187             app.HAMMINGENCODINGDECODINGButton.FontSize = 14;
188             app.HAMMINGENCODINGDECODINGButton.FontWeight = 'bold';
189             app.HAMMINGENCODINGDECODINGButton.Position = [394 267 199 42];
190             app.HAMMINGENCODINGDECODINGButton.Text = {'HAMMING ENCODING &'; '
                    ↪ DECODING'};
191
192             % Create CHARACTERSTUFFINGDESTUFFINGButton
193             app.CHARACTERSTUFFINGDESTUFFINGButton = uibutton(app.UIFigure, 'push'
                    ↪ );
```

```
194             app.CHARACTERSTUFFINGDESTUFFINGButton.ButtonPushedFcn =
                    ↪ createCallbackFcn(app, @CHARACTERSTUFFINGDESTUFFINGButtonPushed
                    ↪ , true);
195             app.CHARACTERSTUFFINGDESTUFFINGButton.BackgroundColor = [0.4824 0.8
                    ↪ 0.2863];
196             app.CHARACTERSTUFFINGDESTUFFINGButton.FontSize = 14;
197             app.CHARACTERSTUFFINGDESTUFFINGButton.FontWeight = 'bold';
198             app.CHARACTERSTUFFINGDESTUFFINGButton.Position = [394 209 199 42];
199             app.CHARACTERSTUFFINGDESTUFFINGButton.Text = {'CHARACTER STUFFING &';
                    ↪  'DESTUFFING'};
200
201             % Create BITSTUFFINGDESTUFFINGButton
202             app.BITSTUFFINGDESTUFFINGButton = uibutton(app.UIFigure, 'push');
203             app.BITSTUFFINGDESTUFFINGButton.ButtonPushedFcn = createCallbackFcn(
                    ↪ app, @BITSTUFFINGDESTUFFINGButtonPushed, true);
204             app.BITSTUFFINGDESTUFFINGButton.BackgroundColor = [0.4784 0.8
                    ↪ 0.2902];
205             app.BITSTUFFINGDESTUFFINGButton.FontSize = 14;
206             app.BITSTUFFINGDESTUFFINGButton.FontWeight = 'bold';
207             app.BITSTUFFINGDESTUFFINGButton.Position = [698 133 199 42];
208             app.BITSTUFFINGDESTUFFINGButton.Text = {'BIT STUFFING &'; 'DESTUFFING
                    ↪ '};
209
210             % Create CRCButton
211             app.CRCButton = uibutton(app.UIFigure, 'push');
212             app.CRCButton.ButtonPushedFcn = createCallbackFcn(app,
                    ↪ @CRCButtonPushed, true);
213             app.CRCButton.BackgroundColor = [0.4824 0.8 0.2863];
214             app.CRCButton.FontSize = 14;
215             app.CRCButton.FontWeight = 'bold';
216             app.CRCButton.Position = [394 153 199 38];
217             app.CRCButton.Text = 'CRC';
218
219             % Create EXITButton
220             app.EXITButton = uibutton(app.UIFigure, 'push');
221             app.EXITButton.ButtonPushedFcn = createCallbackFcn(app,
                    ↪ @EXITButtonPushed, true);
222             app.EXITButton.BackgroundColor = [1 0 0];
223             app.EXITButton.FontSize = 14;
224             app.EXITButton.FontWeight = 'bold';
225             app.EXITButton.FontColor = [1 1 1];
226             app.EXITButton.Position = [504 79 89 38];
227             app.EXITButton.Text = 'EXIT';
228
229             % Create DocumentationButton
230             app.DocumentationButton = uibutton(app.UIFigure, 'push');
231             app.DocumentationButton.ButtonPushedFcn = createCallbackFcn(app,
                    ↪ @DocumentationButtonPushed, true);
232             app.DocumentationButton.BackgroundColor = [1 0.4118 0.1608];
233             app.DocumentationButton.FontSize = 14;
234             app.DocumentationButton.FontWeight = 'bold';
235             app.DocumentationButton.FontColor = [1 1 1];
236             app.DocumentationButton.Position = [37 79 117 38];
237             app.DocumentationButton.Text = 'Documentation';
238
239             % Create FeedbackButton
240             app.FeedbackButton = uibutton(app.UIFigure, 'push');
241             app.FeedbackButton.ButtonPushedFcn = createCallbackFcn(app,
                    ↪ @FeedbackButtonPushed, true);
242             app.FeedbackButton.BackgroundColor = [1 0.4118 0.1608];
243             app.FeedbackButton.FontSize = 14;
244             app.FeedbackButton.FontWeight = 'bold';
245             app.FeedbackButton.FontColor = [1 1 1];
246             app.FeedbackButton.Position = [180 79 117 38];
247             app.FeedbackButton.Text = 'Feedback';
248
249             % Create DeveloperButton
250             app.DeveloperButton = uibutton(app.UIFigure, 'push');
251             app.DeveloperButton.ButtonPushedFcn = createCallbackFcn(app,
                    ↪ @DeveloperButtonPushed, true);
252             app.DeveloperButton.BackgroundColor = [1 0.4118 0.1608];
253             app.DeveloperButton.FontSize = 14;
254             app.DeveloperButton.FontWeight = 'bold';
```

```
            app.DeveloperButton.FontColor = [1 1 1];
            app.DeveloperButton.Position = [328 79 117 38];
            app.DeveloperButton.Text = 'Developer';

            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
    end

    % App creation and deletion
    methods (Access = public)

        % Construct app
        function app = home

            % Create UIFigure and components
            createComponents(app)

            % Register the app with App Designer
            registerApp(app, app.UIFigure)

            if nargout == 0
                clear app
            end
        end

        % Code that executes before app deletion
        function delete(app)

            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
    end
end
```

CODE [Line Coding Page]:

```
classdef app1 < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                        matlab.ui.Figure
        GridLayout                      matlab.ui.container.GridLayout
        LeftPanel                       matlab.ui.container.Panel
        Image2                          matlab.ui.control.Image
        Image                           matlab.ui.control.Image
        LINECODINGSCHEMASLabel          matlab.ui.control.Label
        DifferentialManchesterButton    matlab.ui.control.Button
        InputEditField                  matlab.ui.control.EditField
        InputEditFieldLabel             matlab.ui.control.Label
        ManchesterButton                matlab.ui.control.Button
        PseodoTernaryButton             matlab.ui.control.Button
        AMIButton                       matlab.ui.control.Button
        NRZLButton                      matlab.ui.control.Button
        NRZIButton                      matlab.ui.control.Button
        RightPanel                      matlab.ui.container.Panel
        DecodedBitsEditField            matlab.ui.control.EditField
        DecodedBitsLabel                matlab.ui.control.Label
        UIAxes                          matlab.ui.control.UIAxes
        Label                           matlab.ui.control.Label
        DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label  matlab.
            ↪ ui.control.Label
    end

    % Properties that correspond to apps with auto-reflow
    properties (Access = private)
        onePanelWidth = 576;
    end

    % Callbacks that handle component events
    methods (Access = private)

        % Button pushed function: NRZIButton
        function NRZIButtonPushed(app, event)
bitStr = app.InputEditField.Value;
bits = str2num(bitStr)
%bits = [1 0 1 1 1 0 0 1];
bitrate = 1;
n = 1000;
T = length(bits)/bitrate;
N = n * length(bits);
dt = T/N;

t = 0:dt:T;
x = zeros(1, length(t));
lastbit = 1;

for i = 1:length(bits)
    if bits(i) == 1
        x((i-1)*n+1:i*n) = -lastbit;
        lastbit = -lastbit;
    else
        x((i-1)*n+1:i*n) = lastbit;
    end
    plot(app.UIAxes, t, x, 'Linewidth', 3);
end

counter = 0;
lastbit = 1;
result = zeros(1, length(bits));

for i = 1:length(t)
    if t(i) > counter
        counter = counter + 1;
        if x(i) ~= lastbit
            result(counter) = 1;
            lastbit = -lastbit;
        else
```

```
            result(counter) = 0;
        end
    end
end

result_str = char('0' + result);
app.DecodedBitsEditField.Value = string(result_str);

        end

        % Button down function: UIAxes
        function UIAxesButtonDown(app, event)

        end

        % Button pushed function: NRZLButton
        function NRZLButtonPushed(app, event)
bitStr = app.InputEditField.Value;
bits = str2num(bitStr)
%bits = [1 0 1 1 1 0 0 1];
bitrate = 1;
n = 1000;
T = length(bits)/bitrate;
N = n*length(bits);
dt = T/N;
t = 0:dt:T;
x = zeros(1,length(t));
low = -1;
high = 1;

for i = 1:length(bits)
    if bits(i) == 1
        x((i-1)*n+1:i*n) = high;

    else
        x((i-1)*n+1:i*n) = low;
    end
end

plot(app.UIAxes, x, 'Linewidth', 3);

counter = 0;
high = 1;

result = zeros(1, length(bits));

for i = 1:length(t)
    if t(i) > counter
        counter = counter + 1;
        if x(i) ~= high
            result(counter) = 0;

        else
            result(counter) = 1;
        end
    end
end
result_str = char('0' + result);
app.DecodedBitsEditField.Value = string(result_str);

        end

        % Button pushed function: AMIButton
        function AMIButtonPushed(app, event)
bitStr = app.InputEditField.Value;
bits = str2num(bitStr)
%bits = [1 0 1 1 1 0 0 1];
bitrate = 1;
n = 1000;
T = length(bits)/bitrate;
N = n*length(bits);
dt = T/N;
t = 0:dt:T;
```

```
x = zeros(1,length(t));
lastbit = 1;

for i=1:length(bits)
    if bits(i)==1
        x((i-1)*n+1:i*n) = -lastbit;
        lastbit = -lastbit;
    end
end

plot(app.UIAxes,t, x, 'Linewidth', 3);

counter = 0;
lastbit = 1;

for i = 1:length(t)
    if t(i) > counter
        counter = counter + 1;
        if x(i) == -lastbit
            result(counter) = 1;
            lastbit = -lastbit;
        else
            result(counter) = 0;
        end
    end
end

result_str = char('0' + result);
app.DecodedBitsEditField.Value = string(result_str);

        end

        % Value changed function: DecodedBitsEditField
        function DecodedBitsEditFieldValueChanged(app, event)
            value = app.DecodedBitsEditField.Value;

        end

        % Callback function
        function NRZIDecodeButtonPushed(app, event)

        end

        % Value changed function: InputEditField
        function InputEditFieldValueChanged(app, event)
            value = app.InputEditField.Value;

        end

        % Callback function
        function ControlSliderValueChanged(app, event)
            value = app.ControlSlider.Value;
            % Get the current zoom level from the slider


            % Update the UIAxes zoom level
           % xlim(app.UIAxes, [0, 10] * app.ControlSlider);  % Adjust the limits
               ↪ accordingly
           % ylim(app.UIAxes, [0, 10] * app.ControlSlider);

            % Optionally, you can update other properties based on the zoom level
            % For example, update the axis labels or other visual elements

        end

        % Button pushed function: PseodoTernaryButton
        function PseodoTernaryButtonPushed(app, event)
            bitStr = app.InputEditField.Value;
bits = str2num(bitStr)
%bits = [1 0 1 1 1 0 0 1];
bitrate = 1;
n = 1000;
T = length(bits)/bitrate;
```

```
N = n*length(bits);
dt = T/N;
t = 0:dt:T;
x = zeros(1,length(t));
lastbit = 1;

for i=1:length(bits)
    if bits(i)== 0
        x((i-1)*n+1:i*n) = -lastbit;
        lastbit = -lastbit;
    end
end

plot(app.UIAxes,t, x, 'Linewidth', 3);

counter = 0;
lastbit = 0;

for i = 1:length(t)
    if t(i) > counter
        counter = counter + 1;
        if x(i) == -lastbit
            result(counter) = 1;
            lastbit = -lastbit;
        else
            result(counter) = 0;
        end
    end
end

result_str = char('0' + result);
app.DecodedBitsEditField.Value = string(result_str);

        end

        % Button pushed function: ManchesterButton
        function ManchesterButtonPushed(app, event)
bitStr = app.InputEditField.Value;
bits = str2num(bitStr)
bitrate = 1;
n = 1000;
T = length(bits)/bitrate;
N = n*length(bits);
dt = T/N;
t = 0:dt:T;
x = zeros(1,length(t));

for i = 1:length(bits)
    if bits(i)== 1
        x((i-1)*n+1:(i-1)*n+n/2) = 1;
        x((i-1)*n+n/2:i*n) = -1;
    else
        x((i-1)*n+1:(i-1)*n+n/2) = -1;
        x((i-1)*n+n/2:i*n) = 1;
    end
end

plot(app.UIAxes,t, x, 'Linewidth', 3);


counter = 0;
for i = 1:length(t)
    if t(i) > counter
        counter = counter + 1;

        if x(i) > 0
            result(counter) = x(i);
        else
            result(counter) = 0;
        end
    end
end

```

```
result_str = char('0' + result);
app.DecodedBitsEditField.Value = string(result_str);

        end

        % Callback function
        function IPV4ConversionButtonPushed(app, event)

        end

        % Callback function
        function HammingCodeButtonPushed(app, event)

        end

        % Callback function
        function AnologToDigitalSignalButtonPushed(app, event)

        end

        % Button pushed function: DifferentialManchesterButton
        function DifferentialManchesterButtonPushed(app, event)
bitStr = app.InputEditField.Value;
bits = str2num(bitStr)
bitrate = 1;
n = 1000;
T = length(bits)/bitrate;
N = n*length(bits);
dt = T/N;
t = 0:dt:T;
x = zeros(1,length(t));

lastbit = 1;
for i=1:length(bits)
if bits(i)==0
x((i-1)*n+1:(i-1)*n+n/2) = -lastbit;
x((i-1)*n+n/2:i*n) = lastbit;
else
x((i-1)*n+1:(i-1)*n+n/2) = lastbit;
x((i-1)*n+n/2:i*n) = -lastbit;
lastbit = -lastbit;
end
end

plot(app.UIAxes,t, x, 'Linewidth', 3);


count = 0;
lastbit = 1;
for i = 1:length(t)
if t(i)>count
count = count + 1;
if x(i)==lastbit
result(count) = 1;
lastbit = -lastbit;
else result(count) = 0;
end
end
end

result_str = char('0' + result);
app.DecodedBitsEditField.Value = string(result_str);

        end

        % Changes arrangement of the app based on UIFigure width
        function updateAppLayout(app, event)
            currentFigureWidth = app.UIFigure.Position(3);
            if(currentFigureWidth <= app.onePanelWidth)
                % Change to a 2x1 grid
                app.GridLayout.RowHeight = {480, 480};
                app.GridLayout.ColumnWidth = {'1x'};
                app.RightPanel.Layout.Row = 2;
```

```
                app.RightPanel.Layout.Column = 1;
            else
                % Change to a 1x2 grid
                app.GridLayout.RowHeight = {'1x'};
                app.GridLayout.ColumnWidth = {220, '1x'};
                app.RightPanel.Layout.Row = 1;
                app.RightPanel.Layout.Column = 2;
            end
        end
    end

    % Component initialization
    methods (Access = private)

        % Create UIFigure and components
        function createComponents(app)

            % Create UIFigure and hide until all components are created
            app.UIFigure = uifigure('Visible', 'off');
            app.UIFigure.AutoResizeChildren = 'off';
            app.UIFigure.Position = [100 100 640 480];
            app.UIFigure.Name = 'MATLAB App';
            app.UIFigure.SizeChangedFcn = createCallbackFcn(app, @updateAppLayout
                ↪ , true);

            % Create GridLayout
            app.GridLayout = uigridlayout(app.UIFigure);
            app.GridLayout.ColumnWidth = {220, '1x'};
            app.GridLayout.RowHeight = {'1x'};
            app.GridLayout.ColumnSpacing = 0;
            app.GridLayout.RowSpacing = 0;
            app.GridLayout.Padding = [0 0 0 0];
            app.GridLayout.Scrollable = 'on';

            % Create LeftPanel
            app.LeftPanel = uipanel(app.GridLayout);
            app.LeftPanel.Layout.Row = 1;
            app.LeftPanel.Layout.Column = 1;

            % Create NRZIButton
            app.NRZIButton = uibutton(app.LeftPanel, 'push');
            app.NRZIButton.ButtonPushedFcn = createCallbackFcn(app,
                ↪ @NRZIButtonPushed, true);
            app.NRZIButton.IconAlignment = 'center';
            app.NRZIButton.BackgroundColor = [0.4784 0.8 0.2902];
            app.NRZIButton.FontSize = 14;
            app.NRZIButton.FontWeight = 'bold';
            app.NRZIButton.Position = [60 311 100 23];
            app.NRZIButton.Text = 'NRZ -I';

            % Create NRZLButton
            app.NRZLButton = uibutton(app.LeftPanel, 'push');
            app.NRZLButton.ButtonPushedFcn = createCallbackFcn(app,
                ↪ @NRZLButtonPushed, true);
            app.NRZLButton.BackgroundColor = [0.4784 0.8 0.2902];
            app.NRZLButton.FontSize = 14;
            app.NRZLButton.FontWeight = 'bold';
            app.NRZLButton.Position = [60 270 100 25];
            app.NRZLButton.Text = 'NRZ-L';

            % Create AMIButton
            app.AMIButton = uibutton(app.LeftPanel, 'push');
            app.AMIButton.ButtonPushedFcn = createCallbackFcn(app,
                ↪ @AMIButtonPushed, true);
            app.AMIButton.BackgroundColor = [0.4784 0.8 0.2902];
            app.AMIButton.FontSize = 14;
            app.AMIButton.FontWeight = 'bold';
            app.AMIButton.Position = [60 234 100 25];
            app.AMIButton.Text = 'AMI';

