A MPMC Lab Project Report

On

"PATTERNED LED BLINKING USING 8051 MICROPROCESSOR"

Submitted in Partial Fulfilment of the Requirements

For the award of the Degree of

Bachelor of Technology

In

Electronics & Computer Engineering (ECM)

By

OM JAJU 20311A1901

Y VISHAL 20311A1904

Under the Supervision of

"Mrs NAGA SHAILAJA"

(Assistant Professor)

Department of Electronics & Computer Engineering

Sreenidhi Institute of Science & Technology (Autonomous)

2022-2023

DEPARTMENT OF ELECTRONICS & COMPUTER ENGINEERING

SREENIDHI INSTITUTE OF SCIENCE & TECHNOLOGY

(AUTONOMOUS)

CERTIFICATE

This is to certify that the Python And Shell programming Lab Project work entitled "**Patterned Led Blinking Using 8051 Microprocessor",** submitted by **OM JAJU, Y VISHAL (20311A1901, 20311A1904)** towards partial fulfilment for the award of Bachelor of Technology in **Electronics and Computer Engineering** from Sreenidhi Institute of Science & Technology, Ghatkesar, Hyderabad, is a record of bonafide work done by them during the academic year 2022-2023 under our guidance and evaluation.

Evaluation:

Mrs. Naga shailaja
Assistant Professor
Lab Coordinator

Dr.D.Mohan
HOD, ECM Department

DECLARATION

This is to certify that the work reported in the present project titled **"Patterned Led Blinking Using 8051 Microprocessor"** is a record work done by our team in the **Department of Electronics and Computer Engineering, Sreenidhi Institute of Science and Technology, Yamnampet, Ghatkesar.**

The report is based on the project work done entirely by our team and not copied from any other source.

NAME OF STUDENT	(ROLL NO.)
OM JAJU	20311A1901
Y VISHAL	20311A1904

ACKNOWLEDGEMENT

We owe a great many thanks to a great many people who have helped and supported us throughout the project **Patterened Led Blinking Using 8051 Microprocessor**, which would not have taken shape without their cooperation. Thanks to all. We express our profound gratitude to **Dr.Ch.Siva Reddy,**Principal and indebtedness to our management Sreenidhi Institute Of Science And Technology, Ghatkesar for their Constructive Criticism.we would like to specially thank our Beloved Head of Department ECM, **Dr.D.Mohan** for his guidance,inspiration, and constant encouragement throughout the research work. We would like to express our deep gratitude, **Mrs. Nagashailaja (Assistant Professor)**for their timely guidance. moral support and personal supervision throughout the project.

These few words would never be complete if we were not to mention our thanks to our parents, Department laboratory, staff members and all friends without whose cooperation this project could never become a reality.

NAME OF STUDENT	(ROLL NO.)
OM JAJU	20311A1901
Y VISHAL	20311A1904

ABSTRACT

This project focuses on creating a stimulating LED blinking effect using the Keil IDE and Proteus simulation environment. The objective is to program and simulate the behavior of eight LEDs blinking alternatively, resulting in an engaging visual display.

Using the Keil IDE, the code is developed in the C programming language to control the LEDs. Each LED is connected to an individual I/O pin of the microcontroller. The code utilizes the GPIO functions provided by Keil to toggle the LEDs on and off in a specific pattern.

The Proteus simulation environment is utilized to visualize the LED blinking effect. The circuit setup includes the virtual representation of the microcontroller, LEDs, and necessary components such as resistors and power supply.

The programmed code is simulated in Proteus, allowing for a dynamic visual representation of the LED blinking pattern. By adjusting the timing and sequence of LED states in the code, different patterns and effects can be achieved and observed in real-time simulation.

This project demonstrates the capability of the Keil IDE and Proteus simulation environment in designing and testing complex LED blinking patterns. It provides a platform for exploring creativity in visual effects and serves as a tool for rapid prototyping and testing before actual hardware implementation.

Through this project, one can gain a deeper understanding of microcontroller programming, simulation techniques, and the visualization of LED blinking effects. The combined use of Keil and Proteus offers a comprehensive development and simulation environment for LED-based projects.

