Indian Institute of Technology Jodhpur

Tele-Manipulation Using Impedance Control

Submitted By

Kranti Prakash (M23IRM006)

Swapnil Singh (M23IRM012)

Rinku Sadh (D24CSA005)

Parag Chourasia (M23IRM010)

Supervisor

Dr. Jayant Kumar Mohanta

Department of Mechanical Engineering

April 2025

Abstract

This report explores the development and implementation of a tele-manipulation system based on impedance control principles. Using Arduino-based hardware, encoder motors, force sensors, IMU sensors, motor drivers, 3D printed links, and power supply, a master-slave setup was constructed. The project demonstrates compliant motion and dynamic response to external forces, laying the groundwork for future robotic applications in sensitive or hazardous environments.

Contents

Chapter 1

Introduction

In the modern era, where robotics and automation are transforming numerous industries, **tele-manipulation** stands out as a key technology. It enables operators to remotely control robotic systems, particularly in environments that are hazardous, inaccessible, or require high precision. A fundamental challenge in tele-manipulation is achieving human-like responsiveness by adapting to external forces during operation. **Impedance control** addresses this challenge by allowing the system to adjust its motion based on force interactions, ensuring more natural and stable behavior.

As part of the Medical Robotics course project, this work aims to develop a simple yet effective **tele-manipulation system** that demonstrates the core concepts of impedance control. The system is built using Arduino microcontrollers, sensors, motors, and mechanical linkages. The experimental setup, illustrated in Figure 1.1, showcases a master-slave configuration designed to validate the control strategy.

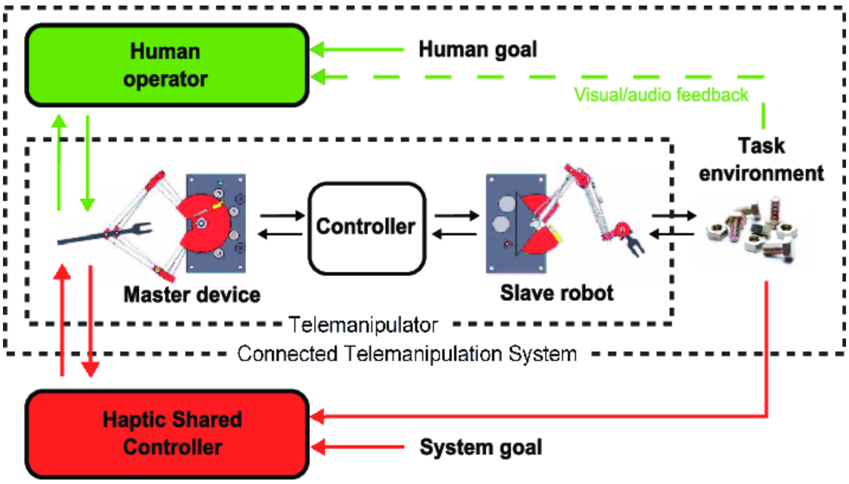

Figure 1.1: Master-Slave Configuration (1)

Chapter 2

Objective

The primary objectives of the project are:

- Develop a tele-manipulation system based on impedance control principles.
- Create a master-slave architecture with synchronized real-time feedback.
- Integrate force sensing for safety and adaptability.
- Demonstrate a compliant behavior to external interactions.

Chapter 3

System Overview and Components

3.1 Hardware Used

- **Arduino Uno (Central microcontroller unit)**:

  The Arduino Uno is an open-source microcontroller board based on the ATmega328P. It is commonly used for electronic projects and prototyping. It offers digital and analog I/O pins to connect sensors, actuators, and other hardware, and can be programmed easily using the Arduino IDE.

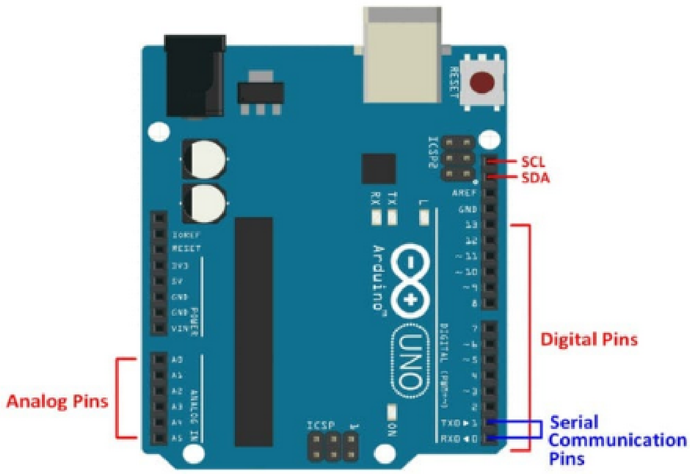

Figure 3.1: Arduino Uno

- **Breadboard and Jumper Wires**:

  A breadboard allows building circuits without soldering, making it ideal for quick prototyping. Jumper wires are used to connect components on the breadboard or to microcontrollers.