            % Create PseodoTernaryButton
            app.PseodoTernaryButton = uibutton(app.LeftPanel, 'push');
```

```
            app.PseodoTernaryButton.ButtonPushedFcn = createCallbackFcn(app,
                ↪ @PseodoTernaryButtonPushed, true);
            app.PseodoTernaryButton.BackgroundColor = [0.4784 0.8 0.2902];
            app.PseodoTernaryButton.FontSize = 14;
            app.PseodoTernaryButton.FontWeight = 'bold';
            app.PseodoTernaryButton.Position = [53 193 116 25];
            app.PseodoTernaryButton.Text = 'PseodoTernary';

            % Create ManchesterButton
            app.ManchesterButton = uibutton(app.LeftPanel, 'push');
            app.ManchesterButton.ButtonPushedFcn = createCallbackFcn(app,
                ↪ @ManchesterButtonPushed, true);
            app.ManchesterButton.BackgroundColor = [0.4784 0.8 0.2902];
            app.ManchesterButton.FontSize = 14;
            app.ManchesterButton.FontWeight = 'bold';
            app.ManchesterButton.Position = [60 150 100 25];
            app.ManchesterButton.Text = 'Manchester';

            % Create InputEditFieldLabel
            app.InputEditFieldLabel = uilabel(app.LeftPanel);
            app.InputEditFieldLabel.HorizontalAlignment = 'right';
            app.InputEditFieldLabel.Position = [24 34 32 22];
            app.InputEditFieldLabel.Text = 'Input';

            % Create InputEditField
            app.InputEditField = uieditfield(app.LeftPanel, 'text');
            app.InputEditField.ValueChangedFcn = createCallbackFcn(app,
                ↪ @InputEditFieldValueChanged, true);
            app.InputEditField.Placeholder = '1 0 1 1 1 0 0 1';
            app.InputEditField.Position = [71 34 123 22];

            % Create DifferentialManchesterButton
            app.DifferentialManchesterButton = uibutton(app.LeftPanel, 'push');
            app.DifferentialManchesterButton.ButtonPushedFcn = createCallbackFcn(
                ↪ app, @DifferentialManchesterButtonPushed, true);
            app.DifferentialManchesterButton.BackgroundColor = [0.4784 0.8
                ↪ 0.2902];
            app.DifferentialManchesterButton.FontSize = 14;
            app.DifferentialManchesterButton.FontWeight = 'bold';
            app.DifferentialManchesterButton.Position = [60 90 100 42];
            app.DifferentialManchesterButton.Text = {'Differential'; 'Manchester'
                ↪ };

            % Create LINECODINGSCHEMASLabel
            app.LINECODINGSCHEMASLabel = uilabel(app.LeftPanel);
            app.LINECODINGSCHEMASLabel.FontSize = 14;
            app.LINECODINGSCHEMASLabel.FontWeight = 'bold';
            app.LINECODINGSCHEMASLabel.Position = [25 352 171 22];
            app.LINECODINGSCHEMASLabel.Text = 'LINE CODING SCHEMAS';

            % Create Image
            app.Image = uiimage(app.LeftPanel);
            app.Image.Position = [109 373 100 100];
            app.Image.ImageSource = 'Logo-PNG-1.png';

            % Create Image2
            app.Image2 = uiimage(app.LeftPanel);
            app.Image2.Position = [37 386 66 74];
            app.Image2.ImageSource = 'images__1_-removebg-preview.png';

            % Create RightPanel
            app.RightPanel = uipanel(app.GridLayout);
            app.RightPanel.Layout.Row = 1;
            app.RightPanel.Layout.Column = 2;

            % Create UIAxes
            app.UIAxes = uiaxes(app.RightPanel);
            title(app.UIAxes, 'OUTPUT FIGURE')
            xlabel(app.UIAxes, 'X')
            ylabel(app.UIAxes, 'Y')
            zlabel(app.UIAxes, 'Z')
            app.UIAxes.ButtonDownFcn = createCallbackFcn(app, @UIAxesButtonDown,
                ↪ true);
```

```
            app.UIAxes.Position = [16 74 389 236];

            % Create DecodedBitsLabel
            app.DecodedBitsLabel = uilabel(app.RightPanel);
            app.DecodedBitsLabel.HorizontalAlignment = 'right';
            app.DecodedBitsLabel.FontSize = 18;
            app.DecodedBitsLabel.Position = [13 228 107 23];
            app.DecodedBitsLabel.Text = 'DecodedBits';

            % Create DecodedBitsEditField
            app.DecodedBitsEditField = uieditfield(app.RightPanel, 'text');
            app.DecodedBitsEditField.ValueChangedFcn = createCallbackFcn(app,
                ↪ @DecodedBitsEditFieldValueChanged, true);
            app.DecodedBitsEditField.Editable = 'off';
            app.DecodedBitsEditField.HorizontalAlignment = 'center';
            app.DecodedBitsEditField.FontSize = 18;
            app.DecodedBitsEditField.Placeholder = 'Decoded Bits will be shown
                ↪ here';
            app.DecodedBitsEditField.Position = [135 224 238 27];

            % Create
                ↪ DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label =
                ↪  uilabel(app.UIFigure);
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ HorizontalAlignment = 'center';
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ FontSize = 8;
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ FontWeight = 'bold';
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ Position = [243 3 348 22];
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ Text = 'DEVELOPED BY: MD SAIFUL ISLAM RIMON (213002039) &
                ↪ NAZMUN NAHAR (191000000)';

            % Create Label
            app.Label = uilabel(app.UIFigure);
            app.Label.HorizontalAlignment = 'center';
            app.Label.FontSize = 8;
            app.Label.FontWeight = 'bold';
            app.Label.Position = [248 28 333 22];
            app.Label.Text = 'PROJECT SUPERVISOR: RUSMITA HALIM CHAITY,  LECTURER
                ↪ , DEPT. OF CSE, GUB';

            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
    end

    % App creation and deletion
    methods (Access = public)

        % Construct app
        function app = app1

            % Create UIFigure and components
            createComponents(app)

            % Register the app with App Designer
            registerApp(app, app.UIFigure)

            if nargout == 0
                clear app
            end
        end

        % Code that executes before app deletion
        function delete(app)

            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
```

```
558     end
559 end
```

CODE [Hamming Encoding & Decoding Page]:

```
classdef hammingcode < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                        matlab.ui.Figure
        Label                           matlab.ui.control.Label
        DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label  matlab.
            ↪ ui.control.Label
        Image3                          matlab.ui.control.Image
        Image2                          matlab.ui.control.Image
        NUMBEROFPARITYBITSEditField     matlab.ui.control.EditField
        NUMBEROFPARITYBITSEditFieldLabel  matlab.ui.control.Label
        LENGTHOFHAMMINGCODEEditField    matlab.ui.control.EditField
        LENGTHOFHAMMINGCODEEditFieldLabel  matlab.ui.control.Label
        CORRECTEDBITSTRINGEditField     matlab.ui.control.EditField
        CORRECTEDBITSTRINGEditFieldLabel  matlab.ui.control.Label
        POSITIONOFERROREditField        matlab.ui.control.EditField
        POSITIONOFERROREditFieldLabel   matlab.ui.control.Label
        P4EditField_2                   matlab.ui.control.EditField
        P4EditField_2Label              matlab.ui.control.Label
        P2EditField_4                   matlab.ui.control.EditField
        P2EditField_4Label              matlab.ui.control.Label
        P1EditField_4                   matlab.ui.control.EditField
        P1EditField_4Label              matlab.ui.control.Label
        ERRORDETECTIONANDCORRECTIONButton  matlab.ui.control.Button
        ERRORDETECTIONLabel             matlab.ui.control.Label
        RECEIVEDBITSTRINGEditField      matlab.ui.control.EditField
        RECEIVEDBITSTRINGEditFieldLabel  matlab.ui.control.Label
        SENDEDBITSTRINGEditField        matlab.ui.control.EditField
        SENDEDBITSTRINGEditFieldLabel   matlab.ui.control.Label
        P4EditField                     matlab.ui.control.EditField
        P4EditFieldLabel                matlab.ui.control.Label
        P2EditField_3                   matlab.ui.control.EditField
        P2EditField_3Label              matlab.ui.control.Label
        P1EditField_3                   matlab.ui.control.EditField
        P1EditField_3Label              matlab.ui.control.Label
        GENERATESENDINGBITSTRINGButton  matlab.ui.control.Button
        BITHAMMINGCODEPARITYBITSANDERRORDETECTIONLabel  matlab.ui.control.Label
        BITSTRINGEditField              matlab.ui.control.EditField
        Image                           matlab.ui.control.Image
        BITSTRINGEditFieldLabel         matlab.ui.control.Label
    end


    properties (Access = public)
        n
        nh % Description
        displayedValue
    end


    % Callbacks that handle component events
    methods (Access = private)

        % Value changed function: P1EditField_3
        function P1EditField_3ValueChanged(app, event)
            value = app.P1EditField_3.Value;

        end

        % Value changed function: P2EditField_3
        function P2EditField_3ValueChanged(app, event)
            value = app.P2EditField_3.Value;

        end

        % Value changed function: P4EditField
        function P4EditFieldValueChanged(app, event)
            value = app.P4EditField.Value;

        end
```

```
71
72          % Value changed function: SENDEDBITSTRINGEditField
73          function SENDEDBITSTRINGEditFieldValueChanged(app, event)
74              value = app.SENDEDBITSTRINGEditField.Value;
75
76          end
77
78          % Value changed function: P1EditField_4
79          function P1EditField_4ValueChanged(app, event)
80              value = app.P1EditField_4.Value;
81
82          end
83
84          % Value changed function: P2EditField_4
85          function P2EditField_4ValueChanged(app, event)
86              value = app.P2EditField_4.Value;
87
88          end
89
90          % Value changed function: P4EditField_2
91          function P4EditField_2ValueChanged(app, event)
92              value = app.P4EditField_2.Value;
93
94          end
95
96          % Value changed function: POSITIONOFERROREditField
97          function POSITIONOFERROREditFieldValueChanged(app, event)
98              value = app.POSITIONOFERROREditField.Value;
99
100         end
101
102         % Value changed function: CORRECTEDBITSTRINGEditField
103         function CORRECTEDBITSTRINGEditFieldValueChanged(app, event)
104             value = app.CORRECTEDBITSTRINGEditField.Value;
105
106         end
107
108         % Button pushed function: GENERATESENDINGBITSTRINGButton
109         function GENERATESENDINGBITSTRINGButtonPushed(app, event)
110             % Convert the entire code to Matlab code
111
112 maxp = 5;
113 a = zeros(1, 50);
114 temp = zeros(1, 70);
115 temp2 = zeros(1, 70);
116 t = 0;
117 i = 0;
118 j = 0;
119 k = 0;
120 nd = 0;
121 n = 0;
122 app.n = n;
123 nh = 0;
124 app.nh = nh;
125 sum = 0;
126 pos = 0;
127
128 nd = 4;
129 a = app.BITSTRINGEditField.Value;
130 a = str2num(a);
131
132 for i = 1:nd
133     for k = 0:maxp-1
134         t = 2^k - 1;
135         if j == t
136             temp(j+1) = 0;
137             j = j + 1;
138         end
139     end
140     temp(j+1) = a(i);
141     j = j + 1;
142 end
143 app.nh = j;
```

```
app.LENGTHOFHAMMINGCODEEditField.Value = string(app.nh);

app.n = app.nh - nd;
app.NUMBEROFPARITYBITSEditField.Value = string(app.n);


b = zeros(1, app.n);
m = app.n - 1;
for k = 0:app.n-1
    t = 2^k - 1;
    i = t;
    while i < app.nh
        for j = 0:t
            sum = sum + temp(i+1);
            i = i + 1;
            if i >= app.nh
                break;
            end
        end
        if i >= app.nh
            break;
        end
        for j = 0:t
            i = i + 1;
            if i >= app.nh
                break;
            end
        end
        if i >= app.nh
            break;
        end
    end
    temp(t+1) = mod(sum, 2);
    sum = 0;

    if(t+1 == 1)
        app.P1EditField_3.Value = string(temp(t+1));
    end

    if(t+1 == 2)
        app.P2EditField_3.Value = string(temp(t+1));
    end

    if(t+1 == 4)
        app.P4EditField.Value = string(temp(t+1));
    end
end

%fprintf('\nHamming code: Sender side:   ');
for i = 1:app.nh
    app.SENDEDBITSTRINGEditField.Value = app.SENDEDBITSTRINGEditField.Value +
        ↪ string(temp(i)) + " ";
    drawnow;
end
app.displayedValue = app.SENDEDBITSTRINGEditField.Value;









        end

        % Button pushed function: ERRORDETECTIONANDCORRECTIONButton
        function ERRORDETECTIONANDCORRECTIONButtonPushed(app, event)

            %fprintf('\nHamming code: Receiver side: ');
a = app.RECEIVEDBITSTRINGEditField.Value;
tempCellArray = strsplit(a, ' ');
temp2 = str2double(tempCellArray);

sum = 0;
n = app.n;
nh = app.nh;
m = n;
```

```
    b = zeros(1, n);
    pos = 0;

    for k = 0:n-1
        t = 2^k - 1;

        for i = t+1:nh %t+1
            for j = 1:t+1
                sum = sum + temp2(i);
                i = i + 1;
                if i >= nh
                    break;
                end
            end

            if i >= nh
                break;
            end

            for j = 1:t+1
                i = i + 1;
                if i >= nh
                    break;
                end
            end

            if i >= nh
                break;
            end
        end
        b(m) = mod(sum, 2);
        sum = 0;
        %fprintf('P%d: %d\n', t+1, b(m));
        if(t+1 == 1)
        app.P1EditField_4.Value = string(b(m));
        end

        if(t+1 == 2)
            app.P2EditField_4.Value = string(b(m));
        end

        if(t+1 == 4)
            app.P4EditField_2.Value = string(b(m));
        end
        m = m - 1;
    end

    for m = 1:n
        pos = pos + b(n-m+1) * 2^(m-1);
    end
    %fprintf('Position of Error: %d\n', pos);

    app.POSITIONOFERRROREditField.Value = string(pos);

    if temp2(pos) == 0
        temp2(pos) = 1;
    else
        temp2(pos) = 0;
    end

    %fprintf('\nHamming code: Receiver side: Error Corrected:  ');

    % for i = 1:nh
    %     fprintf('%d ', temp2(i));
    % end

    % for i = 1:nh
    % app.CORRECTEDBITSTRINGEditField.Value = app.CORRECTEDBITSTRINGEditField.
        ↪ Value + string(temp2(i)) + " ";
    % drawnow;
    % end

    app.CORRECTEDBITSTRINGEditField.Value = app.displayedValue;
```

```
288
289
290     %fprintf('\n----------------------------------\n');
291         end
292
293         % Value changed function: BITSTRINGEditField
294         function BITSTRINGEditFieldValueChanged(app, event)
295             value = app.BITSTRINGEditField.Value;
296
297         end
298
299         % Value changed function: LENGTHOFHAMMINGCODEEditField
300         function LENGTHOFHAMMINGCODEEditFieldValueChanged(app, event)
301             value = app.LENGTHOFHAMMINGCODEEditField.Value;
302
303         end
304
305         % Value changed function: NUMBEROFPARITYBITSEditField
306         function NUMBEROFPARITYBITSEditFieldValueChanged(app, event)
307             value = app.NUMBEROFPARITYBITSEditField.Value;
308
309         end
310
311         % Value changed function: RECEIVEDBITSTRINGEditField
312         function RECEIVEDBITSTRINGEditFieldValueChanged(app, event)
313             value = app.RECEIVEDBITSTRINGEditField.Value;
314
315         end
316     end
317
318     % Component initialization
319     methods (Access = private)
320
321         % Create UIFigure and components
322         function createComponents(app)
323
324             % Create UIFigure and hide until all components are created
325             app.UIFigure = uifigure('Visible', 'off');
326             app.UIFigure.Position = [100 100 640 480];
327             app.UIFigure.Name = 'MATLAB App';
328
329             % Create BITSTRINGEditFieldLabel
330             app.BITSTRINGEditFieldLabel = uilabel(app.UIFigure);
331             app.BITSTRINGEditFieldLabel.HorizontalAlignment = 'right';
332             app.BITSTRINGEditFieldLabel.Position = [33 373 72 22];
333             app.BITSTRINGEditFieldLabel.Text = 'BIT STRING';
334
335             % Create Image
336             app.Image = uiimage(app.UIFigure);
337             app.Image.Position = [-171 -102 979 647];
338             app.Image.ImageSource = 'bg.jpg';
339
340             % Create BITSTRINGEditField
341             app.BITSTRINGEditField = uieditfield(app.UIFigure, 'text');
342             app.BITSTRINGEditField.ValueChangedFcn = createCallbackFcn(app,
                    ↪ @BITSTRINGEditFieldValueChanged, true);
343             app.BITSTRINGEditField.Placeholder = '1 0 1 0';
344             app.BITSTRINGEditField.Position = [120 373 229 22];
345
346             % Create BITHAMMINGCODEPARITYBITSANDERRORDETECTIONLabel
347             app.BITHAMMINGCODEPARITYBITSANDERRORDETECTIONLabel = uilabel(app.
                    ↪ UIFigure);
348             app.BITHAMMINGCODEPARITYBITSANDERRORDETECTIONLabel.
                    ↪ HorizontalAlignment = 'center';
349             app.BITHAMMINGCODEPARITYBITSANDERRORDETECTIONLabel.WordWrap = 'on';
350             app.BITHAMMINGCODEPARITYBITSANDERRORDETECTIONLabel.FontSize = 14;
351             app.BITHAMMINGCODEPARITYBITSANDERRORDETECTIONLabel.FontWeight = 'bold
                    ↪ ';
352             app.BITHAMMINGCODEPARITYBITSANDERRORDETECTIONLabel.Position = [140
                    ↪ 421 362 34];
353             app.BITHAMMINGCODEPARITYBITSANDERRORDETECTIONLabel.Text = '4 BIT
                    ↪ HAMMING CODE PARITY BITS AND ERROR DETECTION';
354
```

