Summer Fellowship 2026

On

DESIGN OF AN INTERNAL BATTERY-POWERED STEPPER SCISSOR LIFT FOR DOBOT MAGICIAN GO

A Technical Report submitted for the FOSSEE Fellowship

Submitted by:

Raj Prajapati

Sardar Vallabhbhai National Institute Of Technology

Under the Guidance of:

Prof. Jayendran Venkateswaran

Department of IEOR

Indian Institute of Technology Bombay

FOSSEE Project, IIT Bombay

Academic Year: 2026

Submitted On 20/07/26

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to all those who provided me the possibility to complete this report. A special gratitude I give to my project guide, **Prof. Jayendran Venkateswaran**, whose contribution in stimulating suggestions and encouragement helped me to coordinate my project successfully.

I am also grateful to the mentors from the department, including the M.Tech students, for their valuable technical support and constructive feedback throughout the project.

I am especially thankful to my **FOSSEE guides, Mr. Sumanto Kar**, for coordinating all official and administrative aspects of the internship and for ensuring smooth communication during its course.

I would also like to acknowledge with much appreciation the crucial role of the **FOSSEE Project, IIT Bombay**, for providing me with this internship opportunity, the necessary resources, and a platform to work on industrial automation and robotics.

Finally, I wish to thank my family and friends for their continuous support and motivation throughout the duration of this project.

Raj Prajapati

ABSTRACT

This report details the design, fabrication, and integration of a custom scissor lift mechanism for the Dobot Magician Go mobile robotic platform.

The project aimed to add controlled vertical actuation to the robot's capabilities, enabling simulation of complex industrial automation tasks. The lift is actuated by a NEMA 17 stepper motor and controlled via an ESP32-C3 microcontroller.

A key achievement of this project is the seamless integration with the Dobot Magician Go, allowing the lift to draw power directly from the robot's internal battery via a DC-DC buck converter. Furthermore, the system receives direct control commands through the robot's internal communication bus, with a secondary remote control capability over Wi-Fi. The mechanism successfully acts as a proof of concept for untethered, mobile vertical manipulation.

Contents

1 INTRODUCTION

1.1 Background of Industrial Automation and Mobile Collaborative Robots

In modern manufacturing, warehousing, and smart logistics, the transition from static automation to highly flexible, mobile systems is a central pillar of the Industry 4.0 paradigm. Traditional robotic arms are fixed to stationary factory floors, limiting their work envelope to a precise, localized radius. To replicate true modern industrial workflows—such as automated sorting, dynamic warehousing, and automated guided vehicle (AGV) integration—educational and research platforms must simulate mobile manipulation.

Figure 1: Simulation of a desktop-class Dobot platform integrated into an automated production setup.

1.2 Introduction to the Dobot Magician Go and Magician Lite Platforms

The **Dobot Magician Go** is an advanced, omnidirectional mobile platform equipped with Mecanum wheels, specialized for autonomous navigation, path planning, and AIdriven vision tasks. This AGV serves as the mobile foundation for the **Dobot Magician Lite**, a lightweight, 4-axis multi-functional intelligent robotic arm designed for highly precise manipulation tasks. By mounting the Magician Lite architecture onto the Magician Go chassis, the system successfully bridges the gap between locomotion and manipulation.

1.3 Problem Statement

While the combination of the Magician Go and Magician Lite offers excellent horizontal dexterity, it faces a critical physical limitation: **a constrained vertical work envelope**. In an optimized industrial automation setup, shelves, conveyors, and storage bays exist at varying heights. The fixed vertical mounting height of the robotic arm significantly limits its ability to pick items from high racks.

Introducing an external lift presents severe engineering challenges:

- **Power Independence:** Standard additions often require external tethered power supplies or separate battery packs, which ruins the platform's mobility.
- **System Integration:** Third-party add-ons typically operate on independent control systems, creating communication latency.
- **Form Factor:** The lift must be lightweight and compact enough to be mounted directly onto the Magician Go chassis without upsetting its center of gravity.

1.4 Project Objectives and Scope

This FOSSEE internship project focuses on the **design, fabrication, and integration of an on-board scissor lift mechanism**. Specific objectives include:

- **Direct Mechanical Mounting:** Engineering a sturdy, stable scissor lift mechanism that fastens directly onto the chassis.

- **Unified Power Draw:** Eliminating external battery packs by tapping directly into the Magician Go's internal lithium battery system.
- **Direct Internal Control:** Developing a communication bridge enabling the Magician Go's internal systems to directly actuate and coordinate the lift.
- **Secondary Wireless Capability:** Integrating a secondary, independent Wi-Fi control subsystem using the ESP32-C3 for testing and remote overrides.

