1/17/2025

Andreea Diana Potra

CAG

MATLAB PROJECT

Diode-Resistor Circuit

This project revolves around interactive simulation and visualization of a diode-resistor circuit, combining a user-friendly GUI with dynamic parameter adjustments, waveform plotting, and voltage transfer characteristic analysis

# Introduction

MATLAB (MATrix LABoratory) is a versatile, high-level programming environment designed for numerical computation, data analysis, visualization, and algorithm development. Created by Cleve Moler in the late 1970s, MATLAB has become a cornerstone in engineering, physics, and applied sciences. Its extensive libraries, combined with powerful built-in functions, allow users to model and simulate complex systems effectively, including electronic circuits. For tasks involving control systems, signal processing, and dynamic simulations, MATLAB offers unmatched flexibility and ease of use.

Electronics is the branch of science and technology that studies and applies the controlled movement of electrons to create devices and systems. From communication technologies to computing and power management, electronics serves as the backbone of modern civilization. A major focus in electronics is the analysis and design of circuits, which serve as the medium for manipulating electrical signals for various purposes.

# A Brief History of Electrical Circuits

The study of electrical circuits has its roots in the 19th century, with foundational contributions by scientists such as:

* **Alessandro Volta** (1800): Developed the first electric battery.
* **Georg Simon Ohm** (1827): Formulated Ohm’s Law, describing the relationship between voltage, current, and resistance.
* **Michael Faraday** (1831): Demonstrated electromagnetic induction, leading to the development of generators and transformers.

The 20th century witnessed exponential growth with the advent of:

* **Vacuum Tubes** (1904): Enabled amplification and rectification of electrical signals.
* **Transistors** (1947): Revolutionized electronics by providing compact, reliable alternatives to vacuum tubes.
* **Integrated Circuits (ICs)** (1958): Combined multiple components on a single chip, reducing size and cost while enhancing performance.

Today, electronic circuits range from basic analog designs to sophisticated digital systems embedded in every aspect of modern technology.

# The Diode -Resistor Circuit

* ***Historical Background***

The diode, a key component in electronic circuits, emerged in the early 20th century with the development of vacuum-tube diodes by John Ambrose Fleming (1904). The invention of semiconductor diodes in the mid-20th century, particularly silicon-based models, revolutionized electronics by providing efficient, compact solutions for rectification, clipping, and voltage regulation.

The resistor, another critical component, has been essential to circuit design since the study of resistive materials in the 19th century. Together, diodes and resistors form the basis for countless applications, from simple signal rectifiers to complex power systems.

* ***Components of the Circuit***

**Diode**:

Allows current flow in one direction, offering properties like rectification and signal clipping.

Models used in this project:

1. **Ideal Diode**: Assumes perfect conduction without voltage drop.
2. **Constant Voltage Drop Model**: Accounts for a small voltage drop (e.g., 0.7V for silicon diodes).
3. **Resistor**: A passive component that limits current flow, enabling voltage regulation and current control.

**Place of This Project in the Field**

This project examines the **Voltage Transfer Characteristics (VTC)** and **Waveform** behavior of a diode-resistor circuit, a crucial concept in understanding nonlinear circuit dynamics. It provides an interactive simulation to study:

-Signal rectification and clipping behavior.

-Voltage transfer and its dependence on diode models.

-The impact of varying sine wave parameters like amplitude and frequency on circuit behavior.

**Context and Relevance**

The diode-resistor circuit is a simplified model that underpins more advanced concepts, such as:

**Rectifiers**: Used in power supplies to convert AC to DC.

**Clipping Circuits**: Employed in signal conditioning to restrict

voltage levels.

**Voltage Regulators**: Found in every electronic device to

maintain stable output voltage.

By focusing on fundamental principles, this project bridges the gap between theoretical understanding and practical application. It offers a learning platform for beginners and a simulation tool for advanced studies in nonlinear circuit behavior.

**Theoretical Basis**

1. **Diode Characteristics**:
   * A diode has two primary states:
     1. **Forward Bias**: Current flows through the diode when the voltage across it exceeds a threshold value ( VTh , typically 0.7V for silicon diodes).
     2. **Reverse Bias**: The diode blocks current flow when the voltage is below ​, except for a small leakage current.
   * The ideal diode equation is given by:

Where:

* : saturation current
* V : voltage across the diode
* n : Ideality factor
* Vt : thermal voltage

2. **Resistor Behavior**:

* A resistor follows Ohm’s Law:

where:

* + VR​: Voltage across the resistor.
  + IR: Current through the resistor.
  + R: Resistance value in ohms (Ω).

