PROJECT BRIEF

PROJECT NAME:	SmartLift: Dual Elevator Control System
GROUP NUMBER :	15
GROUP MEMBERS:	Amit (2023MEB1328), Anand (2023MEB1329), Anirudha (2023MEB1330), Ayush Tiwari (2023MEB1333), Aditya (2023MEB1324), Agrasen (2023MEB1327)

INTRODUCTION

- Modern buildings increasingly rely on elevators for vertical transportation. Efficient elevator control systems are crucial for ensuring safety, optimizing wait times, and improving user convenience. This project presents the development of a dual-elevator control system using Arduino, designed to simulate real-world lift behavior with smart features like floor sensing and load monitoring.

PROBLEM STATEMENT

- Traditional small-scale elevator models often lack real-time feedback mechanisms and safety features such as floor detection and load sensing. This project addresses these gaps by integrating limit switches for accurate floor detection and a load cell for real-time cabin weight monitoring.

PROJECT GOALS

The primary objectives of this project are:

- To design a functional dual-lift control system.
- To implement real-time floor detection.
- To integrate a load cell for monitoring cabin weight and preventing overload.
- To simulate internal and external floor requests through button inputs.
- To implement priority-based lift selection based on distance.

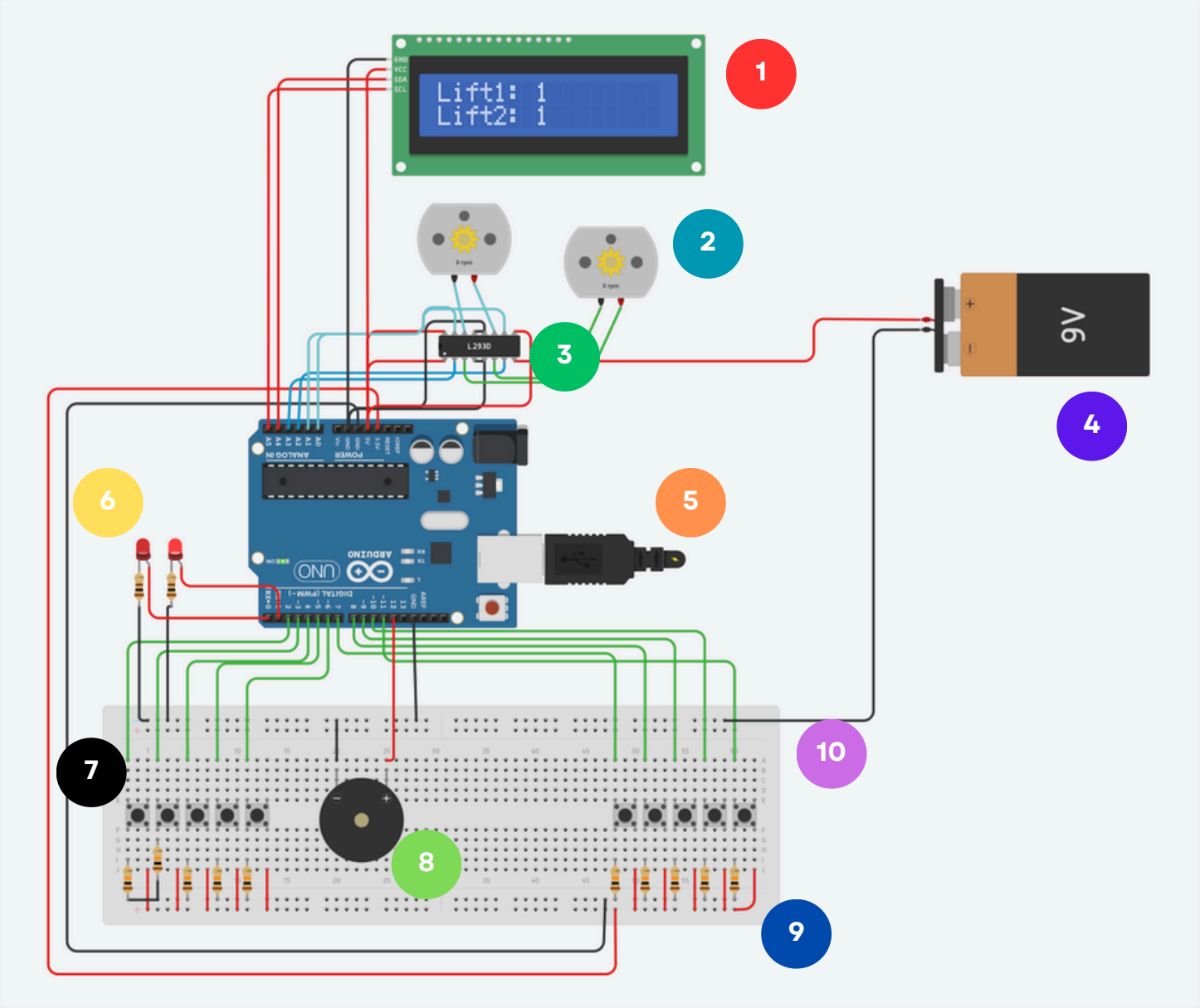
KEY FEATURES

- Dual Elevator Support: Two independently controlled lifts sharing floor requests.
- Load Monitoring: A 1kg load cell ensures safety by measuring cabin weight.
- Priority-Based Lift Selection: The closest lift is dispatched to serve the request.
- LED Indicators: Show active lift during operation.

COMPONENTS USED

- Arduino UNO Microcontroller
- DC Motors
- LM298 Motor Driver
- Load Cell
- Connecting Wires
- Power Supply
- Plywood

CIRCUIT DIAGRAM



COMPONENTS USED

1	LCD DISPLAY	6	LED LIGHTS
2	DC MOTORS	7	PUSH BUTTONS
3	MOTOR DRIVER	8	BUZZER
4	POWER SUPPLY	9	RESISTORS
5	ARDUINO UNO	10	BREADBOARD

PROBLEMS FACED

During development we ran into a few hiccups:

The cars would sometimes “tie” when equally distant from a call and hesitate until we added a timestamp-based tie-breaker

Our call-panel buttons would chatter and register multiple presses before we introduced a 20 MS software debounce