2 MECHANISM SELECTION & DESIGN METHODOLOGY

2.1 Evolution of Vertical Actuation Mechanisms

In mobile robotics, introducing a vertical degree of freedom (Z-axis translation) radically expands the operational workspace. Common industrial methodologies include telescopic masts, linear rails with lead screws, and scissor lift linkages.

2.2 Comparative Technical Analysis

To determine the optimal mechanism for the Dobot Magician Go platform, a comparative matrix was established based on structural rigidity, collapsed-to-extended height ratio, payload capacity, and spatial footprint constraints.

Mechanical Vector	**Telescopic Column**	**Vertical Linear Rail**	**Scissor Lift Linkage**
Collapsed Height	Moderate	Extremely Poor	**Excellent**
Center of Gravity	High	Exceptionally High	**Low**
Payload-to-Weight	Moderate	Poor	**Excellent**
Footprint Efficiency	Compact but tall	Large structural overhead	**Fits within AGV perimeter**

Table 1: Comparative Analysis of Vertical Actuation Mechanisms.

2.3 Rationale for the Scissor Lift Configuration

The selection of the scissor lift configuration is driven by the strict stability criteria of the Dobot Magician Go chassis. A standard vertical rail column introduces a massive cantilevered moment arm and forces a permanently high center of gravity (CoG), causing tipping hazards.

Figure 2: Industrial scissor lifts demonstrating high vertical extension and structural stability.

When collapsed, the scissor mechanism sits flat on the chassis base, maintaining a low CoG during transit. When deployed, it scales up to reach the necessary elevation to pick items from elevated industrial shelves.

3 MECHANICAL DESIGN & KINEMATIC MODELING

3.1 Geometric Architecture

The designed scissor lift utilizes a single-stage symmetric pantograph linkage system driven horizontally by a stepper motor connected to a precision lead screw.

Let L be the uniform length of each pin-to-pin scissor link arm (mm), θ be the instantaneous angle of the link arms relative to the horizontal base plane (rad), H be the overall operational height of the top platform (mm), and x be the horizontal distance between the sliding and fixed ground pins (mm).

The fundamental structural geometry dictating the total height is governed by:

$$H(\theta) = 2 \cdot L \cdot \sin(\theta) \tag{1}$$

Concurrently, the horizontal separation distance is modeled as:

$$x(\theta) = 2 \cdot L \cdot \cos(\theta) \tag{2}$$

3.2 Kinematic Velocity Analysis

To map the relationship between the linear velocity of the actuation drive ($\dot{x}$) and the vertical deployment velocity of the lift ($\dot{H}$), we take the time derivative of the geometric position equations:

$$\dot{H} = \frac{dH}{dt} = 2 \cdot L \cdot \cos(\theta) \cdot \dot{\theta} \tag{3}$$

$$\dot{x} = \frac{dx}{dt} = -2 \cdot L \cdot \sin(\theta) \cdot \dot{\theta} \tag{4}$$

Solving for $\dot{\theta}$ and substituting yields the instantaneous transmission ratio:

$$\dot{H} = -\dot{x} \cdot \cot(\theta) \tag{5}$$

This velocity relationship is highly non-linear. As the lift approaches its fully collapsed state ($\theta \to 0$), the vertical velocity spikes dramatically, requiring careful software acceleration profiling to avoid motor stall.

3.3 Static Force and Torque Analysis

To accurately size the stepper motor driver, a static force analysis using the **Principle of Virtual Work** was performed. Let W represent the total downward force (N) and F_x be the horizontal force required along the lead screw axis:

$$\delta W_{\text{virtual}} = 0 \implies W \cdot \delta H + F_x \cdot \delta x = 0 \tag{6}$$

$$W \cdot (2L\cos(\theta)\delta\theta) + F_x \cdot (-2L\sin(\theta)\delta\theta) = 0 \tag{7}$$

$$F_x(\theta) = W \cdot \cot(\theta) \tag{8}$$

This proves that as the scissor lift collapses down toward the top plate ($\theta \to 0$), the horizontal force required from the stepper motor asymptotically approaches infinity. Consequently, a physical mechanical limit stop is implemented to prevent the mechanism from dropping below a critical threshold angle ($\theta_{\text{min}} = 15°$).

4 ELECTRICAL ARCHITECTURE & CONTROL SYSTEMS

4.1 Control Topology Overview

A **Dual-Channel Control Architecture** was developed. The primary physical layer interfaces with the Dobot's internal communication bus, and the secondary layer provides wireless telemetry.