1. **Input Voltage**:

* A sine wave input is typically applied to study the circuit:

where:

* + A: Amplitude of the input signal.
  + f: Frequency of the input signal.

**Circuit Configurations**

1. **Ideal Diode Model**:
   * The diode is modeled as a perfect conductor in forward bias and an open circuit in reverse bias.
   * Output Voltage:

This configuration clips the negative half of the sine wave, producing a rectified output.

2. **Diode with Constant Voltage Drop**:

* A real diode introduces a voltage drop ( VTh​) even in forward bias.
* Output Voltage:

This results in a clipped and shifted waveform.

3. **Load Resistor**:

* The resistor affects the current flow and voltage distribution in the circuit. The output voltage across the load resistor is influenced by both the diode's behavior and the applied input signal.

# Experimental Results

**Case 1 : Forword Biased - Ideal Diode**

Ex :

The formula state that , which mean .

**Case 2: Reverse Biased – Ideal Diode**

Ex :

The formula state that , which mean

**Case 3:Forword Biased – Constant Voltage Drop**

The formula state that ,which mean .

**Case 4:Reverse Biased – Constant Voltage Drop**

The formula state that ,which mean

In order to show the functionality of the project ,I will take more values into consideration (for the Output the first value will show the ideal diode model and the second one ,the constant voltage drop diode model results) :

|  |  |  |
| --- | --- | --- |
| *Input Voltage*  *(Vin)* | *Expected Output*  *(expected Vout)* | *Measured Output*  *(real Vout)* |
| *Vin= 22V* | Vout =22V (ideal)  Vout=21,3(voltage drop) | Vout=21,82V(ideal)  Vout=21,3(voltage drop) |
| *Vin=-3V* | *Vout=0V(ideal)*  *Vout=0V(voltage drop)* | *Vout=0(ideal)*  *Vout=0(voltage drop)* |
| *Vin=0.5V* | Vout=0.5V(ideal)  Vout=0V(voltage drop) | Vout=0.499V(ideal)  Vout=0V(voltage drop) |
| *Vin=0V* | Vout=0V(ideal)  Vout=0V(voltage drop) | Vout=0v(ideal)  Vout=0V(voltage drop) |

# What does the code do?

1. *GUI initialization:*

|  |  |
| --- | --- |
| Fig = figure('Name', 'Diode-Resistor Circuit', ...  'Units', 'normalized', ...  'Position', [0.2, 0.2, 0.6, 0.6]); | Creates a main window named **"*Diode-Resistor Circuit*"** with normalized dimensions, occupying 60% of the screen both horizontally and vertically |
| backgroundImage = 'download.jpg';  ax = axes('Parent', Fig, 'Position', [0 0 1 1]);  imshow(backgroundImage, 'Parent', ax);  set(ax, 'Visible', 'off'); | Loads and displays an image (download.jpg) as the background of the GUI, filling the entire window |
| uicontrol('Style', 'text', ...  'Units', 'normalized', ...  'Position', [0.3 0.85 0.4 0.1], ...  'String', 'Diode-Resistor Circuit', ...  'FontSize', 18, ...  'FontWeight', 'bold', ...  'ForegroundColor', [1 0.75 0.8], ...  'BackgroundColor', [0.1 0.1 0.3]); | Adds a centered title displaying **"*Diode-Resistor Circuit*"** in bold with a dark background and pink text. |
| uicontrol('Style', 'pushbutton', ...  'Units', 'normalized', ...  'Position', [0.2 0.7 0.6 0.1], ...  'String', 'Documentation', ...  'FontSize', 12, ...  'Callback', 'documentation();'); | Opens a **Documentation.docx** file |
| uicontrol('Style', 'pushbutton', ...  'Units', 'normalized', ...  'Position', [0.2 0.55 0.6 0.1], ...  'String', 'Characteristics', ...  'Callback', 'showCharacteristics();'); | Launches another GUI window for exploring circuit characteristics |
| uicontrol('Style', 'pushbutton', ...  'Units', 'normalized', ...  'Position', [0.80 0.03 0.1 0.06], ...  'String', 'Close', ...  'Callback', 'close();'); | Closes the GUI window when clicked |

