T.C.
IZMIR KATIP CELEBI UNIVERSITY FACULTY OF ENGINEERING AND ARCHITECTURE DEPARTMENT OF MECHATRONICS ENGINEERING

MEE410 INTRODUCTION TO ROBOT TECHNOLOGY PROJECT REPORT

Arda Atak - 210412005

Recep Doğaç Kılıçoğlu – 190412031

Aytekin Dönmez - 190412029

Instructor: Erkin Gezgin

Izmir – 13 January 2024

CONTENTS

A. Introduction

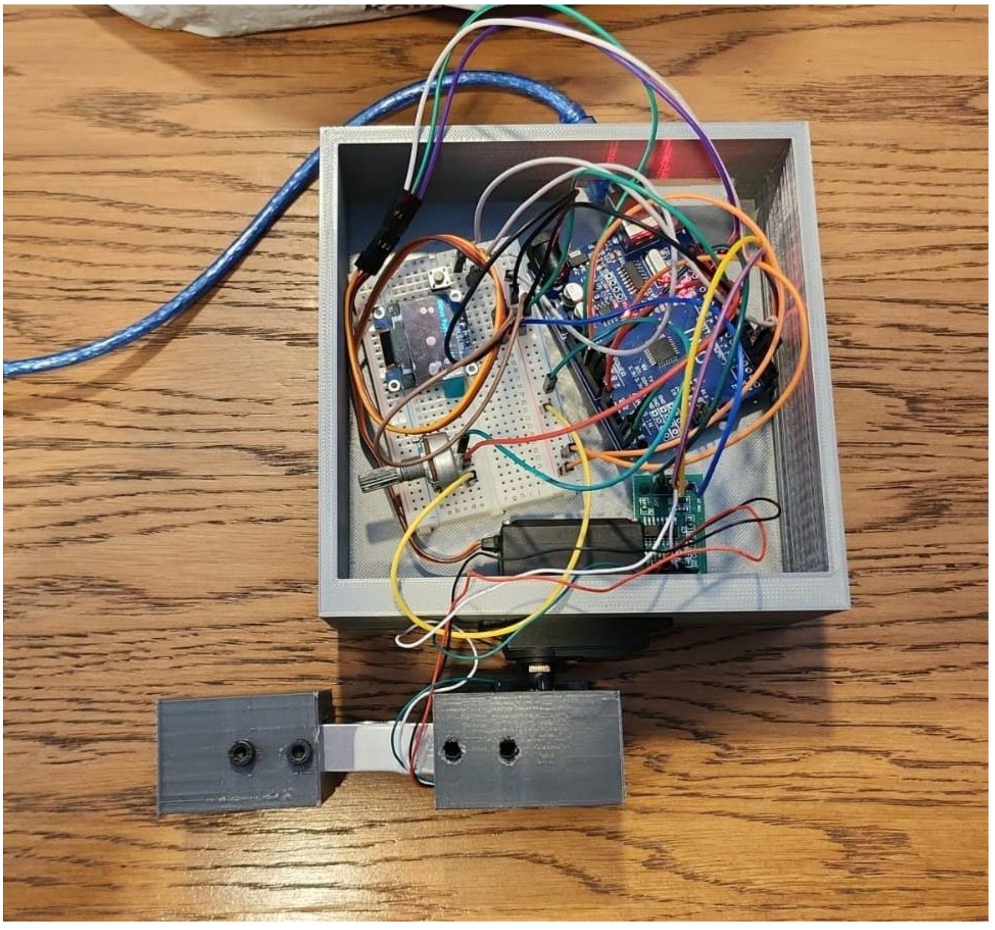
This project focuses on the importance of precise control mechanisms in mechatronic systems, developing an Arduino-based servo motor control system. The primary goal is to provide both force and position control. A potentiometer plays a role in position control. This component detects an angle set by the user, directing the servo motor to move accordingly. Thus, the servo motor precisely guides the attached link to the specific position desired by the user. The use of the potentiometer in this system highlights the importance of precise motion control in mechatronic systems. On the other hand, force control is executed using a load cell. This component measures the force applied to the system and initiates movement when a predefined threshold is reached. If the applied force is below this threshold, the system maintains its current position. This feature is critically important in force-sensitive applications, either to prevent damage to the system or to maintain a specific force level. During the system's operation, the reached position and force values are displayed in real-time on an OLED screen. This screen allows the user to monitor the system's instant status and make adjustments if necessary. This project serves as an example of the integration of technologies used in the control of mechatronic systems, demonstrating how these components work together. Overall, this project represents a significant contribution to the design and control of mechatronic systems, combining theoretical knowledge with practical applications to enhance engineering skills in this field.