4.2 Microcontroller Core: ESP32-C3 Rationale

The computational brain is an **ESP32-C3 RISC-V microcontroller**. It implements an open-source, single-core 32-bit RISC-V processor, selected for its low power consumption, processing performance, and integrated wireless peripherals within an ultra-compact footprint.

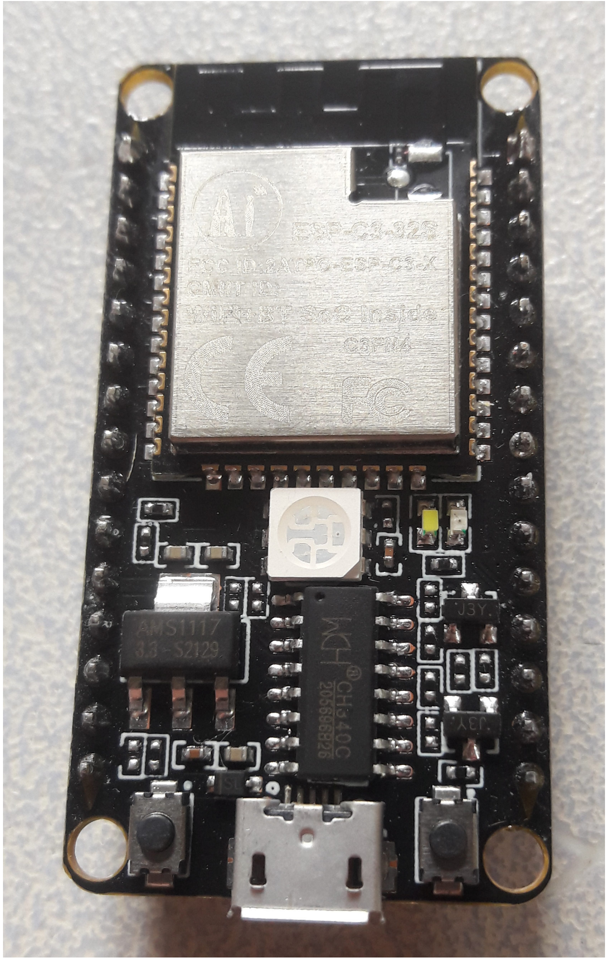


Figure 3: The ESP32-C3 RISC-V development board used for physical execution and processing.

4.3 Wired Communication Subsystem: Dobot Serial Bus Integration

For direct autonomous coordination, the ESP32-C3 is integrated directly into the **Dobot Magician Go internal serial communication bus (UART)**. This bypasses external control interfaces, allowing the robot's main processor to command the lift as a native auxiliary axis.

$$\text{Frame Structure} = [1\ \text{Start}] + [8\ \text{Data}] + [0\ \text{Parity}] + [1\ \text{Stop}] \tag{9}$$

Since both systems operate natively on a 3.3V CMOS logic standard, RX/TX signal lines are tied directly together, eliminating the need for bi-directional logic-level shifters.

4.4 Wireless Actuation Subsystem

In addition to internal bus control, the system runs a secondary wireless protocol stack. The ESP32-C3 establishes an asynchronous HTTP/WebSocket server in SoftAP or Station (STA) mode for low-latency manual control overriding.

4.5 Stepper Step-to-Distance Modeling

To translate structural height requirements (H) commanded by the Dobot into discrete motor step pulses (S_x), the system calculates the required step count by combining lead screw pitch parameters (P) with the native step angle (α) and microstepping divisor (m_s):

$$S_x = \frac{360^\circ \cdot m_s}{\alpha \cdot P} \tag{10}$$

5 STEPPER MOTOR MECHANICS, ELECTROMAGNETICS, AND DRIVER THEORY

5.1 Introduction to Incremental Motion Control

For precise vertical positioning, a standard **NEMA 17** form-factor hybrid stepper motor was implemented. Stepper motors divide a full 360° rotation into a discrete number of equal fractional moves, providing inherent position holding.

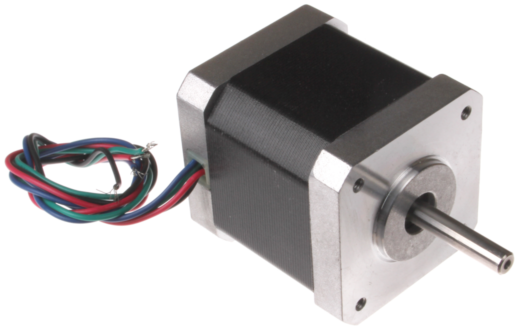


Figure 4: The physical NEMA 17 bipolar stepper motor integrated into the scissor lift.

