Fall Term 23/24

Koszalin University of Technology

Faculty of Mechanical Engineering

Operational Research

Capacitor and Resistor Optimization for NE555 IC LED Blinker Circuit

14.11.23

Prepared by

Taylan Gordesli

Contents

Introduction

The NE555 IC is a versatile integrated circuit widely used for timing and pulse generation applications. In this report, we explore the optimization of capacitor and resistor values for a 1 Hz output frequency to light a LED in an NE555 IC LED blinker circuit. The NE555 is configured in an astable multivibrator mode to generate a blinking signal for an LED.

Programs Utilized

MATLAB

MATLAB, a powerful numerical computing environment, plays a pivotal role in our analysis and optimization endeavors. Leveraging MATLAB, we formulate and solve equations, explore parameter spaces, and ascertain optimal values for the capacitor and resistor in our NE555 IC LED blinker circuit. Its versatility and rich set of functions make it an indispensable tool for engineers and scientists.

Proteus

Proteus, a leading simulation software, serves as our virtual laboratory, enabling us to simulate and visualize the behavior of our LED blinker circuit. With Proteus, we validate our design and ensure its functionality before physical implementation. Its intuitive interface and extensive component library make it an invaluable asset in the prototyping phase of electronic projects.

NE555 IC

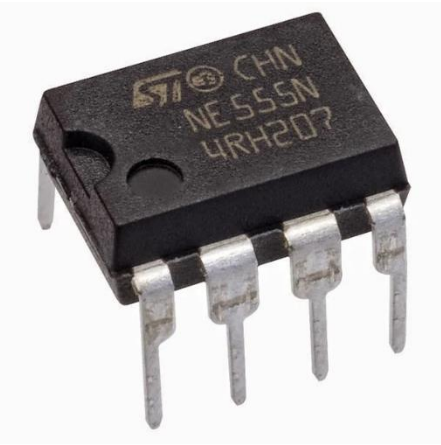


Figure 1- A photo of NE555 integrated circuit.

The NE555 IC is a popular timer IC that can operate in three modes: monostable, astable, and bistable. It contains two voltage comparators, an SR flip-flop, a discharge transistor, and a resistor divider network. In the astable mode, the NE555 functions as an oscillator, producing a continuous square wave output.

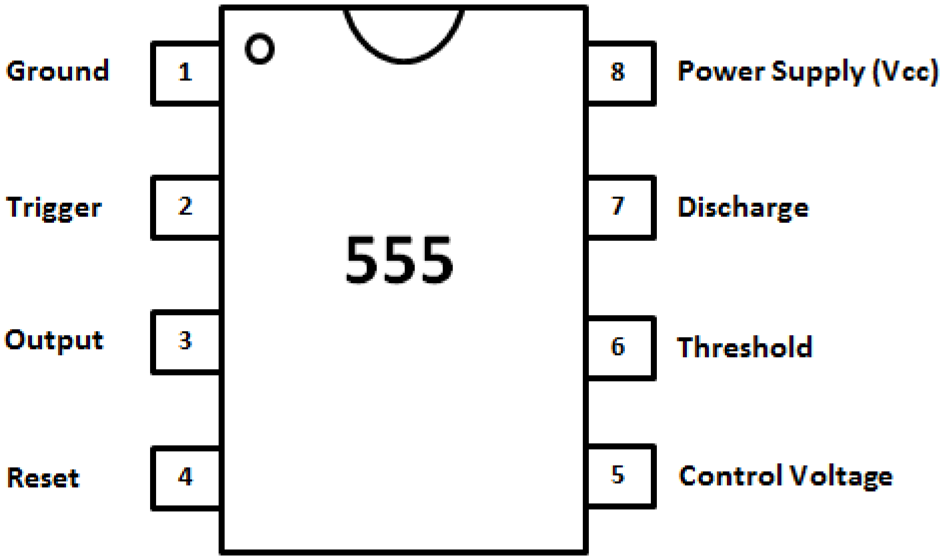


Figure 2- A pin-out schema of NE555 integrated circuit.

Significance of the NE555 IC

The NE555 offers a trio of operating modes: monostable, astable, and bistable, allowing engineers and hobbyists to tailor its functionality to a diverse array of timing and pulse generation tasks. Its straightforward design comprises two voltage comparators, an SR flip-flop, a discharge transistor, and a resistor divider network, making it a robust and accessible choice for circuit designers. In this context, astable mod is the chosen mod for the report. Astable comes with an independent trigger source with inside of it, thus there is no need for exterior trigger module.