```
            % Create GENERATESENDINGBITSTRINGButton
            app.GENERATESENDINGBITSTRINGButton = uibutton(app.UIFigure, 'push');
            app.GENERATESENDINGBITSTRINGButton.ButtonPushedFcn =
                ↪ createCallbackFcn(app, @GENERATESENDINGBITSTRINGButtonPushed,
                ↪ true);
            app.GENERATESENDINGBITSTRINGButton.BackgroundColor = [0.4 0.8
                ↪ 0.1216];
            app.GENERATESENDINGBITSTRINGButton.Position = [370 373 245 23];
            app.GENERATESENDINGBITSTRINGButton.Text = 'GENERATE SENDING BIT
                ↪ STRING';

            % Create P1EditField_3Label
            app.P1EditField_3Label = uilabel(app.UIFigure);
            app.P1EditField_3Label.HorizontalAlignment = 'right';
            app.P1EditField_3Label.Position = [63 310 25 22];
            app.P1EditField_3Label.Text = 'P1';

            % Create P1EditField_3
            app.P1EditField_3 = uieditfield(app.UIFigure, 'text');
            app.P1EditField_3.ValueChangedFcn = createCallbackFcn(app,
                ↪ @P1EditField_3ValueChanged, true);
            app.P1EditField_3.Editable = 'off';
            app.P1EditField_3.Position = [103 310 38 22];

            % Create P2EditField_3Label
            app.P2EditField_3Label = uilabel(app.UIFigure);
            app.P2EditField_3Label.HorizontalAlignment = 'right';
            app.P2EditField_3Label.Position = [63 268 25 22];
            app.P2EditField_3Label.Text = 'P2';

            % Create P2EditField_3
            app.P2EditField_3 = uieditfield(app.UIFigure, 'text');
            app.P2EditField_3.ValueChangedFcn = createCallbackFcn(app,
                ↪ @P2EditField_3ValueChanged, true);
            app.P2EditField_3.Editable = 'off';
            app.P2EditField_3.Position = [103 268 38 22];

            % Create P4EditFieldLabel
            app.P4EditFieldLabel = uilabel(app.UIFigure);
            app.P4EditFieldLabel.HorizontalAlignment = 'right';
            app.P4EditFieldLabel.Position = [63 230 25 22];
            app.P4EditFieldLabel.Text = 'P4';

            % Create P4EditField
            app.P4EditField = uieditfield(app.UIFigure, 'text');
            app.P4EditField.ValueChangedFcn = createCallbackFcn(app,
                ↪ @P4EditFieldValueChanged, true);
            app.P4EditField.Editable = 'off';
            app.P4EditField.Position = [103 230 38 22];

            % Create SENDEDBITSTRINGEditFieldLabel
            app.SENDEDBITSTRINGEditFieldLabel = uilabel(app.UIFigure);
            app.SENDEDBITSTRINGEditFieldLabel.HorizontalAlignment = 'right';
            app.SENDEDBITSTRINGEditFieldLabel.Position = [253 241 129 22];
            app.SENDEDBITSTRINGEditFieldLabel.Text = 'SENDED BIT STRING';

            % Create SENDEDBITSTRINGEditField
            app.SENDEDBITSTRINGEditField = uieditfield(app.UIFigure, 'text');
            app.SENDEDBITSTRINGEditField.ValueChangedFcn = createCallbackFcn(app,
                ↪  @SENDEDBITSTRINGEditFieldValueChanged, true);
            app.SENDEDBITSTRINGEditField.Editable = 'off';
            app.SENDEDBITSTRINGEditField.Position = [389 241 172 22];

            % Create RECEIVEDBITSTRINGEditFieldLabel
            app.RECEIVEDBITSTRINGEditFieldLabel = uilabel(app.UIFigure);
            app.RECEIVEDBITSTRINGEditFieldLabel.HorizontalAlignment = 'right';
            app.RECEIVEDBITSTRINGEditFieldLabel.Position = [33 153 137 22];
            app.RECEIVEDBITSTRINGEditFieldLabel.Text = 'RECEIVED BIT STRING';

            % Create RECEIVEDBITSTRINGEditField
            app.RECEIVEDBITSTRINGEditField = uieditfield(app.UIFigure, 'text');
            app.RECEIVEDBITSTRINGEditField.ValueChangedFcn = createCallbackFcn(
                ↪ app, @RECEIVEDBITSTRINGEditFieldValueChanged, true);
```

```
            app.RECEIVEDBITSTRINGEditField.Placeholder = '1 0 0 1 0 1 1';
            app.RECEIVEDBITSTRINGEditField.Position = [177 153 172 22];

            % Create ERRORDETECTIONLabel
            app.ERRORDETECTIONLabel = uilabel(app.UIFigure);
            app.ERRORDETECTIONLabel.HorizontalAlignment = 'center';
            app.ERRORDETECTIONLabel.FontSize = 14;
            app.ERRORDETECTIONLabel.FontWeight = 'bold';
            app.ERRORDETECTIONLabel.Position = [140 193 362 22];
            app.ERRORDETECTIONLabel.Text = 'ERROR DETECTION';

            % Create ERRORDETECTIONANDCORRECTIONButton
            app.ERRORDETECTIONANDCORRECTIONButton = uibutton(app.UIFigure, 'push'
                ↪ );
            app.ERRORDETECTIONANDCORRECTIONButton.ButtonPushedFcn =
                ↪ createCallbackFcn(app, @ERRORDETECTIONANDCORRECTIONButtonPushed
                ↪ , true);
            app.ERRORDETECTIONANDCORRECTIONButton.BackgroundColor = [0.4 0.8
                ↪ 0.1216];
            app.ERRORDETECTIONANDCORRECTIONButton.Position = [370 153 245 23];
            app.ERRORDETECTIONANDCORRECTIONButton.Text = 'ERROR DETECTION AND
                ↪ CORRECTION';

            % Create P1EditField_4Label
            app.P1EditField_4Label = uilabel(app.UIFigure);
            app.P1EditField_4Label.HorizontalAlignment = 'right';
            app.P1EditField_4Label.Position = [63 123 25 22];
            app.P1EditField_4Label.Text = 'P1';

            % Create P1EditField_4
            app.P1EditField_4 = uieditfield(app.UIFigure, 'text');
            app.P1EditField_4.ValueChangedFcn = createCallbackFcn(app,
                ↪ @P1EditField_4ValueChanged, true);
            app.P1EditField_4.Editable = 'off';
            app.P1EditField_4.Position = [103 123 38 22];

            % Create P2EditField_4Label
            app.P2EditField_4Label = uilabel(app.UIFigure);
            app.P2EditField_4Label.HorizontalAlignment = 'right';
            app.P2EditField_4Label.Position = [63 90 25 22];
            app.P2EditField_4Label.Text = 'P2';

            % Create P2EditField_4
            app.P2EditField_4 = uieditfield(app.UIFigure, 'text');
            app.P2EditField_4.ValueChangedFcn = createCallbackFcn(app,
                ↪ @P2EditField_4ValueChanged, true);
            app.P2EditField_4.Editable = 'off';
            app.P2EditField_4.Position = [103 90 38 22];

            % Create P4EditField_2Label
            app.P4EditField_2Label = uilabel(app.UIFigure);
            app.P4EditField_2Label.HorizontalAlignment = 'right';
            app.P4EditField_2Label.Position = [63 58 25 22];
            app.P4EditField_2Label.Text = 'P4';

            % Create P4EditField_2
            app.P4EditField_2 = uieditfield(app.UIFigure, 'text');
            app.P4EditField_2.ValueChangedFcn = createCallbackFcn(app,
                ↪ @P4EditField_2ValueChanged, true);
            app.P4EditField_2.Editable = 'off';
            app.P4EditField_2.Position = [103 58 38 22];

            % Create POSITIONOFERROREditFieldLabel
            app.POSITIONOFERROREditFieldLabel = uilabel(app.UIFigure);
            app.POSITIONOFERROREditFieldLabel.Position = [277 107 159 22];
            app.POSITIONOFERROREditFieldLabel.Text = 'POSITION OF ERROR';

            % Create POSITIONOFERROREditField
            app.POSITIONOFERROREditField = uieditfield(app.UIFigure, 'text');
            app.POSITIONOFERROREditField.ValueChangedFcn = createCallbackFcn(app,
                ↪  @POSITIONOFERROREditFieldValueChanged, true);
            app.POSITIONOFERROREditField.Editable = 'off';
            app.POSITIONOFERROREditField.Position = [415 107 170 22];
```

```
483
484             % Create CORRECTEDBITSTRINGEditFieldLabel
485             app.CORRECTEDBITSTRINGEditFieldLabel = uilabel(app.UIFigure);
486             app.CORRECTEDBITSTRINGEditFieldLabel.HorizontalAlignment = 'right';
487             app.CORRECTEDBITSTRINGEditFieldLabel.Position = [255 58 151 22];
488             app.CORRECTEDBITSTRINGEditFieldLabel.Text = 'CORRECTED BIT STRING';
489
490             % Create CORRECTEDBITSTRINGEditField
491             app.CORRECTEDBITSTRINGEditField = uieditfield(app.UIFigure, 'text');
492             app.CORRECTEDBITSTRINGEditField.ValueChangedFcn = createCallbackFcn(
                    ↪ app, @CORRECTEDBITSTRINGEditFieldValueChanged, true);
493             app.CORRECTEDBITSTRINGEditField.Editable = 'off';
494             app.CORRECTEDBITSTRINGEditField.Position = [413 58 172 22];
495
496             % Create LENGTHOFHAMMINGCODEEditFieldLabel
497             app.LENGTHOFHAMMINGCODEEditFieldLabel = uilabel(app.UIFigure);
498             app.LENGTHOFHAMMINGCODEEditFieldLabel.HorizontalAlignment = 'right';
499             app.LENGTHOFHAMMINGCODEEditFieldLabel.Position = [209 283 173 22];
500             app.LENGTHOFHAMMINGCODEEditFieldLabel.Text = 'LENGTH OF HAMMING CODE'
                    ↪ ;
501
502             % Create LENGTHOFHAMMINGCODEEditField
503             app.LENGTHOFHAMMINGCODEEditField = uieditfield(app.UIFigure, 'text');
504             app.LENGTHOFHAMMINGCODEEditField.ValueChangedFcn = createCallbackFcn(
                    ↪ app, @LENGTHOFHAMMINGCODEEditFieldValueChanged, true);
505             app.LENGTHOFHAMMINGCODEEditField.Editable = 'off';
506             app.LENGTHOFHAMMINGCODEEditField.Position = [389 283 172 22];
507
508             % Create NUMBEROFPARITYBITSEditFieldLabel
509             app.NUMBEROFPARITYBITSEditFieldLabel = uilabel(app.UIFigure);
510             app.NUMBEROFPARITYBITSEditFieldLabel.HorizontalAlignment = 'right';
511             app.NUMBEROFPARITYBITSEditFieldLabel.Position = [227 316 153 22];
512             app.NUMBEROFPARITYBITSEditFieldLabel.Text = 'NUMBER OF PARITY BITS';
513
514             % Create NUMBEROFPARITYBITSEditField
515             app.NUMBEROFPARITYBITSEditField = uieditfield(app.UIFigure, 'text');
516             app.NUMBEROFPARITYBITSEditField.ValueChangedFcn = createCallbackFcn(
                    ↪ app, @NUMBEROFPARITYBITSEditFieldValueChanged, true);
517             app.NUMBEROFPARITYBITSEditField.Editable = 'off';
518             app.NUMBEROFPARITYBITSEditField.Position = [387 316 172 22];
519
520             % Create Image2
521             app.Image2 = uiimage(app.UIFigure);
522             app.Image2.Position = [33 394 71 80];
523             app.Image2.ImageSource = 'images__1_-removebg-preview.png';
524
525             % Create Image3
526             app.Image3 = uiimage(app.UIFigure);
527             app.Image3.Position = [538 394 93 80];
528             app.Image3.ImageSource = 'Logo-PNG-1.png';
529
530             % Create
                    ↪ DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label
531             app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label =
                    ↪  uilabel(app.UIFigure);
532             app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                    ↪ HorizontalAlignment = 'center';
533             app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                    ↪ FontWeight = 'bold';
534             app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                    ↪ Position = [60 3 521 22];
535             app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                    ↪ Text = 'DEVELOPED BY: MD SAIFUL ISLAM RIMON (213002039) &
                    ↪ NAZMUN NAHAR (191000000)';
536
537             % Create Label
538             app.Label = uilabel(app.UIFigure);
539             app.Label.HorizontalAlignment = 'center';
540             app.Label.FontWeight = 'bold';
541             app.Label.Position = [68 28 499 22];
542             app.Label.Text = 'PROJECT SUPERVISOR: RUSMITA HALIM CHAITY,  LECTURER
                    ↪ , DEPT. OF CSE, GUB';
543
```

```
            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
    end

    % App creation and deletion
    methods (Access = public)

        % Construct app
        function app = hammingcode

            % Create UIFigure and components
            createComponents(app)

            % Register the app with App Designer
            registerApp(app, app.UIFigure)

            if nargout == 0
                clear app
            end
        end

        % Code that executes before app deletion
        function delete(app)

            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
    end
end
```

CODE [Analog to Digital Signal Conversion Page]:

```
classdef ADC < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                matlab.ui.Figure
        Image3                  matlab.ui.control.Image
        Image2                  matlab.ui.control.Image
        Label                   matlab.ui.control.Label
        DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label  matlab.
            ↪ ui.control.Label
        BitCountEditField       matlab.ui.control.EditField
        BitCountEditFieldLabel  matlab.ui.control.Label
        FrequencyEditField      matlab.ui.control.EditField
        FrequencyEditFieldLabel matlab.ui.control.Label
        CONVERTButton           matlab.ui.control.Button
        UIAxes                  matlab.ui.control.UIAxes
    end

    % Callbacks that handle component events
    methods (Access = private)

        % Button pushed function: CONVERTButton
        function CONVERTButtonPushed(app, event)
cla(app.UIAxes);
f = str2num(app.FrequencyEditField.Value) ;
n = str2num(app.BitCountEditField.Value) ;
q = f/2^(n-1) ;
t = 0:0.1:f ;
y = abs((f/2) * sin(t));

x0 = fix(y/q);

y0 = dec2bin(x0,n) ;
y1 = x0 * q;
plot(app.UIAxes,t, y, 'r', 'Linewidth', 3);
%plot (t,y,'r') ;
hold(app.UIAxes, 'on');
%plot (t, y1,'b')  ;
plot(app.UIAxes,t, y1, 'b', 'Linewidth', 3);
        end

        % Value changed function: FrequencyEditField
        function FrequencyEditFieldValueChanged(app, event)
            value = app.FrequencyEditField.Value;

        end

        % Value changed function: BitCountEditField
        function BitCountEditFieldValueChanged(app, event)
            value = app.BitCountEditField.Value;

        end
    end

    % Component initialization
    methods (Access = private)

        % Create UIFigure and components
        function createComponents(app)

            % Create UIFigure and hide until all components are created
            app.UIFigure = uifigure('Visible', 'off');
            app.UIFigure.Position = [100 100 640 480];
            app.UIFigure.Name = 'MATLAB App';

            % Create UIAxes
            app.UIAxes = uiaxes(app.UIFigure);
            title(app.UIAxes, 'ANALOG TO DIGITAL CONVERSION')
            xlabel(app.UIAxes, 'X')
            ylabel(app.UIAxes, 'Y')
            zlabel(app.UIAxes, 'Z')
```

```
            app.UIAxes.Position = [75 140 478 310];

            % Create CONVERTButton
            app.CONVERTButton = uibutton(app.UIFigure, 'push');
            app.CONVERTButton.ButtonPushedFcn = createCallbackFcn(app,
                ↪ @CONVERTButtonPushed, true);
            app.CONVERTButton.BackgroundColor = [0.4784 0.8 0.2902];
            app.CONVERTButton.FontSize = 14;
            app.CONVERTButton.FontWeight = 'bold';
            app.CONVERTButton.Position = [481 94 100 25];
            app.CONVERTButton.Text = 'CONVERT';

            % Create FrequencyEditFieldLabel
            app.FrequencyEditFieldLabel = uilabel(app.UIFigure);
            app.FrequencyEditFieldLabel.HorizontalAlignment = 'right';
            app.FrequencyEditFieldLabel.Position = [48 95 62 22];
            app.FrequencyEditFieldLabel.Text = 'Frequency';

            % Create FrequencyEditField
            app.FrequencyEditField = uieditfield(app.UIFigure, 'text');
            app.FrequencyEditField.ValueChangedFcn = createCallbackFcn(app,
                ↪ @FrequencyEditFieldValueChanged, true);
            app.FrequencyEditField.Placeholder = '10';
            app.FrequencyEditField.Position = [125 95 100 22];

            % Create BitCountEditFieldLabel
            app.BitCountEditFieldLabel = uilabel(app.UIFigure);
            app.BitCountEditFieldLabel.HorizontalAlignment = 'right';
            app.BitCountEditFieldLabel.Position = [271 95 54 22];
            app.BitCountEditFieldLabel.Text = 'Bit Count';