5.2 Internal Electro-Mechanical Architecture

The internal structure combines variable reluctance and a permanent magnet configuration. The stator features wound electromagnetic cores with precision teeth pointing inward toward the rotor axis. The rotor utilizes a multi-tooth hybrid iron cap.

5.3 Magnetic Commutation

When an electric current flows through a specific stator phase winding, it creates a localized electromagnetic field. The rotor experiences a magnetic reluctance torque that forces its teeth to align perfectly with the energized stator teeth.

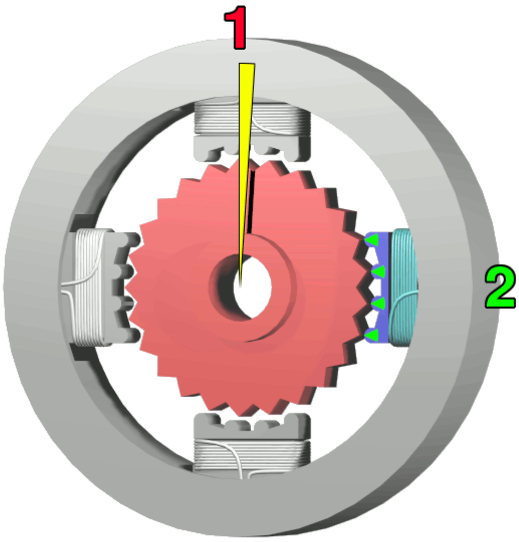


Figure 5: Operational alignment diagram showcasing how localized stator magnetization induces incremental rotational steps.

5.4 Stepper Driver Circuitry: Dual H-Bridge Theory

Because a microcontroller cannot directly output the high current required to drive the motor, a dedicated stepper motor driver utilizing two separate **H-Bridge** circuits handles the phase switching via Power MOSFETs.

Figure 6: Electronic architecture of a two-phase H-Bridge controller.

5.5 Microstepping Theory

To achieve smooth mechanical motion and eliminate low-speed resonance, the driver utilizes microstepping. It feeds the phases with fractional, sine-and-cosine-weighted current levels via high-frequency PWM:

$$I_A = I_{\max} \cdot \cos(\phi) \tag{11}$$

$$I_B = I_{\max} \cdot \sin(\phi) \tag{12}$$

6 CAD DESIGN, MECHANICAL COMPONENTS, AND ADDITIVE MANUFACTURING

6.1 Geometric CAD Modeling

The geometric layout of the scissor lift was modeled parametrically. The structural assembly consists of three distinct 3D printed groups: the bottom base plate, the top platform plate, and the uniform linkage arms.

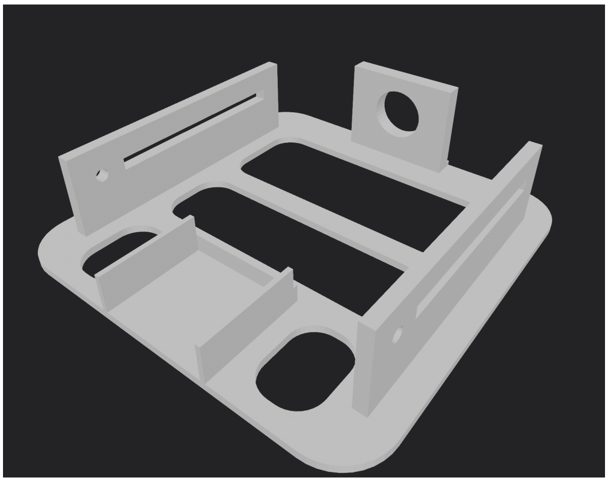


Figure 7: CAD rendering of the main structural base carriage.

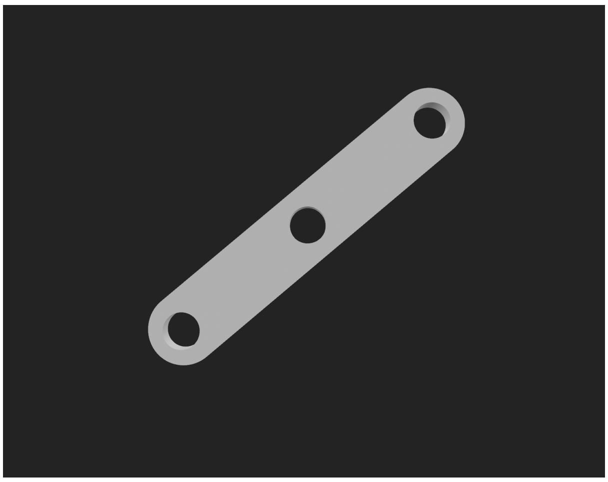


Figure 8: CAD rendering of a symmetric three-hole internal/external scissor linkage arm.

