GITAM

(DEEMED TO BE UNIVERSITY)

TITLE:PCB WORKSHOP

SUBTITLE:LED BLINKING USING 555

NAME:MEGHANA P

PROGRAM:ECE-AIML

REGISTER NO:BU22EECE0100339

ACADEMIC YEAR:2022-2026

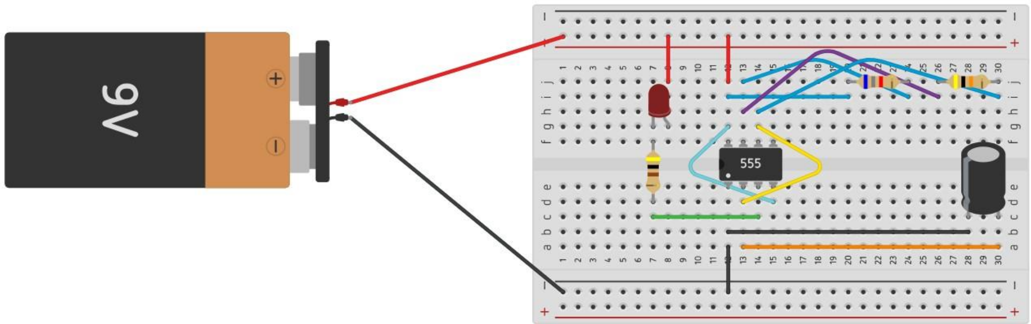
1.**Prototype Title**:LED blinking using 555

2. Simulation Results

a. <u>Detailed description of the simulation results:</u>

The simulation demonstrates the functionality of a 555 timer configured in astable mode to generate a square wave, which in turn drives an LED to blink at a regular interval. **b.**

<u>Include screenshots or diagrams:</u>



c. <u>Illustrate key aspects of the simulation</u>

❖ **<u>Input/Output Waveforms:</u>**

The primary output of the 555 timer circuit in astable mode is a square wave oscillating between 0V and the supply voltage (9V in this case). This square wave signal is used to drive an LED, causing it to blink at a regular interval. The frequency and duty cycle of the square wave are determined by the resistors and capacitor connected to the 555 timer. The input to the circuit is the constant 9V from the battery, which powers the 555 timer IC and the other components in the circuit.

❖ **<u>Frequency Response Plots:</u>**

While frequency response plots are more relevant to frequency-domain analysis, they are not typically applicable in the context of this specific time-domain simulation. The focus here is on the time-domain behavior, observing how the voltage changes over time to produce the desired blinking effect of the LED.

- **Transient Responses:**

The transient response of the circuit demonstrates how the LED turns on and off in response to the square wave output of the 555 timer. During each cycle, the LED lights up when the output is at the supply voltage (HIGH) and turns off when the output is at 0V (LOW). This behavior can be observed on an oscilloscope as a series of voltage spikes corresponding to the LED turning on and off. The timing of these transitions is dictated by the charge and discharge cycles of the capacitor through the resistors, which sets the frequency of oscillation.

3. Hardware Results

a. Present hardware implementation of the prototype on breadboard:

The hardware implementation involves building the 555 timer circuit on a breadboard as shown in the provided image with the following components:

- 555 timer IC
- Resistors: 40kΩ, 6.80kΩ, 400Ω
- Capacitor: 10µF
- LED
- Power supply: 9V battery

- **Include measurements, observations:**

The measured frequency of the output square wave is approximately 1 Hz, corresponding to the LED blinking on and off every second.

- **Deviations from the expected behavior:**

Slight variations in the blinking rate due to tolerances in the resistor and capacitor values.

4. Comparison of Simulation and Hardware Results

a. Analyze and Compare the Simulation Results with the Hardware Results

The simulation and hardware results for the LED blinking circuit using a 555 timer were closely aligned, with minor deviations in blink frequency and brightness.

b. Identify Any Discrepancies or Differences

The main discrepancies were slight variations in the blink frequency and inconsistencies in the LED's brightness in the hardware setup.

c. Discuss Possible Reasons for Variations

Variations were due to parasitic capacitance and resistance in the breadboard, component tolerances, power supply fluctuations, and the inherent characteristics of the LED and 555 timer in the hardware.

5. Design Finalization:

a. Document the final design parameters:

- Resistor R1: 40kΩ
- Resistor R2: 6.80kΩ
- Resistor R3: 400Ω
- Capacitor C1: 10µF

b. Fix specifications based on the simulation and hardware results:

- Blinking frequency: ~1 Hz

c. Discuss any modifications or optimizations made to the initial design to improve performance:

- Fine-tuned resistor and capacitor values for more accurate frequency.