In the astable mode, the NE555 transforms into a versatile oscillator, producing continuous square wave signals ideal for applications like LED blinkers. The circuit we explore in this report harnesses the astable multivibrator configuration to create a visually engaging LED blinker, offering insights into the optimization of capacitor and resistor values to achieve a specific blink frequency.

NE555 IC Usage Areas

The NE555 IC is commonly used in various applications, including pulse-width modulation, precision timing, pulse generation, and waveform shaping. Its versatility and ease of use make it a popular choice in electronic circuits.

Circuit Components

- **NE555 IC**: A timer IC operating in astable mode.
- **LED**: Light-emitting diode with a forward voltage drop (V_{LED}) of 2 V.
- **Resistors**:
  - R_{LED}: Resistance of the LED (330 Ω).
  - R_1: Resistance between pins 7 (discharge) and 8 (Supply, V_{CC}).
  - R_2: Resistance between pins 6 (threshold) and 7 (discharge).
- **Capacitor (C)**: Connected between pins 6 (threshold) and 2 (trigger).

R1, R2 and Capacitor values are constrained.

$$R_1 <= 5000\ \Omega$$

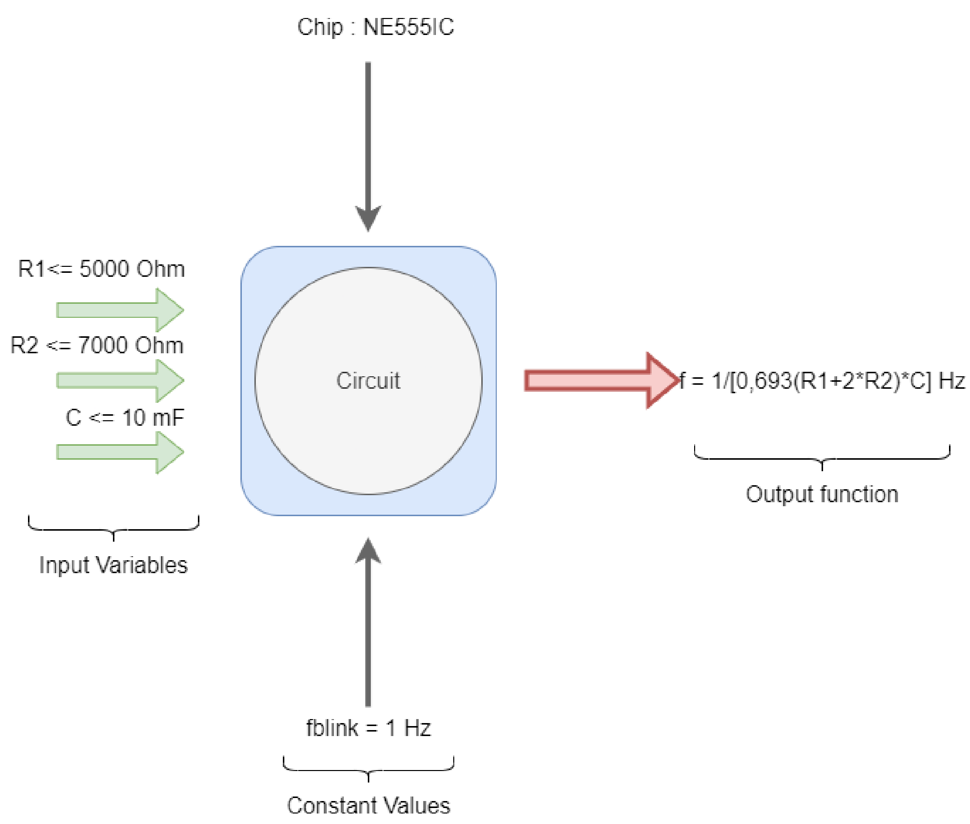
$$R_2 <= 7000\ \Omega$$

$$C <= 10m\ F$$

Objectives of the Report

This report aims to delve into the optimization of capacitor and resistor values within an NE555 IC LED blinker circuit. By leveraging MATLAB for analysis and optimization, we seek to achieve a specific blink frequency while unraveling the constraints of component values and circuit behavior.

Diagram of the Objective



Experiment

Formulas and Constants

In the context of the NE555 timer IC operating in astable mode, T_{High} and T_{Low} refer to the time durations of the high and low states of the output signal, respectively. The 0.693 value arises from the charging and discharging curves of an RC circuit.