6.2 Linear Actuation Mechanics via Threaded Rod

To convert the rotational torque of the stepper motor into the high-force horizontal linear motion needed to drive the sliding pins, a steel threaded power rod assembly was implemented.

The system must be statically self-locking to prevent back-driving when unpowered. This requires the lead angle (λ) to be limited by the coefficient of friction (μ) and thread angle (α):

$$\mu > \tan(\lambda) \cdot \cos(\alpha) \tag{13}$$

6.3 Friction Minimization: Deep Groove Ball Bearings

To prevent binding at the high-load central and slider pivot joints, miniature radial deep groove ball bearings are pressed directly into the linkage bores, replacing sliding friction with rolling friction.

6.4 Material Selection: PLA+ vs. PETG

The prototype components for this project were fabricated using additive manufacturing. **PLA+** was chosen initially for rapid prototyping due to its dimensional stability. However, for a final industrial automation deployment, the component material should change to **PETG** due to its high ductility under dynamic loading, resistance to cyclic fatigue on the retention clips, and superior thermal stability.

7 PHYSICAL ASSEMBLY, POWER REGULATION, AND SYSTEM VALIDATION

7.1 Physical Integration and Structural Optimization

The base and top plates were designed with lightweight structural cutouts rather than solid blocks to optimize print time and minimize filament consumption, reducing the dead-weight payload on the mobile AGV.

