

Politecnico di Torino

Master's Degree in Mechatronic Engineering LM-25

HW & Embedded Systems for Industry 4.0

— September 2026 —

Mechatronic Optimization, Fabrication and Integration of a Robotic Arm for an Educational Humanoid Robot

Supervisor:

Marco Vacca

Candidate:

Giorgio Giacomotti

Abstract

This thesis presents the design, fabrication and validation of an enhanced upper-limb manipulator starting from the open-source humanoid robot PIB, with the objective of improving its mechanical robustness, manufacturability and usability as a low-cost research and educational platform. From a critical analysis of the baseline architecture, the work identifies the principal limitations of the original design, including the restricted capabilities of hobby-grade position-controlled servomotors, limited torque density, thermal saturation under sustained loads, underactuated tendon-driven hand kinematics, transmission losses caused by long Bowden cables and the absence of proprioceptive and tactile feedback. These constraints significantly reduce manipulation accuracy, payload capacity and the suitability of the platform for advanced mechatronic applications.

To address these limitations, the manipulator was redesigned through a series of targeted mechanical and electromechanical improvements. The actuation system was upgraded using higher-performance digital servomotors selected to maximize torque, reliability and thermal performance. Structurally, standard polylactic acid (PLA) was substituted with polyethylene terephthalate glycol (PETG) to exploit its superior ductility and thermal stability, while the fused deposition modeling (FDM) process was optimized using Gyroid triply periodic minimal surfaces (TPMS) to guarantee isotropic strength against dynamic loads.

Proprietary controllers were replaced with a custom, highly transparent embedded architecture to improve kinematic performance and deterministic control. To safely support the upgraded actuators, a modular power supply infrastructure was engineered, physically decoupling the robotic arm from the power generation stage for enhanced portability and thermal management. Robust safety features were integrated directly into the hardware and firmware, including a dedicated emergency stop that triggers a controlled homing sequence to safely bleed off kinetic energy and prevent structural collapse. Additionally, a standalone analog control panel equipped with linear sliders was developed to facilitate direct, manual manipulation of the robotic joints for didactic observation.

To govern the physical plant, custom firmware computes dynamic trapezoidal velocity profiles that mathematically override the rigid internal PID controllers of the servomotors. This trajectory generation successfully prevents kinetic shock to the 3D-printed linkages, eliminates mechanical jitter and significantly increases positional resolution.

Finally, this low-level firmware operates in tandem with a custom 3D Digital Twin. A Python-based serial-to-WebSocket bridge routes optimized telemetry to a WebGL visualizer, enabling real-time spatial synchronization. An intelligent Sync-Lock arbitration protocol guarantees seamless, collision-free control transitions between autonomous digital commands and the manual sliders. The resulting hybrid-control framework establishes a robust, cost-effective and highly observable robotic testbed tailored for advanced didactic experimentation.

Table of Contents

List of Tables xi

List of Figures xiii

List of Acronyms and Abbreviations xv

1 Introduction 1
- 1.1 Background and Didactic Motivation . . . 1
  - 1.1.1 Polymer Additive Manufacturing and Open Hardware in Robotics . . . 1
  - 1.1.2 The Need for Observable Mechatronic Testbeds . . . 2
- 1.2 State of the Art in Open-Source Manipulator Platforms . . . 3
  - 1.2.1 Survey of Open-Source Humanoid and Robotic Arm Frameworks . . . 3
  - 1.2.2 Comparative Analysis: The InMoov Architecture and Practical Bottlenecks . . . 5
  - 1.2.3 Actuation and Transmission Paradigms in Low-Cost Robotics 6
- 1.3 The PIB Platform: Selection Rationale, Objectives and Contributions 8
  - 1.3.1 Baseline System Limitations and Project Scope . . . 9
  - 1.3.2 Summary of Key Technical Contributions . . . 10
- 1.4 Structure of the Thesis . . . 13

2 Hardware Architecture and Mechatronic Limitations of the PIB Manipulator 19
- 2.1 System Overview and Kinematic Framework of the PIB Robotic Arm 20
- 2.2 Control Paradigm Constraints: Closed-Loop Position Control . . . 22
- 2.3 Torque Density Deficits and Thermal Saturation . . . 24
- 2.4 Underactuated Kinematics: Single-Tendon Routing and Passive Spring Return . . . 26
  - 2.4.1 Bowden-Cable-Driven Torso-Mounted Actuation . . . 28

2.4.2 Distal Constraints and Dexterity Limitations 28
2.4.3 Transmission Nonlinearity and Capstan Friction Losses . . . 29
2.4.4 Limitations of the Passive Spring Return 30
2.4.5 Loss of Kinematic Determinism 31
2.5 The Absence of Proprioceptive Telemetry and Tactile Feedback . . 31
2.5.1 The Critical Mechatronic Role of Haptic Feedback 32
2.5.2 Complete Lack of Sensor Integration in the Baseline Project 32

3 Prototype Fabrication, Mechatronic Integration and System Calibration 35
3.1 Actuation Architecture and Electromechanical Upgrades 36
3.1.1 Actuator Upgrades for Distal and Secondary Joints 36
3.1.2 Actuator Upgrades for Primary Joints 38
3.2 End-Effector Kinematics and the Rejection of Antagonistic Tendon Routing 40
3.3 Polymer Thermodynamics and Material Preparation 41
3.3.1 Tribological and Mechanical Superiority of PETG 41
3.3.2 Moisture Absorption Kinetics and Hydrolysis 42
3.3.3 Dual-Environment Manufacturing and Polymer Dehydration 43
3.4 FDM Manufacturing Platform: Upgrades and Calibrations 44
3.4.1 The Anycubic Prototyping Platform and Hardware Upgrades 44
3.4.2 Extrusion Tribology and Nozzle Replacement 46
3.4.3 Empirical Calibration Matrix 46
3.4.4 Extrusion Thermodynamics and Viscoelastic Control 48
3.5 Advanced FDM Slicing Algorithms and Toolpath Optimization . . . 49
3.5.1 Triply Periodic Minimal Surfaces (TPMS): The Gyroid Architecture 49
3.5.2 Anisotropic Load Bearing and Spatial Orientation 50
3.5.3 Kinematic Tuning and Resonance Mitigation 51
3.6 Structural Foundation: Custom Anchoring Architecture 53
3.6.1 Custom Mount Development for Single-Arm Integration . . 53
3.6.2 Anisotropic Load Mitigation in the Primary Support 55
3.6.3 Didactic Advantages of the Open-Architecture Configuration 56
3.7 Physical Assembly and Mechatronic Commissioning 57
3.7.1 Inventory Management and Mechanical Post-Processing . . . 57
3.7.2 Bottom-Up Subassembly and Fastener Economy 58
3.7.3 Empirical Actuator Calibration and Kinematic Bounding . . 58
3.7.4 Tribological Tuning and Dynamic Kinematic Instability . . . 59

4 Custom Electronic Architecture and Power Distribution 63
4.1 Limitations of the Baseline Electronic Architecture 64

4.2 Custom Embedded Control Architecture 66
4.3 Empirical Actuator Calibration and Signal Frequency Optimization 67
4.3.1 The Necessity of Open-Loop Hardware Calibration 67
4.3.2 Pulse Width Frequency Analysis and Resolution Enhancement 69
4.4 High-Current PWM Distribution: The Custom PCA9685 Architecture 71
4.4.1 The Bottleneck of Commercial Breakout Boards 71
4.4.2 PCB Topography Modification and Split-Rail Busbars . . . 72
4.4.3 Spatial Integration and EMI Mitigation 73
4.5 Industrial Power Supply and Safety Infrastructure 75
4.5.1 Primary Enclosure and AC Mains Interfacing 75
4.5.2 Dual-Rail Switch Mode Power Supplies (SMPS) 76
4.5.3 Secondary DC Overcurrent Protection and Internal Routing 78
4.6 The Modular Interconnect and Power/Data Interfaces 78
4.6.1 Architectural Modularity and System Decoupling 79
4.6.2 High-Current DC Transmission and Thermal Mitigation . . 79
4.6.3 Electro-Mechanical Safety: Keying, Locking and Polarity Protection 80
4.6.4 High-Speed Data Routing and Dual-Redundancy Telemetry 81
4.7 Analog Control Interface (Hardware HMI) 82
4.7.1 Kinematic Mapping and Potentiometer Array 82
4.7.2 Mechanical Integration and Prototyping Substrate 83

5 Embedded Software and Digital Twin Architecture 87
5.1 Custom Software Architecture 88
5.1.1 Rationale for a Custom Software Ecosystem 88
5.1.2 The Dual-Layered Hybrid Framework 88
5.1.3 Virtual Commissioning and Safe Development 89
5.2 Low-Level Embedded Firmware (STM32 HAL) 90
5.2.1 Non-Blocking Serial Communication and JSON Parsing . . . 90
5.2.2 Kinematic Mapping and Mechanical Constraints 92
5.2.3 Trajectory Generation and Trapezoidal Velocity Profiles . . . 93
5.2.4 Mitigating Asymptotic Jitter via Dynamic Deadbands 95
5.3 Hardware-Software Safety Interlocks 96
5.3.1 Boot-Up Diagnostics and Controlled Homing 96
5.3.2 Open-Loop Initialization and Conservative Constraints . . . 97
5.4 Hybrid Control Architecture (ADC + DMA) 98
5.4.1 Direct Memory Access and Signal Filtering 99
5.4.2 The Sync-Lock Safety Protocol and Intelligent Hysteresis . . 100
5.5 The Digital Twin and High-Level Middleware 102
5.5.1 Twin-Mode Telemetry and Dual-Frequency Protocol 102
5.5.2 Python Serial-to-WebSocket Bridge 103

5.5.3 WebGL Visualization and URDF Parsing 104

6 Conclusions and Future Developments 107
6.1 Achievements and System Performance 108
6.1.1 Full Mechatronic Integration and Dual-Mode Operation . . . 109
6.1.2 Embedded Architecture and Real-Time Low-Level Control . 110
6.1.3 Modular Infrastructure, Portability and Didactic Observability 112
6.2 Criticalities and Experimental Limitations 113
6.2.1 Procurement Hurdles and Additive Manufacturing Realities 113
6.2.2 Distal Transmission Nonlinearities and Joint Flexibility . . . 114
6.2.3 Dynamic Deceleration and Inductive Back-EMF Overvoltage 114
6.3 Proposed Solutions and Future Improvements 115
6.3.1 Active Power Regeneration and Electrical Protection 115
6.3.2 Mechanical Redesign of Distal Articulations 118
6.3.3 Scalability: Bilateral Expansion and Multi-Driver Support . 118
6.3.4 High-Level Control, Edge AI and Autonomous Manipulation 119
6.4 Educational Reflections and Professional Takeaways 121
6.4.1 Comprehensive Mechatronic Competencies 121
6.4.2 Contributions to the Open-Source Humanoid Community . . 122

Bibliography 125

List of Tables

2.1 Technical comparison and deployment scope of the actuators utilized in the original PIB manipulator project. 21

3.1 Technical comparison of the upgraded servos versus the original design for the Distal and Secondary Joints. 37
3.2 Technical comparison of the upgraded servos versus the original design for the Primary Joints. 39
3.3 Technical specifications and material properties of the Verbatim White PETG filament. 42
3.4 Recommended 3D printing parameters for manufacturing the manipulator components. 48

List of Figures

1.1 Comparative overview of open-source articulated manipulators. . . . 3
1.2 Comparison of full-scale humanoid robotic platforms. 4
1.3 InMoov open right forearm assembly. 5
1.4 InMoov right shoulder worm-gear mechanism. 6
1.5 Comparison of servomotor actuation technologies. 7
1.6 Printable Intelligent Bot complete upper body. 9

2.1 PIB torso-mounted bowden-cable actuation system. 27
2.2 Detailed view of the PIB torso servomotor bank and tendon pulley mechanism. 27
2.3 PIB underactuated right hand mechanism and tendon routing. . . . 29

3.1 Architecture of a 3D-printed servo cycloidal speed reducer. 38
3.2 Effects of moisture absorption on PETG print quality. 43
3.3 FDM 3D printers employed for structural PETG fabrication. 45
3.4 Examples of empirical FDM calibration techniques. 47
3.5 Comparison of FDM infill patterns and density variations. 50
3.6 Example of inter-layer delamination and mechanical failure under tensile stress. 51
3.7 Effect of printer structural resonance on print quality. 52
3.8 Centralized servo bank mount for distal tendon actuation. 54
3.9 SP21 power supply and RJ45 signal cable connector. 54
3.10 Custom load-bearing shoulder support bracket. 55

4.1 Exploded view of elbow and shoulder slip-ring assemblies. 68
4.2 Comparison between the baseline Tinkerforge Servo Bricklet 2.0 and the PCA9685 breakout board. 69
4.3 Mounting location of the PCA9685 node on the carbon-fiber mast, situated behind the shoulder joint and to the rear of the distal actuation servo bank. 74

4.4 Internal layout and rear I/O panel of the IP56 power and control enclosure. 77
4.5 Six-channel linear potentiometer HMI for manual joint manipulation. 82

5.1 Dual-layer framework: host Digital Twin and STM32 communicating via USB-JSON to drive the manipulator over Fast-mode I2C. 89
5.2 Web-based Three.js Digital Twin interface for real-time 3D kinematic visualization and telemetry. 104

6.1 Complete experimental setup integrating the Digital Twin GUI, control enclosure, analog HMI and physical manipulator. 108
6.2 Asymmetric inverting Schmitt trigger circuit and simulated voltage response. 117

List of Acronyms and Abbreviations

ABL	Auto Bed Leveling
ABS	Acrylonitrile Butadiene Styrene
ADC	Analog-to-Digital Converter
API	Application Programming Interfaces
BLDC	Brushless Direct Current motor
CAD	Computer-Aided Design
COTS	Components Of-The-Shelf
CPU	Central Processing Unit
DH	Denavit-Hartenberg
DMA	Direct Memory Access
DOF	Degrees Of Freedom
DOM	Document Object Model
EMA	Exponential Moving Average
EMF	Electromagnetic Field
EMI	Electromagnetic Interference
EXTI	External Interrupt
FOC	Field Oriented Control
HAL	Hardware Abstraction Layer
HMI	Human-Machine Interface
HRI	Human-Robot Interaction
IC	Integrated Circuit

IK	Inverse Kinematics
JSON	JavaScript Object Notation
MTBF	Mean Time Between Failures
NC	Normally Closed
NO	Normally Open
OCP	Over-Current Protection
OTP	Over-Temperature Protection
OVP	Over-Voltage Protection
PCB	Printed Circuit Board
PDB	Power Distribution Board
PIB	Printable Intelligent Robot
PID	Proportional-Integral-Derivative Controller
PLA	Polylactic Acid
PWM	Pulse Width Modulation
QDD	Quasi-Direct Drive
RAS	Robot Assisted Surgery
RC	Radio-Controlled
ROM	Range Of Motion
ROS	Robot Operating System
SCL	Serial Clock
SDA	Serial Data
SDK	Software Development Kit
SMA	Simple Moving Average
SMPS	Switch Mode Power Supply
SRAM	Static Random-Access Memory
STL	Standard Triangle Language
TPMS	Triply Periodic Minimal Surface
UART	Universal Asynchronous Receiver-Transmitter
UHMWPE	Ultra-High-Molecular-Weight Polyethylene
URDF	Universal Robot Description Format

Chapter 1

Introduction

1.1 Background and Didactic Motivation

In recent years, the intersection of accessible digital fabrication and open hardware has reshaped mechanical and electronic prototyping within academic environments. The capacity to rapidly translate algorithmic concepts into physical testbeds is critical for mechatronics education, where multidisciplinary theory must interface directly with physical dynamics, hardware latencies and material constraints.

1.1.1 Polymer Additive Manufacturing and Open Hardware in Robotics

Fused Deposition Modeling (FDM) has transitioned from a tool for aesthetic prototyping into a viable manufacturing process for functional, load-bearing mechanical assemblies. The availability of capable desktop extrusion platforms and engineering-grade thermoplastics allows academic laboratories to produce custom robotic linkages, bracketry and transmission housings at low cost.

Parallel to advancements in additive manufacturing, open-source hardware repositories have expanded access to complex robotic architectures. By releasing Computer-Aided Design (CAD) models, electrical schematics and baseline firmware under permissive licenses, research groups and developer communities provide accessible starting points for robotics education. Rather than relying on capital-intensive industrial cells or closed commercial platforms, universities can deploy, evaluate and modify functional multi-degree-of-freedom (DOF) platforms using standard laboratory resources.

However, utilizing low-cost FDM components in functional robotics introduces pronounced engineering trade-offs. Additively manufactured parts exhibit significant structural anisotropy, where mechanical strength along the vertical build axis (Z) is bounded by interlayer thermal fusion rather than continuous polymer strands. Dynamic loads, cyclic fatigue and localized shear concentrations can trigger structural failure or premature wear at joint interfaces. Successfully deploying FDM within high-torque robotic systems requires systematic optimization of print kinematics, slicer infill topologies and print orientations aligned with primary dynamic stress tensors.

1.1.2 The Need for Observable Mechatronic Testbeds

A primary pedagogical challenge in mechatronics curricula is the prevalence of proprietary "black-box" educational robots. Commercial educational arms often encapsulate their actuators, drive electronics and power distribution within sealed, injection-molded casings. Furthermore, their control stacks typically rely on high-level, vendor-specific Application Programming Interfaces (APIs) that abstract away low-level embedded functions such as pulse-width modulation (PWM) timer configuration, analog-to-digital conversions, interrupt handling and bus communication protocols.

While convenient for high-level algorithmic demonstrations, this encapsulation limits educational effectiveness:

- **Obscured Hardware-Firmware Interactions:** Students cannot inspect or tune the low-level trajectory generation loops, observe the real-time execution of hardware timers or evaluate the computational overhead of non-blocking communications.
- **Hidden Electrical Phenomena:** High transient stall currents, inductive noise, bus voltage drops and thermal limits are concealed behind internal protective circuits, preventing direct empirical study of electrical distribution engineering.
- **Decoupled Physical Realities:** Abstracted simulation environments often treat mechanical linkages as perfectly rigid bodies, neglecting the impacts of joint friction, backlash, structural elasticity and transmission nonlinearities.

Addressing these educational shortcomings requires an observable mechatronic testbed: an open-frame platform where both mechanical and electronic subsystems are accessible for physical inspection and real-time instrumentation. Exposing bare-metal microcontroller peripherals, high-current split-rail power busbars and physical tendon routings enables students to analyze mechatronic principles directly

on physical hardware. Correlating these physical states with a real-time 3D Digital Twin bridges theoretical kinematics with physical implementation, creating an ideal environment for advanced mechatronics coursework and research.

1.2 State of the Art in Open-Source Manipulator Platforms

Open-source robotics has produced a diverse spectrum of platforms aimed at lowering the cost barrier for academic research and education[8]. These systems diverge widely in structural topology, kinematic complexity, actuation technologies and transmission strategies, leading to distinct mechanical trade-offs between biological anthropomorphism, cost and dynamic performance.

1.2.1 Survey of Open-Source Humanoid and Robotic Arm Frameworks

Open-source robotic architectures generally fall into two primary structural categories:

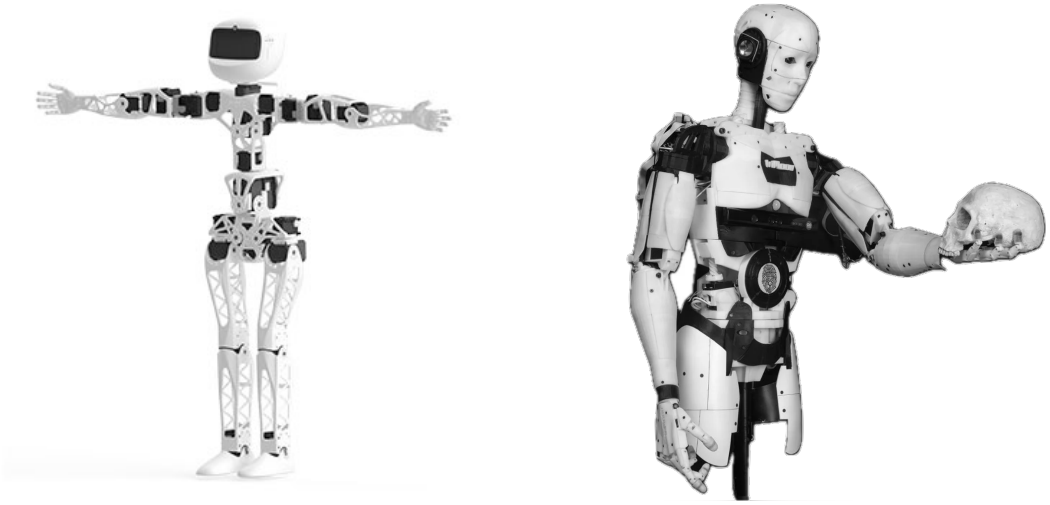
Dedicated Articulated Manipulators:
Platforms such as the BCN3D Moveo[11] (figure: 1.1a), the Thor robotic arm[12] (figure: 1.1b) and the ROBOTIS OpenManipulator[13] (figure: 1.1c) focus strictly on multi-axis serial manipulation. By omitting biological aesthetics, these systems utilize direct joint-axis mounting or localized timing-belt reductions. While structurally stiff and kinematically predictable, their architectures lack biomimetic interest, featuring rigid industrial kinematic links rather than anthropomorphic joints, multi-link underactuated end-effectors or torso-routed transmissions.



(a) Moveo[11] **(b)** Thor[12] **(c)** OpenManipulator[13]

Figure 1.1: Comparative overview of open-source articulated manipulators.

Full-Scale Humanoid Platforms:
Projects such as the Poppy Humanoid[9] (figure: 1.2a) and the InMoov robot[14] (figure: 1.2b) aim to reproduce the human form, providing full-body or upper-torso kinematic trees. While Poppy relies on expensive, daisy-chained smart actuators (Dynamixel servos) and lightweight carbon-filled polyamide structures, its total bill of materials places it outside standard low-cost prototyping budgets. Conversely, community platforms such as InMoov prioritize accessibility by utilizing standard fused deposition modeling (FDM) and commercial off-the-shelf (COTS) radio-controlled (RC) servomotors.



(a) Poppy Humanoid[9] **(b)** InMoov[14]

Figure 1.2: Comparison of full-scale humanoid robotic platforms.

Developing a complete bi-manual humanoid torso introduces severe operational overhead for isolated manipulation studies. Managing two arms, a neck articulation and an active torso multiplies actuator counts, power requirements and mechanical failure points. Consequently, academic research frequently adapts these open-source designs into modular, single-arm benchtop testbeds, isolating the manipulator to study trajectory generation, distal transmissions and embedded control architectures.

1.2.2 Comparative Analysis: The InMoov Architecture and Practical Bottlenecks

The InMoov robot[14] (figure: 1.2b) represents one of the most widely replicated open-source humanoid designs. While visually compelling and biologically anthropomorphic, physical implementations of the platform reveal several mechanical and electromechanical bottlenecks that compromise dynamic performance, motion repeatability and structural longevity.

A primary limitation lies in its distal mass distribution. In the InMoov architecture, the five servomotors dedicated to finger flexion are mounted directly inside the forearm housing, as illustrated in figure 1.3.

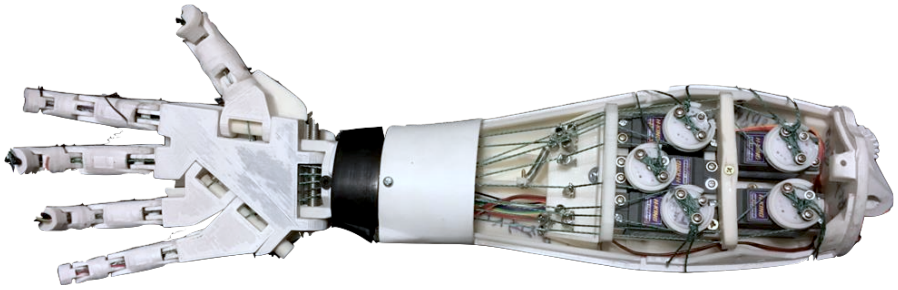


Figure 1.3: InMoov open right forearm assembly. Adapted from *InMoov: Open-Source Life-Size 3D Printed Humanoid Robot.*

While this layout shortens the tendon routing and reduces cable-sheath friction relative to torso-mounted solutions, it concentrates substantial mass at the distal extremity of the kinematic chain. In serial manipulators, distal mass amplifies the reflected moment of inertia experienced by proximal joints:

$$I_{\text{reflected}} = m_{\text{distal}} \cdot L_{\text{arm}}^2 \tag{1.1}$$

This amplified inertia limits angular acceleration, lowers the natural mechanical resonant frequencies of the arm and demands high static holding torques from the elbow and shoulder actuators simply to counteract gravitational sag.