            % Create BitCountEditField
            app.BitCountEditField = uieditfield(app.UIFigure, 'text');
            app.BitCountEditField.ValueChangedFcn = createCallbackFcn(app,
                ↪ @BitCountEditFieldValueChanged, true);
            app.BitCountEditField.Placeholder = '3';
            app.BitCountEditField.Position = [340 95 100 22];

            % Create
                ↪ DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label =
                ↪  uilabel(app.UIFigure);
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ HorizontalAlignment = 'center';
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ FontWeight = 'bold';
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ Position = [60 3 521 22];
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ Text = 'DEVELOPED BY: MD SAIFUL ISLAM RIMON (213002039) &
                ↪ NAZMUN NAHAR (191000000)';

            % Create Label
            app.Label = uilabel(app.UIFigure);
            app.Label.HorizontalAlignment = 'center';
            app.Label.FontWeight = 'bold';
            app.Label.Position = [68 28 499 22];
            app.Label.Text = 'PROJECT SUPERVISOR: RUSMITA HALIM CHAITY,  LECTURER
                ↪ , DEPT. OF CSE, GUB';

            % Create Image2
            app.Image2 = uiimage(app.UIFigure);
            app.Image2.Position = [22 381 71 80];
            app.Image2.ImageSource = 'images__1_-removebg-preview.png';

            % Create Image3
            app.Image3 = uiimage(app.UIFigure);
            app.Image3.Position = [548 381 93 80];
            app.Image3.ImageSource = 'Logo-PNG-1.png';

            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
```

```
    end

    % App creation and deletion
    methods (Access = public)

        % Construct app
        function app = ADC

            % Create UIFigure and components
            createComponents(app)

            % Register the app with App Designer
            registerApp(app, app.UIFigure)

            if nargout == 0
                clear app
            end
        end

        % Code that executes before app deletion
        function delete(app)

            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
    end
end
```

CODE [Character Stuffing & Destuffing Page]:

```
classdef CharacterStuffingDestuffing < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure              matlab.ui.Figure
        Image3                matlab.ui.control.Image
        Image2                matlab.ui.control.Image
        Label                 matlab.ui.control.Label
        DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label  matlab.
            ↪ ui.control.Label
        CharacterStuffingDestuffingLabel  matlab.ui.control.Label
        OUTPUTLabel           matlab.ui.control.Label
        OUTPUTEditField       matlab.ui.control.EditField
        DESTUFFINGButton      matlab.ui.control.Button
        STUFFINGButton        matlab.ui.control.Button
        INPUTEditField        matlab.ui.control.EditField
        INPUTEditFieldLabel   matlab.ui.control.Label
        Image                 matlab.ui.control.Image
        OUTPUTEditFieldLabel  matlab.ui.control.Label
    end


    properties (Access = public)
        globalvariable;
        d;
    end


    % Callbacks that handle component events
    methods (Access = private)

        % Value changed function: OUTPUTEditField
        function OUTPUTEditFieldValueChanged(app, event)
            value = app.OUTPUTEditField.Value;

        end

        % Value changed function: INPUTEditField
        function INPUTEditFieldValueChanged(app, event)
            value = app.INPUTEditField.Value;

        end

        % Button pushed function: STUFFINGButton
        function STUFFINGButtonPushed(app, event)
i = 1;
j = 1;
app.globalvariable = 0;
app.d = blanks(100);
l = 'DLEETX';
sd = blanks(100);
ds = blanks(100);


app.d = app.INPUTEditField.Value;

sd(1:6) = 'DLESTX';
j = 7;

while i <= length(app.d)
    if app.d(i) == 'D' && app.d(i + 1) == 'L' && app.d(i + 2) == 'E'
        sd(j:j+5) = 'DLEDLE';
        j = j + 6;
        i = i + 3;
    else
        sd(j) = app.d(i);
        j = j + 1;
        i = i + 1;
    end
end

```

```
sd(j) = char(0);
ds = sd;

app.globalvariable = j;
ds = strcat(ds, 1);

app.OUTPUTEditField.Value = ds;

        end

        % Button pushed function: DESTUFFINGButton
        function DESTUFFINGButtonPushed(app, event)
ds = app.INPUTEditField.Value;
i = 1;
l = 'DLEETX';
sd(1:6) = 'DLESTX';
j = app.globalvariable;
p = blanks(100); %addedlater
sd = blanks(100);
ds = blanks(100);
k = j;
while j <= length(ds) && ds(j) ~= 0
    if ds(j) == 'D' && ds(j + 1) == 'L' && ds(j + 2) == 'E'
        if ds(j + 3) == 'D' && ds(j + 4) == 'L' && ds(j + 5) == 'E'
            % Remove "DLE DLE" by shifting remaining characters
            ds(k:j) = ds(j + 6:end);
            % Update loop condition and string length
            j = k;
            ds = ds(1:k);
        else
            % Found single "DLE", copy character (check index bounds)
            if i <= length(p)
                ds(k) = p(i);
                i = i + 1;
                k = k + 1;
            end
        end
    else
        % Copy character (check index bounds)
        if i <= length(p)
            ds(k) = p(i);
            i = i + 1;
            k = k + 1;
        end
    end
    j = j + 1;
end

app.OUTPUTEditField.Value = app.d;


        end
    end

    % Component initialization
    methods (Access = private)

        % Create UIFigure and components
        function createComponents(app)

            % Get the file path for locating images
            pathToMLAPP = fileparts(mfilename('fullpath'));

            % Create UIFigure and hide until all components are created
            app.UIFigure = uifigure('Visible', 'off');
            app.UIFigure.Position = [100 100 640 480];
            app.UIFigure.Name = 'MATLAB App';

            % Create OUTPUTEditFieldLabel
            app.OUTPUTEditFieldLabel = uilabel(app.UIFigure);
            app.OUTPUTEditFieldLabel.HorizontalAlignment = 'right';
            app.OUTPUTEditFieldLabel.FontSize = 24;
            app.OUTPUTEditFieldLabel.FontWeight = 'bold';
```

```
            app.OUTPUTEditFieldLabel.Position = [81 317 104 31];
            app.OUTPUTEditFieldLabel.Text = 'OUTPUT';

            % Create Image
            app.Image = uiimage(app.UIFigure);
            app.Image.Position = [-172 -100 985 649];
            app.Image.ImageSource = fullfile(pathToMLAPP, 'bg.jpg');

            % Create INPUTEditFieldLabel
            app.INPUTEditFieldLabel = uilabel(app.UIFigure);
            app.INPUTEditFieldLabel.HorizontalAlignment = 'right';
            app.INPUTEditFieldLabel.FontSize = 24;
            app.INPUTEditFieldLabel.FontWeight = 'bold';
            app.INPUTEditFieldLabel.Position = [108 214 77 31];
            app.INPUTEditFieldLabel.Text = 'INPUT';

            % Create INPUTEditField
            app.INPUTEditField = uieditfield(app.UIFigure, 'text');
            app.INPUTEditField.ValueChangedFcn = createCallbackFcn(app,
                ↪ @INPUTEditFieldValueChanged, true);
            app.INPUTEditField.FontSize = 24;
            app.INPUTEditField.Placeholder = 'UNITEDLEDSTATES';
            app.INPUTEditField.Position = [200 195 361 61];

            % Create STUFFINGButton
            app.STUFFINGButton = uibutton(app.UIFigure, 'push');
            app.STUFFINGButton.ButtonPushedFcn = createCallbackFcn(app,
                ↪ @STUFFINGButtonPushed, true);
            app.STUFFINGButton.BackgroundColor = [0.4784 0.8 0.2902];
            app.STUFFINGButton.FontSize = 18;
            app.STUFFINGButton.FontWeight = 'bold';
            app.STUFFINGButton.Position = [106 72 105 30];
            app.STUFFINGButton.Text = 'STUFFING';

            % Create DESTUFFINGButton
            app.DESTUFFINGButton = uibutton(app.UIFigure, 'push');
            app.DESTUFFINGButton.ButtonPushedFcn = createCallbackFcn(app,
                ↪ @DESTUFFINGButtonPushed, true);
            app.DESTUFFINGButton.BackgroundColor = [0.9412 0.302 0.302];
            app.DESTUFFINGButton.FontSize = 18;
            app.DESTUFFINGButton.FontWeight = 'bold';
            app.DESTUFFINGButton.Position = [400 72 130 30];
            app.DESTUFFINGButton.Text = 'DESTUFFING';

            % Create OUTPUTEditField
            app.OUTPUTEditField = uieditfield(app.UIFigure, 'text');
            app.OUTPUTEditField.ValueChangedFcn = createCallbackFcn(app,
                ↪ @OUTPUTEditFieldValueChanged, true);
            app.OUTPUTEditField.Editable = 'off';
            app.OUTPUTEditField.FontSize = 24;
            app.OUTPUTEditField.Placeholder = 'Output will show here';
            app.OUTPUTEditField.Position = [200 298 361 61];

            % Create OUTPUTLabel
            app.OUTPUTLabel = uilabel(app.UIFigure);
            app.OUTPUTLabel.FontSize = 24;
            app.OUTPUTLabel.FontWeight = 'bold';
            app.OUTPUTLabel.Position = [80 316 104 31];
            app.OUTPUTLabel.Text = 'OUTPUT';

            % Create CharacterStuffingDestuffingLabel
            app.CharacterStuffingDestuffingLabel = uilabel(app.UIFigure);
            app.CharacterStuffingDestuffingLabel.FontSize = 24;
            app.CharacterStuffingDestuffingLabel.FontWeight = 'bold';
            app.CharacterStuffingDestuffingLabel.Position = [147 412 370 31];
            app.CharacterStuffingDestuffingLabel.Text = 'Character Stuffing &
                ↪ Destuffing';

            % Create
                ↪ DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label =
                ↪  uilabel(app.UIFigure);
```

```
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ HorizontalAlignment = 'center';
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ FontWeight = 'bold';
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ Position = [60 3 521 22];
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ Text = 'DEVELOPED BY: MD SAIFUL ISLAM RIMON (213002039) &
                ↪ NAZMUN NAHAR (191000000)';

            % Create Label
            app.Label = uilabel(app.UIFigure);
            app.Label.HorizontalAlignment = 'center';
            app.Label.FontWeight = 'bold';
            app.Label.Position = [68 28 499 22];
            app.Label.Text = 'PROJECT SUPERVISOR: RUSMITA HALIM CHAITY,  LECTURER
                ↪ , DEPT. OF CSE, GUB';

            % Create Image2
            app.Image2 = uiimage(app.UIFigure);
            app.Image2.Position = [43 381 71 80];
            app.Image2.ImageSource = 'images__1_-removebg-preview.png';

            % Create Image3
            app.Image3 = uiimage(app.UIFigure);
            app.Image3.Position = [537 381 93 80];
            app.Image3.ImageSource = 'Logo-PNG-1.png';

            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
    end

    % App creation and deletion
    methods (Access = public)

        % Construct app
        function app = CharacterStuffingDestuffing

            % Create UIFigure and components
            createComponents(app)

            % Register the app with App Designer
            registerApp(app, app.UIFigure)

            if nargout == 0
                clear app
            end
        end

        % Code that executes before app deletion
        function delete(app)

            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
    end
end
```

CODE [IPv4 Conversion Page]:

```
classdef ipv4 < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                matlab.ui.Figure
        Label                   matlab.ui.control.Label
        DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label  matlab.
            ↪ ui.control.Label
        Image3                  matlab.ui.control.Image
        Image2                  matlab.ui.control.Image
        IPv4CONVERSIONLabel     matlab.ui.control.Label
        DecimalIPEditField      matlab.ui.control.EditField
        DecimalIPEditFieldLabel matlab.ui.control.Label
        BinaryIPToDecimalButton matlab.ui.control.Button
        BinaryIPEditField       matlab.ui.control.EditField
        BinaryIPEditFieldLabel  matlab.ui.control.Label
        InputEditField          matlab.ui.control.EditField
        InputEditFieldLabel     matlab.ui.control.Label
        DecimalIPToBinaryButton matlab.ui.control.Button
        Image                   matlab.ui.control.Image
    end

    % Callbacks that handle component events
    methods (Access = private)

        % Value changed function: InputEditField
        function InputEditFieldValueChanged(app, event)
            value = app.InputEditField.Value;

        end

        % Button pushed function: DecimalIPToBinaryButton
        function DecimalIPToBinaryButtonPushed(app, event)
            dec = app.InputEditField.Value;
            octets = str2double(strsplit(dec, '.'));
            binaryIP = dec2bin(octets, 8);
            binaryIP = reshape(binaryIP', 1, []);
            binaryIP = regexprep(binaryIP, '(\d{8})', '$1 ');
            app.BinaryIPEditField.Value = string(binaryIP);

        end

        % Value changed function: BinaryIPEditField
        function BinaryIPEditFieldValueChanged(app, event)
            value = app.BinaryIPEditField.Value;

        end

        % Button pushed function: BinaryIPToDecimalButton
        function BinaryIPToDecimalButtonPushed(app, event)
binaryIP = app.InputEditField.Value;

% Remove whitespaces and split the binary string into octets
binaryOctets = strsplit(strtrim(binaryIP), ' ');

% Convert each binary octet to decimal
decimalOctets = zeros(1, numel(binaryOctets));
for i = 1:numel(binaryOctets)
    decimalOctets(i) = bin2dec(binaryOctets{i});
end

% Join decimal octets into an IPv4 address
ipv4Address = join(string(decimalOctets), '.');
app.DecimalIPEditField.Value = string(ipv4Address);



        end

        % Value changed function: DecimalIPEditField
        function DecimalIPEditFieldValueChanged(app, event)
```

```
            value = app.DecimalIPEditField.Value;

        end
    end

    % Component initialization
    methods (Access = private)

        % Create UIFigure and components
        function createComponents(app)

            % Create UIFigure and hide until all components are created
            app.UIFigure = uifigure('Visible', 'off');
            app.UIFigure.Position = [100 100 640 480];
            app.UIFigure.Name = 'MATLAB App';

            % Create Image
            app.Image = uiimage(app.UIFigure);
            app.Image.Position = [-211 -93 1059 654];
            app.Image.ImageSource = 'bg.jpg';

            % Create DecimalIPToBinaryButton
            app.DecimalIPToBinaryButton = uibutton(app.UIFigure, 'push');
            app.DecimalIPToBinaryButton.ButtonPushedFcn = createCallbackFcn(app,
                ↪ @DecimalIPToBinaryButtonPushed, true);
            app.DecimalIPToBinaryButton.BackgroundColor = [1 0.4118 0.1608];
            app.DecimalIPToBinaryButton.FontWeight = 'bold';
            app.DecimalIPToBinaryButton.Position = [84 117 473 38];
            app.DecimalIPToBinaryButton.Text = {'Decimal IP '; ' To Binary'};

            % Create InputEditFieldLabel
            app.InputEditFieldLabel = uilabel(app.UIFigure);
            app.InputEditFieldLabel.HorizontalAlignment = 'right';
            app.InputEditFieldLabel.FontSize = 18;
            app.InputEditFieldLabel.Position = [92 173 45 23];
            app.InputEditFieldLabel.Text = 'Input';

            % Create InputEditField
            app.InputEditField = uieditfield(app.UIFigure, 'text');
            app.InputEditField.ValueChangedFcn = createCallbackFcn(app,
                ↪ @InputEditFieldValueChanged, true);
            app.InputEditField.FontSize = 18;
            app.InputEditField.Placeholder = '192.168.10.10';
            app.InputEditField.Position = [152 170 397 30];

            % Create BinaryIPEditFieldLabel
            app.BinaryIPEditFieldLabel = uilabel(app.UIFigure);
            app.BinaryIPEditFieldLabel.HorizontalAlignment = 'right';
            app.BinaryIPEditFieldLabel.FontSize = 24;
            app.BinaryIPEditFieldLabel.FontWeight = 'bold';
            app.BinaryIPEditFieldLabel.Position = [22 327 109 31];
            app.BinaryIPEditFieldLabel.Text = 'Binary IP';

            % Create BinaryIPEditField
            app.BinaryIPEditField = uieditfield(app.UIFigure, 'text');
            app.BinaryIPEditField.ValueChangedFcn = createCallbackFcn(app,
                ↪ @BinaryIPEditFieldValueChanged, true);
            app.BinaryIPEditField.Editable = 'off';
            app.BinaryIPEditField.FontSize = 18;
            app.BinaryIPEditField.FontWeight = 'bold';
            app.BinaryIPEditField.Position = [146 311 403 71];

            % Create BinaryIPToDecimalButton
            app.BinaryIPToDecimalButton = uibutton(app.UIFigure, 'push');
            app.BinaryIPToDecimalButton.ButtonPushedFcn = createCallbackFcn(app,
                ↪ @BinaryIPToDecimalButtonPushed, true);
            app.BinaryIPToDecimalButton.BackgroundColor = [0.0941 0.6588 0.0941];
            app.BinaryIPToDecimalButton.FontWeight = 'bold';
            app.BinaryIPToDecimalButton.Position = [84 71 473 38];
            app.BinaryIPToDecimalButton.Text = {'Binary IP '; ' To Decimal'};

            % Create DecimalIPEditFieldLabel
            app.DecimalIPEditFieldLabel = uilabel(app.UIFigure);
```

```
            app.DecimalIPEditFieldLabel.HorizontalAlignment = 'right';
            app.DecimalIPEditFieldLabel.FontSize = 24;
            app.DecimalIPEditFieldLabel.FontWeight = 'bold';
            app.DecimalIPEditFieldLabel.Position = [5 241 126 31];
            app.DecimalIPEditFieldLabel.Text = 'Decimal IP';