B. Manufacturing of the Design

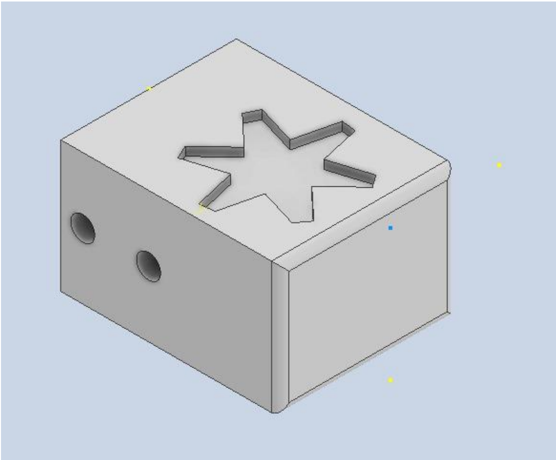
1. Design of Assembly

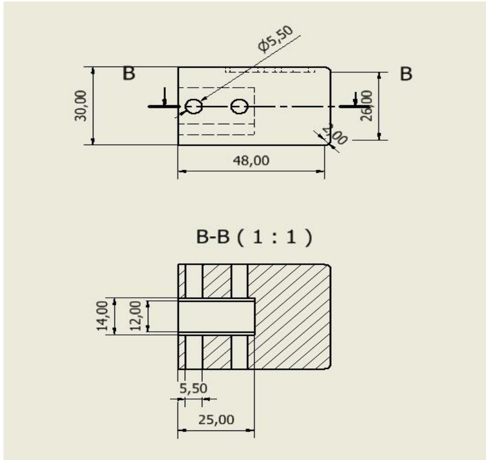
For our manufacturing design, we utilized a 3D printer to fabricate both a casing and a linkage structure. The 3D-printed casing serves as an electrical enclosure housing the Arduino UNO, cables, HX711 load cell amplifier, and a mini breadboard. Additionally, it provides a stable foundation for the servo motor we employed. The linkage structure, printed in two separate pieces, was designed with the intention of precisely positioning the load cell in the center. This configuration allows for the application of force to the linkage, inducing tension on the load cell and enabling accurate weight measurement.

The linkage was securely affixed to the servo motor using a dedicated servo motor apparatus. Through the implementation of our Arduino code, the servo motor is capable of manipulating the linkage in response to applied force, demonstrating its ability to respond dynamically to external forces for precise control and measurement.