1. *Characteristics Window :*

|  |  |
| --- | --- |
| Fig2 = figure('Name', "DR Circuit", ...  'Units', 'normalized', ...  'Position', [.3 .3 .6 .6], ...  'NumberTitle', 'off', ...  'Color', [0.94, 0.9, 0.96]); | A new figure window is created. |
| uicontrol('Style', 'text', ...  'Units', 'normalized', ...  'Position', [0.15 0.90 0.5 0.08], ...  'String', 'DR Circuit', ...  'FontSize', 30, ...  'FontWeight', 'bold', ...  'FontName', 'Georgia', ...  'ForegroundColor', [0.5, 0, 0.5], ...  'BackgroundColor', [0.94, 0.9, 0.96]); | Adds a large, bold title at the top of the window displaying "DR Circuit" |
| imagePath = 'DR\_circuit.jpg';  img = imread(imagePath);  ax = axes('Parent', Fig2, ...  'Position', [0.28, 0.35, 0.28, 0.39]);  imshow(img, 'Parent', ax);  set(ax, 'XTick', [], 'YTick', []); | Loads an image of the circuit diagram DR\_circuit.jpg and displays it in an axes control. |

|  |  |
| --- | --- |
| sineWavePanel = uipanel('Parent', Fig2, 'Title', 'Sine Wave', ...  'FontSize', 12, ...  'FontWeight', 'bold', ...  'Position', [0.08, 0.6, 0.15, 0.3], ...  'BackgroundColor', [0.9, 0.8, 1], ...  'ForegroundColor', [0.5, 0.0, 0.5]); | A panel labeled "Sine Wave" is placed on the left side.  This panel allows users to adjust the sine wave parameters: Amplitude (A), Frequency (f), and Cycles (n). |
| uicontrol('Style', 'text', ...  'Parent', sineWavePanel, ...  'Position', [0.05 0.8 0.3 0.1], ...  'String', 'A:', ...  'FontSize', 10, ...  'ForegroundColor', [0.5, 0.0, 0.5], ...  'BackgroundColor', [0.9, 0.8, 1]);  editA = uicontrol('Style', 'edit', ...  'Parent', sineWavePanel, ...  'Position', [0.4 0.8 0.5 0.1], ...  'BackgroundColor', 'white', ...  'FontSize', 9, ...  'String', '5', ...  'Callback', @(src, event) updatePlot(dropdown)); | Displays a label "A:" and an editable field “editA” for the amplitude. |
| uicontrol('Style', 'text', ...  'Parent', sineWavePanel, ...  'Position', [0.05 0.6 0.3 0.1], ...  'String', 'f:', ...  'FontSize', 10);  editF = uicontrol('Style', 'edit', ...  'Parent', sineWavePanel, ...  'Position', [0.4 0.6 0.5 0.1], ...  'BackgroundColor', 'white', ...  'String', '50', ...  'Callback', @(src, event) updatePlot(dropdown)); | Adds a similar input for frequency “f” with a default value of 50 Hz. |
| uicontrol('Style', 'text', ...  'Parent', sineWavePanel, ...  'Position', [0.05 0.4 0.3 0.1], ...  'String', 'n:', ...  'FontSize', 10);  editN = uicontrol('Style', 'edit', ...  'Parent', sineWavePanel, ...  'Position', [0.4 0.4 0.5 0.1], ...  'String', '3', ...  'Callback', @(src, event) updatePlot(dropdown)); | Adds an input for the number of cycles “n” with a default value of 3. |

|  |  |
| --- | --- |
| diodePanel = uipanel('Parent', Fig2, 'Title', 'Diode Model', ...  'FontSize', 12, ...  'FontWeight', 'bold', ...  'Position', [0.08, 0.3, 0.15, 0.25], ...  'BackgroundColor', [0.9, 0.8, 1]); | A panel labeled "Diode Model" is added below the Sine Wave panel. |
| dropdown = uicontrol('Style', 'popupmenu', ...  'Parent', diodePanel, ...  'String', {'<none>', 'Ideal', 'Constant voltage drop'}, ...  'Units', 'normalized', ...  'Position', [0.2, 0.4, 0.6, 0.1], ...  'Callback', @(src, event) updatePlot(src)); | Dropdown menu to select a diode model:  <none>: No diode behavior.  Ideal: Ideal diode model.  Constant voltage drop: Includes a 0.7V threshold.  Triggers the updatePlot function when changed. |