Furthermore, InMoov relies on 3D-printed worm-gear drives to articulate its primary shoulder and bicep axes. While a worm-gear mechanism provides high reduction ratios in a compact footprint and offers passive non-backdrivability against gravity, additively manufactured polymer implementations suffer from severe drawbacks:

- **High Sliding Friction:** Unlike rolling-element drives, worm gears transmit force through continuous surface-to-surface sliding. Printed thermoplastics (such as PLA or ABS) exhibit high friction coefficients and poor tribological thermal dissipation, leading to high mechanical power losses and localized frictional heating.
- **Rapid Wear and Backlash:** Cyclic loading under cantilevered arm weights rapidly degrades the printed polymer thread profiles, introducing mechanical backlash (lost motion) that compromises kinematic repeatability.
- **Loss of Dynamic Transparency:** The inherent non-backdrivability of the worm gear decouples the motor from external output shocks. Any inadvertent collision or joint stall propagates directly into the printed gear teeth rather than backdriving the motor shaft, resulting in stripped gears and mechanical fractures.

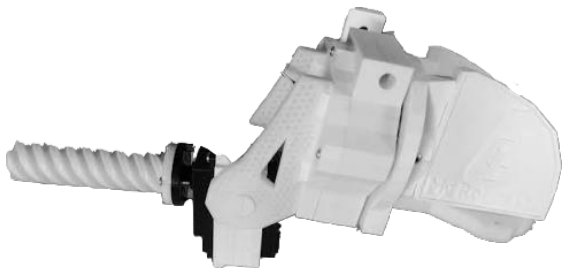


Figure 1.4: InMoov right shoulder worm-gear mechanism. Adapted from *InMoov: Open-Source Life-Size 3D Printed Humanoid Robot.*

These kinematic and tribological constraints demonstrate that prioritizing aesthetic anthropomorphism over mechanical efficiency limits the utility of a platform for precise control and dynamic manipulation tasks.

1.2.3 Actuation and Transmission Paradigms in Low-Cost Robotics

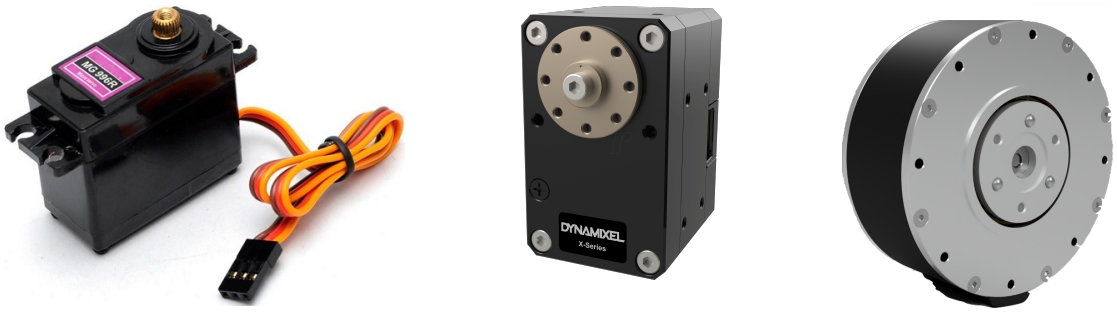
The performance boundaries of accessible robotic platforms are defined primarily by the interplay between their actuators and transmission systems.

Actuation Technologies:

- **Standard RC PWM Servomotors:** Widely adopted due to low unit cost and high availability. However, traditional analog and entry-level digital RC servos operate as closed-loop position controllers with internal, unalterable

proportional-integral-derivative (PID) routines. They accept low-frequency PWM signals (50 Hz), offer no proprioceptive back-channel telemetry (position, current, temperature) and saturate thermally when holding static loads.

- **Smart Serial Bus Actuators:** Integrate brushless or coreless DC motors, absolute magnetic encoders and internal microcontrollers communicating over daisy-chained half-duplex UART or RS-485 buses (e.g., Dynamixel). They allow closed-loop current, velocity and trajectory management with onboard diagnostic feedback. However, their high cost per unit scales poorly in multi-DOF manipulators, conflicting with low-budget academic frameworks.
- **Quasi-Direct Drive (QDD) Brushless Systems:** High-pole-count, low-gear-ratio brushless DC (BLDC) motors paired with Field-Oriented Control (FOC). QDD actuators provide exceptional backdrivability, high bandwidth and force transparency. Nevertheless, the cost of dedicated FOC drivers, high-resolution magnetic encoders and precision cycloidal or planetary gearboxes restricts their deployment in low-cost platforms.



(a) TowerPro MG996R[17] **(b)** Dynamixel X-Series[13] **(c)** BLDC QDD[15]

Figure 1.5: Comparison of servomotor actuation technologies.

Remote Tendon Transmissions and Underactuation:
To decouple actuator mass from moving links, accessible robotic manipulators frequently employ remote cable-conduit (Bowden) transmissions. Centralizing the actuator banks proximally (e.g., within the base or torso) substantially lowers distal mass and minimizes reflected inertia during dynamic motion.

However, guiding flexible tendons through polymer sheaths across multiple articulating joints introduces path-dependent friction losses, sheath compliance and mechanical hysteresis. Governed by the Capstan effect[1, 10], frictional resistance scales exponentially with the cumulative angular deflection of the conduit, causing force transmission efficiency to vary continuously with the arm's spatial posture. When paired with underactuated end-effectors—where a reduced number of actuators drive multi-joint finger mechanisms—these non-linearities complicate tendon

pretensioning and kinematic repeatability. The formal mathematical modeling of these transmission dynamics and their effect on kinematic determinism are detailed in chapter 2.4.3.

1.3 The PIB Platform: Selection Rationale, Objectives and Contributions

Following the evaluation of available open-source humanoid architectures, the PIB (Printable Intelligent Bot) platform[16] was selected as the foundational baseline for this research. This section outlines the engineering rationale behind this selection, identifies the systemic limitations of the stock architecture that motivated a complete mechatronic overhaul and summarizes the primary technical contributions achieved in this thesis.

Platform Selection Rationale: PIB over Anthropomorphic Alternatives

The selection of an open-source robotic framework for advanced mechatronics research requires balancing mechanical feasibility, component accessibility and physical robustness. While biologically inspired platforms such as InMoov have achieved widespread community adoption, empirical evaluations of physical InMoov prototypes—including implementations developed within our departmental research group—revealed critical structural and tribological drawbacks that impede precise dynamic control.

InMoov's reliance on 3D-printed polymer worm-gear pairs for the shoulder and upper-arm joints generates severe sliding friction, rapid tooth wear and mechanical backlash under continuous cantilevered loads. Furthermore, housing five standard RC servomotors directly inside the forearm concentrates significant mass at the distal extremity of the kinematic chain, drastically magnifying joint inertia ($I = mr^2$) and reducing the arm's natural mechanical resonance frequency.

The PIB platform, originally created by isento GmbH and maintained by an active international open-source community, adopts an alternative mechanical design philosophy:

- **Industrial Design Elements:** Rather than relying on direct polymer-on-polymer sliding interfaces, PIB integrates deep-groove ball bearings at all rotational joint pivots, distributed structural ribbing and localized spring assists to handle joint loads.

- **Mass Centralization via Remote Transmission:** Actuators governing the hand and wrist are centralized proximally within the torso block, transmitting force across the arm via Bowden cables. This remote actuation approach keeps distal linkages lightweight, minimizing reflected inertia at the primary joints.
- **Performance over Cosmetic Anthropomorphism:** Unlike InMoov, which prioritizes a fully enclosed, cosmetically anthropomorphic outer shell, PIB features a streamlined, modular kinematic geometry. The platform omits non-functional exterior casings in favor of direct mechanical access to joint mechanisms, fasteners and transmission paths.

This emphasis on functional mechanical performance, modularity and community-driven development made PIB the most suitable framework for investigating low-cost robotic manipulation.



Figure 1.6: Printable Intelligent Bot complete upper body. Adapted from *PIB – Printable Intelligent Bot: Open-Source 3D-Printable Humanoid Robot.*

1.3.1 Baseline System Limitations and Project Scope

Despite its sound design foundation, the baseline PIB architecture relies on several compromises intended to keep fabrication accessible to novice makers, resulting in performance bottlenecks that preclude its use as an advanced mechatronic testbed:

- **Actuation Deficits and Thermal Saturation:** The stock platform utilizes standard hobby-grade servomotors (TowerPro MG996R[17], DSSERVO DS3225[18] and DSSERVO DS5180SSG[19]) operating on closed-loop position control with low torque density. Under sustained static holding loads or rapid dynamic sweeps, these actuators suffer from rapid Joule heating ($P_{\text{loss}} = I^2R$) and lack thermal dissipation paths, leading to magnetic flux loss and premature thermal cutout.

- **Transmission Non-Linearities:** The remote tendon-driven hand operates as an underactuated system where single-acting pull cables flex the fingers against passive mechanical return springs. Force transmission through the flexible Bowden sheaths is governed by the Capstan friction model ($F_{\text{in}} = F_{\text{out}} \cdot e^{\mu\theta}$). Bending along the shoulder, elbow and wrist accumulates significant friction and mechanical hysteresis, reducing fingertip force and causing non-deterministic finger motion.

- **Structural Brittleness:** Standard PIB components are 3D-printed in Polylactic Acid (PLA) using standard rectilinear infill. PLA's low impact resistance (2 to 3 kJ/m^2) and low glass transition temperature ($T_g \approx 55$ to 60°C) make structural linkages susceptible to brittle fracture under dynamic shear and localized motor heat.

- **Closed Proprietary Control Architecture:** Low-level motor control relies on commercial Tinkerforge[20] hardware (Master Brick and Servo Bricklet 2.0). Operating as a computational "black box", the Tinkerforge module abstracts PWM generation to a legacy 50 Hz frequency, hiding silicon peripherals, hardware timers and interrupt structures from the user.

- **Torso Dependency:** The open-source repository treats the robot strictly as a monolithic humanoid torso, lacking provision for an isolated single-arm experimental mount.

The scope of this thesis addresses these limitations by re-engineering the PIB left arm into an autonomous, high-torque, single-limb benchtop manipulator. By completely replacing the proprietary electronics, upgrading the drivetrain and decoupling the mechanical arm from its power generation stage, the project transforms an open-source educational model into an observable, deterministic mechatronic research platform.

1.3.2 Summary of Key Technical Contributions

This work implements systematic upgrades spanning mechanical design, additive manufacturing, power systems, embedded firmware and real-time simulation:

- **Drivetrain and Actuator Overhaul:** The baseline servomotors were replaced with higher-specification digital actuators. Primary joints (shoulder yaw, shoulder pitch, elbow) utilize high-voltage DSSERVO DS51150[21] actuators delivering up to 173 kg · cm of stall torque at 12.6 V across a 270° physical sweep. Secondary and tendon-driven axes utilize DFRobot DC5535[22] coreless actuators providing 35 kg · cm at 7.0 V, improving torque margins and dynamic acceleration across all 12 degrees of freedom.

- **Advanced Material Slicing and Fabrication:** Structural linkages were transitioned from PLA to White Verbatim PETG[23], exploiting its higher ductility, impact strength (8.1 kJ/m^2) and thermal endurance ($T_g \approx 80$°C). Components were sliced using Triply Periodic Minimal Surface (TPMS) Gyroid infill parameterized from 30% (distal links) to 80% (motor brackets), ensuring isotropic load bearing against multi-axial dynamic bending moments[2].

- **Embedded Control and 300 Hz High-Resolution Driving:** The Tinkerforge ecosystem[20] was replaced with a custom embedded controller based on an STM32 microcontroller[24] running bare-metal C via the STMicroelectronics Hardware Abstraction Layer (HAL)[3]. Actuators are governed via an NXP PCA9685 12-bit PWM driver[25] operated at an optimized frequency of 300 Hz ($T \approx 3.33$ ms). This expanded the usable duty cycle window from 10.0% to 60.0%, yielding a 600% increase in positional resolution (improving primary joint discretization from 0.66° to 0.11° per step) and eliminating low-speed motion jitter.

- **Modular Power Electronics and Interconnects:** A dedicated power delivery stage was engineered inside a sealed IP56 industrial enclosure, integrating dual Mean Well switching power supplies[26] (12.0 V @ 30 A and 7.5 V @ 20 A) with secondary DC blade fusing. To handle high transient currents without a custom PCB, the PCA9685 board was modified with soldered 1.5 mm^2 copper busbars to create a split-rail architecture mounted directly to the arm's mast. The manipulator uncouples completely via an industrial 30 A SP21 aviation umbilical[27] and a shielded, dual-redundant RJ45 Fast-mode I2C telemetry connection.

- **Deterministic Firmware, Safety Interlocks and HMI:** The embedded firmware implements a deterministic 300 Hz hardware timer loop executing real-time trapezoidal velocity profiling via discrete Euler integration, overriding the aggressive step-response PIDs of the servos. Asymptotic limit-cycling at targets is suppressed via Dynamic Deadbands ($\epsilon_p = 0.3°$, $\epsilon_v = 5.0°/s$). Operator safety is maintained via a hardware interrupt-driven Controlled Homing sequence. Direct manual control is provided by a 6-channel linear

potentiometer HMI read via non-blocking Continuous Scan ADC with DMA and Exponential Moving Average filtering, arbitrated against digital commands via a derivative-based Sync-Lock state machine.

- **Bidirectional 3D Digital Twin:** The embedded controller interfaces with a high-level Python serial-to-WebSocket bridge, streaming telemetry over a dual-frequency protocol (50 Hz kinematic poses, 1 Hz static parameters). A Three.js WebGL browser application parses the arm's URDF model to mirror physical joint states in real time, supporting virtual commissioning and remote teleoperation.
- **Upstream Open-Source Contributions:** Practical assembly findings, including structural bracket modifications, bearing clearance optimizations and Bowden sheath routing revisions, were communicated to the PIB development team and integrated upstream into the official open-source release.

1.4 Structure of the Thesis

This thesis is organized into six chapters that document the engineering progression from the diagnostic evaluation of the baseline PIB platform to the deployment and evaluation of the upgraded mechatronic manipulator.

Chapter 2: *Hardware Architecture and Mechatronic Limitations of the PIB Manipulator*

Presents an electromechanical and kinematic assessment of the stock PIB robotic arm.
The chapter models the governing dynamics of standard commercial-off-the-shelf (COTS) radio-controlled (RC) servomotors, detailing the algorithmic limitations of unalterable position-controlled proportional-integral-derivative (PID) architectures.
It analyzes the thermodynamic mechanisms of Joule heating ($P_{\text{loss}} = I^2R$) that cause magnetic flux degradation and premature thermal cutouts during static holding loads.
It investigates the distal kinematics of the underactuated hand, formalizing the transmission losses and mechanical hysteresis caused by Bowden cable routing through the Capstan friction model ($F_{\text{in}} = F_{\text{out}} \cdot e^{\mu\theta}$) and single-acting passive return springs.
Finally, it highlights the operational vulnerabilities stemming from the total absence of proprioceptive feedback and tactile telemetry.

Chapter 3: *Prototype Fabrication, Mechatronic Integration and System Calibration*

Details the physical manufacturing, component selection and structural redesign of the left-arm manipulator.
The chapter outlines the establishment of an independent Bill of Materials (BOM) sourced through institutional distributors (RS Components and DigiKey), bypassing proprietary hardware kits.
It justifies the electromechanical selection of high-torque digital actuators: DFRobot DC5535 coreless servos for distal joints and DSSERVO DS51150 actuators for the primary axes. The material preparation of White Verbatim PETG is documented, covering moisture diffusion kinetics, dehydration protocols to prevent hydrolysis and toolpath optimization via Triply Periodic Minimal Surface (TPMS) Gyroid infill for isotropic dynamic load bearing.
The chapter concludes with the development of a custom mounting assembly anchored to a 40x40 mm carbon-fiber mast, open-loop hardware calibration via an independent function generator and tribological tuning of the distal Bowden sheaths and return springs.

Chapter 4: *Custom Electronic Architecture and Power Distribution*

Documents the overhaul of the electrical infrastructure, replacing the proprietary Tinkerforge commercial ecosystem with an open embedded platform.
The chapter details the empirical actuator calibration and pulse-width modulation (PWM) frequency optimization from 50 Hz to 300 Hz on an NXP PCA9685 driver, demonstrating a 600% gain in angular resolution (0.66° down to 0.11° per step). To handle combined stall currents exceeding 50 A without custom PCB fabrication, it presents a physical modification of the PCA9685 board utilizing soldered 1.5 mm^2 copper busbars to establish a split-rail power topology (7.5 V and 12.0 V).
It covers the engineering of an isolated IP56 power enclosure housing dual Mean Well switching mode power supplies (SMPS) with multi-tier AC and DC blade fuse protection. Finally, the chapter details the modular interconnect system consisting of a 30 A SP21 aviation umbilical and a shielded CAT6 RJ45 connection utilizing dual-redundant conductor pairs for 400 kHz Fast-mode I2C telemetry alongside a standalone 6-channel linear potentiometer Human-Machine Interface (HMI).

Chapter 5: *Embedded Software and Digital Twin Architecture*

Examines the dual-layered software ecosystem.
The low-level embedded layer, developed in bare-metal C using the STMicroelectronics Hardware Abstraction Layer (HAL) on an STM32 microcontroller, executes a deterministic 300 Hz hardware timer loop.
It details the non-blocking UART DMA pipeline coupled with the zero-allocation jsmn parser for JSON command tokenization, real-time trapezoidal velocity profiling via discrete Euler integration and the elimination of limit-cycle through Dynamic Deadbands.
The chapter covers hardware-software safety interlocks, including boot diagnostics and an active-low interrupt-driven Controlled Homing sequence (EXTI).
It formalizes the hybrid control architecture, utilizing Continuous Scan ADC with Direct Memory Access (DMA) and Exponential Moving Average (EMA) filtering, governed by the derivative-based Sync-Lock arbitration protocol with an hysteresis timeout.
Lastly, it presents the high-level architecture: a dual-frequency telemetry stream (50 Hz poses, 1 Hz static limits), a multi-threaded Python WebSocket bridge and a Three.js WebGL 3D Digital Twin that parses URDF geometry for bidirectional synchronization.

Chapter 6: *Conclusions and Future Developments*

Synthesizes the technical achievements of the project, verifying dual-mode operation, dynamic motion smoothness, electrical stability and didactic observability.
It provides an analysis of the experimental challenges encountered, including procurement cycles, PETG extrusion wear, distal Bowden non-linearities (underdamped finger flutter and wrist bending stiffness) and the inductive back-EMF spikes that trigger SMPS Over-Voltage Protection (OVP) cutouts during shoulder deceleration.
The chapter details prospective solutions, including an active Schmitt-trigger dynamic braking circuit, low-compliance UHMWPE/Dyneema tendons, decoupled wrist guides and a roadmap for 24-actuator bilateral expansion and high-level edge AI integration (ROS2 MoveIt and computer vision).
It concludes with reflections on mechatronic competencies gained and highlights upstream CAD contributions integrated into the official international PIB platform.

Chapter 2

Hardware Architecture and Mechatronic Limitations of the PIB Manipulator

The PIB (Printable Intelligent Bot) is an open-source humanoid robotics platform developed to improve the accessibility of robotic hardware through additive manufacturing and commercially available off-the-shelf (COTS) components. Originally developed by isento GmbH[16] and subsequently expanded through the open-source community, the platform adopts a design philosophy centered on reducing the cost and complexity traditionally associated with humanoid robotics. To this end, conventional industrial components are replaced, where possible, by 3D-printed polymer structures and commercially available hobby-grade electronics. This architecture makes the PIB particularly suitable for educational applications, maker communities and experimental robotics, while providing a practical platform for investigating humanoid robotic systems.

The emphasis on accessibility, however, necessarily introduces a number of engineering trade-offs. The use of polymer-based structures, hobby-grade actuators and simplified transmission mechanisms affects the mechanical stiffness, load-bearing capability, actuation performance and overall precision of the manipulator. These characteristics are therefore important when assessing the capabilities of the platform and establishing the limitations imposed by its baseline hardware architecture.

This chapter provides an overview of the PIB manipulator and examines the main mechatronic limitations of its hardware architecture, with particular attention to its structural, kinematic and actuation systems and their comparison with solutions adopted in professional robotic platforms.

2.1 System Overview and Kinematic Framework of the PIB Robotic Arm

The physical structure of the PIB manipulator is constructed primarily from 3D-printable thermoplastics, predominantly Polylactic Acid (PLA) and Acrylonitrile Butadiene Styrene (ABS). These materials form both the structural elements of the robotic arm and the external housings for the embedded electronics. The resulting architecture provides a lightweight and readily manufacturable mechanical framework while maintaining the modularity required for assembly and modification.

From a kinematic perspective, the PIB robotic arm adopts a conventional serial manipulator configuration[4]. The forward and inverse kinematics of the right and left arms are formalized using Denavit-Hartenberg (DH) parameters and implemented through the Python-based PIB Software Development Kit (PIB-sdk), which interfaces with the Robotics[16]. Each arm provides a nominal six degrees of freedom (DOF) at the macro-kinematic level, distributed across the shoulder, elbow and wrist joints. This count excludes the more complex micro-kinematics of the underactuated hand.

The mechanical motion of the manipulator is generated entirely by commercially available servomotors designed for radio-controlled (RC) models. This approach avoids the use of industrial-grade brushless DC (BLDC) motors, frameless torque motors and harmonic-drive gearboxes typically found in professional robotic platforms. Instead, it relies on high-torque hobby servos controlled via standard pulse-width modulation (PWM), in which the desired joint positions are encoded as microsecond-level pulse widths generated by an intermediate controller or directly by the embedded system.

Different actuator classes are assigned according to joint-level requirements. Compact, standard-size servos such as the MG996R[17] and DS3225[18] are employed in distal sections of the manipulator where high force requirements are less critical. The MG996R, with a torque capability of approximately 12 kg $\cdot$ cm, is intended for finger actuation, while the DS3225, rated at around 21 kg $\cdot$ cm, is used for wrist motion.

In contrast, the primary joints (shoulder horizontal, shoulder vertical and elbow) require significantly higher torque and greater mechanical robustness. These joints therefore employ the DS5180[19], a large-format DS51-series servo rather than the standard-size servos used elsewhere in the manipulator. The DS5180 provides up to 80 kg · cm of rated torque, with a 65 × 30 × 48 mm housing and a mass of approximately 158 g, providing the increased actuation capacity required at these joints.

The specifications of the primary actuators utilized in the PIB manipulator are critical to understanding the system's physical limitations, as detailed in the technical comparison below:

Table 2.1: Technical comparison and deployment scope of the actuators utilized in the original PIB manipulator project.

	TowerPro MG996R[17]	**DSSERVO DS3225**[18]	**DSSERVO DS5180**[19]
Operating Voltage	4.8 V – 6.6 V	4.8 V – 6.8 V	6.0 V – 8.4 V
Stall Torque (Low Voltage)	≈ 9.4 kgf·cm @ 4.8 V (1.4 A)	21 kgf·cm @ 5.0 V (1.9 A)	85 kgf·cm @ 6.0 V (4.1 A)
Stall Torque (Max Voltage)	11 kgf·cm @ 6.0 V (1.8 A)	24.5 kgf·cm @ 6.8 V (2.3 A)	105 kgf·cm @ 8.4 V (6.5 A)
Operating Speed (Low)	0.19 s / 60° @ 4.8 V	0.15 s / 60° @ 5.0 V	0.24 s / 60° @ 6.0 V
Operating Speed (Max)	0.15 s / 60° @ 6.0 V	0.13 s / 60° @ 6.8 V	0.19 s / 60° @ 8.4 V
Gear Material	Metal	Metal	Stainless Steel / Metal
Rotation Angle	180°	180° or 270°	180° or 270°
Weight	55 g	60 g	158 g
Dimensions (L×W×H)	40.7 × 19.7 × 42.9 mm	40 × 20 × 40.5 mm	65 × 30 × 48 mm
Waterproof Rating	None	IP66	IP67
Scope	Finger concentric movement	Wrist concentric, Elbow lateral movements	Shoulder horizontal, Shoulder vertical, Elbow movements

While the integration of these commercial RC servomotors enables rapid, cost-effective assembly and radically simplifies the electromechanical integration process for hobbyists and students, it introduces profound, insurmountable limitations regarding torque density, control bandwidth, mechanical impedance and sensory feedback. The kinematic model assumes that the robotic links are infinitely rigid and the actuators act as perfect joints capable of reaching any commanded angle. However, the physical reality of the PIB's polymer construction and COTS actuators deviates significantly from this idealized mathematical model. The subsequent sections of this chapter rigorously analyze how the reliance on position-controlled, underactuated and sensor-deprived RC servos curtails the dynamic performance, trajectory fidelity and manipulation reliability of the PIB robotic arm, preventing its application in complex, force-dependent mechatronic tasks.

2.2 Control Paradigm Constraints: Closed-Loop Position Control

A fundamental and systemic limitation of the PIB manipulator arises from the internal, unalterable control architecture of the commercial RC servomotors it employs. In the realm of advanced robotics, such as industrial manipulators designed for physical human-robot interaction, actuators rely on cascaded control[5]. In these sophisticated architectures, torque (or motor current) forms the innermost control loop operating at a very high frequency, followed by an intermediate velocity loop and an outermost position loop. This cascaded structure allows roboticists to implement advanced, compliant control paradigms such as impedance control, admittance control and operational space formulation, where the dynamic relationship and energy exchange between the manipulator and its environment are explicitly managed and mathematically guaranteed to remain stable.

