

Robotic Arm Project Documentation

**Team Members and Roles**

* Team Member 1: Oladosu Olawale (Project Lead):

- Overall project coordination

- Timeline management

- Documentation lead

* Team Member 2: Onah Maxwell Chiedozie (Hardware Specialist):

- 3D printing coordination

- Mechanical assembly

- Material procurement

* Team Member 3 Sogbere Jim-jaja George (Electronics):

- Circuit design

- Component testing

- Power management

* Team Member 4 Amaddin Iyobosa Majid (Programming):

- Arduino code development

- Servo control programming

- Testing protocols

* Team Member 5 Ojima Noah (Design & Testing):

- 3D design

- Testing procedures

- Quality assurance

# Project Overview

The **Robotic Arm Project** is designed to create a functional and versatile robotic arm that can perform precise and controlled movements, with the potential for various applications across industries. This project aims to blend mechanical engineering principles with programming and electronics to build a robotic arm capable of completing complex tasks autonomously or semi-autonomously. The robotic arm will mimic the functionality of a human arm, with several degrees of freedom provided by multiple joints, enabling it to handle objects with precision and adapt to diverse operational environments.

## Objectives

The primary objective of this project is to develop a robotic arm that:

* Demonstrates accuracy and control in its movements.
* Performs repetitive or complex tasks reliably, with minimal human intervention.
* Can be easily customized and scaled to suit specific applications or modified to add more features, such as sensors or grippers.
* Incorporates programmability and flexibility to allow for various control options, from manual joystick control to automated sequences via programming.

## Potential Applications

Robotic arms are widely used in fields such as manufacturing, medicine, agriculture, and research due to their ability to perform tasks that require high precision and endurance. Here are some specific applications for this robotic arm:

1. Manufacturing and Assembly: Picking, placing, assembling, and packaging products on production lines.
2. Medical and Laboratory Assistance: Conducting repetitive tasks in labs, like moving test tubes, handling delicate equipment, or assisting with surgical procedures.
3. Education and Research: Serving as an educational tool in engineering and robotics courses, allowing students and researchers to experiment with control systems, programming, and hardware.
4. Agriculture: Assisting in crop harvesting or precision spraying, which requires a steady and repetitive approach.
5. Home Automation and Assistance: Helping with tasks in home automation, especially for people with disabilities who may need assistance with reaching or grasping objects.

## Key Components of the Project

The project involves integrating various subsystems and components, each of which plays a critical role in the arm’s operation:

* Mechanical Structure: The physical build of the robotic arm, including its base, arm segments, joints, and end effector (such as a gripper). This structure is designed to provide stability and allow for a range of motion similar to that of a human arm.
* Control System: At the core of the control system is the Arduino microcontroller, which processes input signals and controls the servo motors that drive each joint. This control system is programmed to coordinate the movement of the robotic arm, ensuring smooth and precise operation.
* Power System: The power source, including 18650 batteries and a voltage regulator, provides consistent and reliable energy to the entire system. Proper power management is crucial to maintain stable performance and avoid interruptions during operation.
* Programming and User Interface: The Arduino is programmed to control the robotic arm’s movements. The program includes sequences for basic tasks and possibly interfaces like a remote control or smartphone app, allowing users to input commands easily.

## 

## Document Outline

This document provides a comprehensive look at the project specifications, system architecture, components list, testing procedures, troubleshooting logs, and project schedule:

1. Project Specifications: Detailed technical specifications, including dimensions, material requirements, and movement capabilities, ensure that the robotic arm meets design expectations.
2. System Architecture: A layout of the robotic arm's design, illustrating how each component interacts within the control and power systems.
3. Components List: A thorough list of each part used, from the power source to the servo motors, detailing specifications and reasons for selection.
4. Testing Procedures: Step-by-step methods for verifying that each part of the arm functions correctly, from individual motor tests to full-range motion tests.
5. Troubleshooting Logs: Documentation of any issues encountered during assembly and programming, as well as the solutions implemented to resolve them.
6. Project Schedule: A timeline for each phase of development, from initial planning and component sourcing to testing and final assembly.