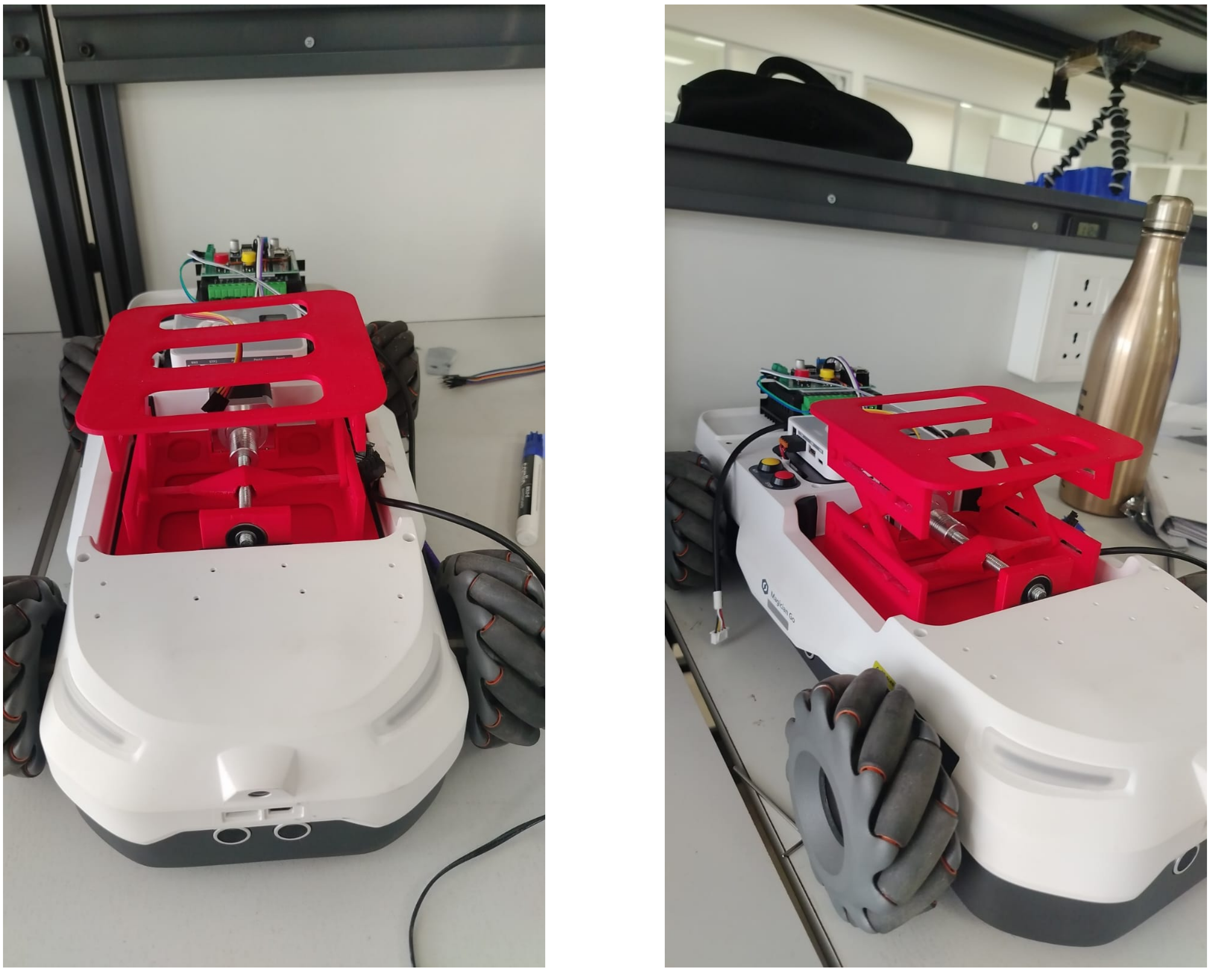


Figure 9: Side profile (left) and isometric view (right) of the assembled lift mechanism.

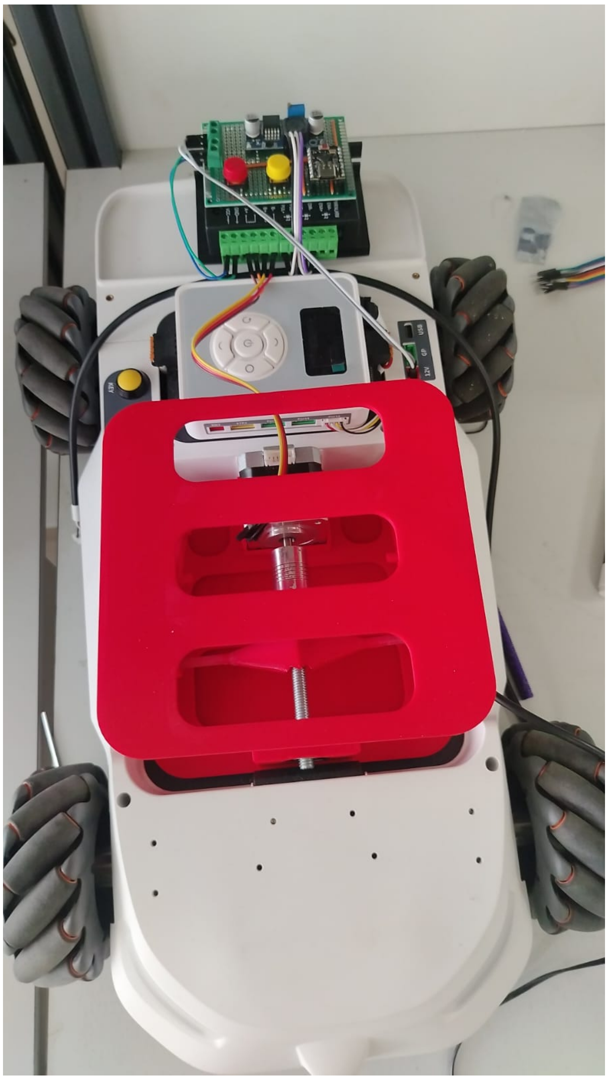
7.2 Power Electronics and Tapping Circuitry

To achieve a completely self-contained mobile system, the lift electronics are powered directly off the Dobot Magician Go's internal battery supply via an adjustable high-efficiency **DC-DC Step-Down Buck Converter**. This switching regulator safely steps down the high voltage to logic levels (5 V/3.3 V) for the ESP32-C3 with minimal heat loss:

$$\eta = \frac{P_{\text{out}}}{P_{\text{in}}} = \frac{V_{\text{out}} \cdot I_{\text{out}}}{V_{\text{in}} \cdot I_{\text{in}}} \tag{14}$$

7.3 Experimental Validation

Testing verified the mechanical performance and integration of the complete assembly during vertical translation tasks.