            % Create DecimalIPEditField
            app.DecimalIPEditField = uieditfield(app.UIFigure, 'text');
            app.DecimalIPEditField.ValueChangedFcn = createCallbackFcn(app,
                ↪ @DecimalIPEditFieldValueChanged, true);
            app.DecimalIPEditField.Editable = 'off';
            app.DecimalIPEditField.FontSize = 24;
            app.DecimalIPEditField.FontWeight = 'bold';
            app.DecimalIPEditField.Position = [146 225 403 71];

            % Create IPv4CONVERSIONLabel
            app.IPv4CONVERSIONLabel = uilabel(app.UIFigure);
            app.IPv4CONVERSIONLabel.FontSize = 24;
            app.IPv4CONVERSIONLabel.FontWeight = 'bold';
            app.IPv4CONVERSIONLabel.Position = [209 431 224 31];
            app.IPv4CONVERSIONLabel.Text = 'IPv4 CONVERSION';

            % Create Image2
            app.Image2 = uiimage(app.UIFigure);
            app.Image2.Position = [22 380 71 84];
            app.Image2.ImageSource = 'images__1_-removebg-preview.png';

            % Create Image3
            app.Image3 = uiimage(app.UIFigure);
            app.Image3.Position = [522 371 100 100];
            app.Image3.ImageSource = 'Logo-PNG-1.png';

            % Create
                ↪ DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label =
                ↪  uilabel(app.UIFigure);
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ HorizontalAlignment = 'center';
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ FontWeight = 'bold';
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ Position = [60 0 521 22];
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.
                ↪ Text = 'DEVELOPED BY: MD SAIFUL ISLAM RIMON (213002039) &
                ↪ NAZMUN NAHAR (191000000)';

            % Create Label
            app.Label = uilabel(app.UIFigure);
            app.Label.HorizontalAlignment = 'center';
            app.Label.FontWeight = 'bold';
            app.Label.Position = [68 25 499 22];
            app.Label.Text = 'PROJECT SUPERVISOR: RUSMITA HALIM CHAITY,  LECTURER
                ↪ , DEPT. OF CSE, GUB';

            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
    end

    % App creation and deletion
    methods (Access = public)

        % Construct app
        function app = ipv4

            % Create UIFigure and components
            createComponents(app)

            % Register the app with App Designer
            registerApp(app, app.UIFigure)

            if nargout == 0
                clear app
```

```
            end
        end

        % Code that executes before app deletion
        function delete(app)

            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
    end
end
```

CODE [Cyclic Redundancy Check (CRC) Page]:

```
classdef crcapp < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                        matlab.ui.Figure
        InputBitStringLabel             matlab.ui.control.Label
        Label                           matlab.ui.control.Label
        DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label  matlab.
            ↪ ui.control.Label
        Image3                          matlab.ui.control.Image
        Image2                          matlab.ui.control.Image
        CyclicRedundancyCheckCRCLabel   matlab.ui.control.Label
        Lamp                            matlab.ui.control.Lamp
        ErrorStatusEditField            matlab.ui.control.EditField
        ErrorStatusEditFieldLabel       matlab.ui.control.Label
        CheckButton                     matlab.ui.control.Button
        ReceivedCodewordEditField       matlab.ui.control.EditField
        ReceivedCodewordEditFieldLabel  matlab.ui.control.Label
        GenerateTransmittingCodewordButton  matlab.ui.control.Button
        TransmittingCodewordEditField   matlab.ui.control.EditField
        TransmittingCodewordLabel       matlab.ui.control.Label
        InputPolynomialinBinaryEditField  matlab.ui.control.EditField
        InputPolynomialinBinaryEditFieldLabel  matlab.ui.control.Label
        InputBitStingEditField          matlab.ui.control.EditField
        Image                           matlab.ui.control.Image
        InputBitStingEditFieldLabel     matlab.ui.control.Label
    end


    properties (Access = public)
        gen_poly
    end

    methods (Access = public)

        function crc_matlab_sender(app)
    % Define the data and polynomial
    data = app.InputBitStingEditField.Value;
    app.gen_poly = app.InputPolynomialinBinaryEditField.Value;

    data_length = length(data);
    N = length(app.gen_poly);

    % Append zeros to the data
    appended_data = [data, repmat('0', 1, N-1)];

    % Perform CRC
    remainder = crc_division(app, appended_data, app.gen_poly);

    % Generate codeword
    codeword = [data, remainder];

    app.TransmittingCodewordEditField.Value = codeword;

        end

        function crc_matlab_receiver(app)
    % Define the data and polynomial
      received_data = app.ReceivedCodewordEditField.Value;
      received_remainder = crc_division(app, received_data, app.gen_poly);

     if any(received_remainder ~= '0')
          app.ErrorStatusEditField.Value = 'Error Detected';
            app.Lamp.Color = [1.00,0.00,0.00];

      else
          app.ErrorStatusEditField.Value = 'No Error Detected';
            app.Lamp.Color = [0.00,1.00,0.00];
      end

      end
```

```
71
72
73  function remainder = crc_division(app, input_data, gen_poly)
74      N = length(gen_poly);
75      temp_data = input_data(1:N);
76
77      for i = 1:length(input_data) - N + 1
78          if temp_data(1) == '1'
79              temp_data = xor_strings(app, temp_data, gen_poly);
80          end
81
82          if i+N <= length(input_data)
83              temp_data = [temp_data(2:end), input_data(i+N)];
84          else
85              temp_data = temp_data(2:end);
86          end
87      end
88
89      remainder = temp_data;
90  end
91
92  function result = xor_strings(app, a, b)
93      result = char(xor(a-'0', b-'0') + '0');
94  end
95
96      end
97
98
99      % Callbacks that handle component events
100     methods (Access = private)
101
102         % Value changed function: InputBitStingEditField
103         function InputBitStingEditFieldValueChanged(app, event)
104             value = app.InputBitStingEditField.Value;
105
106         end
107
108         % Value changed function: InputPolynomialinBinaryEditField
109         function InputPolynomialinBinaryEditFieldValueChanged(app, event)
110             value = app.InputPolynomialinBinaryEditField.Value;
111
112         end
113
114         % Button pushed function: GenerateTransmittingCodewordButton
115         function GenerateTransmittingCodewordButtonPushed(app, event)
116             crc_matlab_sender(app);
117
118
119         end
120
121         % Value changed function: TransmittingCodewordEditField
122         function TransmittingCodewordEditFieldValueChanged(app, event)
123             value = app.TransmittingCodewordEditField.Value;
124
125         end
126
127         % Value changed function: ReceivedCodewordEditField
128         function ReceivedCodewordEditFieldValueChanged(app, event)
129             value = app.ReceivedCodewordEditField.Value;
130
131         end
132
133         % Button pushed function: CheckButton
134         function CheckButtonPushed(app, event)
135             crc_matlab_receiver(app)
136         end
137
138         % Value changed function: ErrorStatusEditField
139         function ErrorStatusEditFieldValueChanged(app, event)
140             value = app.ErrorStatusEditField.Value;
141
142         end
143     end
```

```
144
145     % Component initialization
146     methods (Access = private)
147
148         % Create UIFigure and components
149         function createComponents(app)
150
151             % Create UIFigure and hide until all components are created
152             app.UIFigure = uifigure('Visible', 'off');
153             app.UIFigure.Position = [100 100 640 480];
154             app.UIFigure.Name = 'MATLAB App';
155
156             % Create InputBitStingEditFieldLabel
157             app.InputBitStingEditFieldLabel = uilabel(app.UIFigure);
158             app.InputBitStingEditFieldLabel.HorizontalAlignment = 'right';
159             app.InputBitStingEditFieldLabel.FontSize = 14;
160             app.InputBitStingEditFieldLabel.FontWeight = 'bold';
161             app.InputBitStingEditFieldLabel.Position = [29 366 101 22];
162             app.InputBitStingEditFieldLabel.Text = 'Input Bit Sting';
163
164             % Create Image
165             app.Image = uiimage(app.UIFigure);
166             app.Image.Position = [-166 -76 979 647];
167             app.Image.ImageSource = 'bg.jpg';
168
169             % Create InputBitStingEditField
170             app.InputBitStingEditField = uieditfield(app.UIFigure, 'text');
171             app.InputBitStingEditField.ValueChangedFcn = createCallbackFcn(app,
                    ↪ @InputBitStingEditFieldValueChanged, true);
172             app.InputBitStingEditField.FontSize = 14;
173             app.InputBitStingEditField.FontWeight = 'bold';
174             app.InputBitStingEditField.Placeholder = '1001010';
175             app.InputBitStingEditField.Position = [145 366 127 22];
176
177             % Create InputPolynomialinBinaryEditFieldLabel
178             app.InputPolynomialinBinaryEditFieldLabel = uilabel(app.UIFigure);
179             app.InputPolynomialinBinaryEditFieldLabel.HorizontalAlignment = '
                    ↪ right';
180             app.InputPolynomialinBinaryEditFieldLabel.FontSize = 14;
181             app.InputPolynomialinBinaryEditFieldLabel.FontWeight = 'bold';
182             app.InputPolynomialinBinaryEditFieldLabel.Position = [289 366 182
                    ↪ 22];
183             app.InputPolynomialinBinaryEditFieldLabel.Text = 'Input Polynomial in
                    ↪  Binary';
184
185             % Create InputPolynomialinBinaryEditField
186             app.InputPolynomialinBinaryEditField = uieditfield(app.UIFigure, '
                    ↪ text');
187             app.InputPolynomialinBinaryEditField.ValueChangedFcn =
                    ↪ createCallbackFcn(app,
                    ↪ @InputPolynomialinBinaryEditFieldValueChanged, true);
188             app.InputPolynomialinBinaryEditField.FontSize = 14;
189             app.InputPolynomialinBinaryEditField.FontWeight = 'bold';
190             app.InputPolynomialinBinaryEditField.Placeholder = '1011';
191             app.InputPolynomialinBinaryEditField.Position = [486 366 127 22];
192
193             % Create TransmittingCodewordLabel
194             app.TransmittingCodewordLabel = uilabel(app.UIFigure);
195             app.TransmittingCodewordLabel.FontSize = 18;
196             app.TransmittingCodewordLabel.FontWeight = 'bold';
197             app.TransmittingCodewordLabel.Position = [38 227 113 55];
198             app.TransmittingCodewordLabel.Text = {'Transmitting'; 'Codeword'};
199
200             % Create TransmittingCodewordEditField
201             app.TransmittingCodewordEditField = uieditfield(app.UIFigure, 'text')
                    ↪ ;
202             app.TransmittingCodewordEditField.ValueChangedFcn = createCallbackFcn
                    ↪ (app, @TransmittingCodewordEditFieldValueChanged, true);
203             app.TransmittingCodewordEditField.Editable = 'off';
204             app.TransmittingCodewordEditField.FontSize = 18;
205             app.TransmittingCodewordEditField.FontWeight = 'bold';
206             app.TransmittingCodewordEditField.Placeholder = 'Shows Transmitting
                    ↪ Codeword';
```

```
            app.TransmittingCodewordEditField.Position = [166 227 321 55];

            % Create GenerateTransmittingCodewordButton
            app.GenerateTransmittingCodewordButton = uibutton(app.UIFigure, 'push
                ↪ ');
            app.GenerateTransmittingCodewordButton.ButtonPushedFcn =
                ↪ createCallbackFcn(app,
                ↪ @GenerateTransmittingCodewordButtonPushed, true);
            app.GenerateTransmittingCodewordButton.BackgroundColor = [0.4784 0.8
                ↪ 0.2902];
            app.GenerateTransmittingCodewordButton.FontSize = 14;
            app.GenerateTransmittingCodewordButton.FontWeight = 'bold';
            app.GenerateTransmittingCodewordButton.Position = [180 314 238 25];
            app.GenerateTransmittingCodewordButton.Text = 'Generate Transmitting
                ↪ Codeword';

            % Create ReceivedCodewordEditFieldLabel
            app.ReceivedCodewordEditFieldLabel = uilabel(app.UIFigure);
            app.ReceivedCodewordEditFieldLabel.HorizontalAlignment = 'right';
            app.ReceivedCodewordEditFieldLabel.FontSize = 14;
            app.ReceivedCodewordEditFieldLabel.FontWeight = 'bold';
            app.ReceivedCodewordEditFieldLabel.Position = [37 153 139 22];
            app.ReceivedCodewordEditFieldLabel.Text = 'Received Codeword';

            % Create ReceivedCodewordEditField
            app.ReceivedCodewordEditField = uieditfield(app.UIFigure, 'text');
            app.ReceivedCodewordEditField.ValueChangedFcn = createCallbackFcn(app
                ↪ , @ReceivedCodewordEditFieldValueChanged, true);
            app.ReceivedCodewordEditField.FontSize = 14;
            app.ReceivedCodewordEditField.FontWeight = 'bold';
            app.ReceivedCodewordEditField.Placeholder = '1001010111';
            app.ReceivedCodewordEditField.Position = [191 153 214 22];

            % Create CheckButton
            app.CheckButton = uibutton(app.UIFigure, 'push');
            app.CheckButton.ButtonPushedFcn = createCallbackFcn(app,
                ↪ @CheckButtonPushed, true);
            app.CheckButton.BackgroundColor = [0.4784 0.8 0.2902];
            app.CheckButton.FontSize = 14;
            app.CheckButton.FontWeight = 'bold';
            app.CheckButton.Position = [444 152 100 25];
            app.CheckButton.Text = 'Check';

            % Create ErrorStatusEditFieldLabel
            app.ErrorStatusEditFieldLabel = uilabel(app.UIFigure);
            app.ErrorStatusEditFieldLabel.HorizontalAlignment = 'center';
            app.ErrorStatusEditFieldLabel.FontSize = 18;
            app.ErrorStatusEditFieldLabel.FontWeight = 'bold';
            app.ErrorStatusEditFieldLabel.Position = [38 70 113 55];
            app.ErrorStatusEditFieldLabel.Text = 'Error Status';

            % Create ErrorStatusEditField
            app.ErrorStatusEditField = uieditfield(app.UIFigure, 'text');
            app.ErrorStatusEditField.ValueChangedFcn = createCallbackFcn(app,
                ↪ @ErrorStatusEditFieldValueChanged, true);
            app.ErrorStatusEditField.Editable = 'off';
            app.ErrorStatusEditField.HorizontalAlignment = 'center';
            app.ErrorStatusEditField.FontSize = 18;
            app.ErrorStatusEditField.FontWeight = 'bold';
            app.ErrorStatusEditField.Placeholder = 'Shows Error Status';
            app.ErrorStatusEditField.Position = [166 70 321 55];

            % Create Lamp
            app.Lamp = uilamp(app.UIFigure);
            app.Lamp.Position = [505 78 42 42];
            app.Lamp.Color = [1 0 0];

            % Create CyclicRedundancyCheckCRCLabel
            app.CyclicRedundancyCheckCRCLabel = uilabel(app.UIFigure);
            app.CyclicRedundancyCheckCRCLabel.HorizontalAlignment = 'center';
            app.CyclicRedundancyCheckCRCLabel.WordWrap = 'on';
            app.CyclicRedundancyCheckCRCLabel.FontSize = 14;
            app.CyclicRedundancyCheckCRCLabel.FontWeight = 'bold';
```

```
            app.CyclicRedundancyCheckCRCLabel.Position = [140 433 362 22];
            app.CyclicRedundancyCheckCRCLabel.Text = 'Cyclic Redundancy Check (CRC)';

            % Create Image2
            app.Image2 = uiimage(app.UIFigure);
            app.Image2.Position = [33 394 71 80];
            app.Image2.ImageSource = 'images__1_-removebg-preview.png';

            % Create Image3
            app.Image3 = uiimage(app.UIFigure);
            app.Image3.Position = [538 394 93 80];
            app.Image3.ImageSource = 'Logo-PNG-1.png';

            % Create DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label = uilabel(app.UIFigure);
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.HorizontalAlignment = 'center';
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.FontWeight = 'bold';
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.Position = [60 3 521 22];
            app.DEVELOPEDBYMDSAIFULISLAMRIMON213002039NAZMUNNAHAR191000000Label.Text = 'DEVELOPED BY: MD SAIFUL ISLAM RIMON (213002039) & NAZMUN NAHAR (191000000)';

            % Create Label
            app.Label = uilabel(app.UIFigure);
            app.Label.HorizontalAlignment = 'center';
            app.Label.FontWeight = 'bold';
            app.Label.Position = [68 28 499 22];
            app.Label.Text = 'PROJECT SUPERVISOR: RUSMITA HALIM CHAITY,  LECTURER , DEPT. OF CSE, GUB';

            % Create InputBitStringLabel
            app.InputBitStringLabel = uilabel(app.UIFigure);
            app.InputBitStringLabel.FontSize = 14;
            app.InputBitStringLabel.FontWeight = 'bold';
            app.InputBitStringLabel.Position = [37 366 106 22];
            app.InputBitStringLabel.Text = 'Input Bit String';

            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
    end

    % App creation and deletion
    methods (Access = public)

        % Construct app
        function app = crcapp

            % Create UIFigure and components
            createComponents(app)

            % Register the app with App Designer
            registerApp(app, app.UIFigure)

            if nargout == 0
                clear app
            end
        end

        % Code that executes before app deletion
        function delete(app)