## Project Goals and Vision

The vision for this project is to create a robust robotic arm prototype that demonstrates the team’s ability to integrate hardware and software into a cohesive, functional system. This robotic arm will showcase practical applications and serve as a foundation for future developments in robotics, where it can be adapted or expanded with additional capabilities such as vision systems or AI-driven controls. Ultimately, the project serves as a hands-on learning experience, strengthening the team’s understanding of robotics, control systems, and real-world engineering challenges.

# System Architecture

The robotic arm system architecture is composed of two primary sections: Power System and Control System.

## Power System

## **Components**:

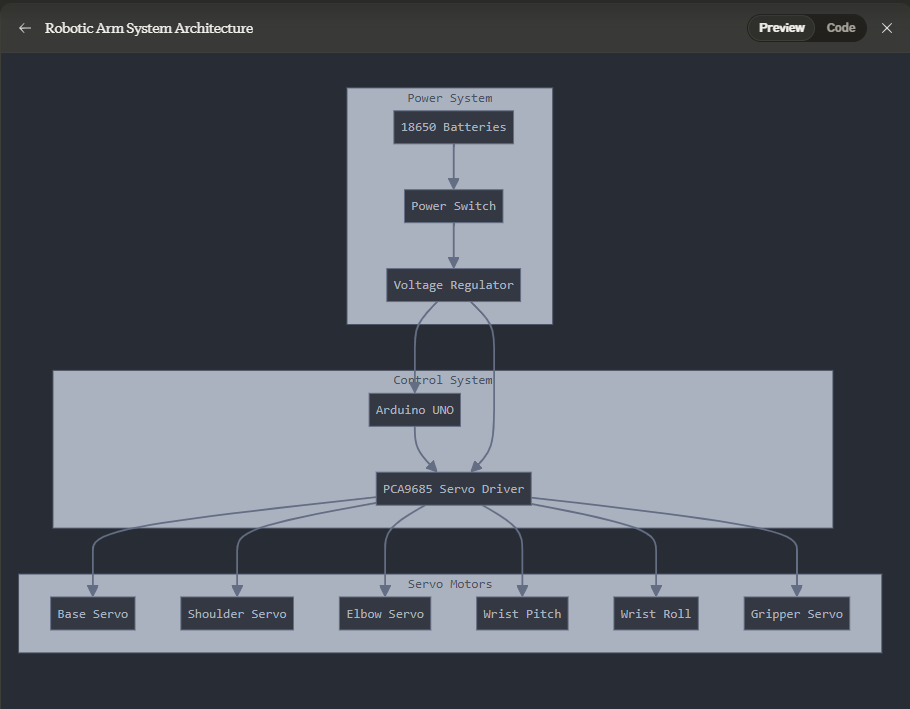
* + **18650 Batteries**: Provides power for the entire system.
  + **Power Switch**: Allows control over power flow to the system.
  + **Voltage Regulator**: Stabilizes the voltage supplied to ensure safe operation.

## Control System

**Components**:

* + **Arduino UNO**: Acts as the main controller, handling commands and sending signals to the servo driver.
  + **PCA9685 Servo Driver**: Enables control of multiple servo motors, facilitating coordinated arm movement.
  + **Servo Motors**:
    - **Base Servo**: Controls the rotation of the base.
    - **Shoulder Servo**: Responsible for the shoulder movement.
    - **Elbow Servo**: Controls the elbow joint for forearm positioning.
    - **Wrist Pitch** and **Wrist Roll**: Allows wrist rotation and bending.
    - **Gripper Servo**: Enables grasping and releasing of objects.

Below is the **"Robotic Arm System Architecture"** [*diagram 1*](#_48wyk5xqiyh7) providing an overview of the core components and their interactions within the robotic arm system, illustrating the flow of power, control signals, and user inputs.