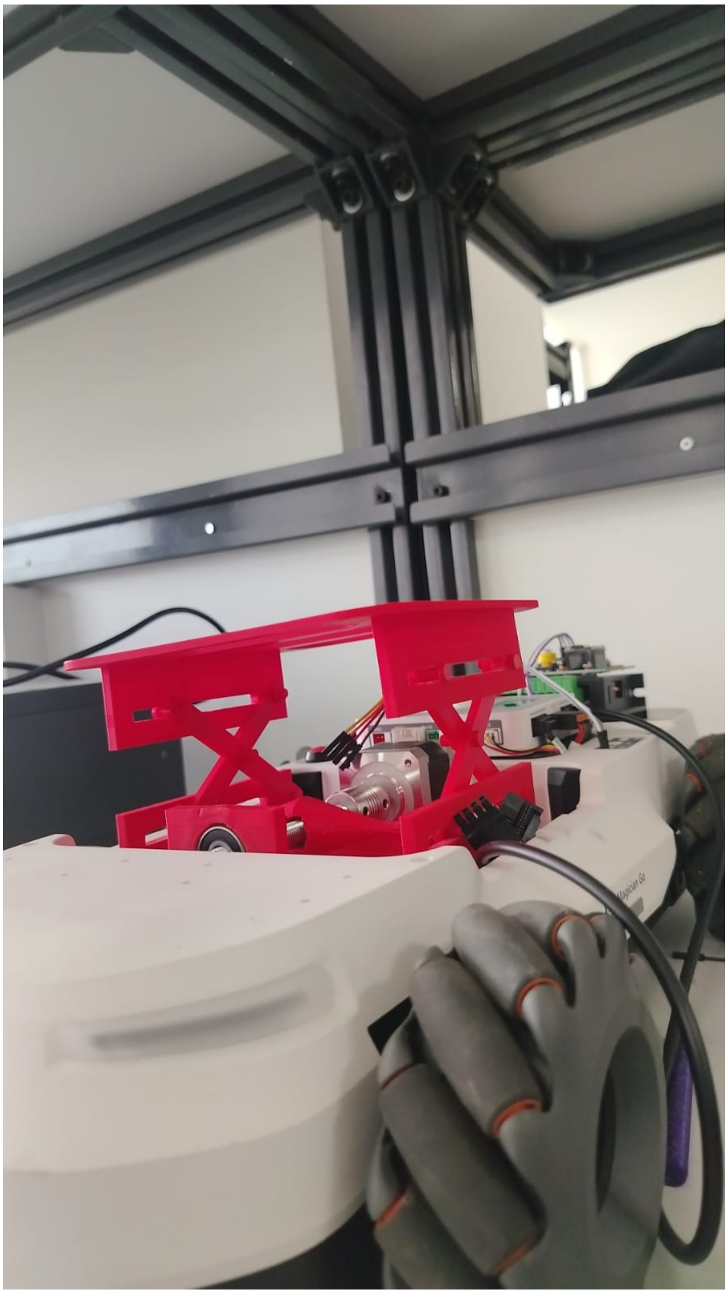


Figure 10: The fully integrated lift system in an extended state, demonstrating successful vertical translation.

The proof of concept successfully met all core performance requirements: wired internal control, low-latency Wi-Fi actuation, and complete power independence.

8 CONCLUSION AND FUTURE SCOPE

8.1 Project Conclusion

The development of the stepper-driven scissor lift for the Dobot Magician Go platform successfully achieved all core objectives of this FOSSEE internship. By bridging the gap between localized robotic manipulation and mobile vertical actuation, the project expands the industrial utility of the Dobot ecosystem. Utilizing an ESP32-C3 microcontroller combined with internal battery integration via a buck converter ensured that the final footprint was completely untethered and mobile. The kinematic design, bearing integration, and dual-communication pipeline validated the mechanism as a highly effective proof of concept.

8.2 Future Scope

While the current prototype functions reliably, transitioning the hardware toward full-scale industrial deployment presents several avenues for future work:

1. **Material Upgrade:** Fully transitioning all 3D-printed structural elements from PLA+ to PETG to ensure long-term cyclic fatigue resistance and thermal durability.

2. **Closed-Loop Feedback:** Upgrading the open-loop stepper drive by integrating rotary encoders on the lead screw or mechanical limit switches on the base plate to enable automatic homing and exact coordinate validation.

3. **ROS2 Integration:** Expanding the ESP32-C3 firmware to directly act as a micro-ROS node, allowing seamless path-planning coordination within larger fleet management environments.