|  |  |
| --- | --- |
| ax\_vtc = axes('Parent', Fig2, 'Position', [0.62, 0.58, 0.35, 0.3]);  plot(ax\_vtc, vin, vout);  xlabel(ax\_vtc, 'Input Voltage (V)');  ylabel(ax\_vtc, 'Output Voltage (V)');  title(ax\_vtc, 'Voltage Transfer Characteristic (VTC)'); | Displays the relationship between input voltage (vin) and output voltage (vout) based on the selected diode model. |
| plotPanel = uipanel('Parent', Fig2, 'Title', 'Waveforms', ...  'Position', [0.59, 0.05, 0.4, 0.45]);  ax1 = axes('Parent', plotPanel, 'Position', [0.1, 0.6, 0.8, 0.35]);  plot(ax1, t, vin);  ax2 = axes('Parent', plotPanel, 'Position', [0.1, 0.15, 0.8, 0.35]);  plot(ax2, t, vout); | Two subplots:  Top Plot (ax1): Time-domain input signal (vin).  Bottom Plot (ax2): Time-domain output signal (vout). |
| returnButton = uicontrol('Style', 'pushbutton', ...  'Parent', Fig2, ...  'Position', [0.08 0.2 0.15 0.05], ...  'String', 'Return', ...  'Callback', @(src, event) resetParameters());  uicontrol('Style', 'pushbutton', ...  'Parent', Fig2, ...  'Position', [0.1 0.09 0.1 0.06], ...  'String', 'Close', ...  'Callback', 'close();'); | **Return Button:** Resets parameters to defaults and updates plots.  **Close Button:** Closes the characteristics window |

|  |  |
| --- | --- |
| Existing errors | Unrecognized function or variable 'dropdown'.  Error in showCharacteristics>@(src,event)updatePlot(dropdown) (line 77)  'Callback', @(src, event) updatePlot(dropdown));  ^^^^^^^^    Error while evaluating UIControl Callback. |

Main window

Documentation

Window

Close

Characteristics

window

# Conclusion

This project focused on designing and implementing a graphical interface for simulating and analyzing a diode rectifier (DR) circuit. The primary objective was to provide users with an interactive environment to visualize the circuit’s behavior, study its voltage transfer characteristics, and experiment with input parameters such as sine wave amplitude, frequency, and diode models.

The expected results were largely achieved, with the interface successfully allowing users to adjust parameters and observe the corresponding circuit behavior in real time. The inclusion of a voltage transfer characteristic (VTC) plot and waveform analysis provided an intuitive way to explore the circuit’s operation under different conditions.

This project has potential applications in educational settings for students and educators studying rectifier circuits. It could also be extended for professional use in prototyping and testing similar circuits in an interactive, visual manner. Further development could expand the tool’s capabilities to include other electronic components and circuit configurations, making it a more comprehensive simulation platform.

Contents

[Introduction 1](#_Toc187960127)

[A Brief History of Electrical Circuits 1](#_Toc187960128)

[The Diode -Resistor Circuit 2](#_Toc187960129)

[Experimental Results 6](#_Toc187960130)

[What does the code do? 7](#_Toc187960131)

[Conclusion 15](#_Toc187960132)

[References 17](#_Toc187960133)

# References

Book: MATLAB for Students - Mihaela Cârligea ,Paul Farago

<https://en.wikipedia.org/wiki/MATLAB>

<https://en.wikipedia.org/wiki/Electronics>

<https://en.wikipedia.org/wiki/Diode>

<https://en.wikipedia.org/wiki/Resistor>

Appendix

%Cag\_project.m

Fig = figure('Name', 'Diode-Resistor Circuit', ...

'Units', 'normalized', ...

'Position', [0.2, 0.2, 0.6, 0.6]);

backgroundImage = 'download.jpg';

ax = axes('Parent', Fig, 'Position', [0 0 1 1]);

imshow(backgroundImage, 'Parent', ax);

set(ax, 'Visible', 'off');

uicontrol('Style', 'text', ...

'Units', 'normalized', ...

'Position', [0.3 0.85 0.4 0.1], ...

'String', 'Diode-Resistor Circuit', ...

'FontSize', 18, ...

'FontWeight', 'bold', ...

'ForegroundColor', [1 0.75 0.8], ...

'BackgroundColor', [0.1 0.1 0.3]);

uicontrol('Style', 'pushbutton', ...

'Units', 'normalized', ...

'Position', [0.2 0.7 0.6 0.1], ...

'String', 'Documentation', ...

'FontSize', 12, ...

'BackgroundColor', [0.1 0.1 0.3], ...

'ForegroundColor', [1 0.75 0.8], ...