In stark contrast, the COTS RC servos utilized in the PIB operate strictly on an isolated, closed-loop position control paradigm. The internal electromechanical architecture of a standard RC servo, such as the MG996R or DS3225, consists of a small brushed DC motor, a multi-stage high-ratio reduction gearbox, a rotary potentiometer mechanically coupled to the output shaft for position feedback and an embedded microcontroller board. The governing electromechanical dynamics of the internal DC motor are classically modeled by the net current balance law (2.1) and the mechanical torque balance law (2.2):

$$v(t) = R\,i(t) + L\,\frac{di(t)}{dt} + K_e\,\dot{\theta}(t) \tag{2.1}$$

$$\tau(t) = J\ddot{\theta}(t) + B\dot{\theta}(t) = K_t i(t) \tag{2.2}$$

Where $v(t)$ represents the applied voltage across the motor terminals, R is the armature resistance, L is the armature inductance, K_e is the back-electromotive force (EMF) constant, $i(t)$ is the armature current, $\tau(t)$ is the generated torque, J is the load inertia (including the reflected inertia of the gearbox), B is the viscous friction coefficient, K_t is the torque constant and $\theta(t)$ is the angular position of the motor shaft.

Despite this underlying electromechanical complexity, the user of the PIB manipulator only has access to a single input parameter: the desired angular position reference. The external high-level controller (e.g., the Raspberry Pi running ROS2 or the PIB-sdk) transmits a PWM signal, typically with a pulse width varying between 500 µs and 2500 µs, which corresponds linearly to an angular range of 0° to 180° or 270°. The internal embedded microcontroller reads this PWM signal,

compares it against the analog voltage returned by the potentiometer and computes a position error. An embedded Proportional-Integral-Derivative (PID) algorithm then calculates the necessary control effort, adjusting the duty cycle of an internal H-bridge to drive the DC motor until the position error approaches zero.

Because this PID control architecture is entirely encapsulated within the hardware of the servo, the specific controller gains (K_p, K_i, K_d) are fixed by the manufacturer during production and remain completely inaccessible to the end-user. This "black-box" configuration dictates that the motor will blindly attempt to minimize the position error regardless of the external environmental forces applied to the manipulator link. If the PIB arm is commanded to move to a spatial coordinate but its path is physically obstructed by a rigid object, the internal position error remains non-zero. In response, the internal PID controller, specifically the integral term, will rapidly accumulate this error, maximizing the duty cycle to the motor to supply maximum current in a futile attempt to reach the commanded setpoint.

The mechatronic consequences of this rigid, unyielding position-control limitation are profound and severely restrict the operational domain of the robot. Primarily, the arm is fundamentally incapable of exhibiting mechanical compliance. In dynamic, unstructured environments, rigid position control without corresponding torque saturation mechanisms leads inevitably to destructive collisions. The servo acts as a source of infinite mechanical impedance; it will attempt to move to the coordinate until either the manipulated object is crushed, the 3D-printed link fractures or the internal metal alloy gears of the servo strip under the extreme stress.

Furthermore, tasks requiring precise force regulation or stable contact transitions (such as wiping a surface, inserting a peg into a tight-tolerance hole or turning a crank) are impossible to stabilize. These tasks require the manipulator to modulate its mechanical impedance according to the generalized dynamic equation:

$$M\ddot{x} + B\dot{x} + Kx = F_{ext} \tag{2.3}$$

Where the mass M, damping B and stiffness K matrices are shaped by the controller in response to the external force F_{ext}. Because the PIB's servos enforce infinite stiffness (K) and do not measure F_{ext}, the manipulator will experience severe dynamic instability upon environmental contact.

Finally, this architecture introduces chronic jitter and limit cycling around the servo's deadband. The DS3225 and MG996R possess deadband widths of 3µs and 5µs, respectively. As the actuator approaches its torque limit (e.g., attempting to hold an 8 kg · cm gravitational payload on a joint powered by a 10 kg · cm MG996R servo), the internal controller struggles to maintain the setpoint. The gravitational disturbance pushes the potentiometer out of the deadband, prompting a sudden

corrective burst of current, which overshoots the target, pushing it out the other side. This results in high-frequency, destructive oscillations (jitter) that degrade trajectory tracking, generate acoustic noise and rapidly accelerate the mechanical wear of both the internal gearbox and the 3D-printed joints. Consequently, the PIB manipulator is strictly confined to kinematic trajectory following in empty space; any application requiring dynamic force interactions is fundamentally precluded by the closed-loop position-only architecture of its constituent RC servos.

2.3 Torque Density Deficits and Thermal Saturation

Beyond the algorithmic limitations of position control, the physical and thermodynamic characteristics of commercial RC servos severely restrict the operational envelope, payload capacity and continuous exertion capabilities of the PIB manipulator. High-performance robotic systems require actuators with high torque density (Nm/kg), i.e., the ability to generate substantial torque while minimizing mass to avoid adding unnecessary inertia to the kinematic chain. In serial manipulators, any excess mass situated at the distal joints (e.g., the wrist and elbow) geometrically amplifies the torque required at the proximal joints (e.g., the shoulder) simply to hold the arm against gravity.

Standard hobby servos utilize low-cost brushed DC motors or simple coreless motors paired with multi-stage spur gearboxes. The torque density and thermodynamic efficiency of these traditional brushed DC motors are drastically inferior to the frameless AC servo motors or high-pole-count brushless DC (BLDC) motors typically found in modern industrial and research arms. An industrial BLDC motor achieves a superior torque-to-weight ratio through the use of rare-earth permanent magnets, advanced field-oriented control (FOC), sinusoidal commutation and optimized thermal pathways that minimize both copper (I^2R) and iron core losses.

In contrast, to generate usable torque, servos like the MG996R and DS3225 must rely on massive mechanical gear reduction ratios (e.g., 275:1 for the DS3225) to convert the high-speed, low-torque rotation of the internal DC motor into a high-torque output. While these high gear ratios succeed in amplifying static torque, they introduce severe mechatronic penalties. High reduction ratios create immense reflected inertia (which scales with the square of the gear ratio), rendering the joints physically non-backdrivable. This eliminates any inherent passive compliance the system might otherwise possess, further exacerbating the collision dangers discussed in the previous section. Moreover, the multi-stage gear trains introduce significant backlash (lost motion) and high mechanical friction, corrupting the kinematic repeatability of the arm.

More critically, these RC servos suffer from acute, unavoidable thermal saturation during prolonged efforts or high-acceleration trajectories. When the PIB arm is commanded to hold a static payload against gravity, the joint velocity is zero. Referring back to the standard DC motor voltage equation (2.1), at a steady-state holding position, the inductive term drops out and the back-EMF (K_e) drops to zero. Consequently, the entire applied voltage is dropped across the internal resistance of the motor windings ($v = R \cdot i$). The electrical power dissipated purely as heat in the motor coils is governed by Joule's First Law: $P_{loss} = I^2R$

To sustain a holding torque hold, the motor must continuously draw a proportional holding current $I_{hold} = \tau_{hold}/K_t$. Because RC servos are packaged in highly compact, enclosed plastic or hybrid plastic-aluminum cases designed for brief, intermittent use in RC vehicles, they possess an exceptionally high thermal resistance (R_{th}). The system lacks active cooling, heat sinks or liquid cooling jackets found in high-performance actuators. As a result, the heat generated by the I^2R losses accumulates rapidly within the stator windings, unable to effectively dissipate into the surrounding environment.

Under prolonged static stress, such as holding an object aloft or during high-frequency dynamic accelerations that demand large transient currents to overcome the rotor and gearbox inertia, the internal temperature of the servomotor elevates exponentially. This thermal accumulation triggers several cascading mechatronic failures:

1. **Magnetic Flux Degradation:** As the temperature of the permanent magnets inside the brushed DC motor rises, their remnant flux density decreases. This physical degradation reduces the motor's torque constant (K_t). Because the internal PID controller still demands the same holding torque to maintain the position setpoint, the motor must draw even more current ($I = \tau/K_t$) to compensate for the weakened magnetic field, which in turn generates even more heat (I^2R), initiating an irreversible thermal runaway loop.

2. **Structural Deformation of the Chassis:** The PIB's structural framework is constructed from 3D-printed PLA or ABS. PLA is notoriously susceptible to heat, possessing a remarkably low glass transition temperature ($T_g \approx 60°C$). As the servos thermally saturate, conductive heat transfer from the servo casing to the 3D-printed mounting brackets will cause the polymer to soften, warp and plastically deform. This permanently compromises the kinematic alignment of the manipulator, introducing massive geometric errors into the Denavit-Hartenberg mathematical model.

3. **Insulation and Electronic Failure:** Continued thermal saturation will eventually melt the thin enamel insulation coating the copper windings, resulting in an internal short circuit or cause the H-bridge MOSFETs on the internal controller board to exceed their maximum junction temperature and burn out, resulting in total actuator loss.

Therefore, while manufacturers boast impressive peak specifications (i.e., the DS3225 claims a stall torque of 25 kg $\cdot$ cm at 6.8V) these figures represent strictly transient ratings. The continuous, thermally stable torque is a mere fraction of this value. The PIB manipulator is thus inherently incapable of executing tasks that require prolonged exertion, lifting moderate payloads or operating at high duty cycles in industrial or continuous-research settings without suffering rapid thermal degradation.

2.4 Underactuated Kinematics: Single-Tendon Routing and Passive Spring Return

The distal end of the PIB manipulator, comprising the hand and wrist assembly, exhibits a highly constrained, underactuated kinematic architecture. In robotics, underactuation is defined as a system paradigm in which the number of actively controlled inputs (i.e., the servomotors) is strictly less than the total number of kinematic degrees of freedom (DOF) in the mechanism[4].

To achieve a humanoid form factor without overburdening the delicate distal structures, the PIB relies on remotely driven tendon architecture for its hand and wrist. While this approach successfully enables a robotic hand to adaptively envelop objects and drastically reduces hardware complexity, the specific implementation chosen for the PIB baseline introduces extreme physical constraints that critically limit both its dexterity and efficiency.

Specifically, the PIB hand implements a single-tendon routing scheme paired with a passive spring return mechanism for each digit. In this configuration, the servomotor actuates a pulley that pulls a flexible tendon routed through a Bowden cable sheath across multiple interphalangeal joints. The tension applied to the tendon generates a unidirectional flexion moment, causing the finger to close around an object. To open the hand, the servomotor reverses its position to release the tension on the cable, relying entirely on a passive torsion spring to provide the restoring force required to extend the finger back to its resting posture.



Figure 2.1: PIB torso-mounted bowden-cable actuation system. Adapted from *PIB – Printable Intelligent Bot: Open-Source 3D-Printable Humanoid Robot.*

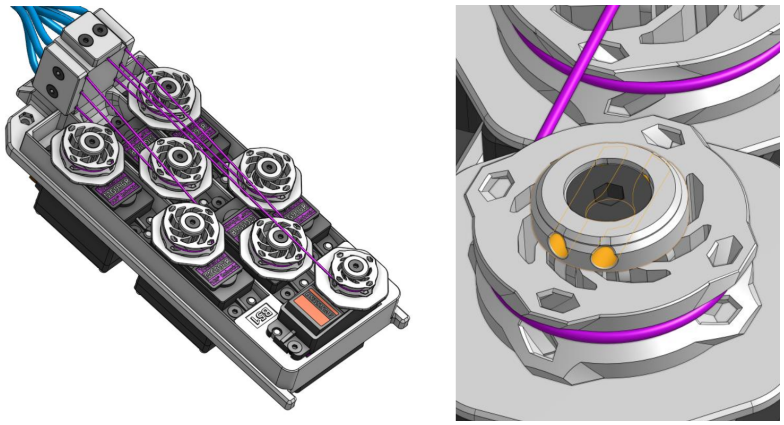


Figure 2.2: Detailed view of the PIB torso servomotor bank and tendon pulley mechanism. Adapted from *PIB – Printable Intelligent Bot: Open-Source 3D-Printable Humanoid Robot.*

While this biomimetic approach successfully minimizes hardware complexity and significantly lowers distal mass, it introduces severe mechanical and control limitations that cripple the manipulator's dexterity, efficiency and reliability.

2.4.1 Bowden-Cable-Driven Torso-Mounted Actuation

One of the most distinctive characteristics of the PIB manipulator is the remote placement of the hand actuation system. Rather than positioning the servomotors inside the arm or forearm, the design centralizes all distal actuation within the robot torso. A bank of seven RC servos is responsible for hand and wrist motion: four servos independently actuate the index, middle, ring and little fingers, two servos control the thumb and a final servo governs wrist flexion. The mechanical force generated by these actuators is transmitted through flexible Bowden cables routed across the entire arm structure, traversing the shoulder assembly, upper arm, elbow and forearm before reaching the tendon anchor points inside the hand.

This architectural choice represents a deliberate mechatronic compromise. By relocating the actuator bank away from the distal extremity, the manipulator significantly reduces the inertial mass of the arm itself. Lower distal mass decreases the reflected inertia experienced by the shoulder and elbow joints, reducing torque requirements during dynamic movements and improving overall kinematic agility. This differs from platforms such as the InMoov humanoid robot[14], where hand-actuation servos are commonly integrated directly into the forearm (figure 1.3). While shorter tendon paths reduce friction and simplify cable routing, they also introduce a substantial distal weight penalty that negatively affects arm responsiveness. The PIB avoids this inertial disadvantage through torso-mounted actuation; however, the exceptionally long Bowden cable paths increase mechanical friction, compliance and tension losses within the transmission system.

2.4.2 Distal Constraints and Dexterity Limitations

Although each finger is assigned an independent actuator, the articulation within each digit remains mechanically coupled. A single tendon spans the metacarpophalangeal and interphalangeal joints of each finger, preventing independent control of individual phalanges. As tension is applied, the finger closes according to a predefined mechanical sequence until external contact constrains its final configuration. This underactuated behavior enables adaptive grasping but inherently limits fine motion control.

More significantly, the four primary fingers operate exclusively within the sagittal plane and lack abduction/adduction degrees of freedom. The fingers cannot spread apart or move laterally toward one another, severely restricting the manipulator's

ability to perform advanced in-hand manipulation or precision fingertip coordination. Tasks such as rolling objects between fingertips or executing adaptive tripod grasps on irregular geometries become mechanically infeasible.

The thumb is the only digit featuring a more advanced articulation scheme. One servo controls thumb flexion similarly to the other fingers, while a second servo actuates the thumb base joint, enabling rotational motion and limited abduction/adduction relative to the palm. This configuration allows the thumb to become partially opposable, enabling basic cylindrical and power grasps. Nevertheless, because the remaining fingers remain constrained to parallel motion planes, the overall dexterity of the hand remains limited, with grasping functionality largely reduced to open/close operations rather than precise object manipulation.

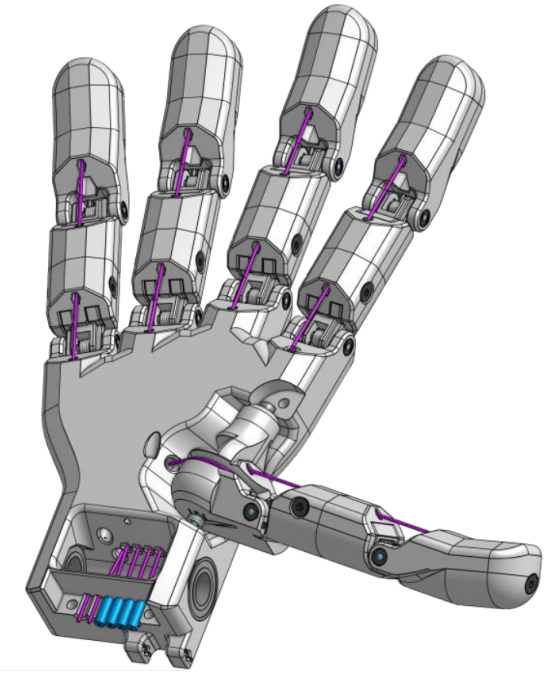


Figure 2.3: PIB underactuated right hand mechanism and tendon routing. Adapted from *PIB – Printable Intelligent Bot: Open-Source 3D-Printable Humanoid Robot.*

2.4.3 Transmission Nonlinearity and Capstan Friction Losses

The routing of tendons through flexible sheaths over articulating joints introduces highly nonlinear transmission dynamics, predominantly governed by the Capstan effect and complex Coulomb friction interactions[1, 10]. Because the tendons must cross shoulder, elbow and wrist joints, as the robotic arm changes its spatial pose,

the cumulative bending angle of the Bowden cable sheath inherently changes. The friction between the internal tendon and the outer sheath is mathematically modeled by the Capstan equation (also known as Eytelwein's formula):

$$F_{in} = F_{out} \cdot e^{\mu\theta} \tag{2.4}$$

Where F_{in} is the input tension applied by the proximal servomotor, F_{out} is the effective tension delivered to the distal finger joint, μ is the coefficient of kinetic friction between the tendon material and the inner lining of the sheath and θ is the cumulative bending angle (in radians) of the tendon routing path.

This exponential friction model dictates a catastrophic loss of mechanical efficiency. As the PIB's wrist and elbow joints flex to reach an object, increasing the cumulative angle θ, the efficiency of force transmission drops precipitously. A substantial and highly variable fraction of the servomotor's finite torque is wasted entirely on overcoming the internal friction of the sheath rather than performing useful work at the fingertip. Furthermore, the friction acts in opposition to the direction of motion. When the motor pulls the tendon (flexion), $F_{in} > F_{out}$. However, when the motor reverses direction to allow the passive spring to extend the finger, the friction direction flips and the spring must overcome both the internal joint friction and the sheath friction ($F_{out} > F_{in}$) to pull the cable back. This directional asymmetry results in massive mechanical hysteresis. Consequently, the mapping between the servomotor's angular position and the actual position of the fingertip becomes uncertain, non-deterministic and configuration-dependent, making accurate inverse kinematic solutions for the fingers very complex.

2.4.4 Limitations of the Passive Spring Return

The architectural choice of utilizing a passive spring for extension, rather than an antagonistic tendon routing (where a second, dedicated tendon actively pulls the finger open), severely curtails the dexterity capabilities of the PIB manipulator.

In a passive spring return system, the restoring force is solely a function of Hooke's Law:

$$F_s = k\Delta x \tag{2.5}$$

For the finger to reliably open, the chosen spring must be stiff enough (k) to overcome the internal friction of the complex tendon system, the friction of the 3D-printed pin joints and the gravitational load of the finger components. However, implementing a high-stiffness spring creates a severe parasitic load during the flexion phase; the servomotor must continuously expend a significant portion of its limited torque just to keep the spring compressed, further diminishing the available grasping force that can be applied to the target object.

Moreover, the absence of antagonistic actuation eliminates the ability to perform simultaneous position and stiffness control at the joint level. In a fully actuated antagonistic system (akin to human bicep/triceps pairs), co-contracting opposing tendons allows the controller to dynamically stiffen a joint without changing its position, providing mechanical stability against external perturbations. The PIB's single-tendon architecture possesses a fixed, highly asymmetric compliance profile: it is highly compliant in extension (governed solely by the weak spring) and rigidly unyielding in flexion (governed by the inextensible tendon and the rigid position-control of the servo). This prevents the fine-tuning of joint mechanical impedance, which is an absolute prerequisite for dexterous, in-hand manipulation of complex objects.

2.4.5 Loss of Kinematic Determinism

Because of the underactuated topology, the final configuration of the phalanges during a grasp is not deterministically controlled by the robot's kinematics, but rather by the external geometric constraints of the target object. As the single tendon is pulled, the proximal joints will rotate until they make physical contact with the object, at which point the joint is blocked. The tendon continues to slide through the routing, propagating the flexion force to the unblocked distal joints until the entire finger envelops the shape. While this "mechanical intelligence" facilitates adaptive power grasps around irregular geometries, it absolutely precludes precision pinch grasps or the independent manipulation of objects within the hand. The manipulator is thus relegated to basic, binary open-close power operations, severely limiting its utility in any complex, precision-dependent mechatronic application.

2.5 The Absence of Proprioceptive Telemetry and Tactile Feedback

The culmination of the hardware limitations detailed in the preceding sections (rigid position-controlled servos, severe thermal limits and the highly nonlinear friction of long Bowden cables) is fundamentally exacerbated by the complete absence of proprioceptive and tactile telemetry in the baseline PIB architecture. In biological systems and advanced robotic manipulators, high-fidelity sensory feedback is an absolute, non-negotiable necessity for executing reliable, safe physical interactions with the environment.

2.5.1 The Critical Mechatronic Role of Haptic Feedback

In robotic manipulation, applying the correct magnitude of gripping force is a delicate, continuous optimization problem. If the applied force is too low, the normal force will not generate sufficient static friction ($\mu_s F_N$) between the fingertips and the object to overcome gravity, resulting in slip and dropping the payload. Conversely, if the applied force is too high, fragile or deformable objects will be crushed or plastically deformed.

Without real-time force feedback, a robotic controller cannot formulate a closed-loop grasping strategy; the system is forced to operate entirely open-loop with respect to applied force. Studies in robot-assisted surgery (RAS) and remote teleoperation have extensively documented the hazards of manipulating objects without haptic feedback[8]. These studies demonstrate that the lack of tactile sensations inevitably results in unintended dynamic mismatches, the excessive application of maximal force, high inconsistency in grasping parameters and ultimately, catastrophic damage to either the manipulated object or the robot itself.

2.5.2 Complete Lack of Sensor Integration in the Baseline Project

Because the PIB relies exclusively on standard COTS RC servos and ubiquitous 3D-printed parts to keep costs low, the baseline system possesses absolutely no hardware capacity to measure the physical effort it exerts on the environment.

There are no strain gauges integrated into the tendon routing, no multi-axis force/torque sensors mounted at the wrist and no tactile piezoresistive sensor arrays embedded on the fingertips. The only sensory data available within the entire actuation chain is the internal analog potentiometer inside the hobby servo, which strictly reports the angular position of the motor's internal output shaft.

When the central controller commands the PIB to close its hand around an object, it does so by blindly commanding a specific geometric angular position for the torso-mounted finger servos based on predetermined waypoints or visual estimation. If the target object is physically larger or stiffer than the geometric aperture defined by the commanded servo angle, the fingers will strike the object before the servo reaches its target setpoint.

As analyzed in section 2.2, the "black-box" PID controller within the servo cannot detect that a physical obstacle has been encountered; it only detects a persistent mathematical position error. Consequently, it will saturate the motor current in

an attempt to force the fingers through the solid object to reach the commanded angle. Because the central controller receives no telemetry indicating an impact or rising physical resistance, it cannot intervene to halt the command.

The manipulator is thus functionally blind to the physical forces of its environment. It cannot utilize mechanical compliance, it cannot perform object slip detection and it cannot dynamically regulate grip pressure. The system is strictly restricted to executing pre-programmed kinematic waypoints in free space. The inevitable result of attempting to grasp an unyielding object without telemetry is mechanical failure: either the fragile payload is destroyed, the internal 3D-printed linkages fracture, the Bowden cables snap under excessive tension or the torso-mounted servos thermally saturate and burn out as they draw stall current indefinitely. This critical limitation renders the baseline PIB wholly inadequate for reliable, closed-loop physical interaction with unstructured environments, dictating the absolute necessity for the mechatronic upgrades.

Chapter 3

Prototype Fabrication, Mechatronic Integration and System Calibration

Following the architectural analysis presented in the previous chapter, this chapter describes the fabrication, modification and integration of the PIB robotic arm prototype. A significant part of the development process involved identifying and sourcing all the hardware required for the modified system while working within a constrained budget. Although pib.rocks provides a complete hardware kit for the PIB, its cost was considered excessive for the scope of this project. The required components were therefore identified individually and sourced from different suppliers, with the aim of obtaining hardware equivalent to or exceeding, the specifications of the original components at a substantially lower cost.

This procurement process required the preparation of a detailed Bill of Materials (BOM), covering the mechanical, electrical and electromechanical components required for the complete assembly. Particular attention was given to actuator specifications, mechanical compatibility, component availability and cost. In addition to the components already required by the original PIB architecture, the BOM had to account for the hardware introduced by the modifications described in this chapter. The selection and procurement of these components represented a significant part of the practical development process, as the compatibility of individual parts had to be verified before fabrication and assembly could proceed.

The hardware was sourced primarily from established industrial distributors, with RS Components used for mechanical hardware and FDM materials and DigiKey used for electronic components and actuators. Additional components were obtained from alternative suppliers where this provided a more suitable combination of price, availability or technical specifications. This approach allowed the complete hardware set to be assembled independently of the official PIB kit while maintaining compatibility with the existing architecture and, where possible, improving its mechanical and electromechanical performance.

Following the procurement phase, the prototype was fabricated and assembled, including the preparation of the selected materials, optimization of the FDM manufacturing process, integration of the new actuators and mechanical components, structural modifications and final system calibration. The following sections describe these activities, beginning with the actuation and electromechanical modifications and continuing through the manufacturing, structural integration and commissioning of the robotic arm.

3.1 Actuation Architecture and Electromechanical Upgrades

3.1.1 Actuator Upgrades for Distal and Secondary Joints

The performance and payload capacity of the manipulator depend significantly on its primary actuators. To overcome the limitations of standard low-cost RC servos, the robotic arm utilizes DFRobot DC5535 servomotors[22]. This specific model was chosen for two main reasons: it provides higher torque than the baseline components and, most importantly, a sufficient quantity was already available in the laboratory inventory. Repurposing these existing actuators eliminated the need for new purchases, ensuring strict adherence to the project's budgetary constraints.