APPENDIX A: ESP32-C3 CONTROL FIRMWARE

The following C++ code utilizes the Arduino framework for the ESP32-C3 microcontroller. It configures the hardware GPIO pins for the stepper motor driver and establishes an asynchronous web server for wireless manual overrides.

```
#include <WiFi.h>
#include <WebServer.h>
#include <AccelStepper.h>

// --- Configuration ---
const char* ssid = "YOUR_WIFI_SSID";
const char* password = "YOUR_WIFI_PASSWORD";

// Pin Definitions
#define STEP_PIN 2
#define DIR_PIN 3
#define BTN_CW 4
#define BTN_CCW 5

// Motor Setup (1 = external stepper driver)
AccelStepper stepper(1, STEP_PIN, DIR_PIN);
float targetSpeed = 800.0; // Steps per second

// Web Server on port 80
WebServer server(80);

// Global state for web commands
enum MotorState { STOPPED, MOVING_CW, MOVING_CCW };
MotorState webState = STOPPED;

// --- HTML for iOS Web App ---
// Contains Apple meta tags to look and feel like a native app
```

```
const char* htmlPage = R"rawliteral(
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
  <meta name="apple-mobile-web-app-capable" content="yes">
  <meta name="apple-mobile-web-app-status-bar-style" content="black-translucent">
  <title>Motor Control</title>
  <style>
    body { font-family: -apple-system, BlinkMacSystemFont, sans-serif; display: flex; flex-direction: column; align-items: center; justify-content: center; height: 100vh; margin: 0; background-color: #1c1c1e; color: white; }
    h1 { margin-bottom: 40px; font-weight: 600; }
    .btn { background-color: #007aff; color: white; border: none; padding: 20px 40px; font-size: 24px; border-radius: 14px; margin: 15px; width: 80%; max-width: 300px; touch-action: manipulation; }
    .btn:active { background-color: #005bb5; }
    .stop-btn { background-color: #ff3b30; }
    .stop-btn:active { background-color: #c92a20; }
  </style>
</head>
<body>
  <h1>Stepper Control</h1>
  <button class="btn" onmousedown="sendCommand('/cw')" ontouchstart="sendCommand('/cw')" onmouseup="sendCommand('/stop')" ontouchend="sendCommand('/stop')">Clockwise</button>
  <button class="btn" onmousedown="sendCommand('/ccw')" ontouchstart="sendCommand('/ccw')" onmouseup="sendCommand('/stop')" ontouchend="
```

```
  sendCommand('/stop')">Anti-Clockwise</button>
  <button class="btn stop-btn" onclick="sendCommand('/stop')">Emergency Stop</button>

  <script>
    function sendCommand(cmd) {
      fetch(cmd);
    }
  </script>
</body>
</html>
)rawliteral";

void setup() {
  Serial.begin(115200);

  // Configure Button Pins
  pinMode(BTN_CW, INPUT_PULLUP);
  pinMode(BTN_CCW, INPUT_PULLUP);

  // Configure Stepper
  stepper.setMaxSpeed(2000.0);
  stepper.setAcceleration(1000.0);

  // Connect to Wi-Fi
  WiFi.begin(ssid, password);
  Serial.print("Connecting to Wi-Fi");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nConnected! IP Address: ");
```

```
  Serial.println(WiFi.localIP());

  // Web Server Routes
  server.on("/", []() {
    server.send(200, "text/html", htmlPage);
  });

  server.on("/cw", []() {
    webState = MOVING_CW;
    server.send(200, "text/plain", "CW");
  });

  server.on("/ccw", []() {
    webState = MOVING_CCW;
    server.send(200, "text/plain", "CCW");
  });

  server.on("/stop", []() {
    webState = STOPPED;
    server.send(200, "text/plain", "STOP");
  });

  server.begin();
}

void loop() {
  server.handleClient();

  // Read physical buttons (LOW means pressed due to INPUT_PULLUP)
  bool cwPressed = (digitalRead(BTN_CW) == LOW);
  bool ccwPressed = (digitalRead(BTN_CCW) == LOW);

```

```
  // Determine current commanded state (Physical overrides Web)
  if (cwPressed || webState == MOVING_CW) {
    stepper.setSpeed(targetSpeed);
    stepper.runSpeed();
  }
  else if (ccwPressed || webState == MOVING_CCW) {
    stepper.setSpeed(-targetSpeed);
    stepper.runSpeed();
  }
  else {
    stepper.setSpeed(0); // Stop moving
  }
}
```

References

[1] Dobot. (n.d.). *Dobot Magician Go User Manual*. Shenzhen Yuejiang Technology Co., Ltd.

[2] Espressif Systems. (2021). *ESP32-C3 Family Datasheet*. Espressif Systems.

[3] Acarnley, P. P. (2002). *Stepping Motors: A Guide to Motor Theory and Practice* (4th ed.). Institution of Engineering and Technology.

[4] Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2005). *Robot Modeling and Control*. John Wiley & Sons.

[5] Texas Instruments. (2019). *Understanding Buck Power Stages in Switchmode Power Supplies*. Application Report SLVA057A.

[6] FOSSEE, IIT Bombay. (n.d.). *Free/Libre and Open Source Software for Education*. Retrieved from `https://fossee.in/`

This link is for my Github Repo

https://github.com/rproot17/Summer-Fellowship-2026-FOSSEE-IIT-Bombay-.git