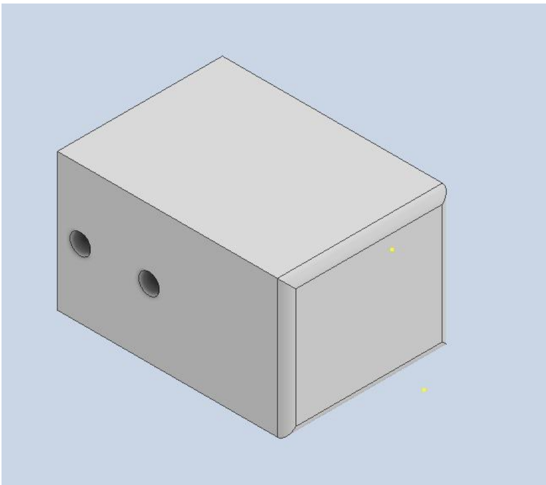
Assembly

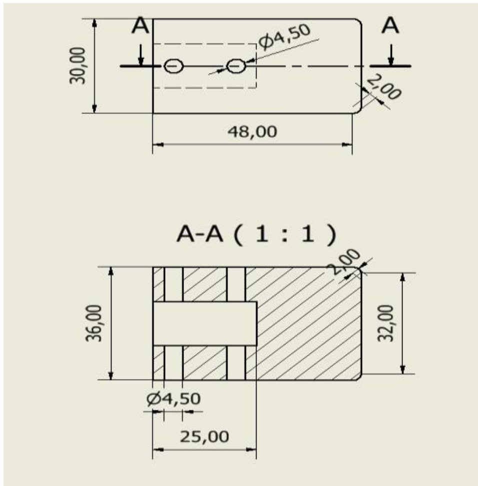




The Part of the Link Where the Servo Motor Will Be Connected

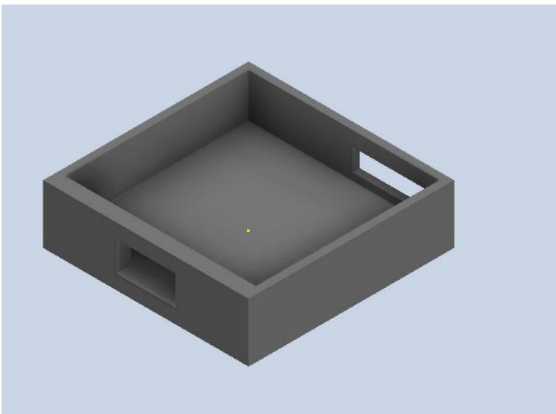
Technical Drawing

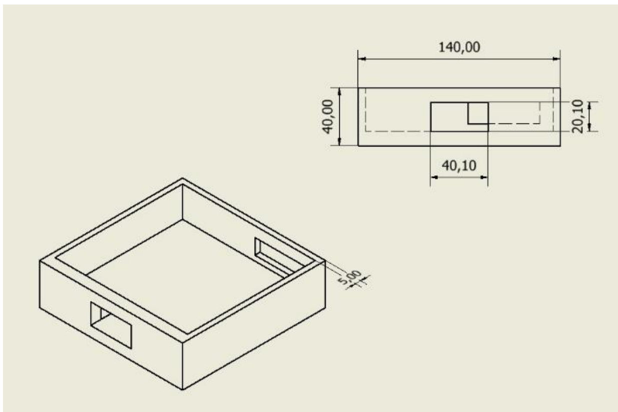




The Other Part of the Link

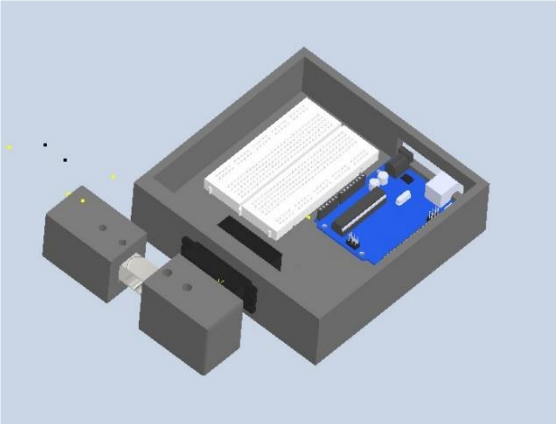
Technical Drawing



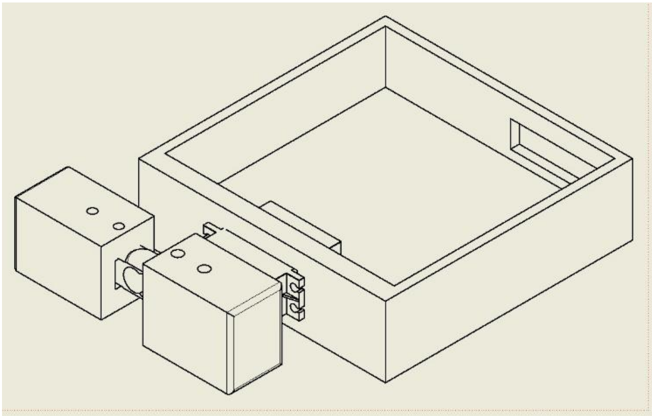


Designed Box

Technical Drawing



Assembly



Technical Drawing

2. Design of Software

In this project, we have two tasks: force control and position control. To accomplish these tasks, we require a load cell and a potentiometer. Additionally, we are employing an OLED display to indicate the current position of the servo motor. The Arduino code for our project is as follows:

File Edit Sketch Tools Help

Arduino Uno

sketch_jan14a.ino

```
#include <Servo.h>
#include "HX711.h"
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

// OLED display settings
#define SCREEN_WIDTH 128 // Width of the OLED display in pixels
#define SCREEN_HEIGHT 64 // Height of the OLED display in pixels
#define OLED_RESET     -1 // Reset pin is not used
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

// Potentiometer settings
int potPin = A0;
int potValue = 0;

// Servo motor settings
Servo servoM;
int servoPos = 0;

// Load cell settings
const int LOADCELL_DOUT_PIN = 3;
const int LOADCELL_SCK_PIN = 2;
HX711 scale;
float loadCellValue = 0;

// Calibration factor
float calibration_factor = 2.207;

// Control modes and threshold values
bool loadCellControlActive = false;
float loadCellControlThreshold = 5.0;
float loadCellReleaseThreshold = 2.0;
```