'Callback', 'documentation();');

uicontrol('Style', 'pushbutton', ...

'Units', 'normalized', ...

'Position', [0.2 0.55 0.6 0.1], ...

'String', 'Characteristics', ...

'FontSize', 12, ...

'BackgroundColor', [0.1 0.1 0.3], ...

'ForegroundColor', [1 0.75 0.8], ...

'Callback', 'showCharacteristics();');

uicontrol('Style', 'pushbutton', ...

'Units', 'normalized', ...

'FontWeight', 'bold', ...

'Fontsize', 12, ...

'BackgroundColor', [1 0.75 0.8], ...

'Position', [0.80 0.03 0.1 0.06], ...

'String', 'Close', ...

'Callback', 'close();');

%documentation.m

function documentation

open('Documentation.docx');

end

%showCharacteristics.m

function showCharacteristics(~, ~)

function [t, vout, vin] = dr(idx, A, f, n, VBias)

if nargin == 4

VBias = 0;

end

T = 1 / f;

stepsize = T / 100;

t = 0:stepsize:n\*T;

vin = A \* sin(2 \* pi \* f \* t);

switch idx

case 2

vout = vin;

vout(vin < VBias) = VBias;

case 3

vout = vin - 0.7;

vout(vin < VBias + 0.7) = VBias;

otherwise

disp('Unknown option');

vout = zeros(size(vin));

end

end

A = 5; f = 50; n = 3; idx = 1;

[t, vout, vin] = dr(idx, A, f, n);

Fig2 = figure('Name', "DR Circuit", ...

'Units', 'normalized', ...

'Position', [.3 .3 .6 .6], ...

'NumberTitle', 'off', ...

'Color', [0.94, 0.9, 0.96]);

uicontrol('Style', 'text', ...

'Units', 'normalized', ...

'Position', [0.15 0.90 0.5 0.08], ...

'String', 'DR Circuit', ...

'FontSize', 30, ...

'FontWeight', 'bold', ...

'FontName', 'Georgia', ...

'ForegroundColor', [0.5, 0, 0.5], ...

'BackgroundColor', [0.94, 0.9, 0.96]);

imagePath = 'DR\_circuit.jpg';

img = imread(imagePath);

ax = axes('Parent', Fig2, ...

'Position', [0.28, 0.35, 0.28, 0.39]);

imshow(img, 'Parent', ax);

set(ax, 'XTick', [], 'YTick', []);

sineWavePanel = uipanel('Parent', Fig2, 'Title', 'Sine Wave', ...

'FontSize', 12, ...

'FontWeight', 'bold', ...

'Position', [0.08, 0.6, 0.15, 0.3], ...

'BackgroundColor', [0.9, 0.8, 1], ...

'ForegroundColor', [0.5, 0.0, 0.5]);

uicontrol('Style', 'text', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.05 0.8 0.3 0.1], ...

'String', 'A:', ...

'FontSize', 10, ...

'HorizontalAlignment', 'right', ...

'ForegroundColor', [0.5, 0.0, 0.5], ...

'BackgroundColor', [0.9, 0.8, 1]);

editA = uicontrol('Style', 'edit', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.4 0.8 0.5 0.1], ...

'BackgroundColor', 'white', ...

'FontSize', 9, ...

'String', '5', ...

'Callback', @(src, event) updatePlot(dropdown));

uicontrol('Style', 'text', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.05 0.6 0.3 0.1], ...

'String', 'f:', ...

'FontSize', 10, ...

'HorizontalAlignment', 'right', ...

'ForegroundColor', [0.5, 0.0, 0.5], ...

'BackgroundColor', [0.9, 0.8, 1]);

editF = uicontrol('Style', 'edit', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.4 0.6 0.5 0.1], ...

'BackgroundColor', 'white', ...

'FontSize', 9, ...

'String', '50', ...

'Callback', @(src, event) updatePlot(dropdown));

uicontrol('Style', 'text', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.05 0.4 0.3 0.1], ...

'String', 'n:', ...

'FontSize', 10, ...

'HorizontalAlignment', 'right', ...

'ForegroundColor', [0.5, 0.0, 0.5], ...

'BackgroundColor', [0.9, 0.8, 1]);

editN = uicontrol('Style', 'edit', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.4 0.4 0.5 0.1], ...

'BackgroundColor', 'white', ...

'FontSize', 9, ...

'String', '3', ...

'Callback', @(src, event) updatePlot(dropdown));

diodePanel = uipanel('Parent', Fig2, 'Title', 'Diode Model', ...

