Mini Crane Project

Brodric Young

December 16, 2025

Contents

List of Tables

List of Figures

1 Project Overview

The goal of this project was to design and build a miniature crane system controlled by an STM32 microcontroller, with the long-term goal of implementing automatic swing compensation for a suspended load. The crane was intended to demonstrate the integration of embedded systems, motor control, sensing, mechanical design, and communication between microcontrollers.

1.1 Github link

The full projects files can be found on Github here:

- Mini Crane Project

1.2 System Overview

The mini crane consists of a wooden mechanical structure, multiple motors, sensors, and two microcontrollers communicating via UART. The system allows manual control through a handheld controller while collecting sensor data for future automation.

1.3 Objectives

- Designing a custom PCB to interface with an STM32 Nucleo board
- Controlling multiple motors (stepper and DC) for crane motion
- Measuring hook position using a rotary encoder
- Detecting load swing using an inertial measurement unit (IMU)
- Using sensors and motors for autonomous swing compensation

2 Specifications

The following parts list outlines the primary electrical, mechanical, and structural components used in the construction of the mini crane system. Components were selected to support motor control, sensor integration, and communication between microcontrollers while remaining modular and serviceable. Emphasis was placed on using commonly available hardware and development boards to allow for rapid prototyping, debugging, and future system expansion.

2.1 Parts List

- STM32 Nucleo development board (main crane controller)
- Custom-designed STM32 breakout PCB (KiCad)
- Arduino microcontroller (handheld controller)
- Two L298N motor driver modules
- NEMA 17 stepper motor (crane rotation)
- Two 10 RPM DC gear motors (winch and trolley drive)
- 1/4 in threaded rod and matching flange nut (trolley linear motion)
- Rotary encoder module (rope length measurement)
- MPU-6050 IMU module (swing detection sensor)
- Handheld controller protoboards with push buttons (motor control), buzzer, and LED (swing compensation indicators)
- UART communication wiring between Arduino and STM32
- 5mm flange coupling connectors (motors)
- 6.35mm flange coupling connectors (threaded rod and rotary encoder)
- Bearing for crane rotation
- Pulleys and rope for winch mechanism
- Wooden structural components (frame and boom)
- 12V power supply (motor driver modules)
- 5V USB power supply for microcontrollers

2.2 System Images and Demonstration Media

This subsection provides external references to visual documentation of the completed mini crane system. High-resolution images showing each major component integrated into the final assembly, as well as a demonstration video of the system in operation, are hosted online in the project Github repository to avoid overloading the report with large media files. These resources offer additional context for the mechanical layout, wiring, and overall system behavior.

- GitHub repository folder containing system images
- Project demonstration video

3 Project Procedures

This section documents the design and implementation of the mini crane system using a combination of pinout diagrams, block diagrams, and written descriptions. Electrical schematics were created in the form of detailed pinout diagrams using Google Sheets to clearly illustrate the connections between the STM32 Nucleo board, motor drivers, sensors, and peripheral devices. This approach allowed for rapid iteration and easy modification as the project evolved.

In addition, a system-level block diagram is provided to describe the overall architecture of the crane, including control flow, power distribution, and communication between subsystems. The written block diagram description complements the visual diagram by explaining the functional role of each subsystem and how they interact during operation. Together, these materials provide a clear representation of both the electrical and logical structure of the project.

Notes:

- The "slew" in the diagrams refers to the NEMA 17 stepper motor
- The "winch" and "trolley" in the diagrams refer to the two 10 RPM DC gear motors
- "L298N [A]" in the diagrams refers to the L298N motor driver module driving the two DC motors
- "L298N [B]" in the diagrams refers to the L298N motor driver module driving the NEMA 17 stepper motor