            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
    end
end
```

2.4 Algorithms

Algorithm 1: DATA TRANSMISSION SIMULATOR (MATLAB APP)

Start the MATLAB environment and initialize the main application window.
Load any necessary resources or libraries.
Create Home Page Interface:
Create a menu-driven interface named "Home Page."
Add buttons for each feature:
- Line Coding
- Hamming Encoding & Decoding
- Analog to Digital Signal Conversion
- Character Stuffing & Destuffing
- IPv4 Conversion
- Bit Stuffing & Destuffing
- CRC Error Detection

Feature Selection Handling:
Implement a callback function for each menu item.
Each callback invokes the corresponding feature module.
Implement Feature Modules: for *each feature module* **do**
    Define functionality for each module.
Create functions for the specific operations.

User Input and Output:
Design input boxes for user data entry.
Design output boxes for displaying results.
Implement a graphical demonstration system for visualizing data.
Execution of Operations: for *each operation* **do**
    Retrieve data from input boxes.
Process data using the respective algorithm.
Display results in output boxes or graphically.

Validation and Error Handling:
Include input validation for data integrity.
Implement error handling and user feedback mechanisms.
Cleanup and Close Application:
Allow users to exit to Home Page or close the application.
Release any allocated resources.

Chapter 3

Performance Evaluation

3.1 Simulation Environment/ Simulation Procedure

1. **Platform:** The code is written in Matlab language, suggesting it is intended to run on an Matlab App Designer. The environment should include all Matlab Packages and App Designer for running the compiled code.

2. **Code Compilation:** Use Matlab App Designer to compile all parts of codes and files first. Otherwise, the program will generate an error like: a file missing.

3. **Execution:** After running once all the files, Now run the homepage only and click every functionality without clicking on Run for every file. Just enjoy!

3.2 Results Analysis/Testing

3.2.1 Home Page

After running the project, at first, we will have graphical interface like this. Here we have to click which transmission technique we are going to use and it will open a new window automatically.

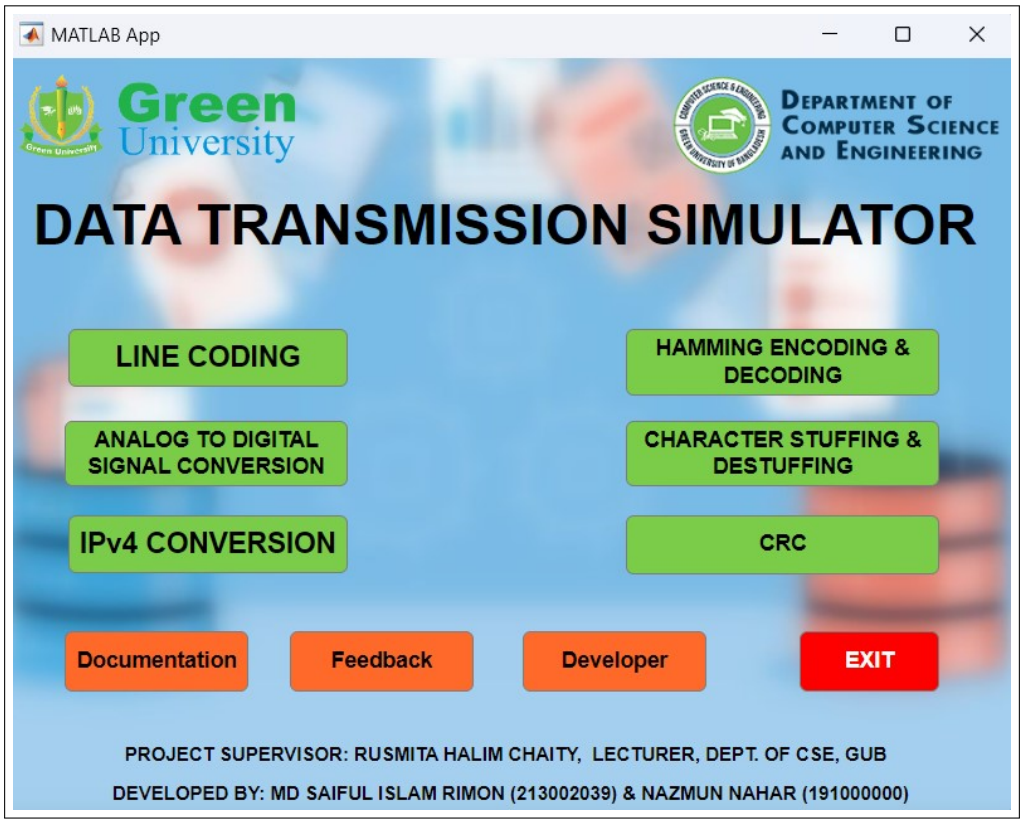

Figure 3.1: Graphical User Interface: Home Page

3.2.2 Line Coding

After clicking on Line Coding button on the home page, we will see a graphical user interface like the picture below. Here, we can input the bit-stream and can try different line coding techniques just by clicking on the buttons for Line Coding techniques.

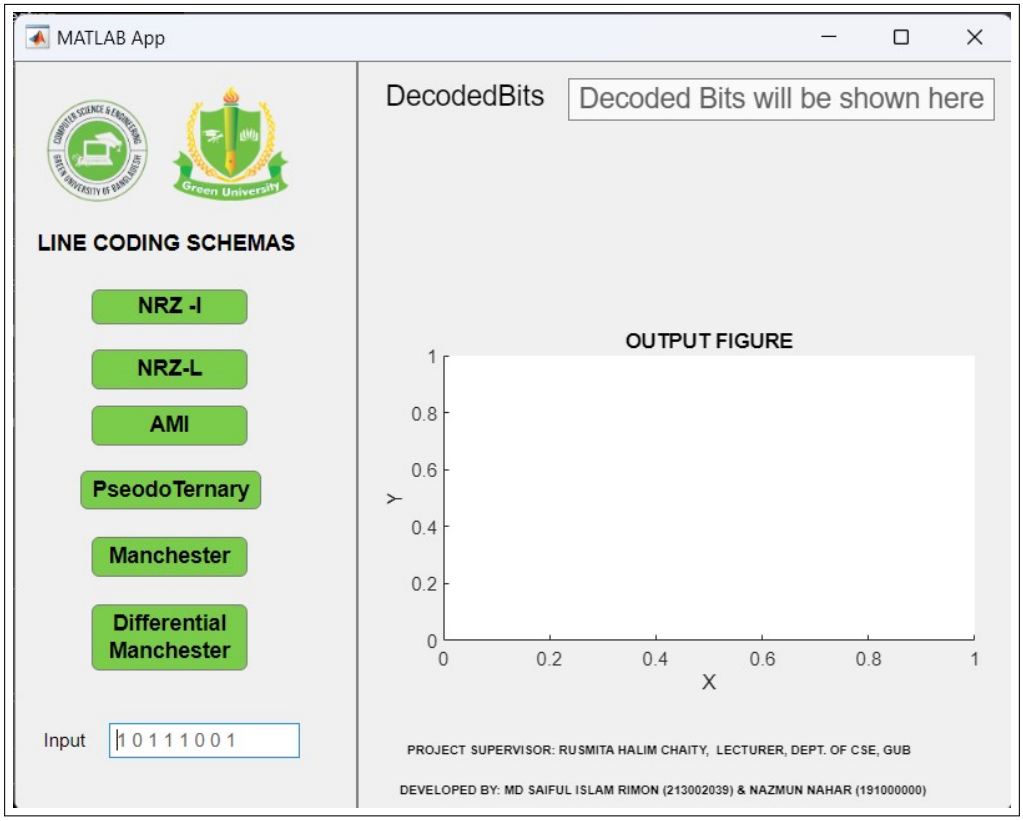

Figure 3.2: Graphical User Interface: Line Coding

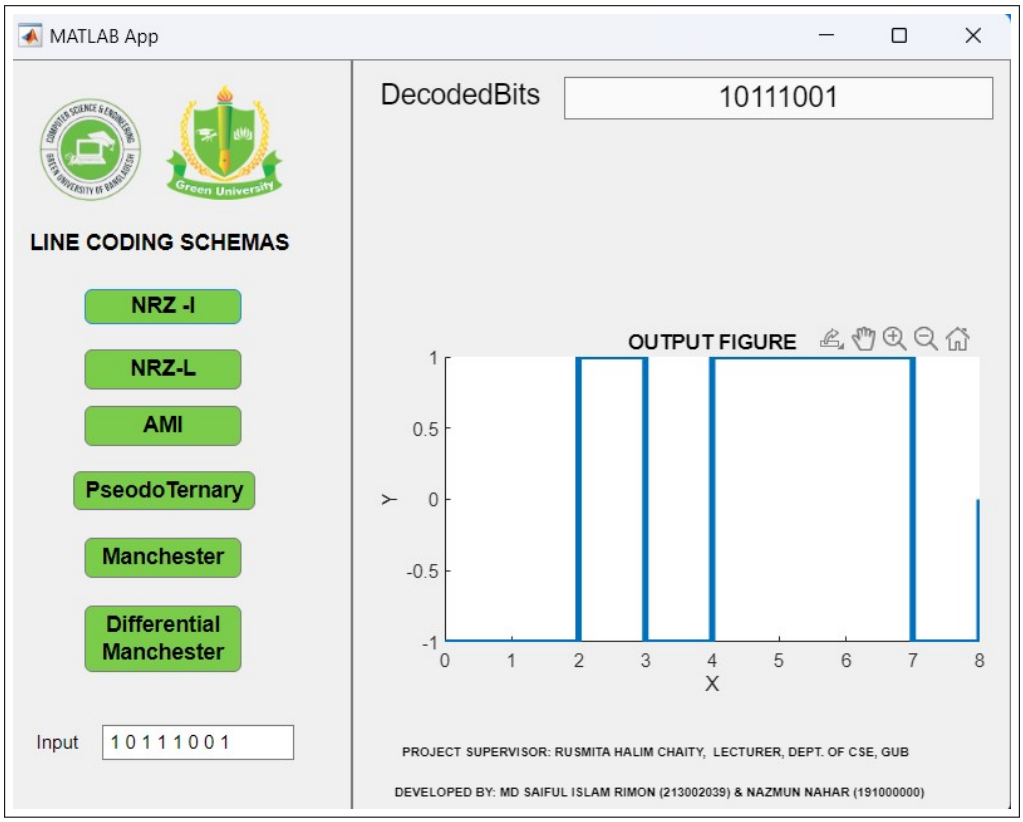

Figure 3.3: Graphical User Interface: Line Coding: NRZ - I

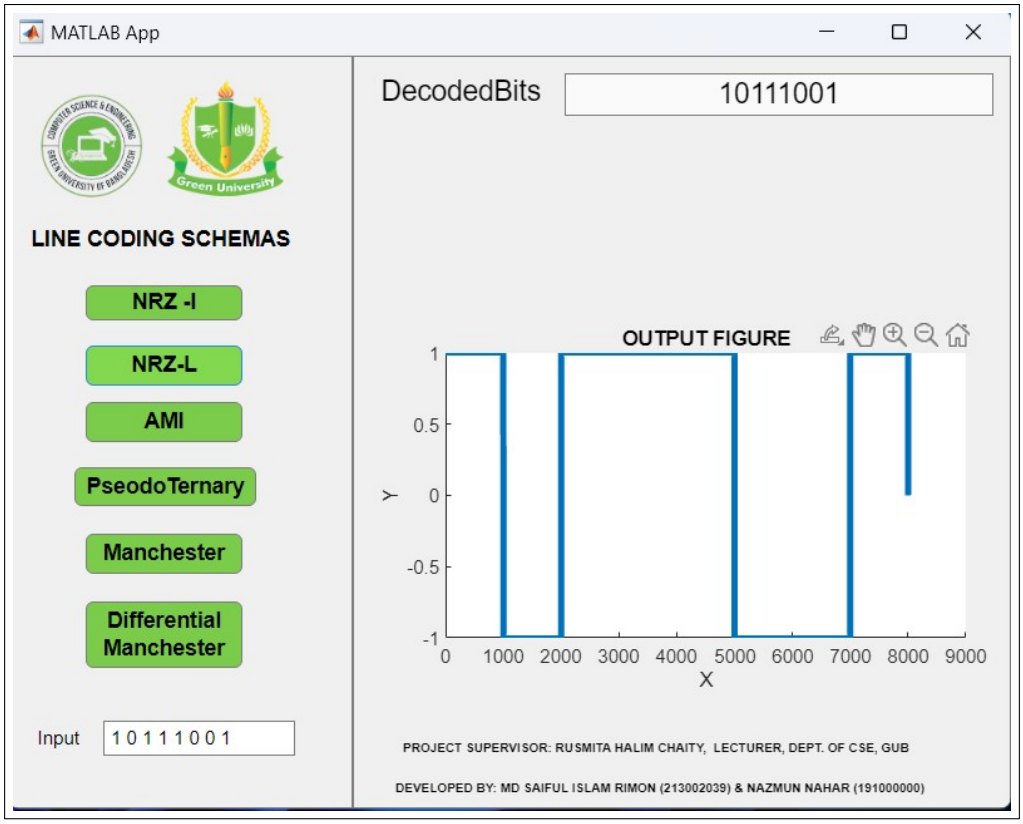

Figure 3.4: Graphical User Interface: Line Coding: NRZ - L

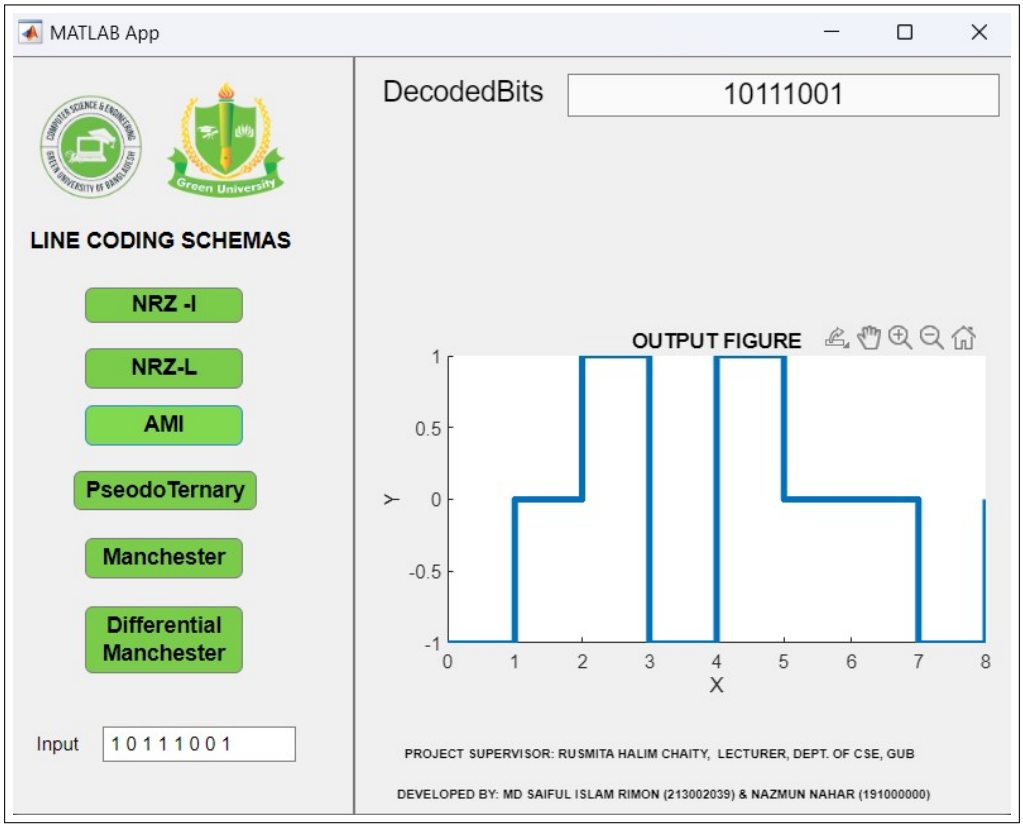

Figure 3.5: Graphical User Interface: Line Coding: AMI

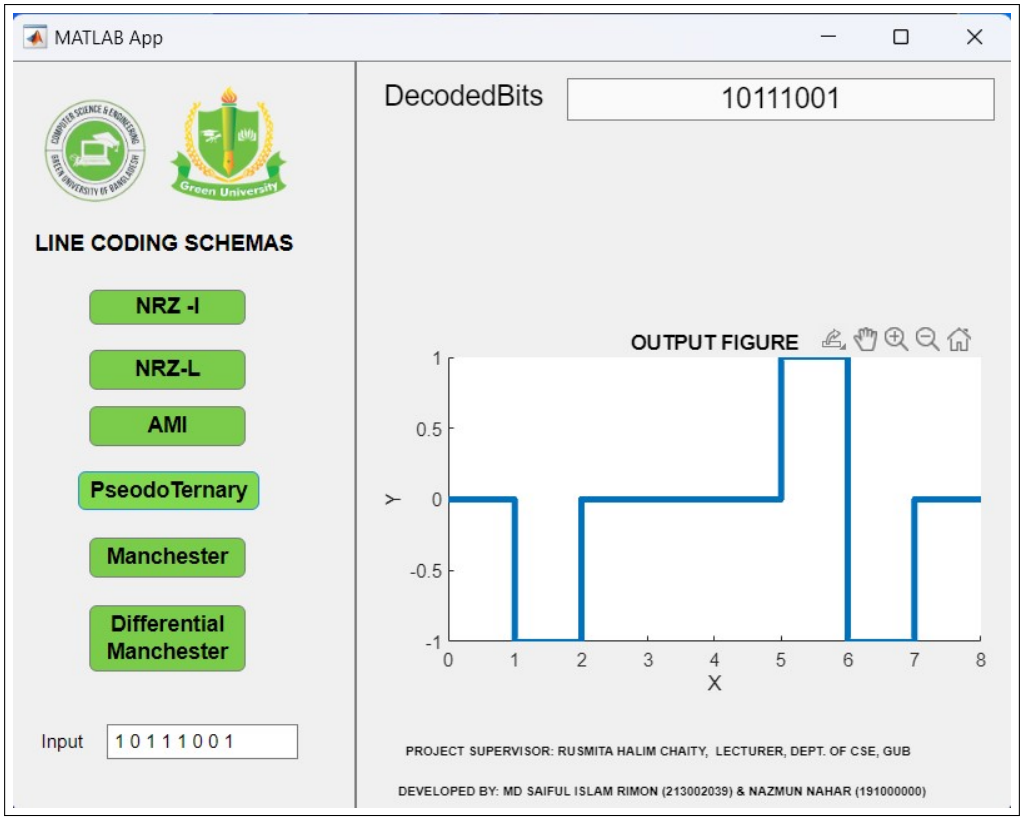

Figure 3.6: Graphical User Interface: Line Coding: PSEUDO TERNARY

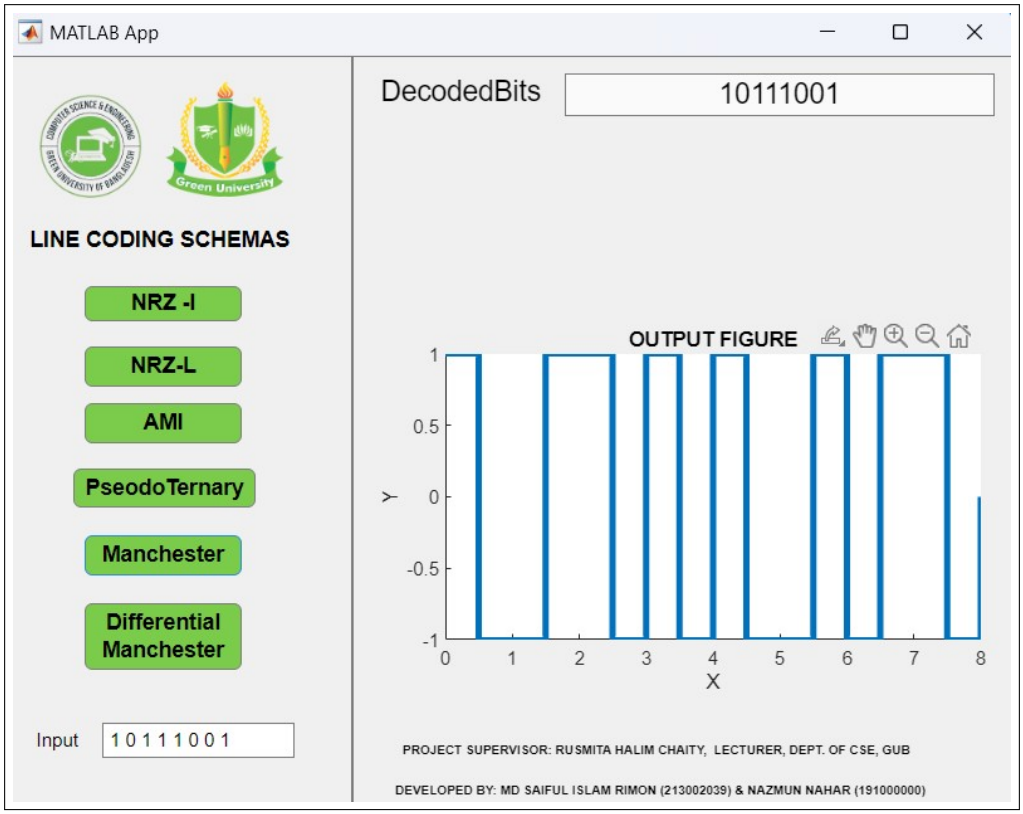

Figure 3.7: Graphical User Interface: Line Coding: MANCHESTER

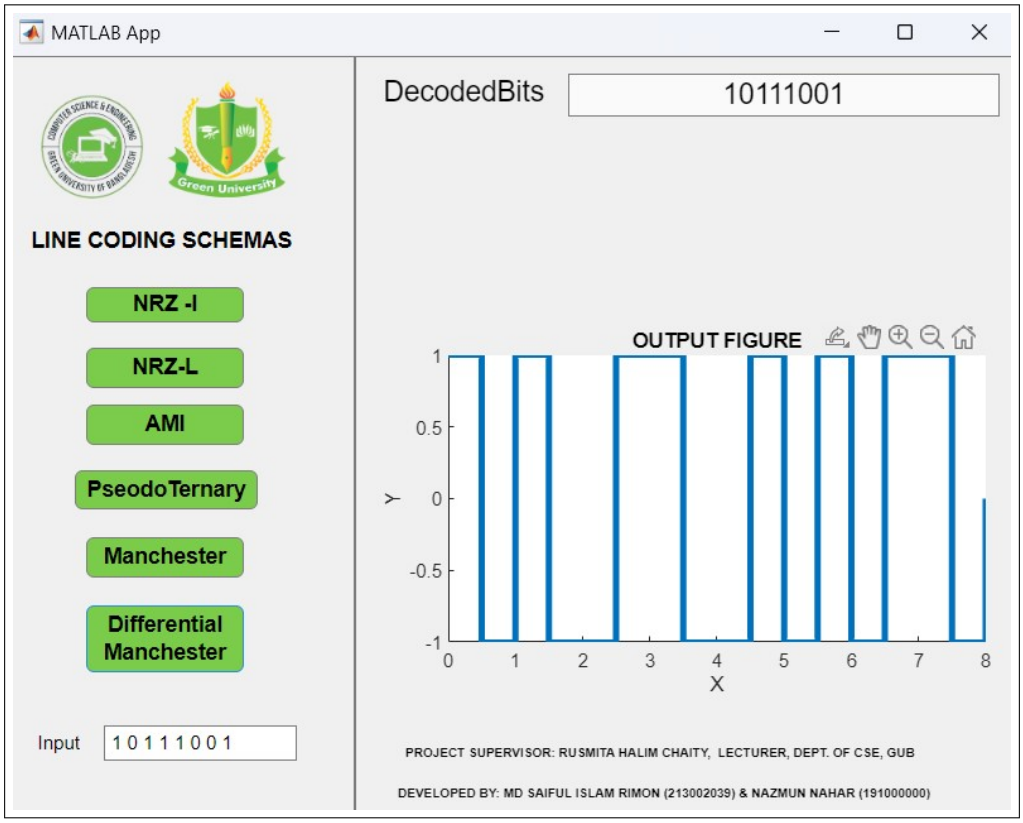

Figure 3.8: Graphical User Interface: Line Coding: DIFFERENTIAL MANCHESTER

3.2.3 Hamming Encoding & Decoding

After clicking on Hamming Encoding & Decoding button on the home page, we will see a graphical user interface like the picture below. Here, we can input the bit-stream and can see encoding and it's parity bits. Then, if the receiver's bit-stream is different, then it will show the position of error and correct the bit-stream. It is the, single bit error detection and correction technique.

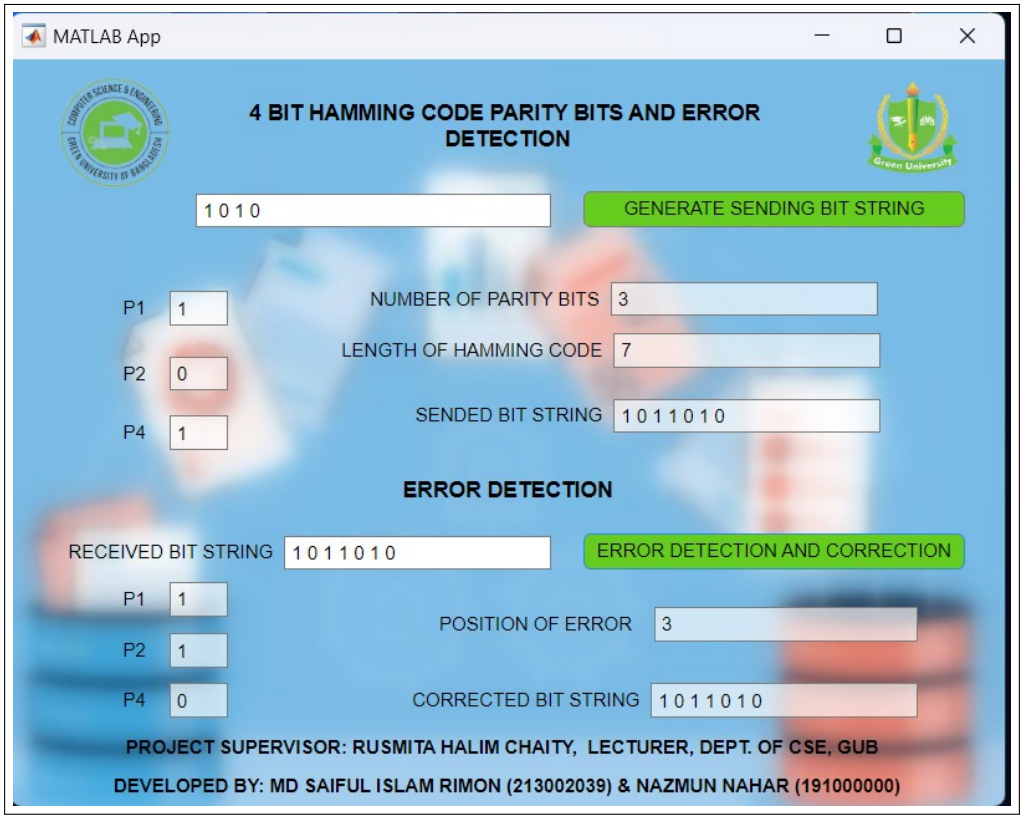

Figure 3.9: Graphical User Interface: Hamming Encoding & Decoding Page

3.2.4 Analog to Digital Signal Conversion

After clicking on the Analog to Digital Signal Conversion button on the home page, we will see a graphical user interface like the picture below. Here, we can input the frequency and the number of bits and it will show both the Analog signal and the converted digital signal.

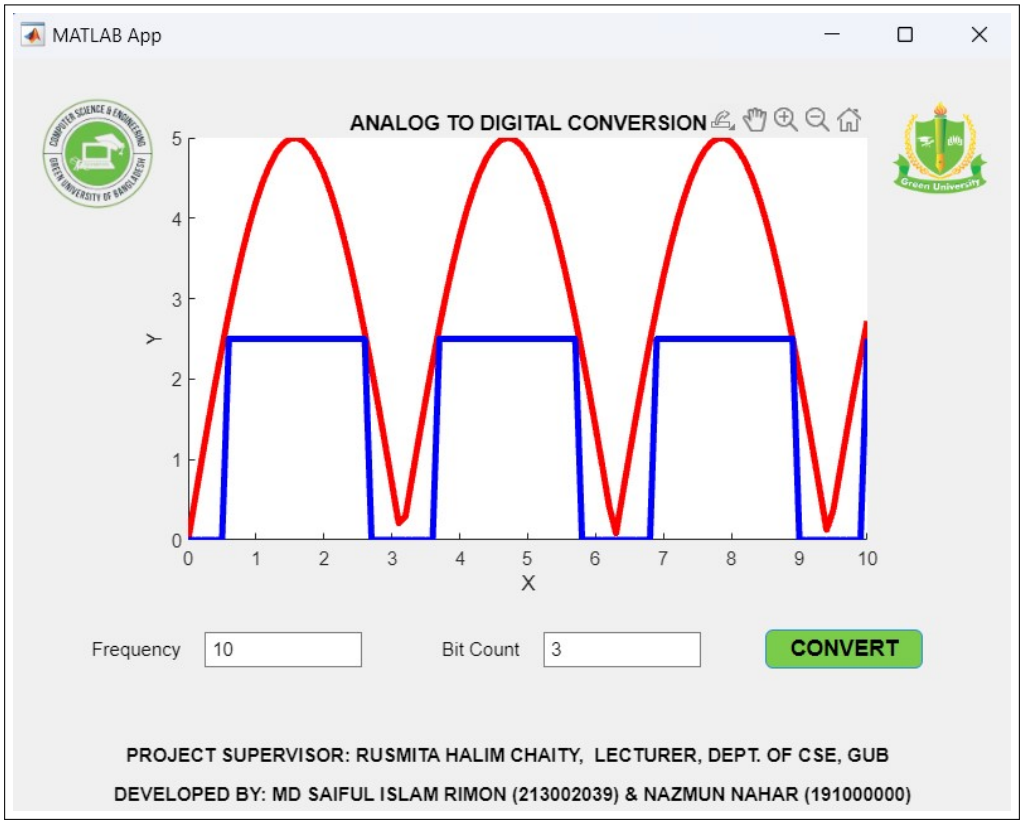

Figure 3.10: Graphical User Interface: Analog to Digital Signal Conversion Page