The provided compartments weren’t sturdy enough, so we manufactured new ones from plywood and also made a lot of important additions to the structure like supports, motor holders, etc.

CONCLUSION

In summary, our two-lift system for a five-story building successfully integrates efficient dispatch logic, overload detection, and precise motion control to deliver reliable, user-friendly performance. By assigning calls to the nearest car—including a simple timestamp tie-breaker—we minimized wait times and avoided conflicts. Load-cell monitoring with led indicators ensures safety by preventing overloading, while button-debounce routines produce smooth, accurate stops and clean input handling. Throughout development, we tackled mechanical, electrical, and firmware challenges to build a robust prototype. This project not only demonstrates practical elevator control strategies but also lays a solid foundation for future enhancements, such as energy-saving modes and remote monitoring.

CONTRIBUTIONS

Anand:
Anand was primarily responsible for building all the physical components of the project. He also contributed significantly to the circuit assembly and provided support in the coding process.

Anirudha:
Anirudha developed the core logic for the dual elevator single-button system. He also assisted in other parts of the coding, contributed to the physical construction of the model, helped with creating the circuits, and edited the final video presentation.

Agrasen:
Agrasen played a key role in assembling the circuit and contributed to the final mechanical assembly of the project. He also supported the coding and physical construction of the miniature lift system. Also helped in manufacturing the new components

Amit:
Amit was involved in setting up the Arduino system and managing the wiring for the entire project. He also handled the mechanical assembly, ensuring that the cabin movements were mechanically aligned with the programmed logic. Additionally, he helped prepare the final report.

Ayush:
Ayush was responsible for writing the final report. Additionally, he supported the coding process, like coding the LCD display, load cell, etc. and contributed to circuit making, as well as assisting with various other aspects of the project.

Aditya:
Aditya provided assistance in building the overall project and supported the team across various stages of development.