INDEX

CHAPTER-1
INTRODUCTION

LED (Light-Emitting Diode) blinking is a widely used technique to create visual effects, indicators, and decorative displays. It involves controlling the on and off states of LEDs in a specific pattern to achieve desired visual patterns and effects. LED blinking projects find applications in various fields, including electronics, lighting systems, advertising, and art installations.

This project focuses on creating a stimulating LED blinking effect using the Keil IDE and Proteus simulation environment. Keil provides a robust integrated development environment for microcontroller programming, while Proteus offers a powerful simulation platform to visualize the behavior of electronic circuits.

The main objective of this project is to program and simulate the behavior of eight LEDs blinking alternatively. By controlling the GPIO pins of the microcontroller using the Keil IDE, the LEDs will be toggled on and off in a specific pattern. The Proteus simulation environment will then provide a virtual representation of the circuit, allowing for real-time visualization of the LED blinking effect.

The project aims to provide a comprehensive understanding of microcontroller programming, simulation techniques, and visual effects using LEDs. It offers a platform for exploring creativity in designing mesmerizing LED blinking patterns and enables rapid prototyping and testing before actual hardware implementation.

Through this project, participants can gain hands-on experience in microcontroller programming, circuit simulation, and visualizing LED effects. The combined use of Keil and Proteus provides a versatile development and simulation environment, fostering innovation and experimentation in the field of LED-based projects.

CHAPTER-2

OBJECTIVE

The main objective of this project is to create a stimulating LED blinking effect using the Keil IDE and Proteus simulation environment. Specifically, the project aims to achieve the following objectives:

- **Programming LED Blinking:** Develop a program using the Keil IDE to control the behaviour of eight LEDs. The program will define a specific pattern for the LEDs to blink alternatively, ensuring the desired visual effect is achieved.
- **Circuit Simulation:** Utilize the Proteus simulation environment to simulate the LED blinking circuit. Set up a virtual representation of the microcontroller, LEDs, resistors, and other necessary components to accurately visualize the behavior of the circuit.
- **Timing and Pattern Control:** Experiment with different timing intervals and patterns to create visually appealing LED blinking effects. Adjust the program parameters in Keil to customize the timing and sequence of LED states, enabling the exploration of various blinking patterns.
- **Real-Time Visualization:** Utilize the Proteus simulation environment to observe the LED blinking effect in real-time. The simulation will provide a visual representation of the LEDs blinking alternatively, allowing for immediate feedback and analysis of the programmed behaviour.
- **Learning and Creativity:** Enhance understanding of microcontroller programming, simulation techniques, and LED effects. Foster creativity and innovation by exploring different LED blinking patterns and visual effects, leading to the development of unique and captivating LED displays.

CHAPTER-3

ANALYSIS

The LED blinking project, implemented using Keil IDE for code development and Proteus for circuit simulation, can be analyzed from the following perspectives:\

- **Code Logic Analysis:** Review the code written in Keil for controlling the LED blinking pattern. Analyze the logic used to alternate the LEDs on and off. Ensure that the code is properly structured and follows best practices, such as using appropriate control structures and clear variable names.
- **Code Efficiency Analysis:** Assess the efficiency of the code in terms of memory usage and processing speed. Analyze the size of the compiled code and evaluate any potential optimizations to reduce memory consumption or improve execution time.
- **Keil IDE Integration:** Evaluate the integration of the code with the Keil IDE. Assess the ease of development, debugging, and code editing capabilities offered by the IDE. Consider features such as syntax highlighting, code completion, and debugging tools to enhance the development process.
- **Proteus Simulation Analysis:** Utilize Proteus for simulating the LED blinking circuit. Analyze the accuracy of the simulation in replicating the behavior of the actual hardware. Evaluate the visual representation of the LEDs and ensure that the simulation accurately reflects the expected blinking pattern.
- **Real-Time Analysis:** Assess the real-time behavior of the LED blinking simulation in Proteus. Analyze the timing and synchronization of the LED states to ensure that they align with the intended pattern. Verify that the simulation accurately captures the timing intervals and transitions of the LEDs.
- **Error Handling Analysis:** Evaluate the code's error-handling capabilities. Assess how the code handles exceptional situations, such as unexpected input or hardware failures. Analyze the error messages or notifications provided by the code to aid in debugging and troubleshooting.