##### Diagram 1: Robotic Arm System Architecture

# Technical Specifications

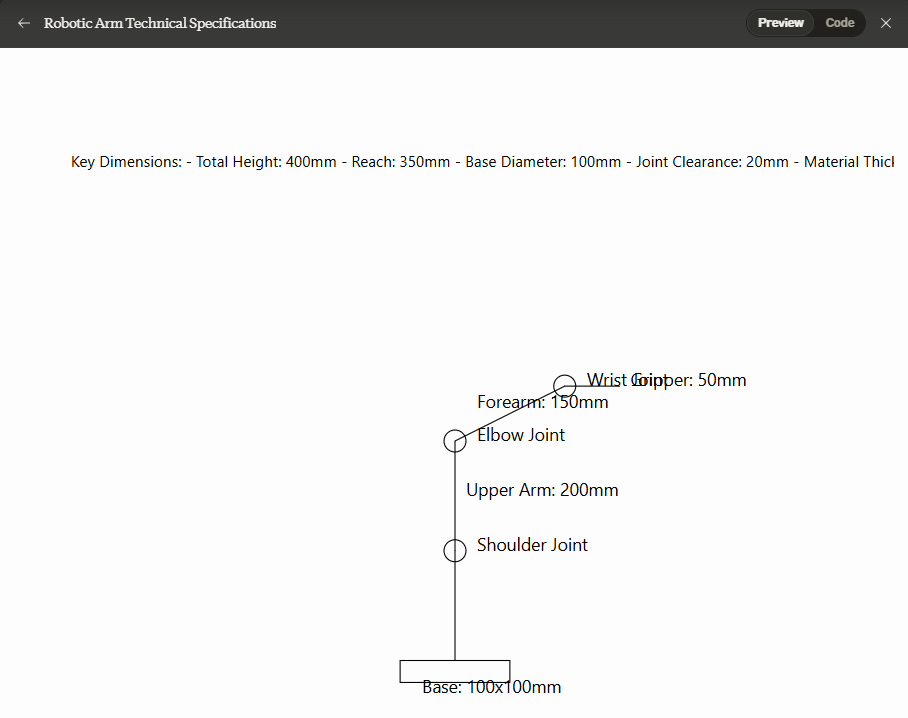
| **Specification** | **Value** |
| --- | --- |
| Total Height | 400mm |
| Reach | 350mm |
| Base Diameter | 100mm |
| Joint Clearance | 20mm |
| Material Thickness | To be specified |

## **Arm Segments**:

## Upper Arm Length: 200mm

## Forearm Length: 150mm

## Wrist Gripper Length: 50mm



##### Diagram 2: Robotic Arm Technical Specifications

# Component List

1. Power Supply

* 2x 18650 Batteries (3.7V): Provide the main power source for the robotic arm. The batteries offer a stable voltage for extended operation.



##### 2x 18650 Batteries (3.7V)

* 1x 18650 Battery Holder (2-cell): Holds the two 18650 batteries securely in place and connects them to the circuit.



##### 1x 18650 Battery Holder (2-cell)

* 1x Battery Charger: Used to recharge the 18650 batteries when they run out.



##### 1x Battery Charger

* 1x Power Switch: Allows manual control to turn the robotic arm on and off.



##### 1x Power Switch

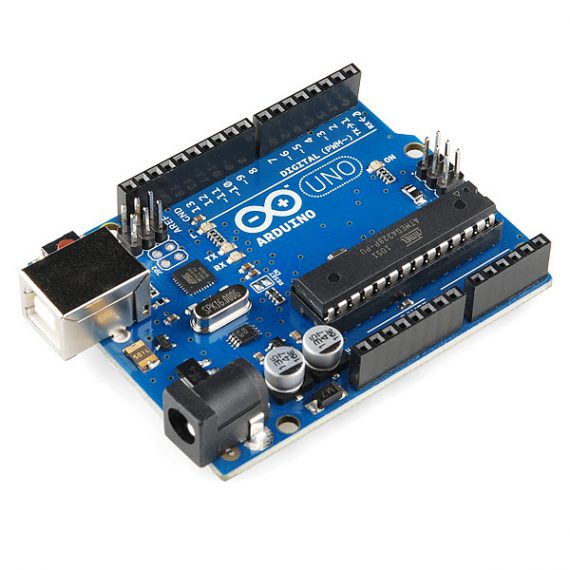
* Voltage Regulator: Ensures a stable voltage level to prevent power surges or drops that might damage sensitive components.