3.2.5 Character Stuffing & Destuffing

After clicking on the Character Stuffing & Destuffing button on the home page, we will see a graphical user interface like the picture below. Here, we can input the string. After clicking the button Stuff, then it show the stuffed string and then if we input the stuffed string and if we click Destuff button, then it will show the destuffed string.

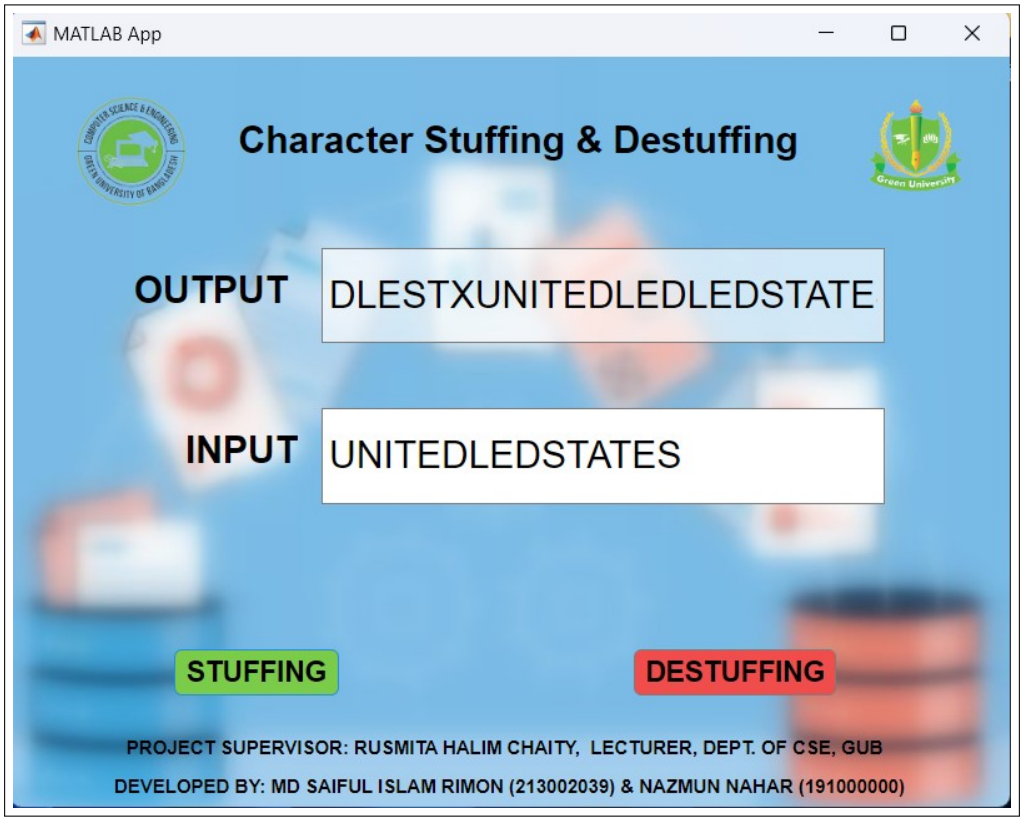

Figure 3.11: Graphical User Interface: Character Stuffing & Destuffing Page: Stuffed String

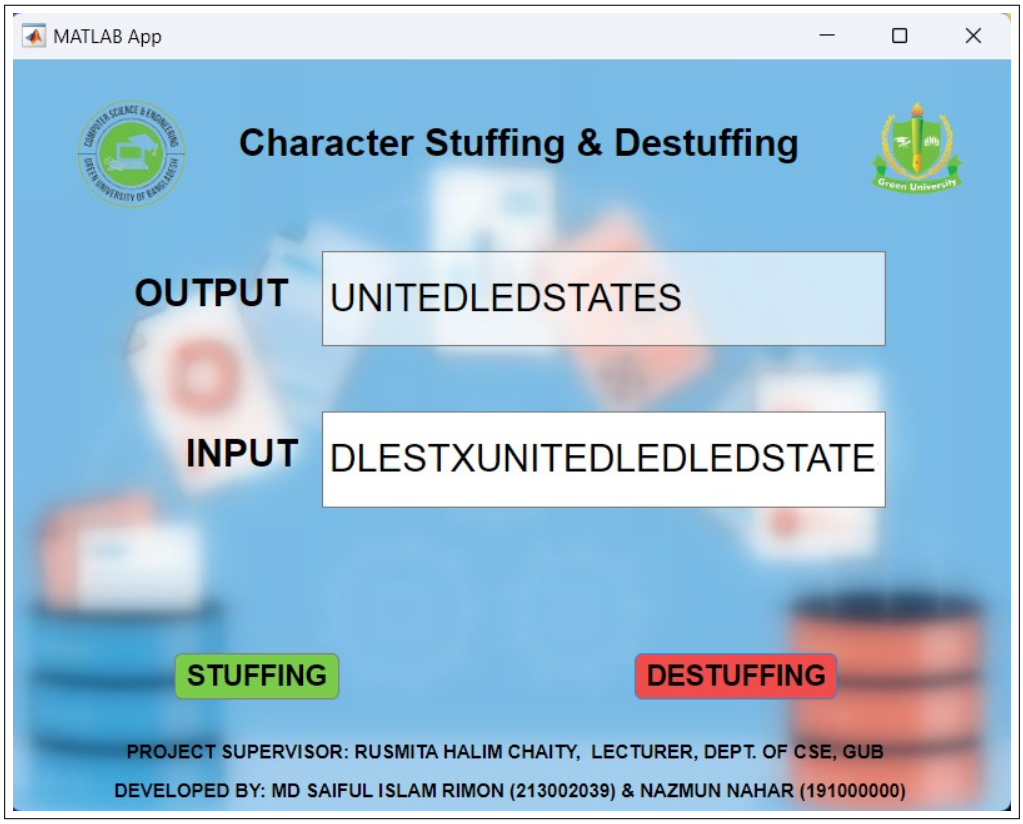

Figure 3.12: Graphical User Interface: Character Stuffing & Destuffing Page: Destuffed String

3.2.6 IPv4 Conversion

After clicking on the IPv4 Conversion button on the home page, we will see a graphical user interface like the picture below. Here, we can input the decimal IP, and convert it into binary IP and vice versa.

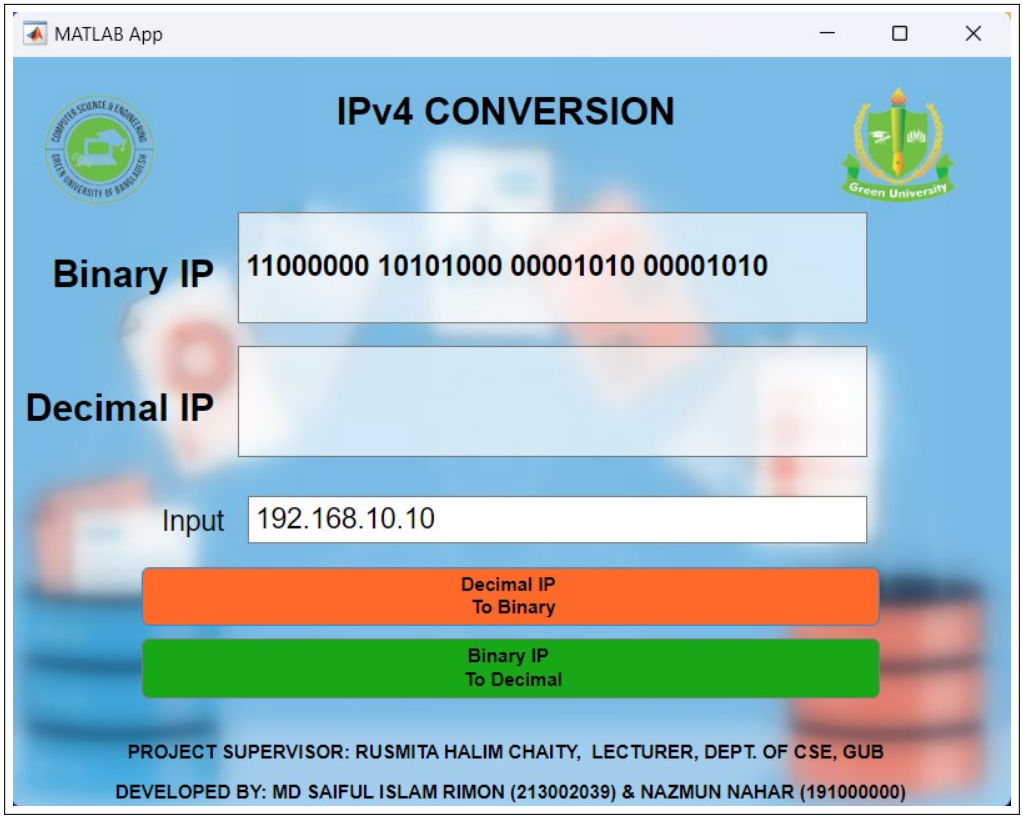

Figure 3.13: Graphical User Interface: IPv4 Conversion Page: Decimal to Binary

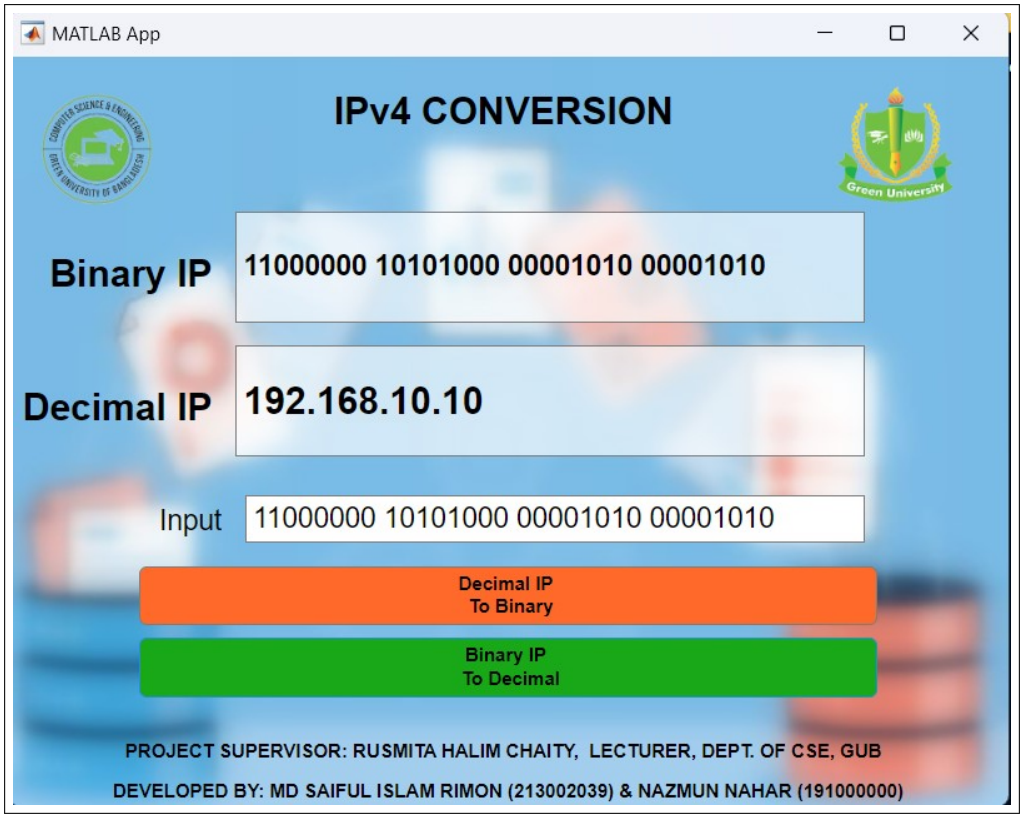

Figure 3.14: Graphical User Interface: IPv4 Conversion Page: Binary to Decimal

3.2.7 Cyclic Redundancy Check (CRC)

After clicking on the Cyclic Redundancy Check (CRC) button on the home page, we will see a graphical user interface like the picture below. Here, we can input the bit-string and the polynomial in binary, then it will generate the transmitted codeword. If the receiver codeword is same with the transmitted codeword, then will show "No Error Detected" and a provide a green light signal. If it is not same, then an error will found and provide a red light signal.

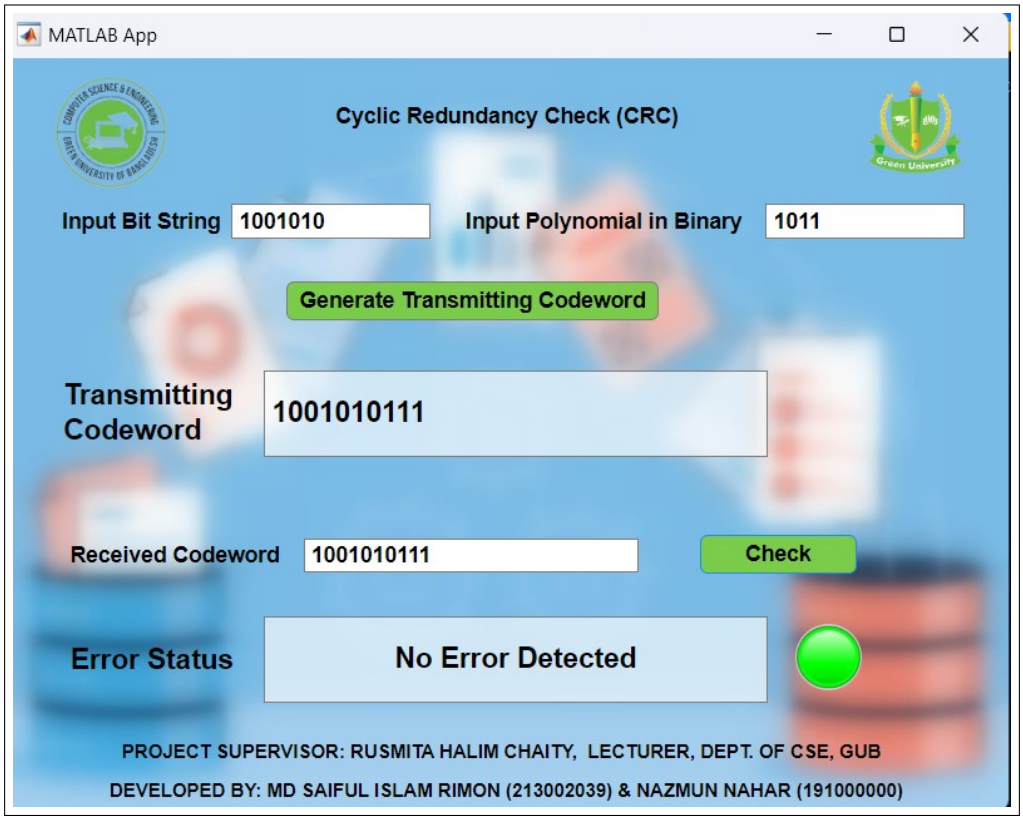

Figure 3.15: Graphical User Interface: Cyclic Redundancy Check (CRC): No Error Detected

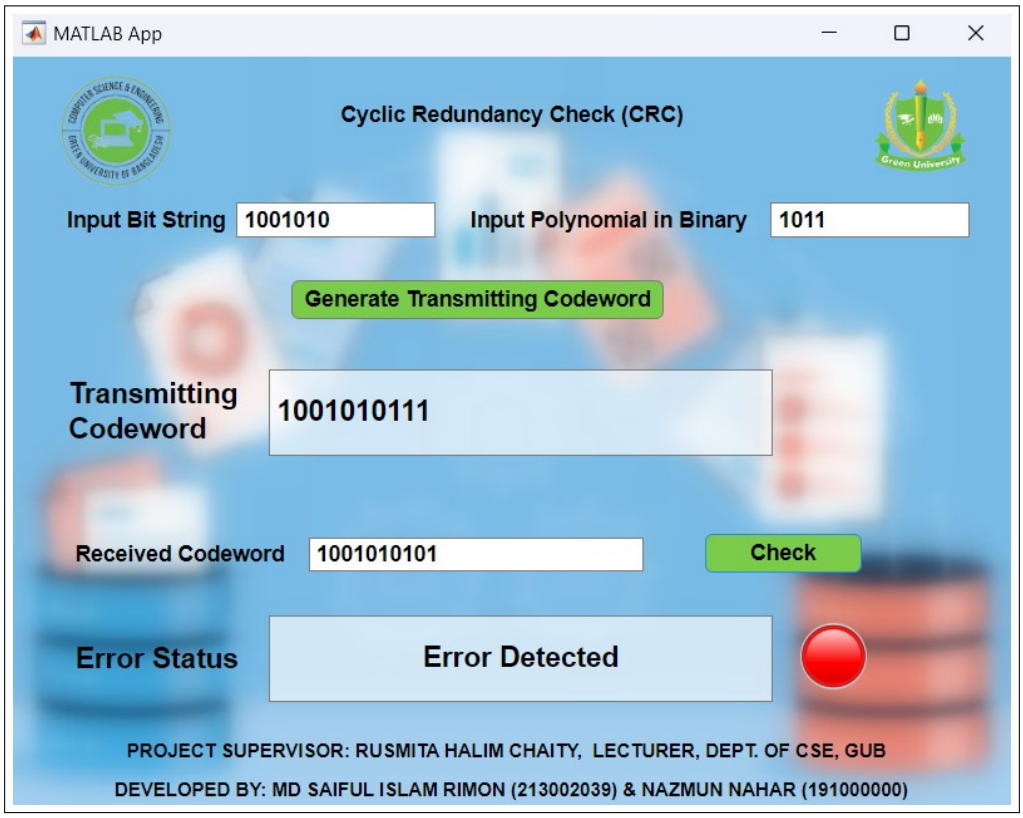

Figure 3.16: Graphical User Interface: Cyclic Redundancy Check (CRC): Error Detected

3.3 Results Overall Discussion

The project successfully delivers a comprehensive and user-friendly Data Transmission Simulator system. Its intuitive interface, combined with essential features like Line Coding, Hamming Encoding-Decoding, Analog to Digital Signal Conversion, Character Stuffing & Destuffing, IPv4 Conversion, Cyclic Redundancy Check (CRC)- demonstrates a well-thought-out design and implementation. The project stands as a robust solution to streamline and simplify data transmission tasks.

3.3.1 Complex Engineering Problem Discussion

The complex engineering problem presented in the project involves developing a Data Transmission Simulator using Matlab language. This task is intricate due to several factors discussed below:

1. Depth of Knowledge Required:

- **Understanding of Digital Communication Concepts:** The project encompasses various elements of digital communication, such as line coding, signal conversion, and error detection. A solid foundation in these topics is necessary to comprehend the underlying principles and objectives of each module.
- **Proficiency in MATLAB Programming:** Implementing these features requires an in-depth understanding of MATLAB's programming environment, including its GUI development tools (such as App Designer or GUIDE), built-in functions, and debugging techniques.
- **Algorithm Development Skills:** The project developer must be capable of translating communication theory into practical algorithms, which requires knowledge of data structures, control flow, and MATLAB's numerical capabilities.

2. Depth of Analysis Required:

- **Algorithmic Efficiency:** Each feature, like Hamming Encoding & Decoding or CRC Error Detection, must be analyzed for algorithmic efficiency. This involves choosing the most appropriate algorithms that balance time complexity and resource usage.
- **User Interface Design:** An analysis of the user interface is required to ensure that the application is intuitive and user-friendly. This includes the layout of input and output boxes, as well as the graphical demonstration system for signals.
- **Error Handling and Data Validation:** The developer needs to analyze the potential points of failure and implement robust error handling and data validation to ensure the application behaves correctly with unexpected or erroneous input.

3. Extent of Applicable Codes:

- **Modular Programming:** The code for each feature (a-g) must be modular, allowing for easy updates and maintenance. The extent of the code will include separate functions or classes for each feature, following MATLAB's best practices for code organization.
- **GUI Component Integration:** The extent of applicable codes also includes the integration of GUI components such as buttons, input boxes, and display panels. It requires understanding of callback functions and event-driven programming in MATLAB.
- **Data Processing and Visualization:** For the graphical demonstration system, the code must include data processing algorithms that can convert raw input into a visual format, using MATLAB's plotting functions and graphics system.
- **Compliance with Standards:** Where applicable, the code must adhere to industry standards for digital communication, particularly for encoding, decoding, and error detection methods.

Addressing these complex problems involves a multi-disciplinary approach that combines technical knowledge with analytical skills. The success of such a project lies in the ability to translate theoretical concepts into practical, working solutions that are both efficient and user-friendly.

Chapter 4

Conclusion

4.1 Discussion

This MATLAB project presents an effective user interface for exploring key digital communication techniques such as Line Coding, Hamming Encoding Decoding, and Analog to Digital Signal Conversion. The inclusion of error detection methods like Hamming codes and CRC is particularly beneficial for understanding data integrity in communication systems. Additionally, the tools for Character and Bit Stuffing Destuffing provide practical insights into data framing, crucial for network communications. The IPv4 Conversion feature aligns with current networking practices, enhancing the software's professional applicability.
The software's design, featuring clear input and output boxes and graphical signal demonstrations, ensures ease of use, making it accessible for learners at various levels. By combining educational value with user-friendly functionality, the program serves as a comprehensive tool for both students and professionals in digital communications.
Future iterations could expand on advanced topics and user feedback integration to further enrich learning experiences and practical application.

4.2 Limitations

1. **Bugs in Hamming Encoding & Decoding:** The current Hamming Encoding & Decoding used in this application has several bugs on the receiver side's parity bit and checking the position of error and correction of the bitstream.
2. **Character Stuffing & Destuffing:** The starting flag and ending flag is fixed here. That's mean, the user cannot set it. And the escaped code is also fixed here, which user cannot change. Also, there are some bugs in destuffing the string.
3. **Error Handling and Validation:** The program has limited error handling capabilities, particularly in dealing with invalid user inputs or overflow scenarios, which could lead to system crashes or incorrect processing.

4. **Hardware and Software Dependency:** Being written in Matlab language, the application is highly dependent on specific software environments, limiting its portability.

4.3 Scope of Future Work

1. **Development of the existing Graphical User Interface (GUI):** Enhance the currently running GUI would greatly enhance user experience and accessibility, making the system more appealing and easier to navigate.

2. **Fixing the algorithmic bugs:** Fixing the algorithmic bugs of Hamming Encoding & Decoding, Character Stuffing & Destuffing and others.

3. **Robust Error Handling and Input Validation:** Developing more sophisticated error handling and validation mechanisms to prevent crashes and ensure system stability, especially under erroneous or unexpected inputs.

4. **Incorporating Advanced Reporting and Analytics:** Adding functionality for detailed reporting and analytics would enable better tracking of parking patterns, financial management, and strategic planning.

5. **Cross-Platform Compatibility:** Rewriting or adapting the code to be compatible with multiple platforms and architectures would significantly increase the system's applicability and reach.

6. **Integration with Additional Services:** Future developments could include integration online version, or mobile version to enhance functionality and user experience.

References