CHAPTER-4

APPROACH

To implement the LED blinking project with Keil and Proteus, you can follow the following approach:

- **Circuit Design:** Begin by designing the circuit layout in Proteus. Add the 8051 microcontroller and connect the eight LEDs to its I/O pins. Include current-limiting resistors for each LED to ensure safe operation. Set up the necessary power supply connections and ensure proper grounding.
- **Code Development:** Open the Keil IDE and create a new project for the 8051 microcontroller. Write the code to control the LED blinking pattern. Use GPIO functions to toggle the I/O pins and alternate the LEDs on and off. Implement a loop to continuously cycle through the LEDs.
- **Compile and Flash:** Compile the code in Keil to check for any syntax or compilation errors. Once the code is error-free, flash the compiled binary file onto the microcontroller within Proteus. This step ensures that the programmed behavior is accurately reflected in the simulation.
- **Proteus Simulation:** Run the simulation in Proteus to observe the LED blinking effect. Check if the LEDs blink alternately as intended. Analyze the timing, transitions, and overall visual effect of the blinking pattern. Adjust the code or circuit if necessary to achieve the desired outcome.
- **Testing and Optimization:** Conduct thorough testing of the LED blinking simulation. Verify that the code operates reliably and consistently. Fine-tune the timing intervals or adjust the code logic as needed to optimize the visual effect. Take note of any issues or areas for improvement during the testing phase.

CHAPTER-5

THE SOURCE CODE AND OUTPUT OF PATTERNED LED

BLINKINGUSING 8051 MICROPROCESSOR

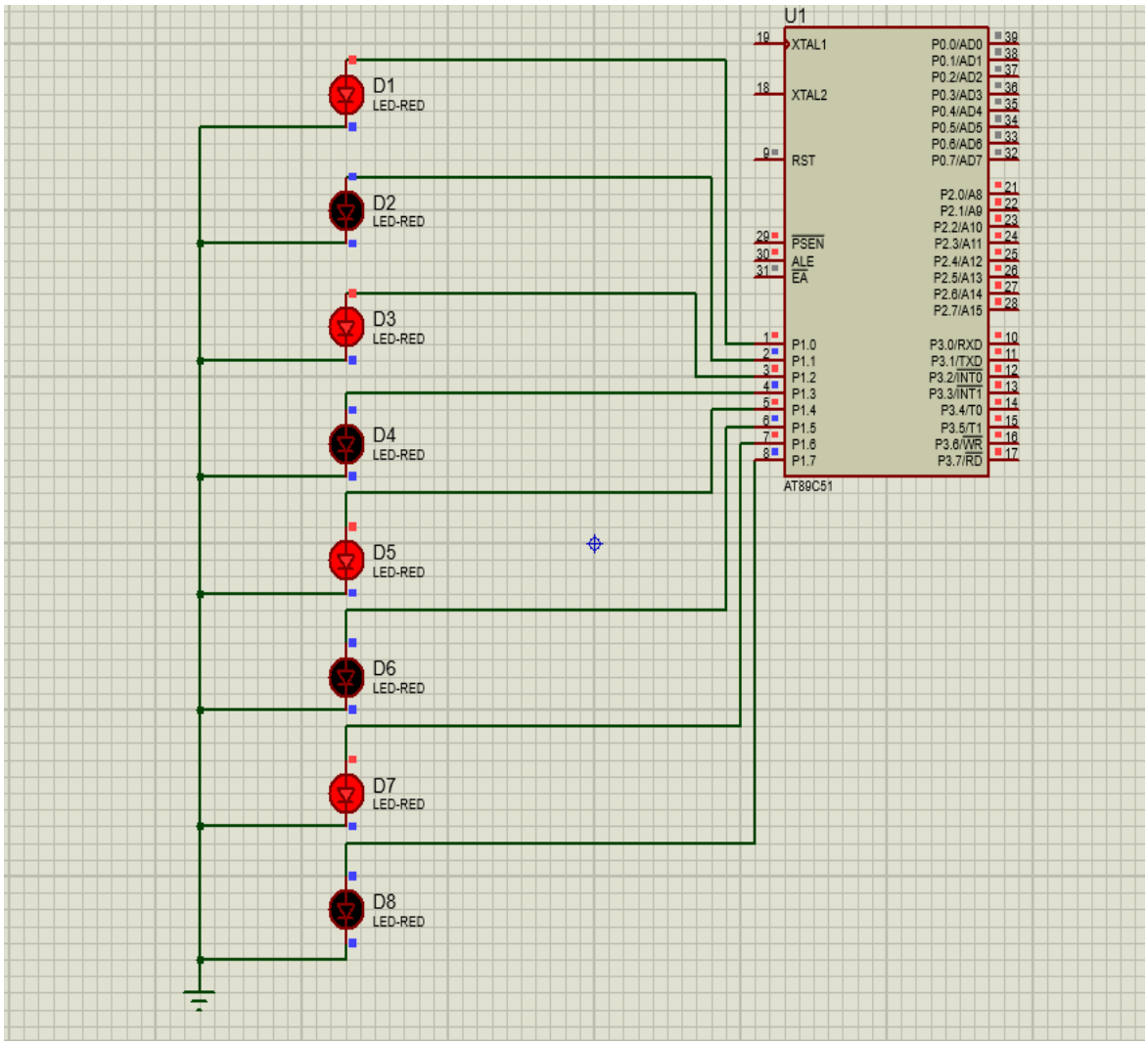
```
#include <reg51.h>

sbit LED1 = P1^0; // Define LED1 pin

sbit LED2 = P1^1; // Define LED2 pin

sbit LED3 = P1^2; // Define LED3 pin

sbit LED4 = P1^3; // Define LED4 pin

sbit LED5 = P1^4; // Define LED5 pin

sbit LED6 = P1^5; // Define LED6 pin

sbit LED7 = P1^6; // Define LED7 pin

sbit LED8 = P1^7; // Define LED8 pin

void delay(unsigned int count){

    unsigned int i, j;
```

```
for (i = 0; i < count; i++)

   for (j = 0; j < 1000; j++);

}

void main(){

    unsigned int i;

  while (1){

        for (i = 1; i <= 8; i++){

            if (i % 2 == 0) // Check if the number is even{

                LED1 = LED3 = LED5 = LED7 = 0; // Turn off odd LEDs

                LED2 = LED4 = LED6 = LED8 = 1; // Turn on even LEDs

            }else{

                LED1 = LED3 = LED5 = LED7 = 1; // Turn on odd LEDs

                LED2 = LED4 = LED6 = LED8 = 0; // Turn off even LEDs

}
```