3.1 Pinout Diagrams

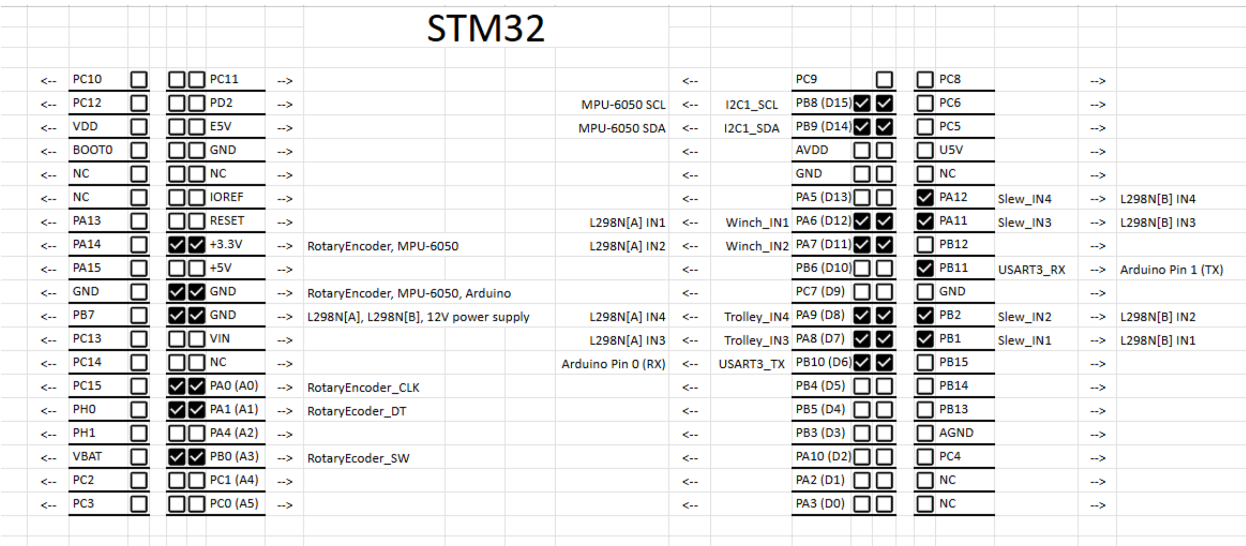

Figure 1: Pinout diagram for the STM32 microcontroller

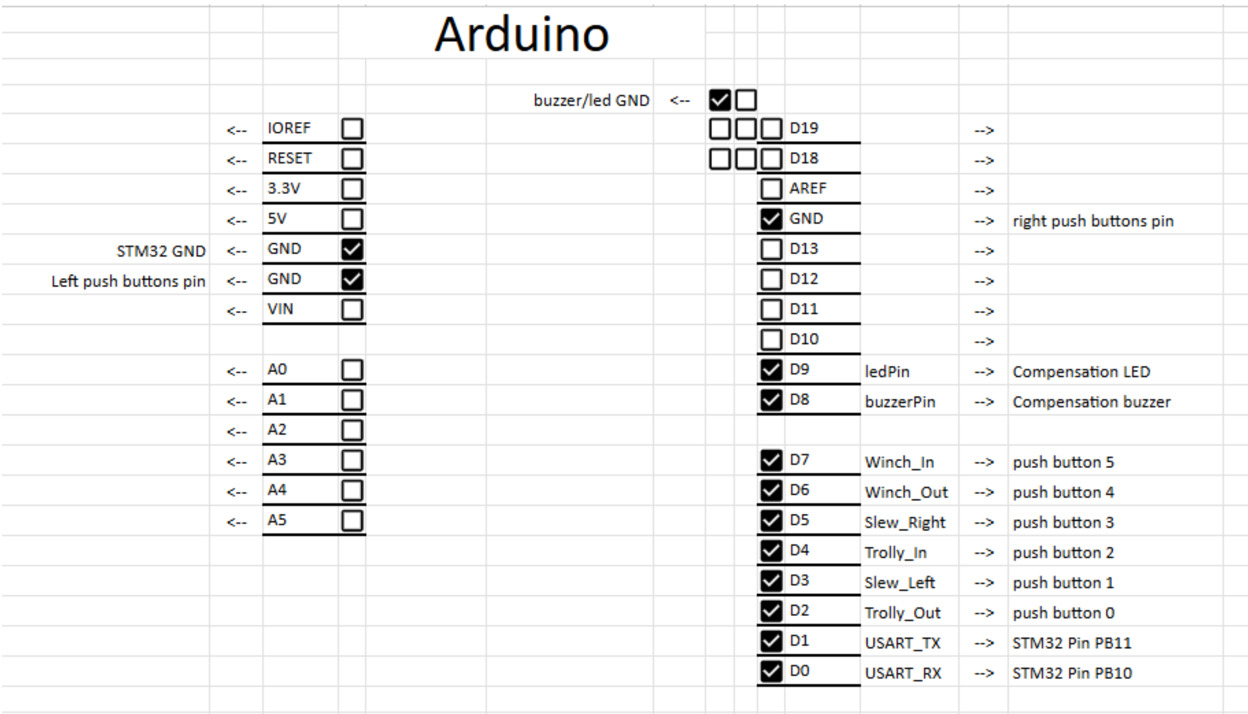

Figure 2: Pinout diagram for the Arduino microcontroller

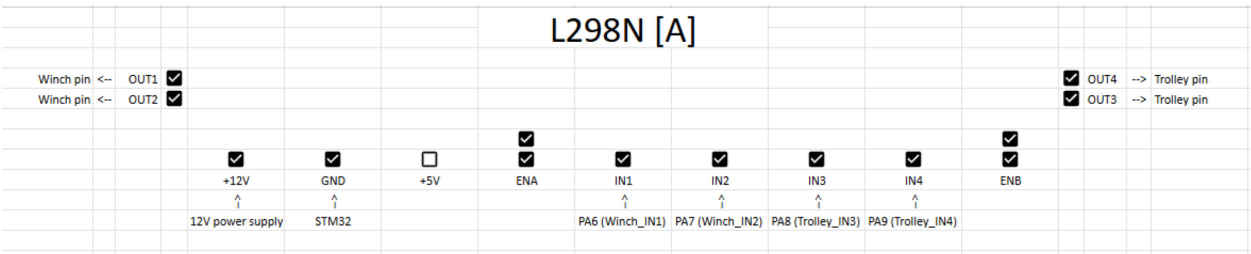

Figure 3: Pinout diagram for the L298N module driving the DC motors

Figure 4: Pinout diagram for the L298N module driving the stepper motor

3.2 Block Diagram

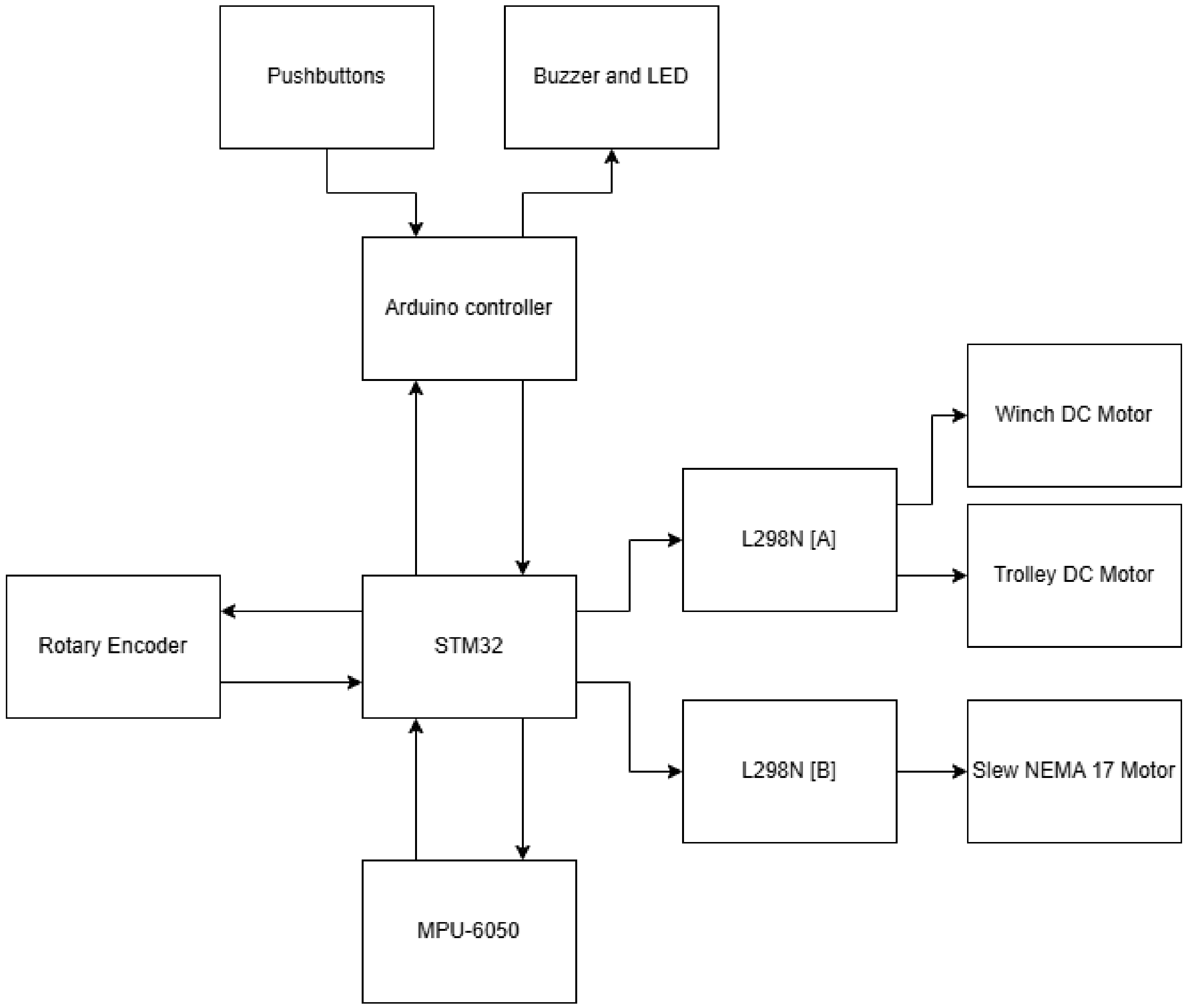

Figure 5: Block diagram of the full system

3.3 Block Diagram Description

- **Handheld Controller**
  - Arduino microcontroller
  - D-pad and additional push buttons
  - Communicates with STM32 via UART
- **STM32 Nucleo Board**
  - Receives commands from handheld controller

- Controls motor driver modules
  - Reads sensor data
  - Outputs diagnostic data via serial terminal (PuTTY)
- **Motor Subsystem**
  - Stepper motor for crane rotation (tread system)
  - DC motor for winch operation (hook up/down)
  - DC motor driving threaded rod for trolley movement
- **Sensor Subsystem**
  - Rotary encoder for rope length measurement
  - MPU-6050 IMU for load swing detection (partially implemented)
- **Indicators**
  - LED indicator for swing compensation status
  - Buzzer for swing compensation notification

4 Code