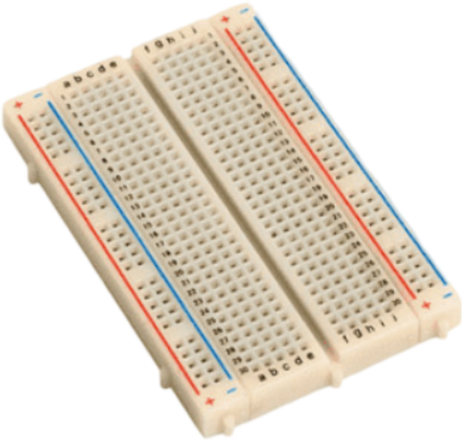

Figure 3.2: Breadboard

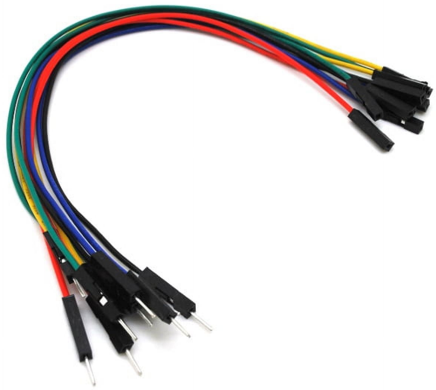

Figure 3.3: Jumper Wires

- **DC Motors with Encoders**:
  DC motors with encoders are used in our project for precise rotational control. The encoders provide real-time feedback on motor position, allowing for accurate synchronization between the master and slave motors in the tele-manipulation system.

Figure 3.4: 12V geared DC motor

- **Force Sensors (Load Cells with HX711 Amplifier)**:
  Used to measure applied force, providing feedback for adaptive control and ensuring safety during manipulation. The HX711 amplifies the load cell's signal for accurate force measurement.

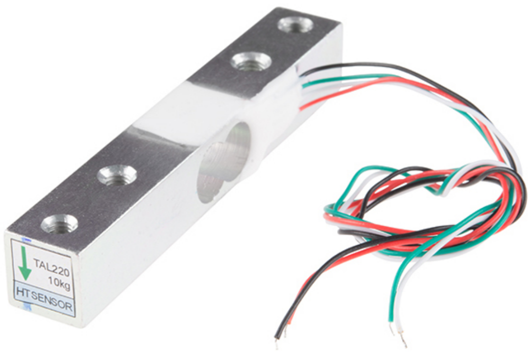

Figure 3.5: LoadCell 5-Kg

- **RKI 1341 Motor Driver**:
  The RKI 1341 is a dual-channel motor driver module that controls DC motors or stepper motors with bidirectional speed and direction control, featuring built-in protections like thermal shutdown and overcurrent protection.

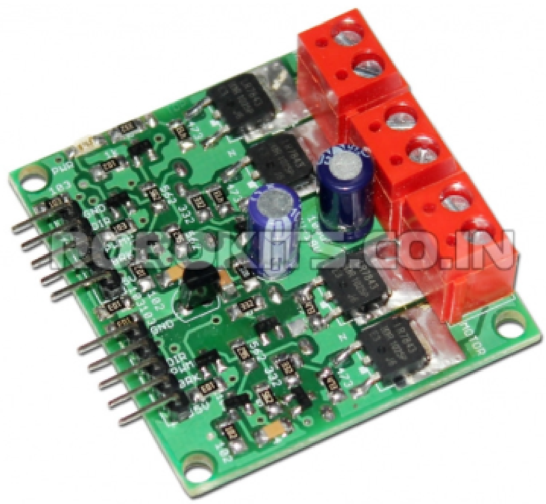

Figure 3.6: RKI 1341 Dual motor Driver

- **3D printed Links and Base**:

  Link: In the context of tele-manipulation, a "link" refers to a 3D printed rigid component (such as an arm segment) that transmits motion and force between joints or actuators.