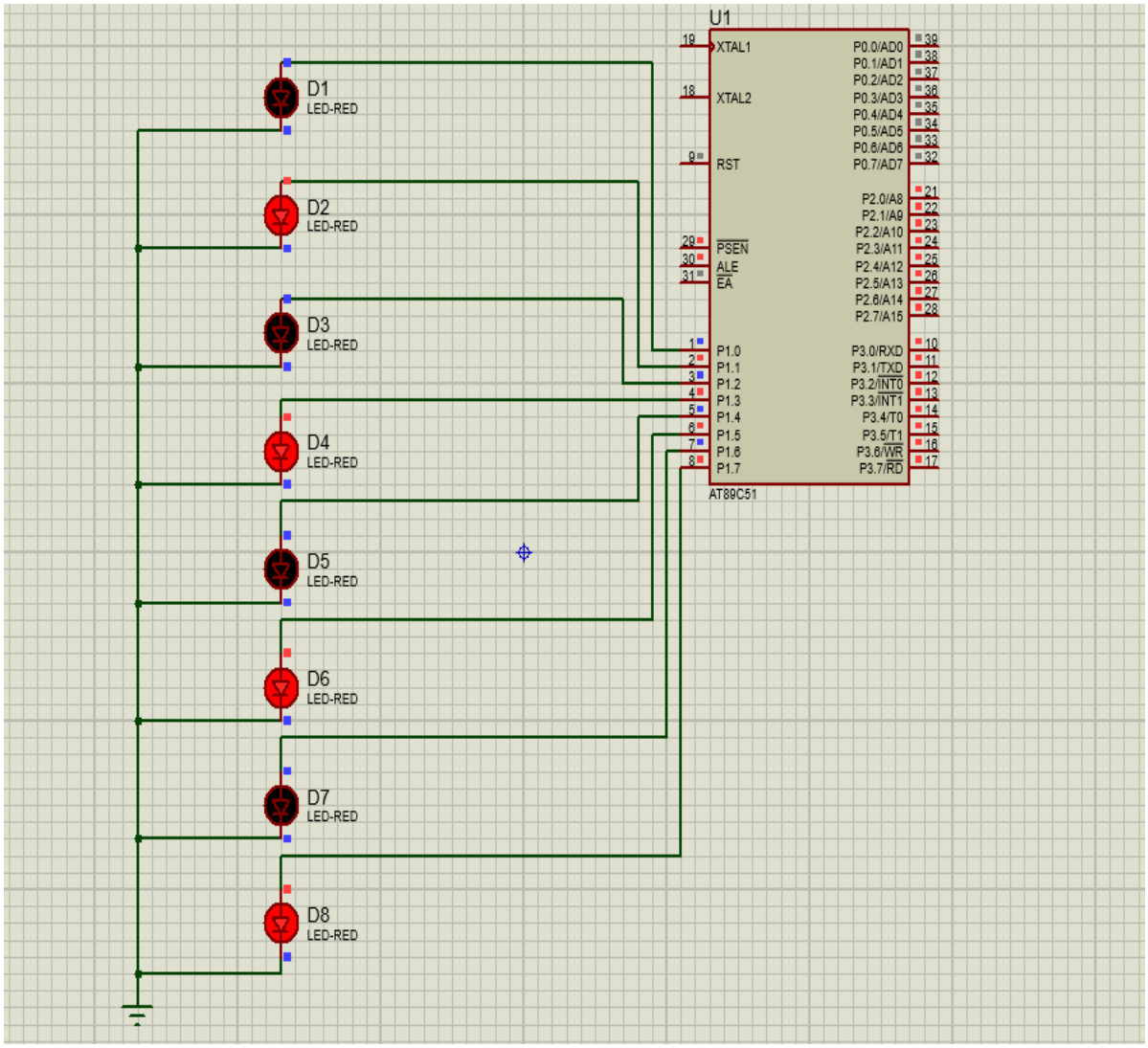
```
        delay(500); // Delay for 500 milliseconds

    }

}}
```