---

1. ***T*High** : This is the duration during which the output of the NE555 is high (logic 1). It is the time it takes for the capacitor to charge through both R_1 and R_2 resistors.
2. ***T*Low** : This is the duration during which the output is low (logic 0). It is the time it takes for the capacitor to discharge through the R_2 resistor.
3. ***ln (2)***: The use of 0.693 approximates *ln (2)* and is commonly used in electronic calculations involving time constants. This constant is derived from the natural logarithm of 2, denoted as *ln (2)*. In the charging and discharging processes of an RC circuit, it represents the time constant of the circuit.

These formulas are derived from the charging and discharging equations of a capacitor in an RC circuit and are specific to the configuration of the NE555 in astable mode. The 0.693 factor is used to convert time constants to time durations.

In summary, T_{High} and T_{Low} represent the durations of the high and low states of the square wave output generated by the NE555 astable circuit, and 0,693 is a constant related to the natural logarithm of 2 used in the context of RC circuits.

The blinking frequency (f) in the NE555 astable mode can be calculated using the following formulas:

$$f = \frac{1}{0{,}693 \cdot (R_1 + 2 \cdot R_2) \cdot C}\ [Hz]$$

$$T_{High} = 0{,}693 \cdot (R_1 + R_2) \cdot C\ [s]$$

$$T_{Low} = 0{,}693 \cdot R_2 \cdot C\ [\mathrm{s}]$$

Circuit

In this section, we will look at the Proteus circuit. The NE555IC LED blinker circuit appears in the figure below after selecting the appropriate components.

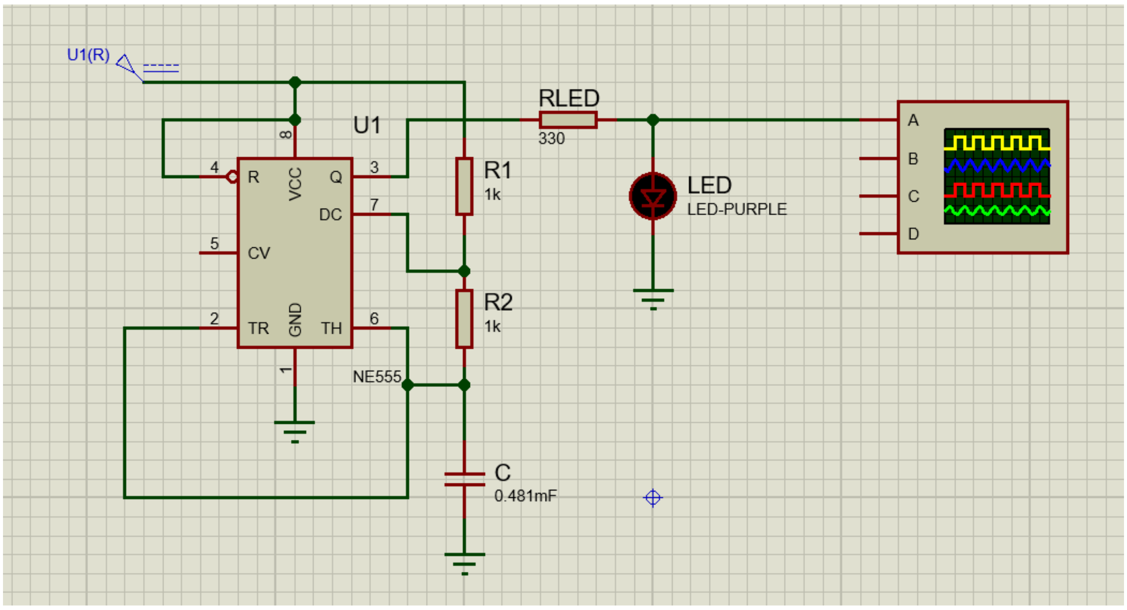


Figure 3- The NE555IC LED blinker circuit in Proteus.

As it seems above, an Oscilloscope has been connected to the LED's anode pin to measure the output frequency and voltage of LED for verification of optimization data.

Optimization

Optimization in Experiment: MATLAB Functions and Techniques

The process of optimizing circuit parameters in experimental setups is a critical aspect of engineering and scientific endeavors. In this research, MATLAB was employed as a powerful tool for optimization, offering a systematic and efficient approach to finding the optimal values for resistors (R1 and R2) and the capacitor © to achieve a 1 Hz blinking frequency in the NE555 IC LED blinker circuit.