From a technical perspective, the DC5535 is a high-performance digital servo based on an internal coreless Direct Current (DC) motor architecture. Unlike conventional brushed DC motors that rely on an iron core to support the windings, the coreless design employs a lightweight, self-supporting winding structure, significantly reducing rotor inertia. This results in rapid angular acceleration and deceleration, enabling responsive, low-latency joint actuation suitable for dynamic trajectory tracking.

Mechanically, it provides a stall torque of up to 35 kg $\cdot$ cm at its peak operating voltage. This represents a substantial and necessary increase compared to the 21 kg $\cdot$ cm offered by the DS3225[18], which was previously considered for wrist and elbow abduction/adduction. The higher torque output is required to handle

the compounded gravitational loads introduced by distal 3D-printed linkages and custom end-effector hardware. In addition, the DC5535 replaces the six MG996R[17] servos originally assigned to finger actuation, simplifying the actuator set while improving overall performance consistency.

To ensure reliable torque transmission under load, the actuator incorporates a fully metal reduction gearbox designed to withstand shock and repetitive stress. Dual ball bearings on the output shaft reduce lateral deflection and friction, contributing to improved positional stability and repeatability.

In contrast to lower-cost servos that rely on analog potentiometer-based feedback, the DC5535 integrates a digital position feedback control system, providing higher resolution, reduced noise sensitivity and improved closed-loop accuracy. This enables more precise joint positioning and enhances control robustness, particularly under varying load conditions.

Finally, the DC5535 offers a high level of environmental protection, with an IP67-rated enclosure that ensures resistance to dust and water ingress. The internal electronics are protected with conformal coating, while critical components are sealed to prevent contamination. The external CNC-machined aluminum casing aids in thermal dissipation, reducing the impact of Joule heating during prolonged operation. Although not intended for submerged use, this level of protection improves reliability in typical laboratory conditions, including exposure to humidity, incidental liquid contact and sustained mechanical loading.

Table 3.1: Technical comparison of the upgraded servos versus the original design for the Distal and Secondary Joints.

	Intended in original design:		**Substitute:**
	TowerPro MG996R[17]	**DSSERVO DS3225**[18]	**DFRobot DC5535**[22]
Operating Voltage	4.8 V – 6.6 V	4.8 V – 6.8 V	4.8 V – 7.4 V
Stall Torque (Low Voltage)	9.4 kgf·cm @ 4.8 V (1.4 A)	24.5 kgf·cm @ 5.0 V (1.9 A)	24.9 kgf·cm @ 5.0 V (1.9 A)
Stall Torque (High Voltage)	11 kgf·cm @ 6.0 V (1.8 A)	28 kgf·cm @ 6.8 V (2.3 A)	35 kgf·cm @ 7.0 V (2.8 A)
Operating Speed (Low)	0.19 s / 60° @ 4.8 V	0.16 s / 60° @ 5.0 V	0.37 s / 60° @ 5.0 V
Operating Speed (High)	0.15 s / 60° @ 6.0 V	0.14 s / 60° @ 6.8 V	0.25 s / 60° @ 7.0 V
Gear Material	Metal	Metal	All-metal
Rotation Angle	180°	180° or 270°	180°
Weight	55 g	60 g	≈75 g
Dimensions (L×W×H)	40.7 × 19.7 × 42.9 mm	40 × 20 × 40.5 mm	40 × 20 × 40.5 mm
Waterproof Rating	None	IP66	IP67
Scope	Finger concentric movement	Wrist concentric, Elbow lateral movements	Finger concentric, Wrist concentric, Elbow lateral movements

3.1.2 Actuator Upgrades for Primary Joints

During the conceptual phase of the prototype's development, a feasibility study was conducted to evaluate the adoption of Brushless DC (BLDC) actuators for the primary joints (shoulder horizontal axis, shoulder vertical axis and elbow). In the original configuration, these joints were driven by DS5180[19] servos, which provide a stall torque of approximately 80 kg · cm but are characterized by relatively large dimensions.

The proposed BLDC solution, potentially paired with custom cycloidal gearboxes, was considered due to its advantages in terms of efficiency, backdrivability and control flexibility. However, this approach was ultimately discarded following both technical and economic assessment. From a mechanical standpoint, cycloidal transmissions are subjected to significant localized shear forces at the contact interfaces. When implemented using polymer-based, additively manufactured components, these stresses can lead to rapid wear, deformation and eventual mechanical failure. In addition, prolonged loading conditions introduce the risk of thermal deformation, which can result in increased friction, loss of motion fidelity and potential seizure of the mechanism. These factors make a fully custom solution unsuitable for reliable operation in the current application context.

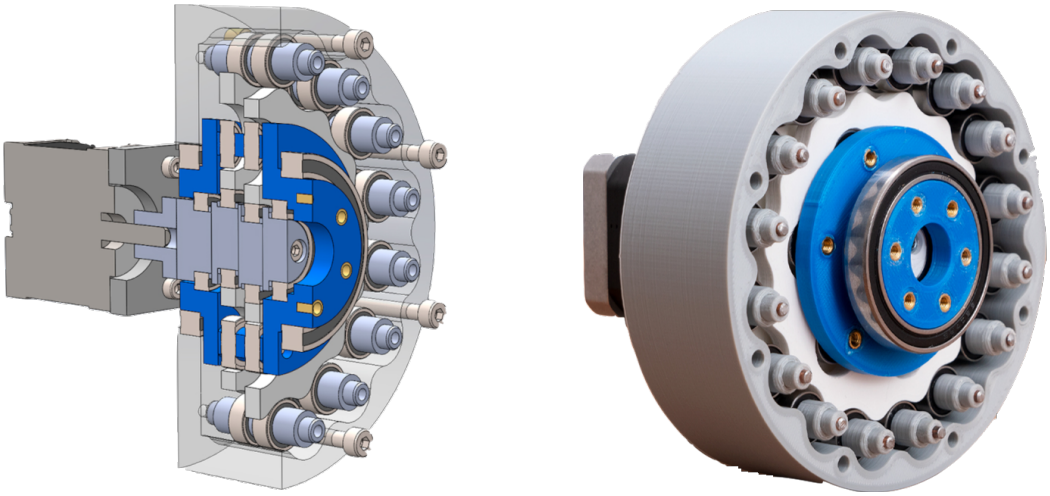


Figure 3.1: Architecture of a 3D-printed servo cycloidal speed reducer. Adapted from *How Cycloidal Drives Work and Designing 3D-Printed Reducers.*

From an integration perspective, BLDC systems require a substantially more complex control architecture compared to conventional servomotors. Accurate operation necessitates Field-Oriented Control (FOC), along with high-resolution position sensing and dedicated motor drivers, significantly increasing system complexity, development time and cost.

As a result, the final design adopts DS51150[21] servos for the primary joints. These actuators provide a stall torque of up to 150 kg · cm, representing a considerable increase over the previously used DS5180 units, while maintaining comparable form factor and mass. The DS51150 supports a wide operating range of up to 270° and operates at 12.6 V, with peak current demands reaching approximately 8.3 A under stall conditions. This increase in power density necessitates corresponding modifications to the power distribution board.

In addition to the improved torque output, the DS51150 incorporates digital control with position feedback based on magnetic field sensing, enhancing accuracy and robustness compared to traditional feedback mechanisms. The actuator also features a full metal gear train and supports firmware-level programmability, enabling adjustment of motion parameters and operational limits. Although the nominal speed is slightly reduced, the significant gain in torque and overall performance makes this solution more suitable for handling the mechanical demands of the manipulator's primary joints.

Table 3.2: Technical comparison of the upgraded servos versus the original design for the Primary Joints.

	Intended in original design:	**Substitute:**
	DSSERVO DS5180[19]	**DSSERVO DS51150**[21]
Operating Voltage	6.0 V – 8.4 V	9.0 V – 12.6 V
Stall Torque (Low Voltage)	85 kgf · cm @ 6.0 V (4.1 A)	150 kgf · cm @ 10.0 V (7.4 A)
Stall Torque (High Voltage)	105 kgf · cm @ 8.4 V (6.5 A)	173 kgf · cm @ 12.6 V (8.3 A)
Operating Speed (Low)	0.24 s / 60° @ 6.0 V	0.24 s / 60° @ 10.0 V
Operating Speed (High)	0.19 s / 60° @ 8.4 V	0.19 s / 60° @ 12.6 V
Gear Ratio / Material	357:1 / Stainless Steel	357:1 / Stainless Steel
Rotation Angle	180° or 270°	180° or 270°
Weight	158 g	175 g
Dimensions (L×W×H)	65 × 30 × 48 mm	65 × 30 × 48 mm
Waterproof Rating	IP67	IP67
Scope	Shoulder horizontal, shoulder vertical, elbow	

3.2 End-Effector Kinematics and the Rejection of Antagonistic Tendon Routing

The distal kinematics of the robotic manipulator—specifically the articulation of the phalanges and the wrist—are driven by a remote underactuated tendon mechanism. Mechanical force generated by the proximally mounted servo bank is transmitted to the hand via flexible Bowden cables traversing the shoulder, upper arm, elbow and forearm. As formalized in section 2.4.3, this transmission path is governed by the Capstan friction model (equation 2.4), which dictates exponential force attenuation as a function of cumulative sheath curvature[1, 10].

In biological systems and advanced robotic end-effectors, joints typically rely on antagonistic configurations where opposing tendon pairs (agonist and antagonist) actively govern flexion and extension. This dual-cable arrangement enables active bi-directional positioning, dynamic co-contraction and programmable joint stiffness. Despite these theoretical kinematic benefits, an antagonistic routing architecture was explicitly investigated and rejected for this prototype based on structural, volumetric and mechanical efficiency constraints.

Implementing an antagonistic cable system would double the transmission infrastructure, expanding the distal routing from seven to fourteen independent Bowden lines across the arm (two antagonistic lines per finger, plus four lines for the thumb and wrist). The internal cross-section of the 3D-printed wrist linkage and arm joint pivots cannot accommodate fourteen individual PTFE sheaths without severely constraining their routing paths. Packaging this bundle within the existing link geometry would force the conduits into contorted paths with tight localized bending radii.

As established by the Capstan relationship analyzed in section 2.4.3, tightening the routing bend radii sharply increases the cumulative angular curvature (θ) along the arm. Under such high cumulative wrap angles, friction across the conduit lining scales exponentially, causing transmission efficiency to collapse. In an antagonistic arrangement, the torso-mounted DFRobot DC5535 servomotors would expend the vast majority of their available 35 $\mathrm{kg \cdot cm}$ torque merely overcoming internal sheath friction and resisting the opposing cable's parasitic drag. Consequently, the residual mechanical tension delivered to the fingertips would be insufficient to execute functional grasping tasks.

Furthermore, an antagonistic configuration would require an active mechanical tensioning mechanism to maintain continuous, equalized preload on both sides of the cable simultaneously, compensating for differential path-length variations during multi-joint articulation and preventing tendon slack or binding.

However, implementing reliable, high-load tensioners using 3D-printed components introduces severe operational vulnerabilities. Maintaining constant cable tension across multi-axis joint movements requires stiff, heavy-duty springs that place continuous mechanical stress on printed assemblies. Over repeated dynamic cycles, 3D-printed tensioning mechanisms are prone to premature wear, friction buildup and mechanical jamming, making it difficult to guarantee long-term reliability and smooth sliding kinematics.

To preserve usable grasping torque, prevent severe transmission hysteresis and maintain electrical and mechanical economy, the single-acting pull tendon architecture was retained. Bi-directional motion is instead achieved by pairing active tendon flexion with precision-calibrated passive return springs integrated directly within the phalange joints.

3.3 Polymer Thermodynamics and Material Preparation

3.3.1 Tribological and Mechanical Superiority of PETG

The structural integrity, operational longevity and kinematic accuracy of an FDM-manufactured robotic system strongly depend on the thermomechanical and tribological properties of the selected filament material. Tribology is the science and technology of interacting surfaces in relative motion, focusing on the study of friction, wear and lubrication. In robotic applications, tribological behavior is particularly important in joints, sliding interfaces, servo mounts and mechanically stressed contact regions, where repetitive motion and vibration can progressively degrade dimensional accuracy and structural reliability.

For the fabrication of the upper limb, Polylactic Acid (PLA), commonly used in hobbyist additive manufacturing, was intentionally excluded in favor of Polyethylene Terephthalate Glycol (PETG). Specifically, the project utilized 3.0 kg of White Verbatim PETG filament sourced from RS Components, resulting in approximately 2.42 kg of structurally optimized printed components directly integrated into the manipulator.

The adoption of PETG was primarily motivated by its superior impact resistance, ductility, thermal stability and improved tribological characteristics compared to PLA. While PLA exhibits high rigidity, it is inherently brittle and prone to catastrophic fracture under impulsive loading conditions. PETG, by contrast, provides greater elasticity and toughness, allowing mechanically stressed components to undergo limited elastic deformation rather than sudden structural failure. This

behavior is particularly advantageous in robotic systems subjected to dynamic loading and repeated accelerations, where localized stress concentrations may otherwise induce crack propagation.

From a tribological perspective, PETG demonstrates improved wear resistance and lower susceptibility to surface degradation under cyclic mechanical interaction. In servo-driven assemblies, microscopic oscillations and vibrations at fastening points can progressively damage brittle materials, leading to loosening, creep or dimensional inaccuracies over time. PETG better tolerates these repeated frictional interactions due to its higher ductility and reduced tendency toward abrasive surface cracking. This contributes to improved long-term stability of mating interfaces, especially in motor housings, articulated link connections and load-bearing support structures.

Table 3.3: Technical specifications and material properties of the Verbatim White PETG filament.

Verbatim White PETG Technical Specifications[23]	
Tensile Strength	50 MPa
Flexural Strength	69 MPa
Flexural Modulus	2050 MPa
Charpy Impact Strength	8.1 kJ/m^2
Glass Transition Temp (T_g)	80°C
Filament Diameter Tolerance	1.75 mm ± 0.05 mm

3.3.2 Moisture Absorption Kinetics and Hydrolysis

While PETG offers exceptional mechanical properties, it is an intrinsically hygroscopic polymer, meaning its chemical matrix actively absorbs ambient atmospheric moisture over time. The moisture absorption kinetics of PETG follow a Fickian diffusion modelmodel[2], where water molecules permeate and become physically trapped between the long polymer chains.

If hygroscopic filament is processed through a 3D printer without prior dehydration, the results are mechanically catastrophic. As the moisture-laden PETG enters the printer's hotend at 235°C, the trapped water molecules undergo instantaneous phase change, boiling violently into steam. This expansion of gas inside the nozzle disrupts the laminar flow of the polymer melt, presenting macroscopically as severe stringing, surface popping, inconsistent extrusion line widths and the formation of microscopic voids within the printed part.

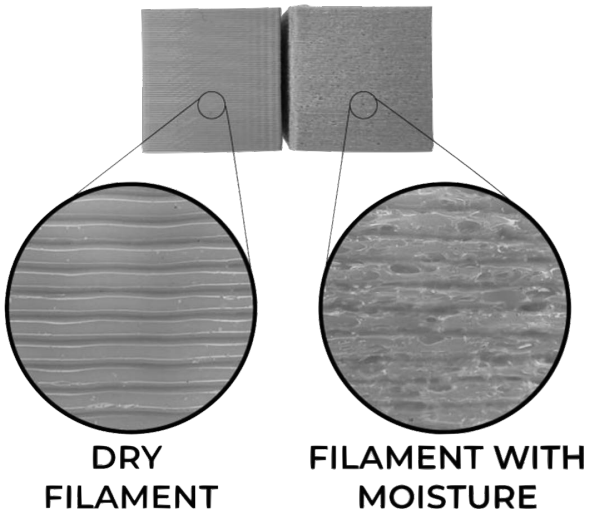


Figure 3.2: Effects of moisture absorption on PETG print quality. Adapted from *Material Guide: Polyethylene Terephthalate Glycol (PETG) and Moisture Dehydration Protocols*.

More critically, the presence of high-temperature steam triggers a chemical degradation process known as hydrolysis. Hydrolysis actively cleaves the covalent bonds of the ester groups within the PETG polymer chains, dramatically reducing the material's molecular weight. This chemical breakdown permanently destroys the polymer's theoretical 50 MPa tensile strength, resulting in brittle, fragile parts that suffer from severe layer delamination[2, 29]. Therefore, absolute dehydration of the filament is a non-negotiable prerequisite for fabricating load-bearing robotic components .

3.3.3 Dual-Environment Manufacturing and Polymer Dehydration

To guarantee the structural integrity of the approximately 2.42 kg of PETG utilized in the arm, a strict dehydration protocol was enforced prior to every fabrication cycle. The manufacturing workload was strategically bifurcated between two environments to optimize efficiency. A portion of the components was fabricated at the university laboratory utilizing a Raise3D Pro3 FDM printer (figure: 3.3b). In this controlled environment, a Creality SpacePi X4 Filament Dryer was employed.

The dryer was directly coupled to the printer via a PTFE filament tube, allowing the PETG to undergo continuous, active desiccation precisely as it was fed into the extruder, ensuring maximum structural quality and zero moisture-induced artifacts.

However, to avoid scheduling constraints associated with shared university equipment, the majority of the robotic manipulator (approximately 2 kg) was manufactured in a home-based environment. Since no dedicated filament dryer was available outside the laboratory, a standard domestic convection oven was used for filament drying. The use of a domestic oven requires particular care, as its temperature control is generally less precise than that of a dedicated drying system and may exhibit significant temperature fluctuations during the heating phase. For PETG, excessive temperatures can soften the filament and cause deformation or adhesion between adjacent layers of the spool, potentially rendering the material unusable.

To minimize this risk, the oven was first preheated without the filament until a stable temperature was reached. The internal temperature was monitored using external thermocouple probes rather than relying solely on the oven's built-in temperature control. Once a stable temperature between 65°C and 70°C was reached, the PETG spool was placed in the oven and dried for approximately 6 hours. This procedure provided a controlled method for reducing moisture in the filament while keeping the material below its glass transition temperature and avoiding thermal deformation.

3.4 FDM Manufacturing Platform: Upgrades and Calibrations

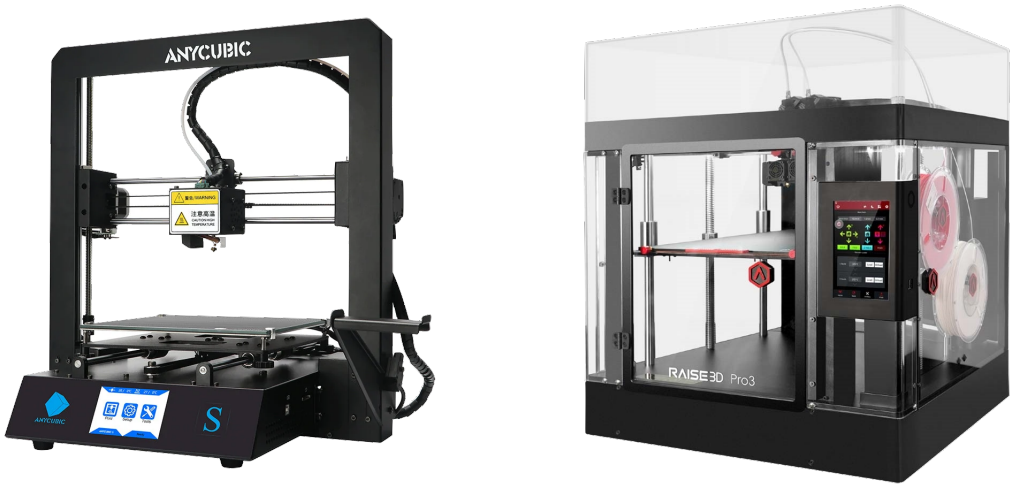
3.4.1 The Anycubic Prototyping Platform and Hardware Upgrades

As previously established, distributing the manufacturing workload to a personal home-based Anycubic FDM printer (figure: 3.3a) was a strategic necessity to circumvent the scheduling constraints of shared university equipment. Operating dual manufacturing environments enabled parallel production, drastically reducing overall machine downtime. Because the home-based printer could be accessed at any hour of the day, it was pushed to operate almost continuously, maintaining an aggressive, non-stop output pace.

However, the consumer-grade Anycubic platform was inherently inferior to the industrial Raise3D Pro3 (figure: 3.3b) utilized in the laboratory. It lacked the mechanical rigidity, thermal consistency and positional accuracy required for the

production of precision robotic components. To improve the print quality and maintain a high production rate, a series of hardware and firmware upgrades was implemented on the Anycubic platform:

- **Chassis Stiffening:** Custom structural reinforcements and rigid tensioning brackets were added to stiffen the gantry frame, attenuating mechanical wobble during rapid axis changes.
- **Stepper Driver Optimization:** The stock stepper motor drivers were upgraded and tuned to deliver higher electrical current. This allowed the motion system to reliably sustain much higher kinematic accelerations and linear speeds without dropping steps or losing spatial coordinates.
- **Thermal and Cooling Enhancements:** The hot-end assembly was completely redesigned to incorporate a more powerful and directed part-cooling airflow. This consistent cooling was critical for rapidly freezing the PETG during complex bridging maneuvers.
- **Firmware Update:** The printer's mainboard firmware was upgraded to integrated advanced PID (Proportional-Integral-Derivative) thermal control for both the hotend and the heated bed to ensure temperature stability and implemented an Auto Bed Leveling (ABL) routine. The ABL system dynamically maps microscopic topological deviations on the build plate, mathematically guaranteeing maximum first-layer adhesion for every continuous print cycle.



(a) Anycubic MegaS[30] **(b)** Raise3D Pro3[31]

Figure 3.3: FDM 3D printers employed for structural PETG fabrication.

3.4.2 Extrusion Tribology and Nozzle Replacement

During the manufacturing phase, after successfully printing approximately 1 kg of PETG components, diagnostic inspection of the Anycubic's hotend revealed that the standard brass extrusion nozzle had suffered extensive geometric degradation. The continuous extrusion of thermoplastic polymers, even non-abrasive filaments like pure PETG, exerts subtle but relentless tribological wear on the internal bore of a brass nozzle. Over hundreds of hours of operation, this friction gradually erodes the inner diameter, widening the 0.4 mm orifice into an unpredictable, asymmetrical oval, while simultaneously flattening the exterior tip.

Operating with a worn nozzle severely degrades the control of the extrusion kinematics. It introduces severe stringing, unpredictable line widths and massive variations in volumetric flow rate, making it impossible to achieve the exact 1.2 mm wall thicknesses required for the robotic linkages. To rectify this failure point and restore absolute mathematical precision to the extrusion path, the worn unit was completely discarded and replaced with a pristine, factory-calibrated 0.4 mm brass nozzle. This maintenance action restored the laminar flow of the melted polymer, ensuring that the slicer's volumetric calculations translated flawlessly into physical plastic.

3.4.3 Empirical Calibration Matrix

At the onset of the manufacturing phase, with the hardware stabilized, a comprehensive suite of empirical calibration tests was executed on the Anycubic platform. The primary objective of this rigorous testing matrix was to establish the absolute peak of the printer's mechatronic capability, specifically targeting the maximum stable printing speeds that could be achieved without sacrificing structural quality or strict dimensional tolerances. Given the large cumulative manufacturing workload required for the multi-kilogram manipulator, optimizing print speeds was a critical necessity to maintain project feasibility.

- **Temperature and Fan Bridging Towers:** Sequential towers were printed, dynamically varying the hotend temperature and part-cooling fan speeds layer-by-layer (figure: 3.4a). This identified the precise thermal threshold where layer adhesion was maximized without inducing excessive polymer oozing and established the optimal fan speed necessary to freeze PETG mid-air during unsupported bridging maneuvers.
- **Speed Test Matrices:** Feed rates were systematically increased to determine the maximum volumetric flow rate the hotend could sustain before the extruder stepper motor began skipping steps due to backpressure, establishing a safe ceiling for rapid infill printing.

- **Bed Flatness and Mesh Leveling:** A high-resolution digital mesh-bed leveling sequence was executed to map the microscopic topological deviations of the heated build plate. The firmware utilizes this mesh to dynamically adjust the Z-axis height in real-time, guaranteeing a geometrically perfect, highly adhesive first layer.

- **Dimensional Tolerance Gauges:** Benchmarking geometries and highly specific clearance gauges were printed and measured with digital calipers. This calibrated the exact horizontal expansion and shrinkage rates of the PETG as it cooled. Achieving absolute dimensional accuracy (within 0.05 mm) was strictly necessary to ensure that the steel bearings, custom rods and servo splines could be smoothly press-fit into the printed arm linkages without requiring destructive post-processing.



(a) Temperature Bridge **(b)** Speed Tower

Figure 3.4: Examples of empirical FDM calibration techniques. Models retrieved from *Printables Open-Source 3D Model Repository.*

Through the iterative optimization of these parameters, the printing process achieved an average speed increase of approximately 20% compared to the default slicer configuration, without compromising structural quality. Ultimately, this exhaustive calibration effort reduced the total printing time by approximately 100 hours, drastically accelerating the overall development cycle and justifying the initial time investment.

3.4.4 Extrusion Thermodynamics and Viscoelastic Control

Based on the calibration data obtained, a dedicated production profile was developed specifically for the Verbatim PETG. The printing process employed a staged thermal and kinematic strategy. During the first six layers, the hotend was maintained at 225°C, the heated bed at 85°C and the print speed limited to 20 mm/s to maximize adhesion.