OUTPUT:

I) ODD LED'S BLINK (1, 3, 5, 7):



II) EVEN LED'S BLINK (2, 4, 6, 8):



CHAPTER-6

CONCLUSION

In conclusion, the LED blinking project implemented using Keil and Proteus offers an effective and efficient approach to achieve an alternating LED blinking pattern. By following the outlined approach, a well-designed circuit can be created in Proteus, and the code can be developed and compiled in the Keil IDE. The Proteus simulation provides a realistic visualization of the LED blinking behavior, allowing for testing, refinement, and optimization.

Through the development and simulation process, we can gain valuable insights and experience in microcontroller programming, circuit design, and simulation techniques. The project enables us to understand the intricacies of controlling LEDs using a microcontroller, timing considerations for creating blinking effects, and the importance of proper circuit design.

The successful implementation of the LED blinking project using Keil and Proteus demonstrates the synergy between software development and hardware simulation. It allows for rapid prototyping, debugging, and refinement without the need for physical components. The project offers a platform for experimentation and creativity, encouraging the exploration of different LED blinking patterns and visual effects.

Furthermore, the documentation of the circuit design, code, and simulation results provides a valuable resource for future reference and replication. It serves as a comprehensive record of the project's development process, allowing for easy sharing, collaboration, and further enhancements.

Overall, the LED blinking project using Keil and Proteus showcases the capabilities of these tools in creating engaging visual effects and demonstrates the potential for innovation in LED-based projects. It serves as a foundation for further exploration and application of microcontroller programming and circuit simulation techniques in various domains.

CHAPTER-7

REFERENCES

[1]https://www.tutorialspoint.com/microprocessor/microcontrollers_8051_architecture.htm

[2] https://electrosome.com/led-blinking-8051-microcontroller-keil-c-tutorial-at89c51/

[3]https://www.circuitstoday.com/blinking-led-using-8051