All source code developed for this project is maintained in a public GitHub repository. Due to the size and modular structure of the codebase, full source listings are not included directly in this report. Instead, this section provides links to the repository and individual source files for reference. This approach allows for easier navigation, version control, and future expansion while keeping the report concise and readable.

4.1 Source Code Repository

The complete source code for the STM32 and the Arduino are available at the following GitHub repositories:

- STM32 code folder
- Arduino code (there is only one file for the arduino)

4.2 STM32 Individual Code Files

- main.c
- main.h
- motors.c
- motors.h
- rotary_encoder.c
- rotary_encoder.h
- arduino_comm.c
- arduino_comm.h
- swing_sensor.c
- swing_sensor.h

5 Conclusion

5.1 Performance Summary

The mini crane system was successfully constructed and demonstrated manual control of multiple crane motions, including hook lifting, trolley movement, and limited base rotation. The rotary encoder functioned correctly and provided real-time feedback of rope length via serial output. Communication between the handheld Arduino controller and the STM32 Nucleo board over UART operated as intended.

5.2 Project Evaluation

The project partially met expectations. While the mechanical structure and electronic control systems operated, several limitations were identified:

- The crane rotation mechanism suffered from alignment and traction issues.
- The stepper motor lacked sufficient torque in some situations.
- The trolley movement was functional but extremely slow due to low motor speed.
- The swing compensation system was not fully implemented due to difficulty reading data from the MPU-6050 sensor.

Despite these challenges, the core system architecture was validated.

5.3 Key Learning Outcomes

This project provided significant learning experiences, including:

- Designing and fabricating a custom PCB independently.
- Working with rotary encoders for position measurement.
- Integrating an IMU for motion sensing.
- Implementing UART communication between microcontrollers.
- Debugging complex electromechanical systems.
- Gaining hands-on experience with mechanical design, alignment, and torque considerations.

Additionally, the project reinforced the importance of careful motor selection and mechanical tolerances.

5.4 Future Improvements

With additional time and resources, the following enhancements would be implemented:

- Successfully implement and calibrate the MPU-6050 swing detection system.
- Complete the autonomous swing compensation algorithm.
- Upgrade motors to higher torque and higher speed models.
- Redesign the crane rotation mechanism for improved alignment and traction.
- Increase mechanical rigidity and optimize pulley systems.
- Refine PCB layout and cable management.

These improvements would move the project closer to a fully autonomous, reliable, and functional mini crane system.