Once the foundational layers were established, the hotend temperature was increased to 235°C to sustain the higher volumetric flow rates associated with elevated print speeds. Simultaneously, the heated bed temperature was reduced to 80°C to prevent localized softening of the lower layers. Speeds were differentiated according to functional importance: external walls were deposited at 30 mm/s to preserve accurate tolerances, while internal sections were accelerated up to 60 mm/s to reduce total manufacturing time.

Table 3.4: Recommended 3D printing parameters for manufacturing the manipulator components.

Printing Parameters:	
Nozzle Diameter	0.4 mm
Layer Height	0.2 mm
Wall Thickness (Perimeters)	1.2 mm (3 Concentric Shells)
Infill Architecture	Gyroid TPMS
Infill Density	Variable: from 30% to 80%
Hotend Extrusion Temperature	235°C (initial layers 225°C)
Heated Bed Temperature	80°C (initial layers 85°C)
Linear Speeds	Infill speed: 60 mm/s Wall speed: 30 mm/s Travel speed: 120 mm/s
Part Cooling Fan Speed	Adaptive: from 20% to 80%
Extruder Retraction Distance	6.0 mm @ 40 mm/s

3.5 Advanced FDM Slicing Algorithms and Toolpath Optimization

3.5.1 Triply Periodic Minimal Surfaces (TPMS): The Gyroid Architecture

The internal topology of a 3D-printed component dictates its ultimate mass, flexibility and resistance to complex stress vectors. Traditional FDM slicer settings default to rectilinear grids or hexagonal honeycomb infill patterns. While these geometries provide exceptional compressive strength along the vertical Z-axis, they are highly anisotropic; they offer minimal shear resistance and structural stability against lateral, multi-directional forces. Because a robotic arm is a dynamic cantilevered system operating in a three-dimensional workspace, the internal stresses propagating through its linkages (bending moments, torsional strain, sheer off-axis loads) are constantly shifting orientation. To combat these multi-axial stress vectors, the Gyroid infill pattern was explicitly selected for all structural components of the manipulator. The Gyroid is a Triply Periodic Minimal Surface (TPMS) mathematically approximated by the implicit trigonometric equation:

$$\sin x \cos y + \sin y \cos z + \sin z \cos x = 0 \tag{3.1}$$

This generates a continuous, interconnected lattice of infinitely curving surfaces. From a mechanical engineering perspective, the Gyroid architecture is immensely superior for robotics because it provides near-perfect isotropic strength[2], meaning the printed part exhibits uniform resistance to mechanical deformation regardless of the angle of the applied force.

Furthermore, unlike rectilinear grid infills that possess discrete, overlapping crossing points on the same layer, the Gyroid toolpath is entirely continuous and never intersects itself on a single 2D plane. This avoids the microscopic "speed bumps" of deposited plastic that cause the brass nozzle to violently collide and chatter as it moves. By eliminating these internal nozzle collisions, the Gyroid pattern allows the printer to sustain much higher kinematic velocities (60 mm/s) without injecting destructive mechanical vibrations into the printer frame.

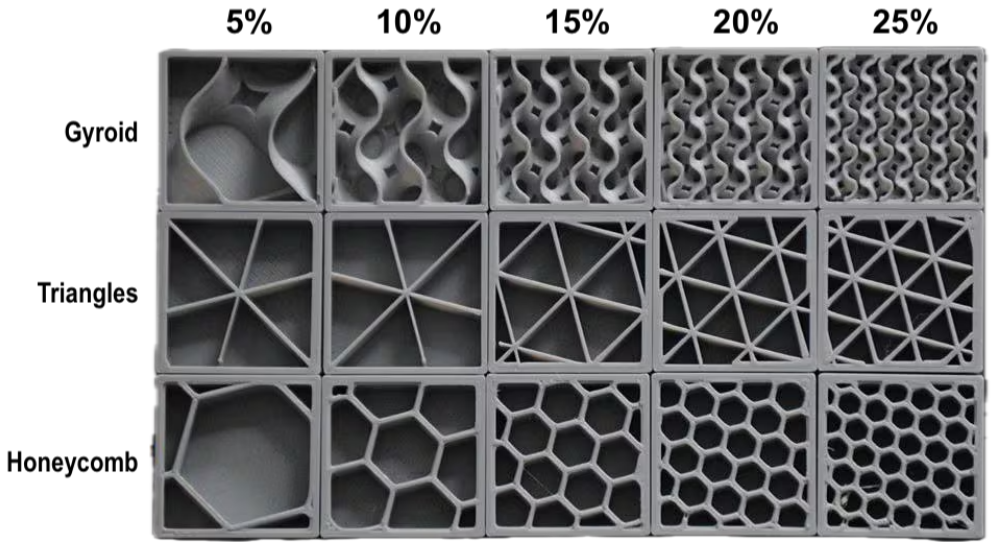


Figure 3.5: Comparison of FDM infill patterns and density variations. Adapted from *Material Guide: Polyethylene Terephthalate Glycol (PETG) and Moisture Dehydration Protocols*.

The infill density was highly parameterized, scaling dynamically between 30% and 80% depending on the specific stress profile of each individual component. For the distal forearm linkages, where minimizing mass to reduce rotational inertia is critical, a 30% Gyroid density provided an optimal balance of rigidity and lightweighting. Conversely, for the primary shoulder brackets and the central motor block—which must anchor the extreme continuous torque of the primary load-bearing actuators—the Gyroid density was increased to 80%. At 80%, the complex curves pack so densely that they simulate near-solid isotropic polymer behavior, resulting in an exceptionally high modulus of elasticity that ensures zero parasitic flex during maximum actuator exertion.

3.5.2 Anisotropic Load Bearing and Spatial Orientation

Unlike isotropic materials such as injection-molded plastics or milled aluminum, parts fabricated via FDM are fundamentally anisotropic; their mechanical properties are highly dependent on the direction of the extruded layers. Tensile strength in the XY plane (parallel to the continuous deposited strands) is typically 4 to 5 times greater than in the Z-axis[2] (perpendicular to the layers). The Z-axis represents the inherent weak point of any FDM print, relying entirely on the secondary thermal fusion of stacked layers rather than unbroken covalent polymer bonds.

If a robotic linkage is subjected to mechanical tension or bending forces that pull perpendicular to the layer lines, the part will rapidly succumb to interlayer cleavage fracture and split in half (figure: 3.6). Therefore, the spatial orientation of every individual component on the virtual build plate was meticulously engineered prior to slicing. Before initiating the manufacturing process, the expected force tensors acting on each linkage during dynamic operation were intuitively computed. Every major load-bearing arm, servo bracket and tensioning mount was manually rotated in 3D space to ensure that the primary axes of expected mechanical stress were aligned strictly parallel to the XY printing plane.

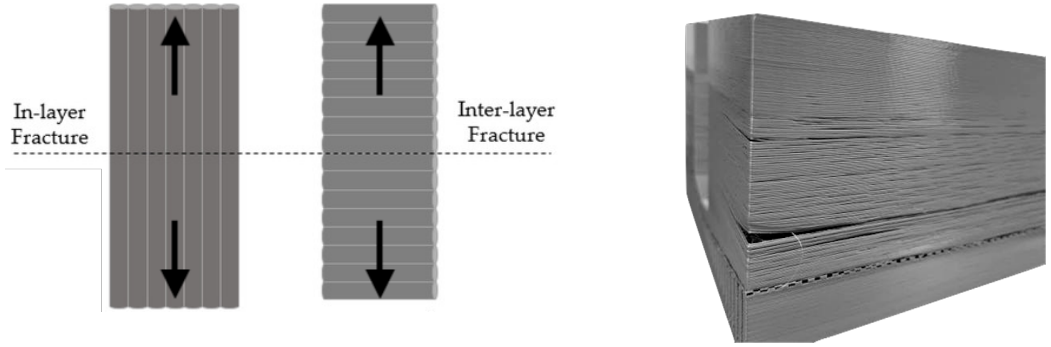


Figure 3.6: Example of inter-layer delamination and mechanical failure under tensile stress. Adapted from *Additive Manufacturing Technologies*.

Because structural integrity superseded material economy and print time, components were frequently oriented at highly unconventional angles on the build plate. This deliberate misalignment often necessitated the generation of significantly more sacrificial support structures than the absolute geometric minimum, simply to guarantee that the layer adhesion vectors were optimized against the physical loads. Furthermore, in instances where a complex component was subjected to severe, multi-directional stress vectors that a single monolithic FDM print could not satisfy, the part was purposefully divided into multiple discrete sub-components. These sub-components were then individually printed in their respective optimal orientations and subsequently assembled using mechanical fasteners, thereby guaranteeing absolute strength across multiple stress axes and greatly reducing the probability of catastrophic shear failure during heavy lifting operations.

3.5.3 Kinematic Tuning and Resonance Mitigation

Despite the structural reinforcements applied to the Anycubic printer, the rapid kinetic movements required to print the complex Gyroid infill at 60 mm/s introduced dynamic artifacts into the prints, a phenomenon commonly referred to as "ringing"

or "ghosting". Ghosting manifests visually as faint, repeating ripple patterns on the surface of the printed part, trailing behind sharp corners or sudden changes in toolpath trajectory. The underlying physics of this defect are governed by the conservation of momentum and mechanical resonance. When the printer's firmware commands a high-acceleration deceleration, the physical mass of the extruder behaves as a damped harmonic oscillator, physically vibrating at the machine's natural resonant frequency and permanently casting these waves into the solidifying PETG. In a precision robotic manipulator, these are highly destructive geometric deviations that corrupt strict dimensional tolerances.

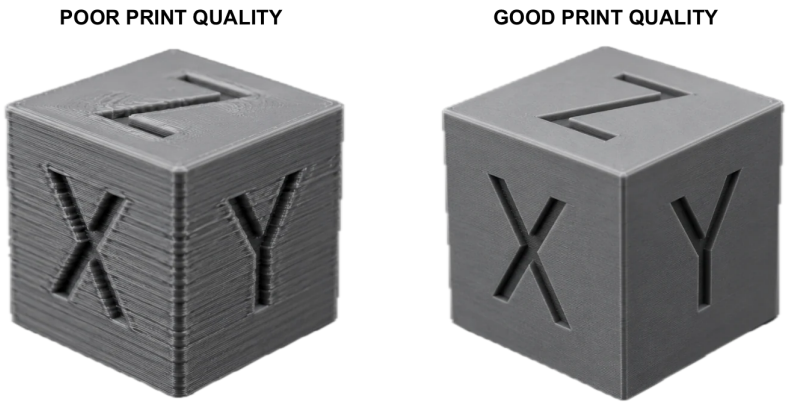


Figure 3.7: Effect of printer structural resonance on print quality. Adapted from *Material Guide: Polyethylene Terephthalate Glycol (PETG) and Moisture Dehydration Protocols.*

To mitigate these ringing artifacts without resorting to reductions in print speed, a targeted kinematic tuning strategy was implemented. First, the baseline acceleration and jerk parameters were intelligently reduced to smooth out the trapezoidal motion planner, preventing the stepper motors from injecting high-energy impulse shocks into the frame. Second, the custom mainboard firmware was optimized to better manage these adjusted dynamic limits. This combination effectively attenuated the machine's natural resonances, guaranteeing that the robotic arm's PETG linkages were fabricated with absolute geometric fidelity and smooth dimensional interfaces, despite operating at rapid linear print speeds.

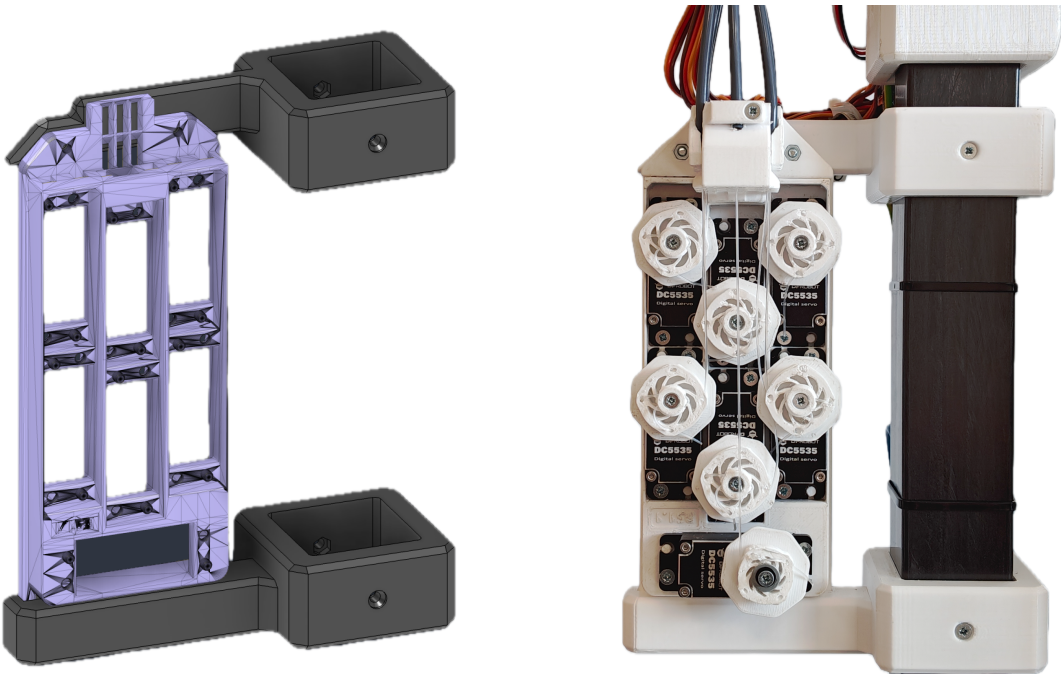
3.6 Structural Foundation: Custom Anchoring Architecture

3.6.1 Custom Mount Development for Single-Arm Integration

To maintain forward compatibility (i.e., ensuring that the manipulator can easily be integrated into a full-body assembly in the future without requiring the redesign of proximal linkages) the original PIB base geometry was deliberately retained. The manipulator is externally supported by a repurposed 40x40 mm square carbon fiber tube available in the laboratory.

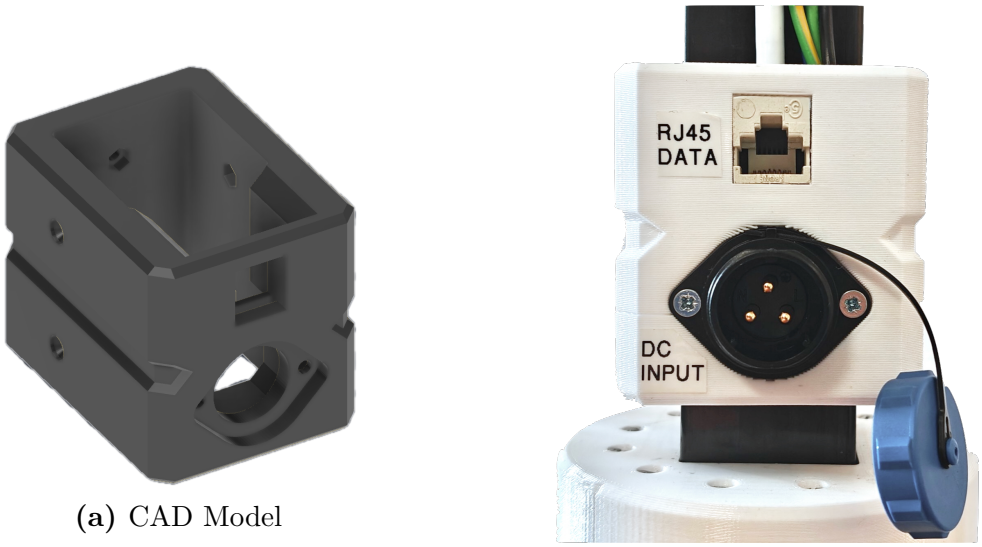
Because extruded carbon fiber tubing often exhibits slight dimensional variance and irregular cross-sectional profiles across its length, standard fixed-dimension rigid mounting brackets were avoided; rigidly toleranced 3D-printed attachments simply would not fit reliably over the varying profile of the tube. Instead, custom attachments designed to hold the servo bank (figure: 3.8), power supply and signal cable connectors (figure: 3.9) were modeled in CAD and 3D printed in White Verbatim PETG.

These attachments feature a square internal bore designed to be slightly larger than the pole's nominal dimensions, allowing them to slide freely along the length of the tube during the initial setup. To secure them, the design utilizes a friction-fit clamping mechanism. M3 metric nuts are embedded into the interior geometry of the attachments. When M3 screws are driven in from the outside, they engage these internal nuts and protrude through the plastic so that the tips of the screws dig directly into the carbon pole. This allows the brackets to be firmly clamped in place exactly where needed, accommodating the dimensional irregularities of the tube without relying on tight surface-to-surface tolerances.



(a) CAD Model **(b)** Physical Implementation

Figure 3.8: Centralized servo bank mount for distal tendon actuation.



(a) CAD Model **(b)** Physical Implementation

Figure 3.9: SP21 power supply and RJ45 signal cable connector.

3.6.2 Anisotropic Load Mitigation in the Primary Support

A complex structural challenge during the foundation's development was the engineering of the main load-bearing arm support (figure: 3.10). The design process began by importing the baseline PIB STL mesh into Autodesk Fusion 360, where the geometry was modified to guarantee spatial compatibility and maximum volumetric efficiency, ensuring the mount did not physically interfere with the arm's required operational workspace.



(a) CAD Model **(b)** Physical Implementation

Figure 3.10: Custom load-bearing shoulder support bracket.

Because the robotic arm acts as a dynamic, multi-link cantilevered mass, it does not exert force in a single, predictable vector. The inertia generated by rapid kinematic movements—particularly during high-acceleration trajectory reversals—creates highly variable, multi-directional force tensors concentrated directly at the shoulder joint. In standard FDM additive manufacturing, components are highly anisotropic, meaning their structural limits are fundamentally dictated by the orientation of the Z-axis layer lines.

To prevent catastrophic shear failure under these complex dynamic loads, this critical mount was purposefully printed in an unconventional spatial orientation on the build plate. This deliberate misalignment maximized the contact area between the Z-axis layers, strategically aligning the polymer's strongest isotropic vectors (the continuous extruded filament paths) directly against the most severe calculated bending moments. By optimizing the print trajectory to oppose the primary inertial vectors, the probability of interlayer delamination was significantly reduced, yielding a support structure capable of anchoring the high-torque actuators safely.

3.6.3 Didactic Advantages of the Open-Architecture Configuration

One of the most significant advantages of omitting the full torso enclosure is the complete exposure of the actuator and electronic subsystems that, in the original PIB architecture, would normally remain hidden inside the torso shell. In particular, the centralized seven-servo actuator bank (figure: 3.8b) responsible for hand and wrist actuation through Bowden cable transmission remains fully accessible and visible during operation.

This open-architecture configuration provides substantial didactic and experimental benefits. By exposing the complete tendon-routing pathway, Bowden cable guides, mechanisms and actuator assemblies, the system allows direct observation of the kinematic behavior of the hand during grasping and manipulation tasks. Students can visually analyze cable tension variations, nonlinear friction phenomena and dynamic interactions between the servomotors and the tendon-driven finger mechanisms in real time.

The exposed configuration also greatly simplifies maintenance, calibration and iterative development. Manual adjustment of tendon pretension and cable routing can be performed without disassembling external covers or structural panels. Similarly, the accessibility of the electronics and power distribution system facilitates rapid debugging and future integration of additional sensing hardware such as current monitoring circuits or force-feedback instrumentation.

From an educational perspective, the absence of a closed torso shell transforms the manipulator into a highly transparent mechatronic platform, where the relationship between actuation, transmission mechanics and robotic motion can be directly observed and studied. This significantly enhances the platform's value for laboratory demonstrations, control experimentation and advanced robotics research.

3.7 Physical Assembly and Mechatronic Commissioning

The physical realization of the custom manipulator marks the critical transition from theoretical kinematic models and digital CAD environments into the physical mechatronic domain. Comprising approximately 100 individual FDM-manufactured components for the left arm alone, the assembly was executed over several non-consecutive weeks. This extended timeline was dictated by the inherent realities of rapid prototyping: iterative tolerance adjustments, the inevitable breakage of fragile linkages during the press-fitting of steel bearings and the continuous on-demand 3D printing of revised components to accommodate the upgraded servomotors.

3.7.1 Inventory Management and Mechanical Post-Processing

To manage the logistical complexity of the build, a strict inventory protocol was established before any physical integration commenced. Every printed component was cataloged and physically tagged with an alphanumeric code using masking tape, cross-referenced against the master Bill of Materials (BOM). This granular tracking was absolutely critical to prevent catastrophic assembly errors, particularly given the high volume of visually similar, geometrically mirrored linkages.

Extensive mechanical post-processing was mandatory for nearly every printed part to bridge the gap between FDM extrusion artifacts and the strict dimensional tolerances required for smooth kinematic motion. Components underwent rigorous manual sanding, the deburring of internal bores to safely accommodate rotating shafts and the precise removal of sacrificial support structures. In instances where large, time-intensive prints suffered from minor localized layer delamination, the parts were structurally recovered using cyanoacrylate adhesives and thermal welding rather than being discarded, maximizing production yield and minimizing material waste.

Furthermore, because standard FDM cannot produce reliable internal threads capable of withstanding dynamic loads, M3 nuts were embedded directly into almost every large structural component. This was achieved through localized thermal yielding: a soldering iron was utilized to briefly heat the metal nut, softening the surrounding PETG matrix and allowing the nut to permanently fuse into the polymer as it cooled, creating a high-strength captive thread capable of anchoring the high-torque actuators.

3.7.2 Bottom-Up Subassembly and Fastener Economy

The integration process followed a strict bottom-up methodology, beginning with the highly intricate distal phalanges of the hand, progressing upwards through the metacarpals and elbow and culminating at the heavy-duty shoulder joints. Each major joint mechanism was constructed as an isolated, independent subassembly. Critical rotating components, specifically the primary steel ball bearings, were thoroughly packed with lubricating grease to minimize dynamic friction prior to being press-fit into their polymer housings. Once the discrete subassemblies were mechanically validated in isolation, they were mated together via structural interconnects.

A pragmatic, cost-driven compromise was necessary regarding the hardware fasteners. In professional mechatronics, M3 hex-head cap screws are the industry standard due to their superior torque-transfer capabilities and resistance to cam-out under high-vibration loads. However, procuring them in the required quantities strictly through the designated supplier (RS Components) would have violated the project's rigid financial constraints. Consequently, the assembly utilized a heterogeneous sorted mix of M3 countersunk screws and nuts, alternating between Pozidriv, Phillips and flathead drives based entirely on localized availability. While this aesthetic deviation diverges from industrial norms, it proved entirely functionally viable, ensuring structural rigidity while successfully preserving the project budget.

3.7.3 Empirical Actuator Calibration and Kinematic Bounding

A critical mechatronic challenge in commissioning the manipulator involved the mechanical zeroing (homing) and kinematic bounding of the digital actuators. Unlike standard analog servos that possess physical internal hard-stops to define their absolute limits, the high-torque digital servos utilized in this project rotate continuously when unpowered, making physical dead-reckoning of their absolute position impossible.

To resolve this, the internal gear mechanisms, designed to function as friction-clamped slip-rings, were left deliberately loose during the initial physical build. Once the entire robotic arm was fully assembled and mounted to its carbon fiber support, but strictly prior to being wired to the final PCA9685 PWM drivers, an independent laboratory function generator was utilized to manually drive the control signals.

By injecting raw PWM signals directly into each isolated actuator, the physical range of motion (ROM) for every joint was empirically mapped. The external brackets were manually rotated and friction-locked only when the actuator's electrical center

aligned with a functional kinematic resting angle for the arm. During this manual sweeping phase, the specific PWM pulse widths corresponding to the arm's home position (which fundamentally differed from the servo's internal absolute zero), as well as the absolute mechanical minimum and maximum limits, were precisely recorded. This empirical mapping was essential to establish hard-coded physical constraints, preventing the manipulator from actively driving itself into catastrophic structural collisions once autonomous control was enabled.

Furthermore, this isolated testing phase with the function generator was leveraged to characterize the dynamic electrical response of the servos. Because the system utilizes digital actuators, the operating frequency can far exceed the legacy 50Hz standard associated with analog RC servos. Empirical testing determined that driving the actuators at a 300Hz PWM frequency yielded optimal mechanical responsiveness. Crucially, elevating the base frequency inherently expanded the usable duty cycle window, directly translating into a significantly higher resolution for the positional control signal.

The comprehensive electronic implementation of this 300Hz control scheme, alongside the integration of the empirically derived kinematic limits into the embedded homing software, will be extensively detailed in chapter 4 (Custom Electronics) and chapter 5 (Embedded Software Architecture).

3.7.4 Tribological Tuning and Dynamic Kinematic Instability

The final phase of commissioning involved the routing of the Bowden cables from the torso-mounted servo bank to govern the hand and finger movements. Upon initial kinematic testing, it was discovered that the originally specified pneumatic Teflon tubing (used for the outer Bowden sheath) was excessively compliant. Under servo tension, the soft Teflon tube physically buckled and compressed, acting as a parasitic mechanical spring that absorbed the actuator's linear displacement before any useful force could reach the fingertips. Attempts to mitigate this energy loss by injecting fluid lubricants into the sheath catastrophically exacerbated the issue, trapping particulate debris and increasing static friction (*stiction*).