'FontSize', 12, ...

'FontWeight', 'bold', ...

'Position', [0.08, 0.3, 0.15, 0.25], ...

'BackgroundColor', [0.9, 0.8, 1], ...

'ForegroundColor', [0.5, 0.0, 0.5]);

dropdown = uicontrol('Style', 'popupmenu', ...

'Parent', diodePanel, ...

'String', {'<none>', 'Ideal', 'Constant voltage drop'}, ...

'Units', 'normalized', ...

'Position', [0.2, 0.4, 0.6, 0.1], ...

'Callback', @(src, event) updatePlot(src));

returnButton = uicontrol('Style', 'pushbutton', ...

'Parent', Fig2, ...

'Units', 'normalized', ...

'Position', [0.08 0.2 0.15 0.05], ...

'String', 'Return', ...

'FontSize', 12, ...

'BackgroundColor', [0.1 0.1 0.3], ...

'ForegroundColor', [1 0.75 0.8], ...

'Callback', @(src, event) resetParameters());

uicontrol('Style', 'pushbutton', ...

'Units', 'normalized', ...

'FontWeight', 'bold', ...

'Fontsize', 12, ...

'BackgroundColor', [1 0.75 0.8], ...

'Position', [0.1 0.09 0.1 0.06], ...

'String', 'Close', ...

'Callback', 'close();');

ax\_vtc = axes('Parent', Fig2, 'Position', [0.62, 0.58, 0.35, 0.3]);

plot(ax\_vtc, vin, vout);

xlabel(ax\_vtc, 'Input Voltage (V)');

ylabel(ax\_vtc, 'Output Voltage (V)');

title(ax\_vtc, 'Voltage Transfer Characteristic (VTC)');

title(ax\_vtc, 'Voltage Transfer Characteristic (VTC)', ...

'Color', [0.4, 0.1, 0.2], ...

'FontSize', 12, ...

'FontWeight', 'bold');

plotPanel = uipanel('Parent', Fig2, 'Title', 'Waveforms', ...

'Position', [0.59, 0.05, 0.4, 0.45], ...

'BackgroundColor', [0.9, 0.8, 1], ...

'ForegroundColor', [0.4, 0.1, 0.2]);

ax1 = axes('Parent', plotPanel, 'Position', [0.1, 0.6, 0.8, 0.35]);

plot(ax1, t, vin);

ylabel(ax1, "v\_I\_N", 'Color', [0.1, 0.3, 0.7]);

xlabel(ax1, 'Time', 'Color', [0.1, 0.3, 0.7]);

ax2 = axes('Parent', plotPanel, 'Position', [0.1, 0.15, 0.8, 0.35]);

plot(ax2, t, vout);

ylabel(ax2, "v\_O\_U\_T", 'Color', [0.1, 0.3, 0.7]);

xlabel(ax2, 'Time', 'Color', [0.1, 0.3, 0.7]);

function updatePlot(dropdown)

idx = dropdown.Value;

A = str2double(editA.String);

f = str2double(editF.String);

n = str2double(editN.String);

[t, vout, vin] = dr(idx, A, f, n);

plot(ax1, t, vin);

plot(ax2, t, vout);

plot(ax\_vtc, vin, vout);

title(ax\_vtc, 'Voltage Transfer Characteristic (VTC)', ...

'Color', [0.4, 0.1, 0.2], ...

'FontSize', 12, ...

'FontWeight', 'bold');

xlabel(ax\_vtc, 'Input Voltage (V)', 'Color', [0.1, 0.3, 0.7]);

ylabel(ax\_vtc, 'Output Voltage (V)', 'Color', [0.1, 0.3, 0.7]);

xlabel(ax1, 'Time', 'Color', [0.1, 0.3, 0.7]);

ylabel(ax1, 'v\_I\_N', 'Color', [0.1, 0.3, 0.7]);

xlabel(ax2, 'Time', 'Color', [0.1, 0.3, 0.7]);

ylabel(ax2, 'v\_O\_U\_T', 'Color', [0.1, 0.3, 0.7]);

end

function resetParameters()

dropdown.Value = 1;

editA.String = '5';

editF.String = '50';

editN.String = '3';

updatePlot(dropdown);

end

end

function updatePlot(dropdown)

idx = dropdown.Value;

A = str2double(editA.String);

f = str2double(editF.String);

n = str2double(editN.String);

function showCharacteristics(~, ~)

function [t, vout, vin] = dr(idx, A, f, n, VBias)

if nargin == 4

VBias = 0;

end

T = 1 / f;

stepsize = T / 100;

t = 0:stepsize:n\*T;

vin = A \* sin(2 \* pi \* f \* t);

switch idx

case 2

vout = vin;

vout(vin < VBias) = VBias;

case 3

vout = vin - 0.7;

vout(vin < VBias + 0.7) = VBias;

otherwise

disp('Unknown option');

vout = zeros(size(vin));

end

end

A = 5; f = 50; n = 3; idx = 1;

[t, vout, vin] = dr(idx, A, f, n);

Fig2 = figure('Name', "DR Circuit", ...