File Edit Sketch Tools Help

Arduino Uno

sketch_jan14a.ino

```
float loadCellReleaseThreshold = 2.0;

void setup() {
  Serial.begin(9600);
  servoM.attach(9);

  scale.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);
  scale.set_scale(calibration_factor);
  scale.tare();

  // Initialize OLED display
  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) { // OLED display I2C address
    Serial.println(F("SSD1306 allocation failed"));
    for(;;);
  }
  display.clearDisplay();
  display.setTextSize(1);      // Normal 1:1 pixel scale
  display.setTextColor(SSD1306_WHITE); // White text color
}

void loop() {
  // Read the load cell value
  loadCellValue = scale.get_units();

  // Check the load cell control mode
  if (loadCellValue > loadCellControlThreshold) {
    loadCellControlActive = true;
  } else if (loadCellValue < loadCellReleaseThreshold) {
    loadCellControlActive = false;
  }

  // If load cell control mode is active, control the servo motor based on load cell data
  if (loadCellControlActive) {
```

File Edit Sketch Tools Help

Arduino Uno

sketch_jan14a.ino

```
  } else if (loadCellValue < loadCellReleaseThreshold) {
    loadCellControlActive = false;
  }

  // If load cell control mode is active, control the servo motor based on load cell data
  if (loadCellControlActive) {
    // Add logic here to control the servo motor based on the load cell data
    // For example, you can set the position of the servo motor based on loadCellValue
    // servoM.write(positionBasedOnLoadCell);
  } else {
// Otherwise, control the servo motor based on the potentiometer data
potValue = analogRead(potPin);
servoPos = map(potValue, 0, 1023, 0, 180);
servoM.write(servoPos);
}

// Update the position value on the OLED display
display.clearDisplay();
display.setCursor(0,0);
display.print("Servo Position: ");
display.println(servoPos);
display.display();

// Print values to the serial monitor (for debugging)
Serial.print("Potentiometer Value: ");
Serial.print(potValue);
Serial.print(" Servo Position: ");
Serial.print(servoPos);
Serial.print(" Load Cell Value: ");
Serial.println(loadCellValue);

delay(15);
}
```

Code Description:

1. Libraries:

- ‘Servo.h’: This library is used to control servo motors.
- ‘HX711.h’: This library is used to communicate with the load cell and obtain weight measurements.
- ‘Wire.h’: This library supports the I2C communication protocol.
- ‘Adafruit_GFX.h’ and ‘Adafruit_SSD1306.h’: These libraries are used to control the OLED display.

2. OLED Display Settings:

- Configure the width (SCREEN_WIDTH), height (SCREEN_HEIGHT), and reset pin (OLED_RESET) of the OLED display.
- Create a ‘display’ object with the ‘Adafruit_SSD1306’ class.

3. Potentiometer Settings:

- Specify that the signal pin of the potentiometer is A0.
- Use the ‘potValue’ variable to store the read value of the potentiometer.

4. Servo Motor Settings:

- Set the signal pin of the servo motor to 9.
- Use the ‘servoPos’ variable to store the position of the servo motor.

5. Load Cell Settings:

- Specify the DOUT and SCK pins of the load cell.
- Initialize the ‘scale’ object with the HX711 library and assign a calibration factor.
- Use the ‘calibration_factor’ variable to ensure accurate measurements from the load cell.
- The ‘loadCellValue’ variable stores the weight value obtained from the load cell.

6. Control Modes and Threshold Values:

- The ‘loadCellControlActive’ variable indicates whether the load cell control mode is active or not.
- The ‘loadCellControlThreshold’ and ‘loadCellReleaseThreshold’ variables specify the weight threshold values for starting and ending the control mode.

7. **Setup Function:**

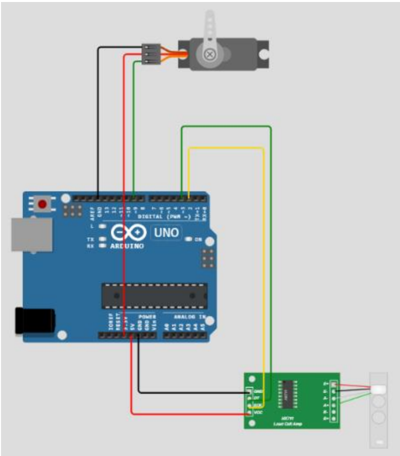
- Initialize serial communication (9600 bps).
- Attach the servo motor.
- Initialize the load cell and perform a reset.
- Initialize the OLED display and check for any initialization errors.