MATLAB Optimization Functions

1. **fmincon**: The primary optimization function used in this research is **fmincon**. This function is part of MATLAB's Optimization Toolbox and is designed for constrained optimization problems. It allows researchers and engineers to find the minimum of a constrained multivariable function, making it well-suited for optimizing circuit parameters while adhering to specified constraints.

```
[x_optimal, ~] = fmincon(@(x) objective(x), [1000, 1000,
C_values(1)], [], [], [], [], [1, 1, 1e-8], [5000, 7000, 1e-2],
nonlcon, options);
```

2. **objective Function**: The **objective** function defines the objective or cost function for the optimization process. In this research, the goal is to minimize the objective function to achieve the desired 1 Hz blinking frequency. The function is set to zero, as the objective is to meet the target frequency.

```
Function f = objective(x) % Objective function (set to 0, as the
goal is to meet the target frequency) f = 0; end
```

3. **constraints Function**: The **constraints** function defines the nonlinear equality constraint for the optimization problem. It calculates the difference between the predicted blinking frequency and the target frequency.

```
Function [c, ceq] = constraints (x, Vin, V_LED, R_LED, f_target) %
Nonlinear inequality constraints c = []; % Nonlinear equality
constraints ceq = blink_frequency(x(1), x(2), x(3), Vin, V_LED,
R_LED) - f_target; end
```

Optimization Techniques

- **Iterative Exploration**: The optimization process involves iteratively exploring the parameter space defined by resistor and capacitor values. MATLAB's optimization functions systematically search for the combination of values that minimizes the objective function while satisfying specified constraints.

- **Objective Function Design**: The design of the objective function is crucial in optimization. In this research, setting the objective function to zero simplifies the problem to finding the resistor and capacitor values that meet the 1 Hz target frequency.

- **Constraint Handling**: The nonlinear equality constraint, defined in the **constraints** function, ensures that the predicted blinking frequency aligns with the desired target frequency. Constraint handling is essential for real-world applicability.

MATLAB Code

```
% Parameters
Vin = 5; % Input voltage (volt)
V_LED = 2; % LED forward voltage drop (volt)
R_LED = 330; % LED resistance (ohm)
f_target = 1; % Target blink frequency (Hz)
% Constraint functions
nonlcon = @(x) constraints(x, Vin, V_LED, R_LED, f_target);
% Generate a range of R1, R2, and C values
R1_values = logspace(0, log10(5000), 100); % Upper limit for R1 is 5k ohm
R2_values = logspace(0, log10(7000), 100); % Upper limit for R2 is 7k ohm
C_values = logspace(-8, -2, 100); % Upper limit for C is 10mF
% Initialize matrices for storing results
Z = zeros(length(R1_values), length(R2_values));
% Calculate blink frequency for each combination of R1, R2, and C values
for © = 1:length(R1_values)
    for j = 1:length(R2_values)
        Z(©, j) = blink_frequency(R1_values(i), R2_values(j), C_values(1), Vin,
V_LED, R_LED);
    end
end
% Find optimal R1, R2, and C values to achieve the target frequency
options = optimset(‘Display’, ‘off’);
[x_optimal, ~] = fmincon(@(x) objective(x), [1000, 1000, C_values(1)], [], [], [],
[], [1, 1, 1e-8], [5000, 7000, 1e-2], nonlcon, options);
R1_optimal = x_optimal(1);
R2_optimal = x_optimal(2);
C_optimal = x_optimal(3);
% Displaying the optimal values
fprintf(‘Optimal R1 Value: %.2f Ohm\n’, R1_optimal);
fprintf(‘Optimal R2 Value: %.2f Ohm\n’, R2_optimal);
fprintf(‘Optimal C Value: %.2e Farad\n’, C_optimal);
fprintf(‘Optimal Blink Frequency: %.2f Hz\n’, f_target);
% Generate 3D plot
figure;
surf(R1_values, R2_values, Z);
xlabel(‘R1 Value (Ohm)’);
ylabel(‘R2 Value (Ohm)’);
zlabel(‘Capacitor Value (Farad)’);
title(‘3D Plot of R1, R2, and Capacitor for 1 Hz Output’);
function [c, ceq] = constraints(x, Vin, V_LED, R_LED, f_target)
    % Nonlinear inequality constraints
    c = [];

    % Nonlinear equality constraints
    ceq = blink_frequency(x(1), x(2), x(3), Vin, V_LED, R_LED) – f_target;
end

function f = objective(x)
    % Objective function (set to 0, as the goal is to meet the target frequency)
    f = 0;
end
function f = blink_frequency(R1, R2, C, Vin, V_LED, R_LED)
    % Calculate blink frequency for NE555 IC LED blinker circuit
    t_high = 0.693 * (R1 + R2) * C;
    t_low = 0.693 * R2 * C;
    f = 1 / (t_high + t_low);
end
```