To resolve this kinematic failure, the entire outer Bowden routing was upgraded to highly rigid Polytetrafluoroethylene (PTFE) tubing, commonly utilized as low-friction filament guides in 3D printer extruders, while retaining the internal nylon tendon. While this successfully restored deterministic force transfer, the fingers remained mechanically stiff. Consequently, the hand was entirely disassembled to replace the original passive return springs with weaker, lower-stiffness variants.

This final tribological tuning successfully restored fluid, full-range articulation to the hand. However, replacing the springs with weaker variants resulted in a highly underdamped kinematic behavior. This behavior manifests as visible structural flutter or "wobbling", in the fingers whenever the primary arm executes rapid macroscopic movements or high-acceleration trajectory reversals. The implications of this dynamic instability, alongside proposed control solutions, will be further analyzed in the concluding evaluation of the system.

Chapter 4

Custom Electronic Architecture and Power Distribution

The mechanical modifications and actuator upgrades described in the previous chapter established the physical basis for the robotic manipulator, but also introduced new requirements for its electronic control and power infrastructure. The original PIB architecture[16] relies on a commercial hardware ecosystem that abstracts most of the low-level functions required to operate the robot. This approach is convenient for a standard configuration, but it provides limited access to the underlying signal generation, actuator control and power distribution and becomes less suitable when the original hardware configuration is modified.

The custom manipulator developed for this project uses a single arm with heterogeneous digital servomotors operating at different voltage levels, current requirements and angular ranges. These changes required the original electronic architecture to be replaced with a dedicated embedded system capable of providing direct control of the actuators while safely managing their power requirements. Particular attention was therefore given to actuator calibration, PWM generation, high-current distribution, power supply design, electrical protection and the physical separation of power and communication interfaces.

The resulting architecture is based on an STM32 microcontroller[24], a PCA9685 PWM driver[25], dedicated power supplies for the different actuator groups and a modular interconnection system between the robotic arm and the external power

enclosure. A hardware Human-Machine Interface (HMI) was also developed to provide direct manual control of the manipulator and to support the hybrid control strategy described in the following chapter.

This chapter presents the design and implementation of this custom electronic architecture. It first discusses the limitations of the baseline electronic system and the requirements that motivated its replacement. The actuator calibration procedure and PWM frequency optimization are then presented, followed by the design of the high-current PWM distribution stage and the dedicated power and safety infrastructure. The modular power and data interfaces are subsequently described, together with the hardware HMI developed for direct manual operation.

4.1 Limitations of the Baseline Electronic Architecture

The baseline PIB project relies primarily on the Tinkerforge hardware ecosystem for low-level actuator control, together with Luxonis sensing[16, 20]. This commercial architecture provides a convenient plug-and-play solution and abstracts many of the complexities associated with embedded system development. The Tinkerforge platform, for example, provides dedicated master and servo driver modules that interface with the higher-level control software and generate the signals required by the actuators. However, this level of abstraction becomes limiting when the system is modified beyond its original configuration.

From an economic perspective, the use of the proprietary Tinkerforge ecosystem was not compatible with the cost constraints of the project. The use of dedicated master bricks and servo drivers for the complete robotic platform would have represented a substantial portion of the available budget. More importantly, from an educational and mechatronic perspective, the commercial architecture hides several low-level aspects of the control system. Signal generation, hardware timer operation, interrupt timing and part of the actuator control process are handled internally by the commercial hardware and its associated software interface. This prevents direct access to the low-level mechanisms responsible for generating and controlling the actuator signals.

For an educational robotic platform, direct access to these functions is particularly relevant. The objective was not only to operate the manipulator, but also to provide direct control and observation of the electronic processes involved in its operation. In particular, the custom architecture required the possibility of controlling the PWM generation, modifying its operating parameters and directly managing the microcontroller hardware responsible for the timing of the control signals. Relying

entirely on a closed commercial driver would therefore limit the possibility of experimentally investigating and optimizing these aspects. The decision to replace the original electronic infrastructure was consequently driven not only by cost, but also by the need for a more transparent and accessible control architecture.

A second limitation arose from the difference between the original PIB configuration and the manipulator developed in this project. The baseline control architecture was designed for the complete humanoid robot and its associated bilateral kinematic structure. The software and control system were therefore developed around the assumption that the complete robot, including both arms and the remaining body subsystems, would be present. In the present configuration, only a single left arm is implemented. This raised uncertainties regarding the compatibility of the original high-level software with an isolated arm, particularly in the absence of the telemetry and communication exchanges associated with the missing torso and contralateral limb. Rather than adapting the original architecture around these uncertainties, the electronic and control system was redesigned specifically for the single-arm configuration.

Furthermore, the mechanical and electromechanical modifications introduced in chapter 3 significantly changed the requirements imposed on the actuator control system. The original PIB architecture uses a set of commercially available servomotors with different roles within the manipulator, including the MG996R[17], DS3225[18] and DS5180[19] models. In the modified configuration, these actuators were replaced or supplemented by two main actuator types: the DFRobot DC5535[22] and the DSSERVO DS51150[21]. The resulting system is therefore characterized by heterogeneous actuators with different operating voltages, current requirements, angular ranges and torque capabilities.

The DS51150 servomotors are used for the primary load-bearing joints and require substantially higher electrical power than the smaller actuators. They also provide an operating range of up to 270°, compared with the approximately 180° range associated with the standard servomotors used elsewhere in the system. The DC5535 actuators, in contrast, are used for the tendon-driven hand and wrist mechanisms and operate at a lower voltage and current level. These differences make it unsuitable to treat all actuators as identical loads or to rely directly on the original low-current servo driver configuration.

In particular, the original Tinkerforge servo drivers[20] were intended to control conventional hobby servomotors and provide a high-level interface for their operation. The modified manipulator instead requires direct management of actuators with substantially different electrical and kinematic characteristics. Attempting to retain the original control system would therefore require the software and hardware

abstraction layers to accommodate different actuator ranges and power requirements while also ensuring that the increased current demand of the DS51150 servos did not exceed the capabilities of the original distribution hardware.

The combination of these factors led to the replacement of the baseline electronic architecture. The limitations of the commercial ecosystem, the uncertainty associated with controlling an isolated single arm and the heterogeneous actuator configuration all motivated the development of a custom embedded system. This architecture provides direct control over the actuator signals and their operating limits while allowing the power distribution and electrical protection systems to be designed specifically around the requirements of the modified manipulator.

4.2 Custom Embedded Control Architecture

The limitations of the baseline electronic architecture led to the adoption of a custom embedded control system. This approach addressed both the technical incompatibilities identified in the previous section and the need to maintain the project within the available budget. At the same time, replacing the proprietary commercial hardware with a custom solution provided direct control over the signal generation, actuator limits, power distribution and low-level control functions of the manipulator.

The resulting electronic architecture, described throughout this chapter, is organized around five main engineering areas:

1. **Empirical Actuator Calibration and Signal Optimization:** Before integrating the central digital control logic, the unpowered digital actuators required rigorous hardware-in-the-loop calibration. This phase empirically established safe physical kinematic boundaries and experimentally elevated the base PWM driving frequency to 300 Hz, effectively unlocking a 600% increase in the manipulator's positional resolution.

2. **High-Current PWM Distribution:** To bridge the gap between low-voltage digital control and high-current actuator demands without fabricating a costly custom Printed Circuit Board (PCB), a strategic hardware modification was performed on an I2C PWM driver. This transformed a standard PCA9685 logic board[25] into a robust, split-rail (7.5V and 12V) copper-busbar distribution node mounted directly on the arm's chassis to drastically minimize electromagnetic interference (EMI).

3. **Industrial Power Supply and Safety Infrastructure:** The high-torque digital servos demand significant transient currents that far exceed typical hobbyist thresholds. To satisfy these demands safely, a dedicated power supply

infrastructure was engineered. This integrates dual high-capacity Switch Mode Power Supplies (SMPS)[26] housed within an IP56-rated protective enclosure, reinforced by a redundant, multi-tiered AC and DC fusing strategy.

4. **Modular Interconnect and Dual-Redundant Telemetry:** To mechanically decouple the dynamic robotic arm from the static power infrastructure, a highly modular interconnect system was developed. This utilizes keyed 30A SP21 aviation connectors[27] with heavy-gauge 2.5 mm^2 cabling for thermal mitigation, operating in parallel with a dual-redundancy RJ45 Ethernet link to ensure fail-safe I2C data transmission.

5. **Analog Control Interface (Hardware HMI):** To facilitate direct didactic observation and manual kinematic testing without relying on an external PC, the system incorporates a custom hardware-level control panel. Comprising an array of strategically mapped linear potentiometers interfacing directly with the embedded STM32 microcontroller, this HMI establishes the hardware foundation for the hybrid (manual/autonomous) software control architecture that will be comprehensively explored in chapter 5.

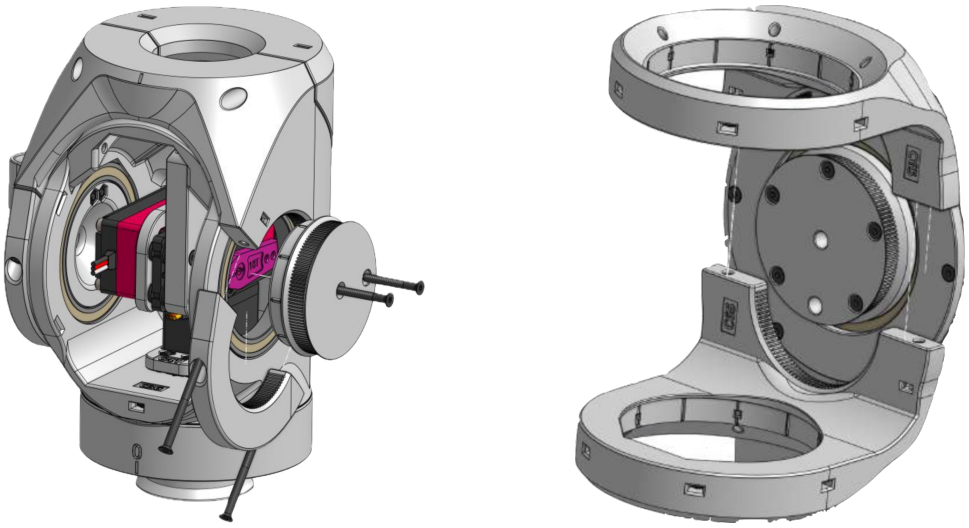
4.3 Empirical Actuator Calibration and Signal Frequency Optimization

Before finalizing the integration of the central microcontroller and power distribution hardware, it was imperative to establish a rigorous electro-mechanical baseline for the robotic arm. As introduced in chapter 3, the custom manipulator utilizes highly geared digital servomotors. A fundamental mechatronic challenge associated with these specific COTS digital actuators is that they lack any external alignment marks or physical internal hard-stops to indicate their absolute zero position when unpowered. Consequently, a dedicated hardware-in-the-loop calibration phase was executed utilizing an independent laboratory function generator. This methodology allowed for the precise mapping of electrical PWM signals to the physical kinematic boundaries of the manipulator prior to the implementation of any embedded control software.

4.3.1 The Necessity of Open-Loop Hardware Calibration

During the physical assembly phase, the internal gear mechanisms, functioning as friction-clamped slip-rings as portrayed in figure 4.1, were intentionally left loose. If the mechanical linkages were rigidly clamped to the servo splines while unpowered, there would be no guarantee that the actuator's internal electrical range of motion would align safely with the joint's physical workspace. A misalignment

would result in the servo attempting to drive past inherent structural constraints (such as localized collisions between the bicep and forearm or the snapping of over-tensioned Bowden cables), leading to immediate stall currents and potential structural fracture.



(a) Elbow Joint **(b)** Shoulder Joint

Figure 4.1: Exploded view of elbow and shoulder slip-ring assemblies. Adapted from *PIB – Printable Intelligent Bot: Open-Source 3D-Printable Humanoid Robot.*

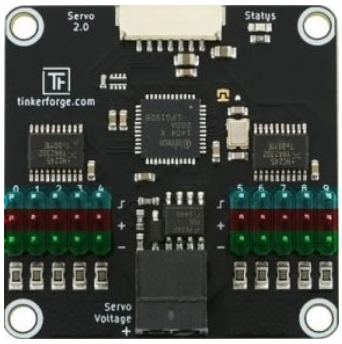
To systematically resolve this, a laboratory signal generator was utilized to inject a highly controlled, synthetic square wave directly into each isolated actuator. Rather than attempting to align the theoretical electrical center of the servo with the center of the joint, the calibration focused on boundary alignment. By generating a specific boundary PWM pulse (e.g., the absolute minimum pulse width), the actuator was driven to one of its extreme electrical limits. While the actuator actively held this powered state, exerting maximum holding torque, the physical 3D-printed joint was manually rotated to its corresponding safe mechanical boundary.

The external friction collars were then firmly tightened. This empirical calibration guaranteed that the actuator possessed an adequate, precisely mapped margin of rotation before encountering any structural interference. Crucially, as a result of this boundary-matching methodology, the functional "home" position of the robotic arm does not necessarily coincide with the mathematical center (or absolute zero) of the actuators. These positional offsets were recorded for later integration into the software architecture.

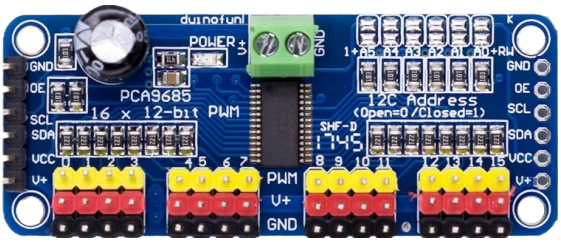
4.3.2 Pulse Width Frequency Analysis and Resolution Enhancement

A critical divergence from the baseline PIB control architecture involved the optimization of the PWM driving frequency to maximize the positional resolution of the arm. The original open-source design relied exclusively on the Tinkerforge Servo Bricklet 2.0[20] (figure: 4.4a) to govern the actuators. While the Tinkerforge Bricklet is a capable commercial module offering a highly abstracted software API that handles internal motion planning and acceleration profiles it fundamentally acts as a computational "black box". It is designed to default to standard legacy RC servo parameters, typically driving the PWM signals at a baseline frequency of 50 Hz, which corresponds to a total signal period of 20.0 milliseconds.

While this abstracted approach is sufficient for basic hobbyist applications, it actively restricts the engineering student's ability to optimize the control loop for high-performance digital actuators. To gain absolute deterministic control over the low-level kinematics, the custom architecture replaced the Tinkerforge ecosystem with an NXP PCA9685[25], a 16-channel, I2C-controlled bare-metal PWM driver featuring a hardware 12-bit timer.



(a) Tinkerforge Servo Bricklet 2.0[20]



(b) NXP PCA9685 Breackout Board[25]

Figure 4.2: Comparison between the baseline Tinkerforge Servo Bricklet 2.0 and the PCA9685 breakout board.

A 12-bit architecture dictates that the controller divides the total period of the PWM signal into $2^{12} = 4096$ discrete digital steps. From a control systems perspective, utilizing a 50 Hz frequency introduces severe quantization limitations. The effective control resolution of the servomotor is defined not merely by the timer's 4096-step limit, but by the functional percentage of the duty cycle window that the standard 0.5 ms to 2.5 ms pulse occupies.

Taking the primary load-bearing actuator, the DSSERVO DS51150[21] (capable of 270° rotation), as the mathematical model, the duty cycle D at 50 Hz is calculated as:

$$D_{\min} = \frac{0.5\,ms}{20\,ms} = 2.5\% \qquad D_{\max} = \frac{2.5\,ms}{20\,ms} = 12.5\% \tag{4.1}$$

At 50 Hz, the entire 270° angular sweep of the actuator is compressed into a remarkably narrow 10.0% window of the total PWM period. Consequently, the controller can only utilize 10% of its available 12-bit resolution ($4096 \cdot 0.10 \approx 410 steps$). This yields a highly constrained angular resolution of $270°/410 \approx 0.66°$ per step, directly resulting in coarse, "jittery" angular movements when attempting to execute slow, precise trajectories.

Because the custom manipulator utilizes modern digital actuators equipped with internal microprocessors, they are not constrained to the legacy 50 Hz refresh rate expected by standard hobbyist controllers. By driving the PCA9685 directly via the STM32, the driving frequency was systematically increased while observing dynamic stability with the signal generator. It was experimentally determined that a 300 Hz control frequency ($T \approx 3.33ms$) offered the optimal balance of mechanical responsiveness and signal integrity. Recalculating the resolution at 300 Hz reveals a massive mechatronic advantage:

$$D_{\min} = \frac{0.5\,ms}{3.33\,ms} = 15\% \qquad D_{\max} = \frac{2.5\,ms}{3.33\,ms} = 75\% \tag{4.2}$$

By elevating the frequency to 300 Hz, the usable duty cycle window expands to 60.0% of the total period. The PCA9685 can now utilize $4096 \cdot 0.60 \approx 2458 steps$ to define the actuator's position. This yields a vastly superior angular resolution of $270°/2458 \approx 0.11°$ per step. This 600% improvement in control definition mathematically validates the abandonment of the Tinkerforge Bricklet, directly facilitating the fluid, high-resolution kinematic trajectories required for an advanced didactic demonstrator.

Following the stabilization of the 300 Hz control frequency, the absolute functional bounds for both actuator types were empirically recorded:

- **DSSERVO DS51150** (270° Primary Joints):
  Freq: 300 Hz Duty Min: 15% (0.5 ms) Duty Max: 75% (2.5 ms)
- **DFRobot DC5535** (180° Tendon Transmission):
  Freq: 300 Hz Duty Min: 9% (0.3 ms) Duty Max: 80% (2.67 ms)

While these values represent the absolute electro-mechanical limits of the bare actuators, the complex geometry of the 3D-printed chassis inherently imposes tighter asymmetrical physical restrictions. Consequently, the specific sub-limits required to safely halt the manipulator before it strikes its own structural boundaries were precisely documented. These empirical sub-limits establish the foundation of the hard-coded safety constraints embedded within the STM32 microcontroller firmware, which will be detailed in chapter 5.

4.4 High-Current PWM Distribution: The Custom PCA9685 Architecture

The transition from the commercial Tinkerforge ecosystem to a bare-metal microcontroller architecture necessitated a robust method for distributing both the high-frequency PWM logic signals and the massive electrical currents required by the actuators. While the generation of the 300 Hz control signal is elegantly handled by the NXP PCA9685 I2C driver, bridging the physical gap between this low-voltage logic IC and the high-power mechanical joints presented a profound mechatronic bottleneck.

4.4.1 The Bottleneck of Commercial Breakout Boards

To accelerate development, the project utilized a commercially available PCA9685 breakout board, which conveniently features 16 output channels with pre-soldered, standard 3-pin (Signal, V+, GND) headers. In a traditional hobbyist configuration, a single power supply is connected to the board's main terminal block and the internal copper traces of the Printed Circuit Board (PCB) distribute this voltage to all 16 servo headers simultaneously.

However, a rigorous analysis of the actuator power demands immediately disqualified this standard topology. The custom manipulator integrates heterogeneous actuators with severe current draws:

- **DSSERVO DS51150:** (3 units)
  stall torque of 173 kg · cm at 12.6 V, drawing a peak stall current of 8.3 A per actuator.
- **DFRobot DC5535:** (9 units)
  stall torque of 35 kg · cm at 7.0 V, drawing a peak stall current of 2.8 A per actuator.

If the manipulator were to experience a localized multi-joint stall, the theoretical transient current could exceed 50 A. Standard commercial breakout boards are engineered for micro-servos and typically feature thin copper traces (usually 1 oz/ft² thickness) designed to carry no more than 2 A to 3 A globally. Furthermore, the onboard filtering capacitors and trace geometries fundamentally restrict the V+ rail to a maximum of 6.0 V.

Attempting to push 12.6 V and tens of amperes through the stock PCB would result in immediate catastrophic failure due to Joule heating $P_{loss} = I^2R$, vaporizing the internal copper traces. The conventional engineering solution to this problem is the fabrication of a large, custom Power Distribution Board (PDB) equipped with heavy-duty MOSFETs and thick copper pours. However, designing and manufacturing a custom PDB would significantly increase the project's cost and physical footprint.

Alternatively, routing all 12 individual servo power and signal cables directly back to the main power supply box was considered and subsequently rejected. Extending unshielded, high-current power lines parallel to 3.3 V logic-level PWM signals over a two-meter umbilical run would induce severe Electromagnetic Interference (EMI), crosstalk and parasitic capacitance, leading to signal degradation and erratic actuator behavior (jittering).

4.4.2 PCB Topography Modification and Split-Rail Busbars

To solve this power distribution crisis while maintaining spatial efficiency and adhering to the strict project budget, a strategic hardware modification was executed directly on the commercial PCA9685 breakout board. The objective was to completely isolate the fragile PCB traces from the high-current demands of the servos, effectively transforming the logic board into a decentralized, high-capacity distribution node.

The physical modification began by surgically severing the factory V+ power supply traces on the PCB that fed the 16 servo headers. By cutting these microscopic traces, the servo power pins were electrically isolated from both the board's logic circuitry and its primary power terminal, instantly bypassing the 6.0 V and low-ampacity limitations of the stock hardware.

To deliver the necessary current directly to the servos, copper wire with a cross-sectional area of 1.5 mm² was manually soldered horizontally across the back of the protruding servo pin headers. These solid copper lines act as heavy-duty busbars, capable of sustaining massive transient currents without generating significant thermal losses.

Crucially, because the manipulator requires two distinct voltage domains to maximize actuator performance 12 V for the primary joints and 7.5 V for the tendon transmission, the busbar modification was spatially divided to create a dual-rail topology:

- **The Low-Voltage Domain (7.5 V):** Channels 0 through 11 were bridged with a dedicated copper busbar and hardwired to the 7.5 V umbilical line. This bank is dedicated to the 9 DFRobot DC5535 actuators governing the hand and wrist. This configuration deliberately leaves 3 spare channels, providing immediate fail-over redundancy if a specific PWM pin is damaged.
- **The High-Voltage Domain (12.0 V):** Channels 12 through 15 were bridged with a separate copper busbar and hardwired directly to the 12 V umbilical line. This bank is explicitly reserved for the 3 heavy-duty DS51150 actuators governing the shoulder and elbow kinematics, leaving 1 spare redundant channel.

This architectural "hack" originated as a rapid-prototyping solution to test the actuators without waiting for custom PCB fabrication. However, exhaustive thermal testing revealed that the 1.5 mm^2 copper busbars performed exceptionally well under sustained dynamic loads. No thermal hot-spots were detected on the modified board itself; instead, the thermal bottleneck shifted to the thin, factory-crimped extension wires of the DS51150 servos, which, while warming under heavy load, safely remained within operational margins. Consequently, this highly efficient, low-cost modification was permanently adopted into the final mechatronic build.

4.4.3 Spatial Integration and EMI Mitigation

The physical placement of this modified PCA9685 distribution node was a critical architectural decision that directly impacted the electro-mechanical stability of the entire system. To mitigate the severe risks of Electromagnetic Interference (EMI) and signal degradation associated with routing power and logic over long wire runs, the modified board operates as a completely decentralized node. It is mounted directly onto the robot's supporting vertical carbon fiber pole, situated immediately behind the primary shoulder joint as shown in figure 4.3.

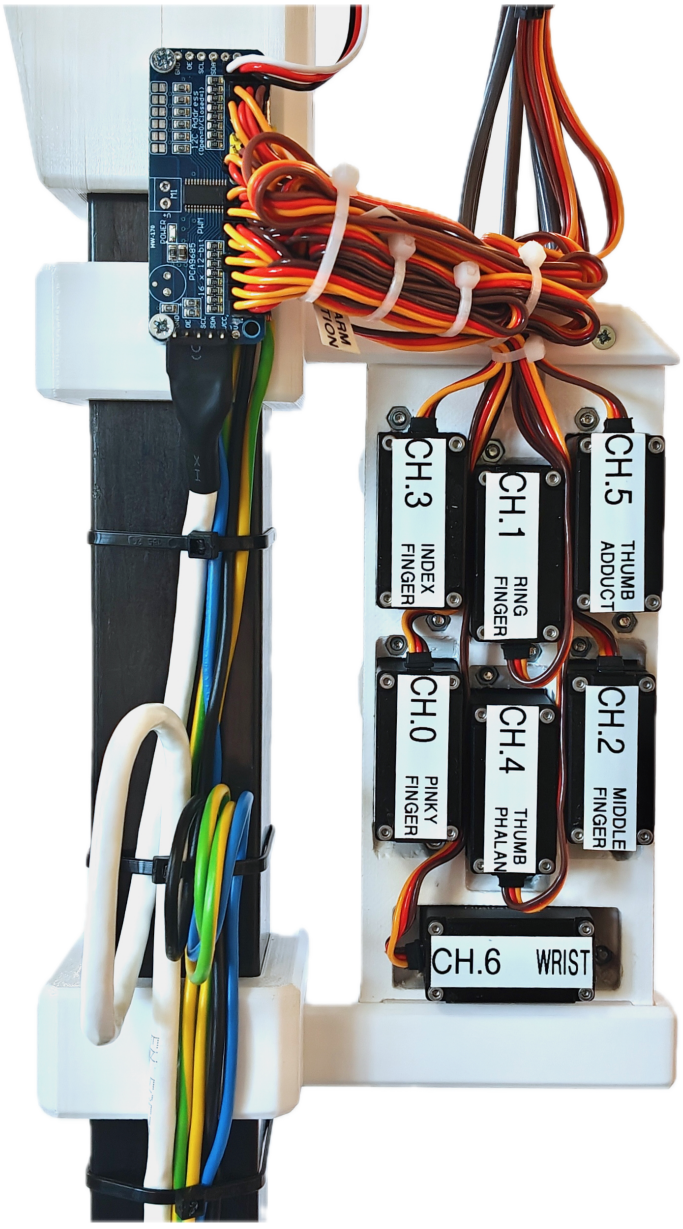


Figure 4.3: Mounting location of the PCA9685 node on the carbon-fiber mast, situated behind the shoulder joint and to the rear of the distal actuation servo bank.