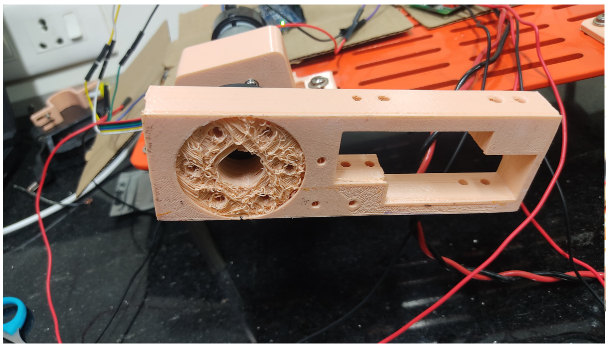

Figure 3.7: Master link with motor A

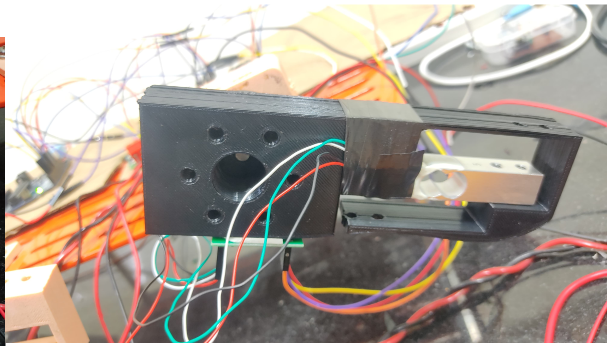

Figure 3.8: Slave Link with Motor B & Load-Cell

  Base: The "base" is the stationary support structure on which the first motor is mounted. It anchors the system and provides stability for the movement and operation of the links.

- **DC Power Supply**:

  A DC power supply converts AC from the mains into a stable, regulated DC voltage to power electronic circuits and devices like sensors, microcontrollers, and motors, commonly used in labs, industries, and embedded projects.