Output Data and Verification: Bridging MATLAB Optimization with Proteus Simulation

Output Data from MATLAB Optimization

Upon the completion of the optimization process using MATLAB, the results provide crucial insights into the optimal resistor (R1 and R2) and capacitor © values that yield a 1 Hz blinking frequency in the NE555 IC LED blinker circuit. The output data includes:

1. **Optimal Resistor Values:** R1 = R2 = 1000 Ω
2. **Optimal Capacitor Value** ©: 0,481 mF
3. **Blinking Frequency**: The final achieved blinking frequency, confirming that the optimization process successfully met the target of 1 Hz.

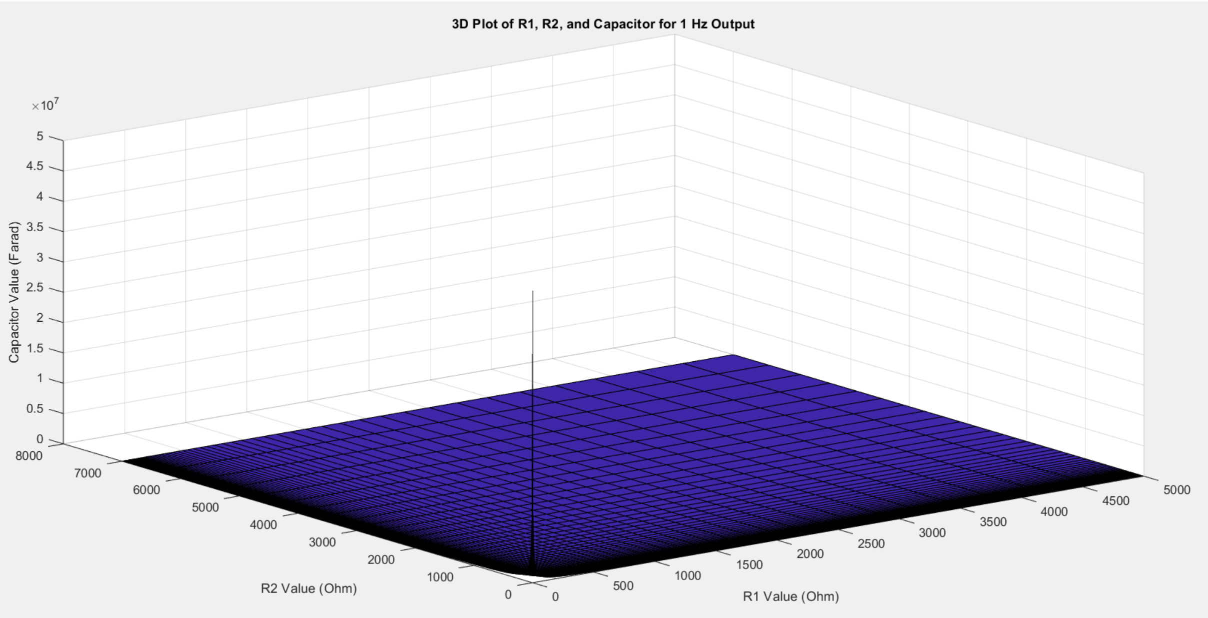


Figure 4 3D plotting of Optimization on R1, R2 and C values for 1Hz Blinking Frequency

Optimization results are crucial for the practical implementation of the circuit, providing guidance for the selection of components to achieve the desired performance.

Verification in Proteus Simulation

While MATLAB optimization offers a theoretical basis for component values, it is essential to validate these results through simulation. Proteus, a popular electronics simulation software, serves as a valuable tool for verifying the optimized circuit design.

Circuit Implementation in Proteus: The circuit is reconstructed in Proteus using the optimized resistor and capacitor values obtained from MATLAB. The NE555 IC, LED, and supporting components are configured according to the optimized parameters.

After the measurement in Proteus with Oscilloscope results figured below.