This strategic, localized placement is fundamental to preserving signal integrity. The positional control of the digital servos relies on high-frequency (300 Hz) digital square waves. If these low-voltage logic signals were routed from a centralized box through a two-meter unshielded umbilical alongside high-current DC power lines, the parallel wire runs would act as antennas. The resulting parasitic capacitance

and inductive crosstalk would round off the sharp rising and falling edges of the PWM square waves, inducing severe timing jitters within the servos' internal microcontrollers. By positioning the PCA9685 node directly at the shoulder in close proximity to the dense hand and forearm actuator banks, the PWM signal wires are kept as short as physically possible, guaranteeing high-resolution kinematic response.

This specific spatial topology aligns perfectly with the overarching didactic and experimental goals of the platform. Mounted on the rear of the pole, the distribution node remains partially concealed from frontal view to maintain a clean aesthetic silhouette, yet it is fully exposed and accessible from the rear. This "open-box" architecture drastically simplifies system maintenance. If a specific PWM channel were to fail, the exposed headers allow for immediate physical rewiring to one of the redundant spare channels. Furthermore, it facilitates rapid hardware debugging and live voltage probing, serving as a highly observable, practical case study in decentralized embedded power distribution without ever requiring the structural disassembly of the robot.

4.5 Industrial Power Supply and Safety Infrastructure

The mechanical capabilities of the custom manipulator—specifically its reliance on heterogeneous, high-torque digital servomotors—generate severe electrical demands that fundamentally cannot be satisfied by standard hobbyist power bricks or generic laboratory bench supplies. During rapid kinematic reversals or localized multi-joint stalls, the transient current spikes across the entire 12-DOF arm can theoretically exceed 50 amperes. To safely and reliably deliver this level of power, a dedicated, industrial-grade power supply infrastructure was designed and fabricated, prioritizing both electro-mechanical stability and operator safety.

4.5.1 Primary Enclosure and AC Mains Interfacing

Because the power delivery system interfaces directly with high-voltage 220V alternating current (AC) mains, ensuring the galvanic isolation and physical safety of the operator (and students utilizing the didactic platform) was the absolute primary design constraint. To achieve this, the entire power infrastructure is housed within a heavy-duty, IP56-rated rectangular electrical junction box. Measuring 300 x 220 x 120 mm, this standardized industrial enclosure provides a rigid, non-conductive polymer chassis that completely shields the user from live AC voltages while offering ample internal volume for hardware mounting and convective heat dissipation.

The physical interface to the laboratory AC mains is localized to a single panel-mounted IEC C14 power inlet on the exterior of the box. This modular approach allows the system to utilize standard, universally available computer power cables, enhancing the platform's portability. The C14 inlet integrates an illuminated toggle switch providing clear visual feedback of the system's live status and a primary 6A glass cartridge fuse. This 6A primary fuse acts as the first critical layer of the safety architecture; it limits the total theoretical power draw to approximately 1320W ($220V \cdot 6A$), establishing a definitive hardware cutoff that protects the upstream laboratory circuits in the event of a catastrophic internal short or dielectric breakdown within the primary power supplies.

4.5.2 Dual-Rail Switch Mode Power Supplies (SMPS)

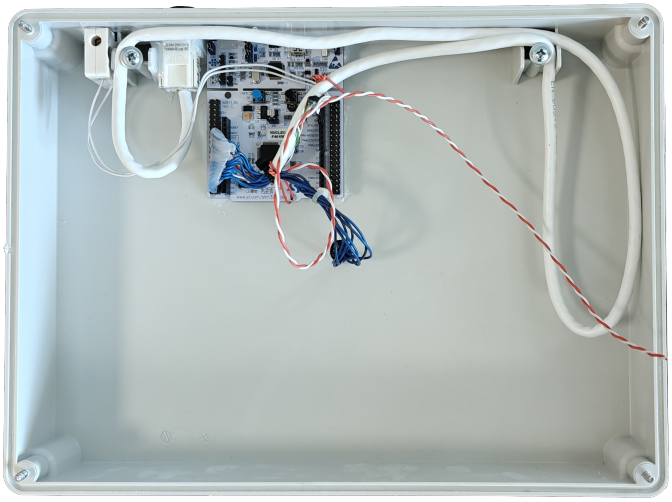
To step the 220V AC mains down to the specific direct current (DC) voltages required by the heterogeneous actuator banks, two independent Mean Well Switch Mode Power Supplies (SMPS)[26] are mounted within the enclosure. Mean Well units were selected primarily for their cost-effectiveness and high power density, providing the substantial current ratings required by the actuator banks within a compact physical form factor.

The dual-rail architecture consists of:

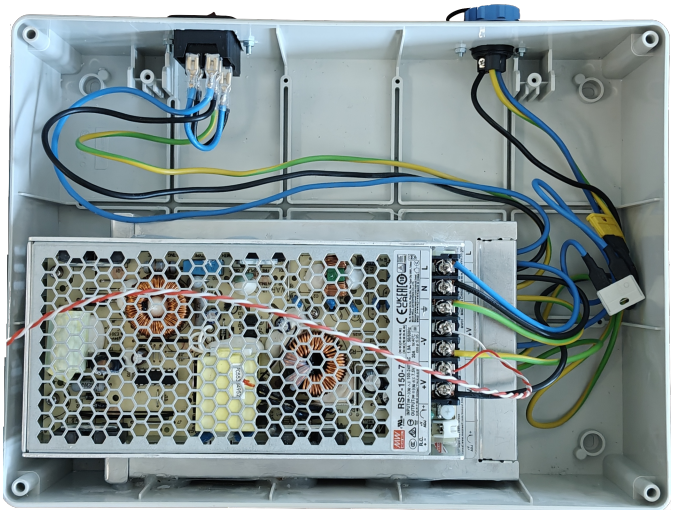
- **The High-Voltage Rail (12.0 V):** Powered by a 30A Mean Well SMPS, dedicated exclusively to providing the massive torque required by the three DSSERVO DS51150 actuators governing the heavy proximal joints (shoulder and elbow).
- **The Low-Voltage Rail (7.5 V):** Powered by a 20A Mean Well SMPS, tailored specifically to the nominal operating voltage of the nine DFRobot DC5535 actuators responsible for the tendon-driven hand and wrist kinematics.

While these two power supplies operate on distinct DC voltage levels, their negative output terminals (0V) are immediately bonded together at the source. Establishing this shared, common ground is a mandatory mechatronic requirement; it ensures that the low-voltage logic signals (such as the 3.3V I2C communication from the microcontroller) and the high-current PWM pulses share a unified electrical reference point.

Furthermore, both Mean Well units feature sophisticated embedded protection circuitry, including Over-Voltage Protection (OVP), Over-Temperature Protection (OTP) and active Over-Current Protection (OCP), effectively safeguarding the power supplies themselves against internal damage during prolonged robotic stall events.



(a) Upper section: STM32 Nucleo and HMI wiring



(b) Lower section: Mean Well SMPS units and DC blade fuses.



(c) Rear I/O panel: AC mains inlet, emergency homing button, SP21 power coupling, RJ45 data jack and Mini-USB interface.

Figure 4.4: Internal layout and rear I/O panel of the IP56 power and control enclosure.

4.5.3 Secondary DC Overcurrent Protection and Internal Routing

While the internal OCP of the Mean Well units protects the power supplies from destroying themselves, it does not adequately protect the physical wire routing. An industrial 30A power supply will flawlessly push 29.9 amperes continuously through a short-circuited wire without tripping its internal protections. If a dead short circuit were to occur downstream in the robot's wiring, this massive current would cause rapid Joule heating, instantly melting the wire insulation and initiating a thermal runaway or electrical fire.

To mitigate this specific risk, a secondary safety layer was introduced on the DC side of the circuit. Standard automotive-grade DC blade fuses were wired in series with the positive outputs of both power supplies. A 25A fuse was placed on the 12V rail and a 20A fuse was placed on the 7.5V rail. Automotive fuses were explicitly chosen for their fast-acting thermal properties, low cost and reliability in handling DC loads. These fuses are mathematically sized to be slightly lower than the absolute maximum rating of their respective power supplies, ensuring they sacrifice themselves to sever the circuit before the internal wiring can thermally degrade during a dead short.

Finally, the internal DC power routing within the IP56 enclosure is executed using multi-stranded copper wire with a cross-sectional area of 1.5 mm^2. In standard residential AC wiring, 1.5 mm^2 is typically rated for 10A to 16A. However, in this specific DC application, the 1.5 mm^2 gauge is entirely sufficient to handle the required currents. This is justified by the extremely short physical wire runs (less than 15 centimeters between the SMPS terminals and the output connectors) and the fact that the cables are routed as single conductors in the open air of the large enclosure which prevents thermal bundling and minimizes parasitic voltage drop across the internal distribution network.

4.6 The Modular Interconnect and Power/Data Interfaces

A fundamental challenge in the design of a tethered robotic manipulator is the engineering of a reliable interconnect system that bridges the external power supply box with the physical base of the robotic arm. In many academic and hobbyist prototypes, transmission lines are permanently hardwired between the control boards and the actuators to minimize costs. However, this approach creates severe logistical limitations and introduces mechanical weak points where the cables continually flex against rigid chassis boundaries. To elevate the platform to a

professional didactic standard, a completely modular, fully detachable architecture was developed, isolating the power transmission from the high-speed data telemetry through heavy-duty panel-mounted interfaces (see figures 3.9 and 4.4c).

4.6.1 Architectural Modularity and System Decoupling

The primary mechanical objective for the power and data interface was the physical decoupling of the robotic arm from the external IP56 power enclosure. By completely severing the permanent hardwired link between the power generation stage and the actuation stage, the system transforms into a modular, highly portable unit.

This architectural choice allows the arm assembly to be quickly disconnected and safely stored or transported independently from the bulky 220V power supply box. Furthermore, this modularity inherently future-proofs the platform's hardware ecosystem. By establishing a standardized external connection, the current robotic arm can easily be powered by entirely new control boxes or different power sources developed in future revisions. Conversely, the existing IP56 control box can seamlessly drive new robotic arms, upgraded linkages or entirely different automated platforms in the future without requiring destructive rewiring. To achieve this bidirectional modularity, heavy-duty panel-mounted interface connectors were explicitly installed at two strategic locations: on the exterior chassis of the IP56 power box and directly at the structural base of the robot's supporting pole.

4.6.2 High-Current DC Transmission and Thermal Mitigation

Transmitting the required DC currents from the control box to the arm base safely over a distance necessitates rigorous conductor sizing. To bridge the 2.0-meter physical span between the subsystems, a robust 3x2.5 mm^2 multi-conductor copper cable was explicitly implemented.

While the internal wiring within the IP56 enclosure safely utilizes thinner 1.5 mm^2 conductors due to extremely short wire runs in open air, deploying that same gauge for the external tether presented a critical thermal hazard. The external cable tightly bundles all three primary power lines (the 12.0V power rail, the 7.5V power rail and the shared 0V common ground) inside a single protective PVC sleeve. Under prolonged operational use and particularly during continuous multi-joint kinematic maneuvers, the combined actuator banks draw sustained high currents.

This electrical load generates continuous Joule heating ($P_{loss} = I^2 R_{wire}$) along the length of the conductors. Because the tight PVC outer sleeve isolates the wires from ambient air and severely restricts convective cooling, the cable is subjected

to amplified thermal bundling effects. If a 1.5 mm^2 cable were utilized, the total line resistance over the 2-meter run—while theoretically too low to induce severe voltage drops or logic-level brown-outs—would generate sufficient localized heat to risk critical overheating over prolonged periods, potentially degrading the internal wire insulation and initiating a short circuit.

By proactively utilizing a heavy-gauge 2.5 mm^2 cross-sectional area, the internal line resistance is significantly minimized. This effectively mitigates dangerous heat accumulation within the cable sleeve, ensuring a safe, thermally stable power transmission from the SMPS to the PCA9685 distribution node, even during continuous, heavy-duty demonstrative use.

4.6.3 Electro-Mechanical Safety: Keying, Locking and Polarity Protection

To physically terminate this heavy-gauge DC cable, industrial SP21[27] aviation connectors were selected. Rated for 30A of continuous current, these 3-pole circular connectors drastically exceed the maximum potential draw of the combined actuator banks, providing an immense thermal and electrical safety margin.

Because the system is designed as an educational demonstrator intended for use in dynamic and often crowded laboratory environments, the physical interfaces were engineered to inherently prevent catastrophic operator errors. If a user were to accidentally invert the power rails (e.g., feeding 12.0V into the 7.5V DC5535 servo network) the actuators would instantly sustain irreversible electrical damage. To completely eliminate this risk, the SP21 connectors feature highly asymmetrical mechanical keying. This ensures that the male and female couplings can only physically mate in one specific orientation, permanently locking the power rails to their correctly assigned pins.

Additionally, careful consideration was given to the physical exposure of the conductive pins. In a busy cluttered mechatronics laboratory an active power cable dropped on a workbench presents a severe short-circuit hazard, especially given the high amperage of the power supplies. To mitigate this, the connectors actively feeding live current (the panel mount on the IP56 box and the terminating end of the transmission cable) utilize deeply recessed female socket contacts. Conversely, the connectors strictly receiving power (the panel mount on the robot base) utilize exposed male pin contacts. This physical shrouding ensures that even if the live cable is mishandled or dropped on a conductive surface, the recessed power pins cannot accidentally bridge and short-circuit.

4.6.4 High-Speed Data Routing and Dual-Redundancy Telemetry

While the SP21 interfaces handle the raw power transmission, the low-level logic signals must be routed in parallel. The control architecture relies on a 400kHz Fast-mode I2C communication bus to transmit the target PWM duty cycles from the central STM32 microcontroller (housed inside the IP56 box) to the modified PCA9685 distribution node (mounted on the robot's shoulder).

To achieve this data transfer across the modular gap, standard panel-mounted female RJ45 jacks were installed adjacent to the SP21 power connectors on both the box and the robot base as depicted in figures 3.9 and 4.4c. This allows the logic telemetry to be carried by a standard, universally available Ethernet cable (CAT5e/CAT6). Utilizing commercial Ethernet cables is highly advantageous: they are heavily shielded against electromagnetic interference (crucial when running adjacent to a high-current DC umbilical), inexpensive to replace if damaged and feature a secure plastic retention clip.

The I2C protocol and logic power technically require only four discrete conductors: Ground (GND), +3.3V Logic Power, Serial Clock (SCL) and Serial Data (SDA). However, a standard RJ45 cable contains eight internal conductors. Rather than leaving four wires floating and unused, a dual-redundancy mapping strategy was implemented. The eight available conductors were explicitly paired together in parallel (2xGND, 2x+3.3V, 2xSCL, 2xSDA).

This specific mechatronic "hack" fundamentally doubles the cross-sectional area of the logic transmission lines, but more importantly, it introduces a physical fail-over mechanism. The delicate copper contacts inside RJ45 connectors are notoriously prone to bending or oxidizing under frequent insertion cycles and the internal wires can micro-fracture under repetitive flexing. By wiring the signals in redundant pairs, if any single contact fails or a wire breaks, the parallel conductor instantaneously maintains the data link. This redundancy exponentially increases the Mean Time Between Failures (MTBF) of the control signal, ensuring uninterrupted, deterministic control over the manipulator.

4.7 Analog Control Interface (Hardware HMI)

While the primary objective of the custom embedded architecture is to facilitate high-level, autonomous control via serial communication with a Digital Twin, relying exclusively on an external PC for kinematic manipulation presents severe didactic and operational limitations. If the serial link is interrupted or if a user wishes to empirically test a mechanical joint without booting the entire software stack, the manipulator would be effectively paralyzed. To resolve this, a dedicated, standalone Human-Machine Interface (HMI) was engineered. Mounted directly to the exterior of the IP56 power enclosure, this physical control panel provides immediate, deterministic control over the robotic arm's low-level kinematics.

4.7.1 Kinematic Mapping and Potentiometer Array

To provide fluid, intuitive manual control over the complex 12-DOF manipulator, the HMI was designed around an array of high-precision linear potentiometers (sliders). Rather than enforcing a strict 1:1 mapping of sliders to servomotors—which would be overwhelmingly complex for a human operator to coordinate manually—a macroscopic kinematic mapping strategy was implemented.

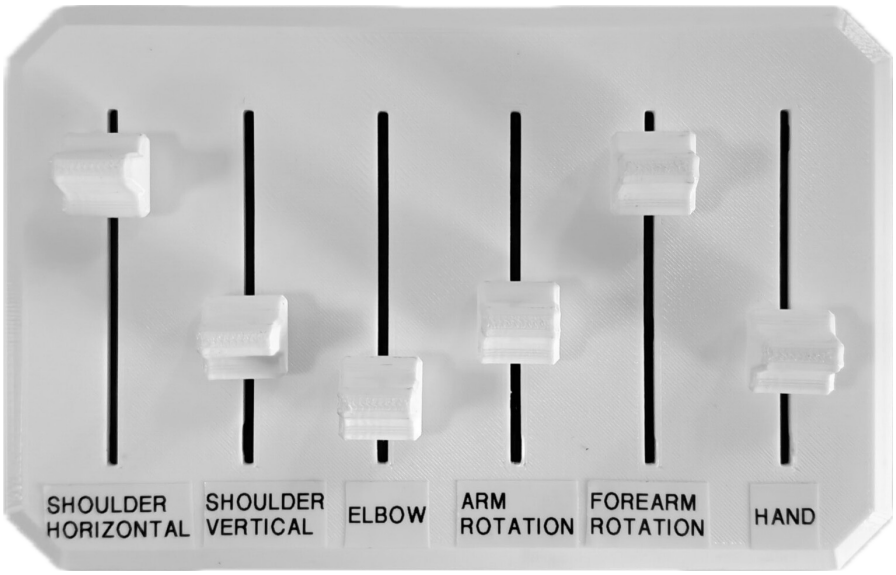


Figure 4.5: Six-channel linear potentiometer HMI for manual joint manipulation.

The control panel features a total of six independent sliders, strategically assigned to govern the primary functional axes of the arm:

1. **Shoulder Horizontal (Azimuth):** Governing the base yaw.
2. **Shoulder Vertical (Elevation):** Governing the primary lifting torque.
3. **Elbow Flexion/Extension:** Controlling the mid-arm articulation.
4. **Arm Rotation:** Enabling the longitudinal twist of the upper arm.
5. **Forearm Rotation (Pronation/Supination):** Controlling the orientation of the wrist.
6. **Global Hand Articulation:** A singular macroscopic slider mapped to simultaneously actuate all five fingers for immediate grasping maneuvers (omitting isolated wrist actuation in this manual state to streamline user operation).

The hardware utilized for this array consists of six 10 $k\Omega$ linear potentiometers, explicitly selected for their extended 75 mm physical stroke. This long mechanical travel is critical for a didactic interface; it provides the operator with a high degree of tactile resolution, allowing for minute, granular adjustments to the arm's spatial posture that would be impossible with standard rotary knobs or short-throw sliders.

4.7.2 Mechanical Integration and Prototyping Substrate

The physical fabrication of the HMI panel required a robust structural substrate to anchor the potentiometers securely against the repetitive physical force exerted by the operator. Due to specific supply-chain constraints during the build phase, a monolithic printed circuit board (PCB) of adequate dimensions to house the wide 75 mm slider array was unavailable.

To overcome this, a rapid-prototyping solution was executed: two standard perforated fiberglass prototyping boards were mechanically and thermally joined together using structural solder welds. This composite "perf-board" functions strictly as a rigid mechanical chassis for the sliders; it contains no active logic components, microcontrollers or signal-processing ICs.

To protect the exposed electronics and seamlessly integrate the HMI with the aesthetic of the system, a custom housing was designed in CAD and FDM-printed using White Verbatim PETG. This casing encapsulates the raw perf-board, firmly anchoring the assembly to the top lid of the IP56 power supply box. Furthermore, custom ergonomic slider knobs were parametrically designed and 3D printed, transforming the bare metal potentiometer tangs into a professional, user-friendly control surface.

Electrical Topology and ADC Interfacing

Electrically, the HMI operates as a parallel array of six voltage divider networks. The entire potentiometer bank is powered directly by the STM32 Nucleo microcontroller housed securely beneath the lid inside the IP56 enclosure. A centralized wiring harness passes through a sealed aperture in the center of the enclosure's lid, delivering the +3.3V logic supply and the 0V reference ground from the STM32 to the external board.

As the user physically manipulates a slider, the internal wiper sweeps across the resistive carbon track, outputting a highly linear, continuous analog voltage ranging precisely between 0V and 3.3V. These six independent analog signal lines are routed back down through the lid and directly connected to six designated Analog-to-Digital Converter (ADC) input pins on the STM32.

Because the sliders are powered by the exact same voltage rail used as the absolute reference for the STM32's internal ADCs, the system acts ratiometrically. This ensures that any slight fluctuations or noise in the 3.3V supply line cancel out mathematically during the analog-to-digital conversion, resulting in incredibly stable, jitter-free positional readings from the human interface.

Prelude to the Hybrid Control Architecture

While the hardware implementation of the analog HMI is relatively straightforward, its true mechatronic value lies in how it interacts with the overarching system logic. The HMI is not merely a fallback system; it is designed to operate concurrently with the high-level serial communication from the PC, acting as the primary, high-priority control override.

To prevent catastrophic signal collisions (e.g., when the serial port commands the arm to move left while the physical slider commands it to move right) a sophisticated state-machine was required. The STM32 continuously monitors the analog voltage levels of the HMI array. Rather than utilizing a physical toggle switch to change between "Manual" and "Serial" modes, the embedded software actively traces the temporal derivative of the sliders' positions. The moment the software detects intentional human interaction (a delta in the potentiometer voltage), it instantaneously suspends the serial command execution and dynamically hands full control over to the physical HMI.

The complex implementation of this hybrid control architecture, alongside the Direct Memory Access (DMA) pipelines required to read these analog voltages without halting the main control loop, fundamentally shifts the project from a hardware challenge to an embedded software challenge. This software framework will be comprehensively detailed in chapter 5.

Chapter 5

Embedded Software and Digital Twin Architecture

The preceding chapter detailed the hardware architecture, electronic integration and mechanical modifications required to realize the customized robotic platform. However, the capabilities of this hardware ultimately depend on the software responsible for its low-level control, communication, safety management and user interaction.

This chapter presents the design, implementation and evaluation of the custom software architecture developed for the robotic arm. The main objective was to replace the original proprietary control stack with a transparent and configurable software framework adapted to the modified hardware. The resulting system is organized into two main operational domains: the low-level embedded firmware, responsible for real-time actuator control and safety functions and the high-level software layer, which provides communication, user interaction and the Digital Twin interface.

The development of this architecture was driven by the specific requirements of the modified manipulator, including the use of different actuators, the custom electronic architecture and the need for direct access to low-level control functions. The following sections describe the resulting software architecture, beginning with its design rationale and overall organization, followed by the implementation of the embedded firmware, safety mechanisms, hybrid control interface and Digital Twin.

5.1 Custom Software Architecture

5.1.1 Rationale for a Custom Software Ecosystem

The original robotic project provides a standard software suite intended to drive the standard hardware configuration. However, utilizing this stock software presented several critical limitations for the scope of an M.Sc. thesis. Primarily, the proprietary low-level firmware on the Tinkerforge Servo[20] is obscuring the underlying control loops, signal routing and kinematic calculations. This lack of transparency severely limits the potential for in-depth mechatronic analysis and optimization.

Furthermore, the physical reality of the constructed prototype necessitated a bespoke approach. Because only the left half of the robot was fabricated and several standard actuators were substituted with alternative components to reduce costs and explore different performance metrics, the mechanical plant fundamentally deviated from the stock design. Relying on a black-box software suite designed for the complete kinematics and original actuators posed a significant risk of unpredictable behavior, unstable control dynamics and potential mechanical collisions.

By developing a custom software architecture from the ground up, several critical advantages were realized:

- **Total Transparency and Control:** Every discrete signal, from the incoming serial payload down to the individual PWM (Pulse Width Modulation) ticks sent to the PCA9685[25] driver, is explicitly defined and accessible. This allows for a much more detailed level of control and highly optimized performance tuned specifically to the newly selected actuators.
- **Cost-Effectiveness:** The custom architecture leverages open-source libraries and lightweight protocols, completely avoiding the need for expensive commercial control software or proprietary middleware.
- **Educational Value and Configurability:** As an academic project, the primary objective is to lay a modular foundation. The custom codebase is heavily documented and structurally decoupled, ensuring that future researchers can easily modify kinematic parameters, integrate new sensors or swap hardware without rewriting the entire control stack.