'Units', 'normalized', ...

'Position', [.3 .3 .6 .6], ...

'NumberTitle', 'off', ...

'Color', [0.94, 0.9, 0.96]);

uicontrol('Style', 'text', ...

'Units', 'normalized', ...

'Position', [0.15 0.90 0.5 0.08], ...

'String', 'DR Circuit', ...

'FontSize', 30, ...

'FontWeight', 'bold', ...

'FontName', 'Georgia', ...

'ForegroundColor', [0.5, 0, 0.5], ...

'BackgroundColor', [0.94, 0.9, 0.96]);

imagePath = 'DR\_circuit.jpg';

img = imread(imagePath);

ax = axes('Parent', Fig2, ...

'Position', [0.28, 0.35, 0.28, 0.39]);

imshow(img, 'Parent', ax);

set(ax, 'XTick', [], 'YTick', []);

sineWavePanel = uipanel('Parent', Fig2, 'Title', 'Sine Wave', ...

'FontSize', 12, ...

'FontWeight', 'bold', ...

'Position', [0.08, 0.6, 0.15, 0.3], ...

'BackgroundColor', [0.9, 0.8, 1], ...

'ForegroundColor', [0.5, 0.0, 0.5]);

uicontrol('Style', 'text', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.05 0.8 0.3 0.1], ...

'String', 'A:', ...

'FontSize', 10, ...

'HorizontalAlignment', 'right', ...

'ForegroundColor', [0.5, 0.0, 0.5], ...

'BackgroundColor', [0.9, 0.8, 1]);

editA = uicontrol('Style', 'edit', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.4 0.8 0.5 0.1], ...

'BackgroundColor', 'white', ...

'FontSize', 9, ...

'String', '5', ...

'Callback', @(src, event) updatePlot(dropdown));

uicontrol('Style', 'text', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.05 0.6 0.3 0.1], ...

'String', 'f:', ...

'FontSize', 10, ...

'HorizontalAlignment', 'right', ...

'ForegroundColor', [0.5, 0.0, 0.5], ...

'BackgroundColor', [0.9, 0.8, 1]);

editF = uicontrol('Style', 'edit', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.4 0.6 0.5 0.1], ...

'BackgroundColor', 'white', ...

'FontSize', 9, ...

'String', '50', ...

'Callback', @(src, event) updatePlot(dropdown));

uicontrol('Style', 'text', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.05 0.4 0.3 0.1], ...

'String', 'n:', ...

'FontSize', 10, ...

'HorizontalAlignment', 'right', ...

'ForegroundColor', [0.5, 0.0, 0.5], ...

'BackgroundColor', [0.9, 0.8, 1]);

editN = uicontrol('Style', 'edit', ...

'Parent', sineWavePanel, ...

'Units', 'normalized', ...

'Position', [0.4 0.4 0.5 0.1], ...

'BackgroundColor', 'white', ...

'FontSize', 9, ...

'String', '3', ...

'Callback', @(src, event) updatePlot(dropdown));

diodePanel = uipanel('Parent', Fig2, 'Title', 'Diode Model', ...

'FontSize', 12, ...

'FontWeight', 'bold', ...

'Position', [0.08, 0.3, 0.15, 0.25], ...

'BackgroundColor', [0.9, 0.8, 1], ...

'ForegroundColor', [0.5, 0.0, 0.5]);

dropdown = uicontrol('Style', 'popupmenu', ...

'Parent', diodePanel, ...

'String', {'<none>', 'Ideal', 'Constant voltage drop'}, ...

'Units', 'normalized', ...

'Position', [0.2, 0.4, 0.6, 0.1], ...

'Callback', @(src, event) updatePlot(src));

returnButton = uicontrol('Style', 'pushbutton', ...

'Parent', Fig2, ...

'Units', 'normalized', ...

'Position', [0.08 0.2 0.15 0.05], ...