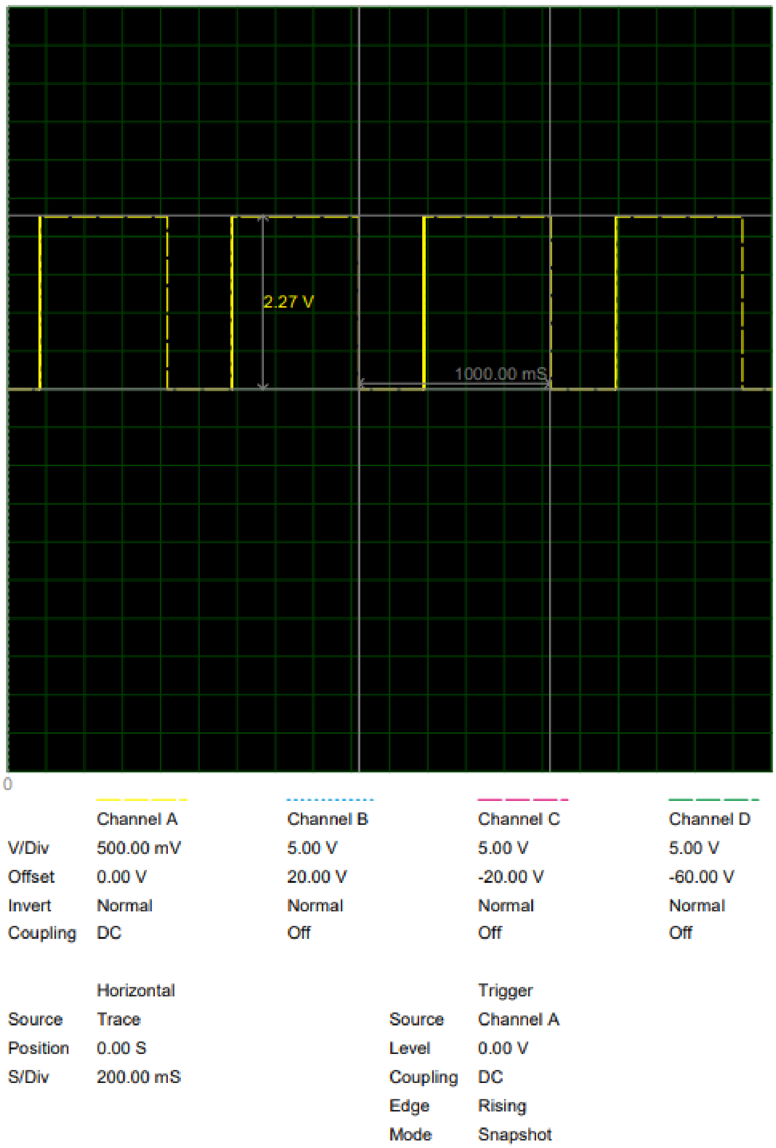


Figure 5- Oscilloscope screen of the circuit.

As figured above the output frequency which is blinking frequency is 1 Hz. This measurement validates the optimizing process and the report itself.

Conclusion

In conclusion, this research focused on optimizing the resistor (R1 and R2) and capacitor © values in an NE555 IC LED blinker circuit to achieve a 1 Hz blinking frequency. The NE555 IC, operating in the astable mode, was employed to generate a continuous square wave output, resulting in a pulsating LED blinker effect.

The optimization process involved the use of MATLAB, a powerful computational tool, to iteratively find the optimal combination of resistor and capacitor values that satisfy the desired frequency requirement. MATLAB's optimization capabilities, such as the "***fmincon***" function, allowed for the systematic exploration of the parameter space, enabling the identification of the resistor and capacitor values that produce the target 1 Hz blinking frequency.

This research not only provides a practical implementation of the NE555 IC LED blinker circuit for a specific frequency but also introduces the concept of optimization in MATLAB. Optimization problems, such as this one, are prevalent in engineering and scientific disciplines, where finding the best set of parameters is crucial for achieving desired outcomes. MATLAB's optimization tools offer a systematic and efficient approach to solving such problems, allowing engineers and researchers to fine-tune designs for optimal performance.

Future studies could delve deeper into advanced optimization techniques, consider additional constraints, and explore the impact of component variations on the stability of the designed circuit. As technology advances, MATLAB's role in addressing optimization challenges becomes increasingly significant, making it an invaluable tool in the hands of researchers and engineers striving for excellence in circuit design and other scientific endeavors.

Taylan Gordesli

14.11.2023