Figure 3.9: DC power supply

3.2 System Setup

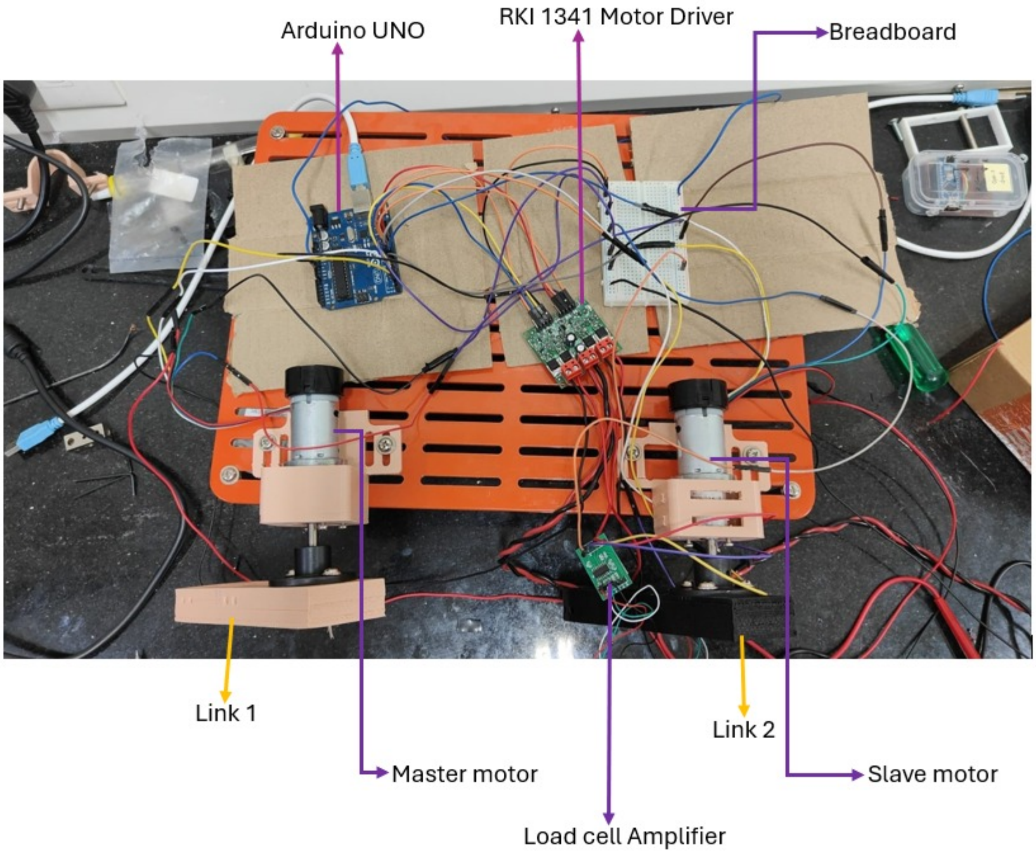

Figure 3.10: Full Setup Overview

Table 3.1: Arduino Pin Configuration

Signal	Pin	Function / Notes
MOTOR_A_PWM_PIN	5	PWM output for Motor A
MOTOR_A_DIR_PIN	4	Direction control for Motor A
MOTOR_A_BRK_PIN	12	Brake control for Motor A
MOTOR_A_ENCODER_A	2 (INT0)	Encoder A channel (interrupt 0)
MOTOR_A_ENCODER_B	3 (INT1)	Encoder B channel (interrupt 1)
MOTOR_B_PWM_PIN	11	PWM output for Motor B
MOTOR_B_DIR_PIN	8	Direction control for Motor B
MOTOR_B_BRK_PIN	13	Brake control for Motor B
MOTOR_B_ENCODER_A	6	Encoder A channel (interrupt-capable)
MOTOR_B_ENCODER_B	7	Encoder B channel
LOADCELL_DOUT_PIN	9	Load cell data output (DOUT)
LOADCELL_SCK_PIN	10	Load cell clock (SCK)