##### 1x Voltage Regulator - Buck Converter

2. Microcontroller

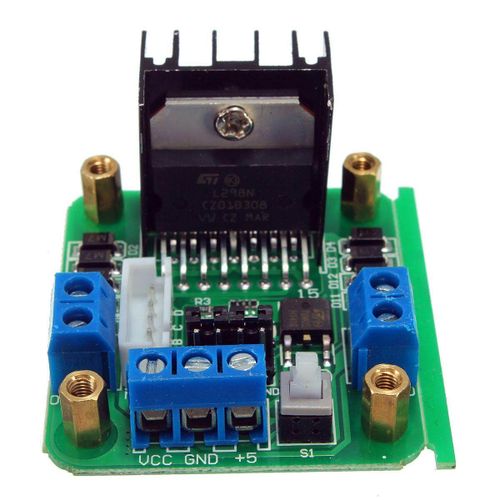
* Arduino UNO R3: The primary control unit for the robotic arm, responsible for processing commands and controlling the servo motors based on user input or predefined programs.



##### 1x Arduino UNO R3

3. Servo Motor Driver

* L298N Motor Driver Board: Enables control of multiple servo motors simultaneously, ensuring smooth and precise movement for each joint of the robotic arm. This driver board is critical for handling the power requirements of the servo motors efficiently.



##### 1x MG996R Servo Motors

4. Servo Motors

* 6x MG996R Servo Motors: High-torque servo motors chosen for their strength, especially suitable for handling the robotic arm’s base and main joints. Each servo is assigned to a specific joint for coordinated movement:
  + Base: Controls the rotation of the entire arm on its base.
  + Shoulder: Enables lifting and lowering of the arm.
  + Elbow: Allows the arm to bend, providing reach and flexibility.
  + Wrist Pitch: Controls the up-and-down movement of the wrist.
  + Wrist Roll: Enables rotation of the wrist for precise orientation of the end effector.
  + Gripper: Operates the end effector to pick up, hold, and release objects.



##### 6x MG996R Servo Motors

5. Additional Components

* 1x Breadboard: Used for prototyping the circuit connections.
* 20x Male-to-Female Jumper Wires & 20x Male-to-Male Jumper Wires: For connecting components on the breadboard and making connections to the Arduino and servo driver.
* Hard Jumper Wires (Pack): Provide stable and durable connections for critical parts of the circuit.
* Power Distribution Board: Distributes power efficiently to all components, preventing overloading of any individual component.

# Step-by-Step Build Process

1. **Assembly of Power System**
   * Connect the 18650 batteries to the voltage regulator and power switch.
   * Connect the output to the Arduino UNO and PCA9685 servo driver to provide stable power.
2. **Control System Setup**
   * Install the Arduino UNO and connect it to the PCA9685 driver.
   * Configure the PCA9685 to control multiple servos, aligning with the planned servo positions.
3. **Servo Motor Installation**
   * Attach the servos to their designated points:
     + Base, Shoulder, Elbow, Wrist Pitch, Wrist Roll, Gripper.
   * Connect each servo motor to the PCA9685 according to the control structure.
4. **Programming**
   * Develop and upload the control program onto the Arduino UNO.
   * Test each servo individually to confirm accurate movement.

# Testing and Troubleshooting

## **Initial Testing**:

* + Power on the system and test each servo motor for smooth operation.
  + Adjust configurations in the Arduino program to fine-tune movement.

## **Troubleshooting Logs**:

* + Record any issues with servo alignment, power fluctuations, or control delays.
  + List steps taken to resolve issues, such as code adjustments, component replacements, or recalibration.

# Schedule and Milestones

* **Planning Phase**: Documenting and gathering initial components - [Dates Here]
* **Assembly Phase**: Constructing power and control systems - [Dates Here]
* **Testing Phase**: Initial testing and troubleshooting - [Dates Here]
* **Documentation Phase**: Finalizing documentation and review - [Dates Here]

# Conclusion and Final Checks

After completing the above steps, ensure that each section is tested and functional. Mark all completed tasks in the documentation, including notes on troubleshooting and successes.

## Document Checklist:

* Task Documentation
* System Diagrams
* Testing Records
* Updated Schedules