'String', 'Return', ...

'FontSize', 12, ...

'BackgroundColor', [0.1 0.1 0.3], ...

'ForegroundColor', [1 0.75 0.8], ...

'Callback', @(src, event) resetParameters());

uicontrol('Style', 'pushbutton', ...

'Units', 'normalized', ...

'FontWeight', 'bold', ...

'Fontsize', 12, ...

'BackgroundColor', [1 0.75 0.8], ...

'Position', [0.1 0.09 0.1 0.06], ...

'String', 'Close', ...

'Callback', 'close();');

ax\_vtc = axes('Parent', Fig2, 'Position', [0.62, 0.58, 0.35, 0.3]);

plot(ax\_vtc, vin, vout);

xlabel(ax\_vtc, 'Input Voltage (V)');

ylabel(ax\_vtc, 'Output Voltage (V)');

title(ax\_vtc, 'Voltage Transfer Characteristic (VTC)');

title(ax\_vtc, 'Voltage Transfer Characteristic (VTC)', ...

'Color', [0.4, 0.1, 0.2], ...

'FontSize', 12, ...

'FontWeight', 'bold');

plotPanel = uipanel('Parent', Fig2, 'Title', 'Waveforms', ...

'Position', [0.59, 0.05, 0.4, 0.45], ...

'BackgroundColor', [0.9, 0.8, 1], ...

'ForegroundColor', [0.4, 0.1, 0.2]);

ax1 = axes('Parent', plotPanel, 'Position', [0.1, 0.6, 0.8, 0.35]);

plot(ax1, t, vin);

ylabel(ax1, "v\_I\_N", 'Color', [0.1, 0.3, 0.7]);

xlabel(ax1, 'Time', 'Color', [0.1, 0.3, 0.7]);

ax2 = axes('Parent', plotPanel, 'Position', [0.1, 0.15, 0.8, 0.35]);

plot(ax2, t, vout);

ylabel(ax2, "v\_O\_U\_T", 'Color', [0.1, 0.3, 0.7]);

xlabel(ax2, 'Time', 'Color', [0.1, 0.3, 0.7]);

function updatePlot(dropdown)

idx = dropdown.Value;

A = str2double(editA.String);

f = str2double(editF.String);

n = str2double(editN.String);

[t, vout, vin] = dr(idx, A, f, n);

plot(ax1, t, vin);

plot(ax2, t, vout);

plot(ax\_vtc, vin, vout);

title(ax\_vtc, 'Voltage Transfer Characteristic (VTC)', ...

'Color', [0.4, 0.1, 0.2], ...

'FontSize', 12, ...

'FontWeight', 'bold');

xlabel(ax\_vtc, 'Input Voltage (V)', 'Color', [0.1, 0.3, 0.7]);

ylabel(ax\_vtc, 'Output Voltage (V)', 'Color', [0.1, 0.3, 0.7]);

xlabel(ax1, 'Time', 'Color', [0.1, 0.3, 0.7]);

ylabel(ax1, 'v\_I\_N', 'Color', [0.1, 0.3, 0.7]);

xlabel(ax2, 'Time', 'Color', [0.1, 0.3, 0.7]);

ylabel(ax2, 'v\_O\_U\_T', 'Color', [0.1, 0.3, 0.7]);

end

function resetParameters()

dropdown.Value = 1;

editA.String = '5';

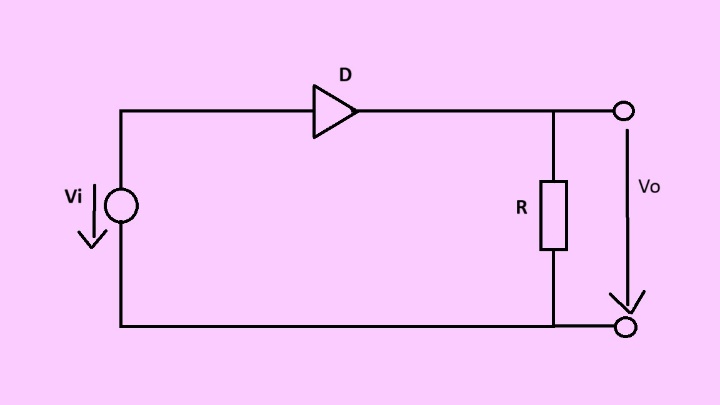
editF.String = '50';

editN.String = '3';

updatePlot(dropdown);

end

end



A close up of a painting

Description automatically generated