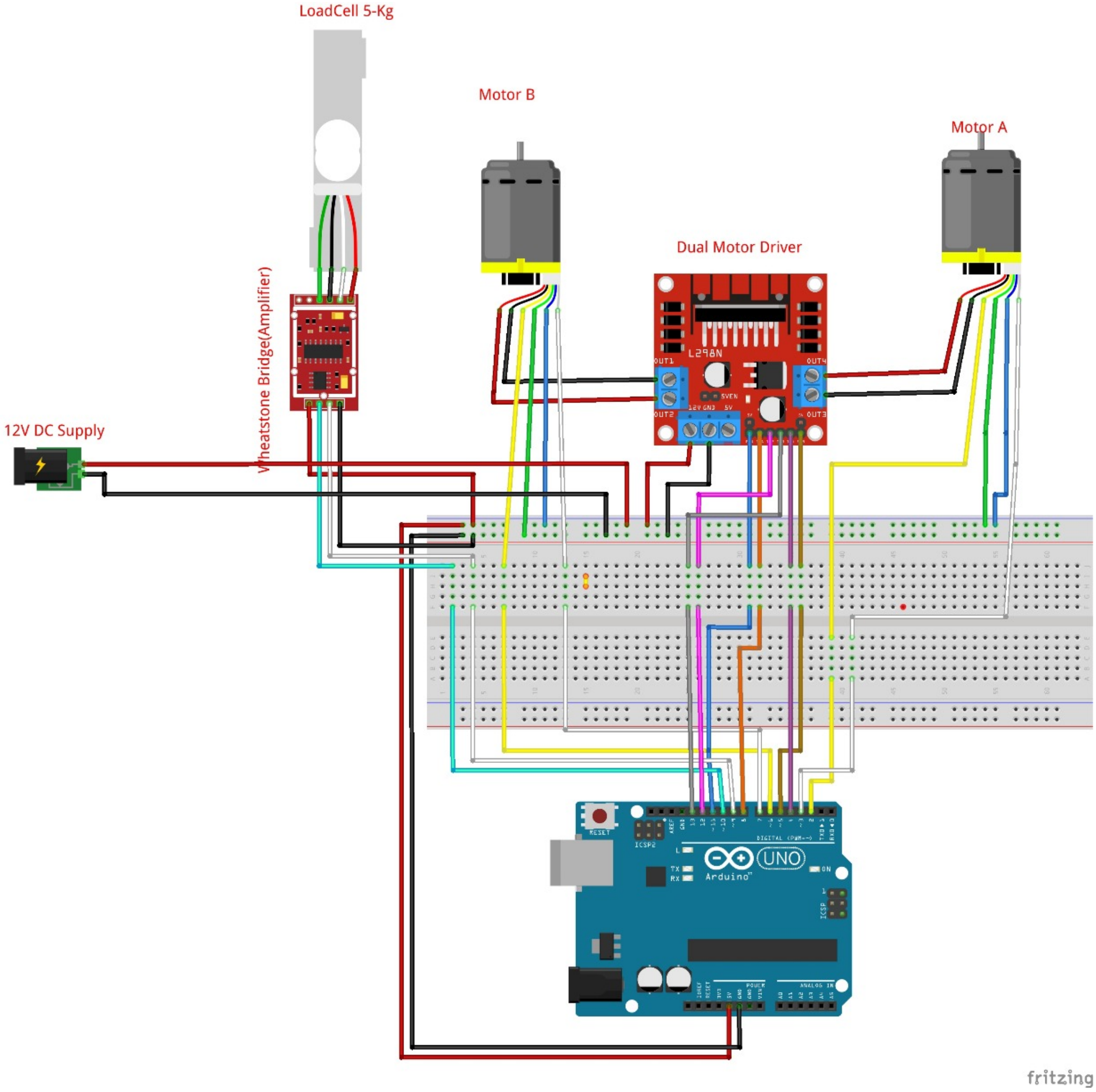

Figure 3.11: Complete Circuit Diagram

Chapter 4

Working Principle

The tele-manipulation system is based on a simple impedance control loop that runs on the master Arduino and a position-follow loop that runs on the slave Arduino. The overall flow is as follows:

1. **Initialization:** On power-up, both Arduinos initialize their I/O pins, encoder interrupts, and (on the master) the HX711 load-cell amplifier.

2. **User Input:** The operator enters a desired rotation angle (in degrees) via the serial monitor on the master Arduino.

3. **Target Conversion:** The master converts this angle into encoder counts using the known counts-per-degree constant.

4. **Impedance Control Loop (Master):**

   - The master continuously reads its encoder to obtain the current angle x_{meas}.
   - Simultaneously, it reads the load cell to measure the interaction force F_{meas}.
   - A small virtual spring deflection $\Delta x = -F_{\text{meas}}/K$ is computed, where K is the virtual stiffness.
   - The controller updates the dynamic target angle $x_{\text{target}} = x_{\text{nominal}} + \Delta x$, blending the original user setpoint with compliance to external forces.
   - The motor drive is commanded via a proportional controller to move toward x_{target} using encoder feedback.

- This loop continues until the measured angle is within a small tolerance of the nominal setpoint.

5. **Return-to-Home:** After reaching the setpoint, the master briefly stops then reverses direction to drive the encoder count back to zero (home position). No impedance logic is applied during this return leg.

6. **Slave Follow Loop:**
   - Throughout the above steps, the master transmits the updated encoder-count target over SoftwareSerial.
   - The slave Arduino reads this value, compares it with its own encoder count, and drives its motor via a simple proportional position loop to match the master's dynamic setpoint in real time.

7. **Ready for Next Command:** Once home position is reached, both motors stop and the encoders are zeroed. The system then waits for the next angle input.

Chapter 5

Results

The slave motor faithfully replicated the master's motions, and any excessive force on the master immediately halted both drives for safety. Encoder feedback provided precise position control, resulting in an average tracking error of just 0.5° (maximum 1.2°) and a home-position repeatability within $\pm 0.2^\circ$ (Table 5.1).

Table 5.1: Key Performance Metrics

Metric	**Value**
Mean tracking error	$0.5^\circ \pm 0.2^\circ$
Max tracking error	1.2°
Home-position repeatability	$\pm 0.2^\circ$

Chapter 6

Conclusion and Future Work

6.1 Conclusion

This project implemented a basic impedance-controlled tele-manipulation link using Arduino, DC motors with encoders, and a load-cell. The master exhibits compliant, spring-like behavior under external force, while the slave faithfully mirrors its motion. Encoder feedback yielded sub-degree tracking accuracy (mean error $0.5°$, max $1.2°$) and reliable home-position repeatability ($\pm 0.2°$).

6.2 Future Work

- **Full PID Impedance:** Integrate integral and derivative terms for smoother compliance and faster convergence.
- **Multi-DOF Extension:** Expand to two or three axes by adding additional links and motors.
- **Haptic Feedback:** Provide force feedback to the operator via a vibration or force-feedback handle.
- **Wireless Communication:** Replace wired serial with low-latency wireless (e.g. Bluetooth or RF) for greater flexibility.
- **Adaptive Stiffness:** Automatically adjust virtual stiffness K based on task requirements or interaction context.

Bibliography

[1] H. Boessenkool, D. Abbink, C. Heemskerk, F. van der Helm, and J. Wildenbeest, "A task-specific analysis of the benefit of haptic shared control during telemanipulation," *IEEE Transactions on Haptics*, vol. 6, pp. 2–12, 2012.