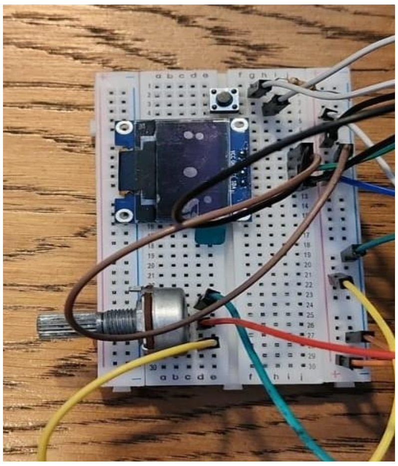
8. **Loop Function:**

- Read the weight value from the load cell.
- Determine the load cell control mode.
- If the load cell control mode is active, control the servo motor based on the load cell data.
- If the control mode is not active, control the servo motor based on potentiometer data.
- Update the servo motor position on the OLED display.
- Print values to the serial monitor.

3. Design of Electronics

In this project, we utilized an Arduino UNO, a Servo Motor, a Load Cell, an HX711 load cell amplifier, an OLED display, and a potentiometer. Initially, we established and soldered the connections between the load cell and the HX711 load cell amplifier. Subsequently, we accurately integrated the Servo Motor with the Arduino UNO using the appropriate connection pins. The Servo Motor was supplied with a 3.3V power source. Following this, we established the connection between the HX711 load cell amplifier and the Arduino UNO. Afterwards, we positioned a potentiometer and an OLED display on a mini breadboard, completing the necessary connections for both. This configuration allowed us to adjust the future position of our servo motor using the potentiometer whenever force is applied to the load cell.





Electrical Design

C. How Does the Design Work ?

This project focuses on the importance of precise control mechanisms in mechatronic systems, developing an Arduino-based servo motor control system. The primary goal is to provide both force and position control. A potentiometer, used for position control, detects an angle set by the user and directs the servo motor to move accordingly. This ensures that the servo motor precisely guides the attached link to the desired position. The use of the potentiometer highlights the importance of precise motion control in mechatronic systems. For force control, a load cell measures the applied force and initiates movement when a predefined threshold is reached, maintaining the current position if the force is below this threshold. This is crucial in force-sensitive applications to prevent damage or maintain specific force levels. During operation, position values are displayed in real-time on an OLED screen, allowing users to monitor and adjust the system's status.

For our manufacturing design, we used a 3D printer to create a casing and a linkage structure. The casing, housing the Arduino UNO, cables, HX711 load cell amplifier, and a mini breadboard, also provides a stable base for the servo motor. The linkage structure, designed to position the load cell precisely at the center, enables accurate weight measurement when force is applied. It's securely attached to the servo motor, allowing for dynamic response to external forces through our Arduino code.

We utilized an Arduino UNO, Servo Motor, Load Cell, HX711 load cell amplifier, OLED display, and potentiometer for this project. Connections were established and soldered between the load cell and the HX711 amplifier, and the Servo Motor was integrated with the Arduino UNO. A potentiometer and an OLED display were positioned on a mini breadboard, completing the necessary connections. This setup allows for the adjustment of the servo motor's position using the potentiometer in response to force applied to the load cell.

D. Bill of Materials (BOM)

Material	Quantity	Price
Load Cell	1	77,40 TL
HX711 Amplifier	1	20.14 TL
Arduino UNO	1	183,13 TL
Servo Motor	1	164,82 TL
M5 Imbus Bolt	2	15.00 TL
M4 Imbus Bolt	2	15.00 TL
Mini Breadboard	1	25.00 TL
I2C SSD1306 OLED Display	1	86.99 TL
10K Potentiometer	1	6,52 TL
	TOTAL	594 TL

E. Conclusion

This project has successfully demonstrated the integration and application of precise control mechanisms in a mechatronic system using an Arduino-based framework. The innovative implementation of a potentiometer for position control and a load cell for force control within this system highlights the significant advancements in precision and adaptability in engineering design. The servo motor, a pivotal component of this system, showcases remarkable accuracy and efficiency in reaching the user-defined positions, underlining the effectiveness of the potentiometer in real-world applications. Moreover, the incorporation of a load cell has provided an essential capability for force-sensitive tasks, ensuring that the system operates within safe force thresholds, thereby enhancing the system's reliability and application scope. Furthermore, the real-time display of position and force metrics on the OLED screen has not only added a layer of user engagement but also provided a critical tool for immediate feedback and system adjustment. This feature underscores the project's commitment to user-centric design and operational transparency. In conclusion, this project embodies a significant stride in the field of mechatronic systems. It not only bridges the gap between theoretical knowledge and practical application but also paves the way for future innovations in the domain. The system's ability to provide precise control in both position and force aspects serves as a testament to the potential of integrated technologies in revolutionizing the field of mechatronics. This project thus stands as an exemplary model for future endeavors in the engineering domain, inspiring continued exploration and advancement in the mechatronic systems.