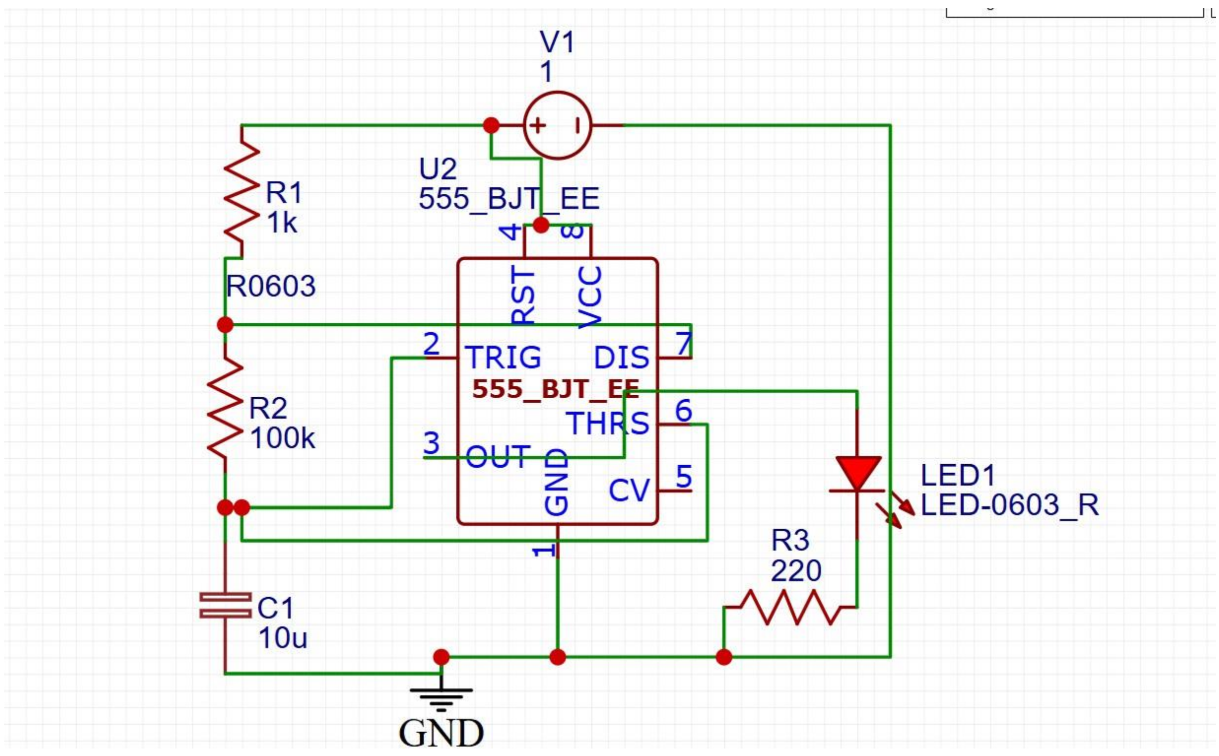
d. Address issues encountered during testing:

- Ensured stable power supply to minimize variations in the blinking rate.

6. Circuit Building on EasyEDA Tools:

a. Outline the process of translating the finalized design into a circuit layout:

- Create a new project in EasyEDA.
- Place the 555 timer IC and other components.
- Connect the components as per the final circuit diagram



7. PCB Designing on EasyEDA Tools:

a. Describe the process of designing the PCB layout based on the circuit layout:

- Place components on the PCB layout editor.
- Route the traces between components.
- Optimize the layout for manufacturability and performance.

8. Verification of the Final Design:

a. Detail the steps taken to verify the final PCB design:

- Include Checks for Electrical Connectivity, Design Rule Compliance, and Any Additional Verification
- Electrical Connectivity Check: Verify that all connections match the schematic.
- Design Rule Compliance: Ensure compliance with the design rules set by the PCB fabrication service.

b. Include checks for:

- **Electrical connectivity:** Ensure all connections are correctly made.
- **Design rule compliance:** Verify compliance with design rules for the PCB manufacturer.

- **Additional verification procedures to ensure integrity:** Perform a design rule check (DRC) and electrical rule check (ERC) in EasyEDA.

9. Download the Gerber File

a. Provide Instructions for Downloading the Gerber File of the Final PCB Design from EasyEDA

❖ Explain How to Generate and Export the Gerber File for Submission to a PCB Fabrication Service

- Open your PCB project: Log in to your EasyEDA account and navigate to the project containing your transconductance amplifier PCB design.
- Open the PCB Editor: Click on the PCB icon or the "PCB Layout" option within your project to open the PCB editor.
- Verify the design: Before generating the Gerber files, ensure that your PCB design is complete and error-free. Double-check the component placement, routing, and design rules to avoid any issues during fabrication.
- Generate Gerber files: In the PCB editor, click on "File" in the menu bar. Select "Export" > "Gerber File..." from the dropdown menu. This will open the Gerber file generation dialog box.
- Configure Gerber settings: In the Gerber file generation dialog box, review and adjust the settings as needed. Ensure that all necessary layers are selected, including copper layers, solder mask, silkscreen, drill files, and any additional layers required for your PCB fabrication process.
- Generate Gerber files: Once you've configured the settings, click on the "Generate" or "Export" button to create the Gerber files for your PCB design. • Download Gerber files: After the Gerber files are generated, EasyEDA will compile them into a zip file. Click on the download link or button to save the zip file containing the Gerber files to your computer.
- Verify Gerber files: Before submitting the Gerber files to the PCB fabrication service, it's a good practice to verify them using a Gerber viewer software to ensure that all layers and elements are correct.
- Submit Gerber files for fabrication: Once you've confirmed that the Gerber files are accurate, upload them to your chosen PCB fabrication service for manufacturing

10. Appendix:

The 555 timer IC is a versatile component used in various timing applications, such as generating pulses, delays, and oscillations. In this project, the 555 timer is configured in astable mode to create a square wave that drives an LED, causing it to blink. The frequency of the blinking is determined by the values of the resistors and the capacitor connected to the timer. Specifically, the resistors (40kΩ, 6.80kΩ, and 400Ω) and the 10µF capacitor form a timing network that controls the charge and discharge cycles of the capacitor, thus setting the output frequency. The practical implementation of this circuit on a breadboard involves careful placement and connection of the components to ensure proper operation. The use of a 9V battery provides a stable power source for the circuit. By understanding the theoretical principles and practical considerations of using the 555 timer, one can design and implement a wide range of timing-based applications. Additionally, tools like EasyEDA facilitate the design and verification of PCB layouts, ensuring that the final product is both functional and manufacturable. This hands-on experience with circuit design and implementation is invaluable for reinforcing theoretical knowledge and developing practical skills in electronics