5.1.2 The Dual-Layered Hybrid Framework

To balance the strict timing requirements of physical motor control with the computational demands of user interfacing and telemetry, the architecture was explicitly bifurcated into two distinct but synchronized domains:

1. **Low-Level Embedded Firmware (Bare-Metal C):** Running on the STM32 microcontroller[24] utilizing the Hardware Abstraction Layer (HAL), this domain is strictly responsible for real-time, deterministic operations. It handles the continuous polling of analog inputs via Direct Memory Access (DMA), parses incoming serial commands, computes trajectory generation to ensure smooth mechanical actuation and enforces critical hardware-software safety interlocks.

2. **High-Level Middleware and Digital Twin (Python / WebGL):** Running on a host PC, this domain abstracts the complexities of the physical robot into a highly visual, user-friendly interface. A Python-based asynchronous bridge manages the serial communication, translating high-level user commands into byte-sized packets, while a WebGL front-end provides a 3D visual representation of the robot's spatial pose in real-time.

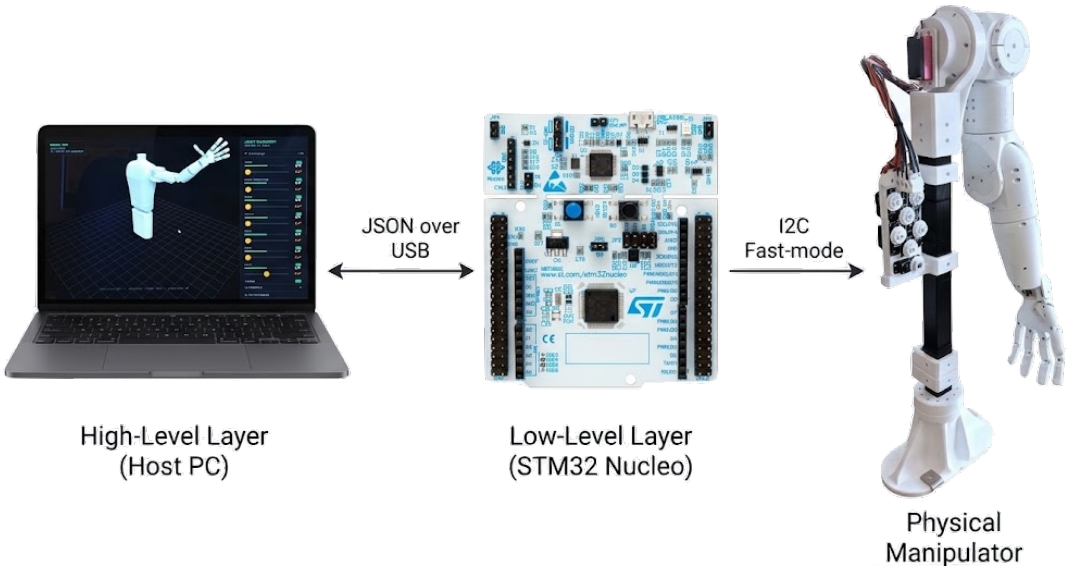


Figure 5.1: Dual-layer framework: host Digital Twin and STM32 communicating via USB-JSON to drive the manipulator over Fast-mode I2C.

5.1.3 Virtual Commissioning and Safe Development

Beyond providing an immediate visual representation of the project for demonstration purposes, the high-level Digital Twin served as a vital engineering tool throughout the development life cycle.

Working with powerful robotic actuators inherently carries the risk of hardware damage or personal injury, especially when testing preliminary, unverified firmware. The integration of the Digital Twin enabled a process akin to "Virtual Commissioning". It allowed the low-level communication protocols, JSON parsing logic and kinematic calculations on the STM32 to be rigorously tested and visualized in the browser without the physical robot being attached or powered.

By decoupling the software logic from the physical plant, developers could inject simulated target angles, observe the mathematical response of the embedded firmware via the 3D model and identify logical faults — all while the physical servos remained safely de-energized. This methodology drastically accelerated the development pipeline and ensured that when physical power was finally introduced, the control algorithms were already proven to be stable and predictable.

5.2 Low-Level Embedded Firmware (STM32 HAL)

The core operational domain of the robotic platform is the embedded firmware running on the STM32 microcontroller. While the high-level Digital Twin handles the computationally expensive tasks of user interfacing and 3D rendering, the STM32 is responsible for the strict, deterministic execution of physical control loops. Built entirely in C utilizing the STMicroelectronics Hardware Abstraction Layer (HAL)[3], this low-level architecture directly interfaces with the silicon peripherals (e.g., Timers, I2C buses, Analog-to-Digital Converters ADCs) to translate abstract logical commands into precise electromechanical action.

The primary objective of this firmware layer is to ensure highly responsive, real-time control while executing complex kinematic mathematics, all within the strict memory and processing constraints of an embedded environment.

5.2.1 Non-Blocking Serial Communication and JSON Parsing

In a real-time robotic control system, one of the most significant bottlenecks is communication. Receiving, buffering and parsing string-based data over a Universal Asynchronous Receiver-Transmitter (UART) interface can consume excessive clock cycles. If the central processing unit (CPU) is forced to block (wait) while a message is received, the high-frequency motor control loops will inevitably stutter, leading to erratic physical movements and potentially dangerous loss of synchronization.

To circumvent this, the serial communication architecture was designed to be strictly asynchronous and non-blocking. Utilizing the STM32's Direct Memory Access (DMA) controller in conjunction with hardware interrupts, incoming byte

streams from the host PC are automatically routed into a circular ring buffer in the microcontroller's SRAM. This autonomous hardware-level routing completely offloads the reception task from the CPU. The main control loop only interacts with the buffer once a terminal character triggers a flag, signaling that a complete data payload is ready for processing.

To maximize structural flexibility and compatibility with the web-based Digital Twin, JavaScript Object Notation (JSON) was chosen as the standard data payload format. Because dynamic memory allocation (`malloc` and `free`) is strictly avoided in embedded control loops to prevent heap fragmentation, the firmware integrates `jsmn`[28], a minimalist, zero-allocation JSON parser. `jsmn` tokenizes the incoming DMA ring buffer in-place, allowing the system to instantly decode highly configurable, multi-axis target trajectories on the fly.

The control protocol dictates a specific JSON schema that the system is capable of parsing. A complete command string encapsulates several parameters, both mandatory and optional:

`"m"` **(Motor ID):** Mandatory integer identifying the target actuator.

`"p"` **(Position):** Mandatory float defining the target angle in degrees.

`"v"` **(Velocity):** Optional float that updates the maximum cruising velocity.

`"a"` **(Acceleration):** Optional float that updates the maximum acceleration limit.

`"t"` **(Telemetry Focus):** Optional integer. For instance, sending `{"t": 3}` instructs the STM32 to shift its debug telemetry focus to motor 3.

An example of a fully populated payload sent from the Digital Twin to the STM32 would appear as: `{"t":1, "m":1, "v":100.0, "a":50.0, "p":90.0}`.
In this transaction, the payload commands actuator 1 to navigate toward a target angle of 90.0° while updating its kinetic profile to a maximum cruising velocity of 100.0°/s and an acceleration limit of 50.0°/s^2. Concurrently, the inclusion of `"t":1` redirects the STM32's debug telemetry pipeline to isolate and stream the real-time internal states of motor 1.

Furthermore, the outbound serial communication (from the STM32 back to the PC) was designed to operate in two distinct modes, depending on the required development or operational context:

1. **Debug Mode:** In this state, the STM32 transmits a highly detailed string containing the internal mathematical states of a single specified motor. The payload includes the target position, current physical position, instantaneous velocity, instantaneous acceleration and the raw PWM output. This mode was crucial for tuning the closed-loop control mathematics.
2. **Twin Mode:** In standard operation, detailed single-motor metrics are suppressed. Instead, the STM32 broadcasts a single, optimized string containing an array of the current positional angles for all 12 motors simultaneously. This ensures the high-level WebGL software receives a complete, synchronized snapshot of the robot's pose to update the 3D visualizer at a high refresh rate.

5.2.2 Kinematic Mapping and Mechanical Constraints

Once the JSON payload is parsed, the firmware possesses the logical target angles for the joint, expressed in degrees (e.g., $\theta_{target} = 90°$). However, the physical hardware, specifically the PCA9685 I2C driver, does not interpret degrees. It requires discrete 12-bit integer values ranging from 0 to 4095, representing the required pulse width (duty cycle) to physically actuate the servo motor.

To translate the logical degrees into physical hardware signals, a direct linear mathematical mapping was implemented within the firmware. Because the physical sweep of each motor is strictly constrained by mechanical stops, a precise calibration protocol was established to define the safe operational envelope for every individual joint.

For each actuator, a configuration structure (`motor_config_t`) defines three critical parameters based on manual physical calibration:

- **Minimum Extent:** θ_{min} P_{min}
  The lowest safe physical angle and its corresponding PWM tick value.
- **Maximum Extent:** θ_{max} P_{max}
  The highest safe physical angle and its corresponding PWM tick value.
- **Home Waypoint:** θ_{home} P_{home}
  A pre-defined neutral coordinate.

Where θ represents the physical angle in degrees and P represents the corresponding 12-bit PWM tick value.

When the STM32 receives a new θ_{target}, the required PWM output P_{target} is calculated using standard linear interpolation along the single operational segment defined by the minimum and maximum extents:

$$P_{target} = P_{min} + \frac{(\theta_{target} - \theta_{min}) \cdot (P_{max} - P_{min})}{(\theta_{max} - \theta_{min})} \tag{5.1}$$

The primary function of this calibration protocol is absolute mechanical protection. The 3D-printed components of the robotic arm are relatively fragile and highly susceptible to self-destruction if the servos are driven past their physical limits. Immediately following the linear calculation, strict mathematical saturation (clamping) is applied. If a user or the Digital Twin erroneously requests a target angle that forces P_{target} outside the safe bounds of P_{min} or P_{max}, the firmware overrides the command arresting the mathematical progression exactly at the safe physical limit.

The third calibrated point, the Home Waypoint, does not alter the linear mathematical interpolation. Rather, it serves as a pre-defined, safe neutral posture. This specific coordinate on the linear segment represents the default state the arm assumes during the initial power-up sequence. Crucially, it is also the designated fallback position to which the arm automatically returns if the hardware Emergency Stop (the red button on the back of the control box visible in figure 4.4c) is triggered, ensuring the system can always be systematically brought back to a known, safe configuration regardless of its previous state.

5.2.3 Trajectory Generation and Trapezoidal Velocity Profiles

As detailed in chapter 4, the baseline PIB architecture utilizes proprietary Tinkerforge Servo Bricklets (figure 4.4a), which natively compute speed curves and manage kinematic trajectory generation. However, the strategic decision to discard these expensive, black-box modules in favor of a cost-effective PCA9685 PWM driver fundamentally shifted the kinematic responsibility back to the master microcontroller.

Without the Tinkerforge middleware, the STM32 interfaces directly with the bare commercial servomotors. These actuators rely solely on their internal PID (Proportional-Integral-Derivative) controllers, which are inherently designed to minimize position error as rapidly as possible. Consequently, if a bare servo receives a single step-input command to move from 0° to 90°, it will instantaneously apply maximum torque to accelerate to its physical speed limit, coast and then brake violently upon reaching the target.

To protect the physical plant, the firmware was engineered to mathematically override the internal servo PID controllers by acting as a high-frequency trajectory generator. Instead of sending the final target angle to the PCA9685 driver, the STM32 calculates a trapezoidal velocity profile[4] and feeds the servomotors microscopic, dynamically updating waypoints. Because the servo only ever sees a target that is a fraction of a degree away from its current position, it is forced to move smoothly, effectively transforming a rigid position-controlled device into a smooth, velocity-and-acceleration-controlled system.

This trajectory generation is orchestrated by a precise hardware timer interrupt set to a control frequency of 300 Hz (`CTRL_FREQ_HZ`). Every 3.33 milliseconds (dt), the main control loop executes the mathematical update step (`Motor_UpdateStep`) for all 12 motors simultaneously.

For a given motor, the instantaneous positional error e is calculated as:

$$e = \theta_{target} - \theta_{current} \tag{5.2}$$

To determine whether the motor should accelerate towards the target or begin decelerating to prevent overshoot, the firmware calculates the theoretical stopping distance d_{stop} required to arrest the current kinetic energy, governed by the user-defined maximum acceleration limit a_{max}:

$$d_{stop} = \frac{v_{current}^2}{2 \cdot a_{max}} \tag{5.3}$$

The state machine then evaluates the trajectory phase:

1. **Acceleration / Cruise Phase:** If $|e| > d_{stop}$, the motor has not yet reached its braking zone. The velocity $v_{current}$ is incrementally increased by $a_{max} \cdot dt$, explicitly clamped by the maximum allowed cruising velocity v_{max}.
2. **Deceleration Phase:** If $|e| \leq d_{stop}$, the motor is approaching the target. The velocity is incrementally reduced, mathematically ensuring it approaches zero exactly as $\theta_{current}$ aligns with θ_{target}.

Finally, Euler integration updates the physical state variables for the current time step:

$$v_{k+1} = v_k + (a \cdot dt) \qquad \theta_{k+1} = \theta_k + (v_k \cdot dt) \tag{5.4}$$

This dynamically calculated θ_{k+1} is then passed to the linear mapping algorithm (detailed in section 5.2.2) to be converted into the immediate PWM output.

5.2.4 Mitigating Asymptotic Jitter via Dynamic Deadbands

While the trapezoidal trajectory generation perfectly maps the theoretical kinematic motion, deploying discrete-time mathematics to govern a continuous physical system introduces a specific computational challenge at the termination of the movement.

As the calculated distance to the target decreases, the theoretical deceleration curve asymptotically approaches zero. However, forcing the microcontroller to compute this trajectory down to a mathematical position error of exactly 0.000° is highly inefficient and mathematically problematic. Because the positional integration occurs in discrete 3.33 ms time steps, the calculated position will inevitably step past the absolute zero-point by an infinitesimal fraction of a degree.

The 300 Hz control loop immediately detects this microscopic mathematical overshoot and calculates a tiny reverse velocity to correct it. Because the physical servos are highly responsive and faithfully execute exactly what the STM32 commands, it is the control algorithm itself that inadvertently forces the hardware into a perpetual, high-frequency oscillation (limit cycling) back and forth across the target coordinate. In physical observation, this mathematically induced jitter manifests as an auditory buzzing, causing the servomotor to draw continuous stall current and rapidly leading to thermal saturation.

To resolve this computational edge-case without sacrificing the macroscopic fluidity of the trajectory, the firmware implements Dynamic Deadbands at the termination of the `Motor_UpdateStep` function. This logic functions conceptually similarly to a soft-thresholding operator (such as those utilized in LASSO regression for L1 regularization), where values falling below a negligible threshold are explicitly zeroed out to enforce stability and computational sparsity.

The firmware defines two strict empirical tolerances:

- A spatial tolerance: $\epsilon_p = 0.3°$
- A kinetic tolerance: $\epsilon_v = 5.0°/s$

Before calculating the next kinetic step, the algorithm evaluates the current mathematical state against these deadbands:

$$\text{If } |e| < \epsilon_p \text{ AND } |v_{\text{current}}| < \epsilon_v \tag{5.5}$$

If both conditions are met (i.e., the joint calculation is mathematically close enough to the target and the theoretical velocity is negligible) the step calculation is forcefully bypassed. The firmware applies a "kinematic lock":

$$\theta_{current} = \theta_{target} \qquad v_{current} = 0.0 \tag{5.6}$$

By explicitly overriding the asymptotic tail of the deceleration curve and snapping the logical position directly to the target, the computational jitter is eliminated entirely. The resulting PWM signal stabilizes perfectly, silencing the servomotors upon reaching their destinations, preventing unnecessary current draw and safeguarding the actuators against thermal degradation during extended didactic demonstrations.

5.3 Hardware-Software Safety Interlocks

The implementation of advanced trajectory generation and precise linear mapping detailed in section 5.2 establishes the mathematical foundation for smooth kinematic control. However, these theoretical models rely on the assumption of standard, uninterrupted operation. In a physical robotic system possessing significant mass and torque, edge cases, such as user error, communication timeouts or sudden hardware interruptions can lead to catastrophic mechanical failure.

To ensure the physical plant and the operator remain protected, a robust layer of safety interlocks was engineered directly into the STM32 Hardware Abstraction Layer (HAL). This architecture specifically addresses the unique limitations of utilizing consumer-grade actuators within a heavy, underactuated humanoid framework.

5.3.1 Boot-Up Diagnostics and Controlled Homing

In traditional industrial robotics, safety mechanisms typically involve a hardwired contactor that instantly severs the high-voltage power bus to the servo drives. Applying this conventional methodology to the modified PIB platform, however, introduces a severe safety hazard.

Because the commercial servomotors utilized in this project lack internal electromagnetic brakes, they possess zero holding torque when unpowered. If the main DC power supply were to be suddenly cut while the arm is fully extended, the entire upper-limb structure would instantly become backdrivable, collapsing violently under its own mass and gravity. This uncontrolled descent would likely shatter the 3D-printed PETG linkages and potentially cause physical harm to the operator.

To circumvent this, the primary hardware safety trigger, implemented as a red pushbutton on the rear panel of the control enclosure (figure 4.4c), is wired as a Normally Closed (NC) circuit routed directly as a logic-level hardware interrupt to the STM32 (`EXT_BTN_Pin`). When the button is struck, rather than killing the power, the firmware triggers a Controlled Homing state (`ROBOT_HOMING`).

During Controlled Homing, the software state machine immediately supersedes any incoming JSON trajectories from the Digital Twin or physical HMI inputs. It overrides the target coordinates for all 12 actuators, forcing them to the pre-calibrated Home waypoint (detailed in section 5.2.2. Crucially, this return-to-home maneuver is executed at an artificially restricted, heavily dampened velocity. By maintaining active power to the servos while algorithmically forcing a slow retreat to a neutral posture, the robot safely bleeds off its kinetic energy without structural collapse.

A crucial mechatronic design consideration for this architecture was system resilience. Incorporating hardware safety loops often introduces single points of failure. In a rigid safety implementation, if the external NC circuit were to be damaged (e.g., a severed wire or a broken contact inside the red button), the microcontroller would interpret this as a permanently triggered emergency state, effectively "bricking" the robotic platform and preventing any didactic use.

To prevent a minor peripheral fault from rendering the entire system unusable, the firmware executes a strict boot-up diagnostic sequence. Upon initialization, the STM32 evaluates the state of the external button circuit. If the circuit is detected as open during boot (i.e., the button is currently physically depressed or the wiring is compromised), the firmware logs the fault and allows the system to boot normally, but it explicitly ignores any further input from the external button for that session, disabling its homing trigger. To ensure that operator safety is not compromised when the external button is bypassed or fails, the system features a redundant fallback. The onboard blue user button (B1), integrated directly onto the STM32 Nucleo development board, is hardcoded to initiate the exact same Controlled Homing sequence. This dual-layered approach guarantees that a manual, hardware-level controlled homing intervention is always viable, preserving both the safety of the operator and the operational uptime of the educational platform.

5.3.2 Open-Loop Initialization and Conservative Constraints

Perhaps the most critical mechatronic limitation of the chosen hardware architecture is the lack of true absolute proprioceptive feedback to the master controller. While the servos utilize internal potentiometers for their own local closed-loop PID control, they do not broadcast this positional data back to the STM32 over the standard PWM lines.

Consequently, upon initial power-up, the embedded firmware is functionally blind to the true physical pose of the manipulator. In order for the 300 Hz trajectory generator (section 5.2.3) to compute a smooth trapezoidal velocity profile, it intrinsically requires two variables: a target position and a current position. Because

the STM32 cannot read the true current position, the firmware is hard-coded to assume that, at boot, every actuator is perfectly aligned with its Home coordinate (current=home).

This open-loop initialization creates a profound operational hazard. If the physical arm is left in an extended or arbitrary pose while unpowered and the system is subsequently booted, the STM32 will instantly output the PWM signals corresponding to the Home position. Because this output is treated as the starting point of the trajectory, no smoothing is applied to this initial signal. The servo's internal black-box PID controller will receive a massive instantaneous step-input error and will saturate its internal motor drivers to reach the Home position as aggressively as possible. This violent snap can instantly strip the internal gearboxes or shear the printed tendon pulleys.

To mitigate this limitation—which cannot be resolved via software due to the black-box nature of the commercial servo firmware—a strict operational protocol was established: manual zeroing is an absolute prerequisite. The operator must physically manually backdrive the unpowered arm into its approximate neutral Home posture before energizing the system.

Once safely booted in the correct posture, the firmware enforces a secondary layer of protection via conservative kinetic constraints. The structural limits of the arm (the absolute maximum velocities and accelerations that the joints can withstand without failing) are strictly hard-coded in the C firmware. However, at startup, the initial v_{max} and a_{max} variables are clamped to highly conservative baseline values (e.g., 20% of maximum velocity, 50% of maximum acceleration).

If an advanced demonstration requires higher dynamic performance, these operational constraints can be dynamically ramped up during runtime. The system accepts optional "v" and "a" parameters embedded within the incoming JSON serial payload (e.g., `{"m":3, "p":90.0, "v":150.0, "a":80.0}`). The software parses these new kinetic boundaries, verifies that they do not exceed the hard-coded structural limits designed to protect the hardware and updates the local limits on the fly. This architecture ensures the robot powers up safely in a highly restricted state, requiring explicit software intent to unlock its full dynamic capabilities.

5.4 Hybrid Control Architecture (ADC + DMA)

As established in chapter 4, the robotic platform features a custom analog Human-Machine Interface (HMI) consisting of an array of linear potentiometers. This physical control panel allows students and operators to directly manipulate the kinematics of the arm for didactic observation without relying on the external PC or

the Digital Twin. However, integrating this hardware interface introduced a significant control theory challenge: the system now possessed two independent "masters" attempting to dictate the target position of the robotic joints simultaneously.

If the high-level Digital Twin commands a joint to 90° via the serial port, while the physical slider is currently resting at a position corresponding to 45°, the embedded system must possess a deterministic logic to resolve the conflict. Without explicit arbitration, a "signal collision" occurs, where the 300 Hz control loop rapidly alternates between the two discordant targets, causing severe mechanical jitter, immense stall currents and potential destruction of the 3D-printed linkages.

To resolve this, a sophisticated hybrid control architecture was developed, utilizing hardware-level memory pipelines and intelligent software state machines to allow seamless, collision-free transitions between autonomous digital control and manual physical override.

5.4.1 Direct Memory Access and Signal Filtering

The physical HMI is read by the STM32 via its internal Analog-to-Digital Converter (ADC). Six linear potentiometers are mapped to a subset of critical joints (section 4.7.1), producing varying analog voltages (0V to 3.3V) that the 12-bit ADC converts into discrete integer values ranging from 0 to 4095.

In a standard embedded application, reading multiple analog channels is achieved by forcing the CPU to explicitly poll each pin sequentially, waiting for the electrical conversion to complete before moving to the next. In a real-time robotic context, this blocking architecture is highly detrimental. Polling six distinct analog channels would consume thousands of clock cycles, introducing severe latency that would fundamentally disrupt the strict 3.33 ms timing of the kinematic trajectory generator (detailed in section 5.2.3).

To eliminate this CPU overhead, the architecture leverages Continuous Scan Mode coupled with Direct Memory Access (DMA). The ADC peripheral is configured at the hardware level to autonomously and continuously cycle through the six designated analog channels. As soon as a conversion is complete, the DMA controller intercepts the data and routes the 12-bit integer directly into a dedicated array in the SRAM (`adc_raw_buffer[6]`). This entire data pipeline occurs in the background at the silicon level; the CPU is never interrupted and expends zero processing time gathering the sensor data. It simply accesses the `adc_raw_buffer` memory addresses whenever it is ready to process the user input.

However, raw analog signals are inherently volatile. Electrical interference from the high-current DC motor supplies, parasitic capacitance and the physical carbon-wiper imperfections of the affordable potentiometers introduce high-frequency noise into the ADC readings. If passed directly to the motors, this electrical noise would translate into erratic micro-movements (*jitter*) at the robotic joints.

To sanitize the human input, the firmware applies an Exponential Moving Average (EMA) low-pass filter to the DMA buffer during the main control loop. The EMA was selected over a standard Simple Moving Average (SMA) because it requires significantly less memory (no need to store historical arrays) while providing excellent smoothing.

For each channel i, the smoothed signal S at the current time step t is calculated as:

$$S_{t,i} = \alpha \cdot R_{t,i} + (1 - \alpha) \cdot S_{t-1,i} \tag{5.7}$$

Where:

- R_t is the instantaneous raw ADC reading from the DMA buffer.
- S_{t-1} is the smoothed value from the previous control cycle.
- α is the smoothing coefficient (hardcoded as `EMA_ALPHA=0.1f`).

This aggressive alpha value heavily attenuates high-frequency electrical spikes, yielding a perfectly stable, noise-free numerical representation of the physical slider's position while maintaining adequate responsiveness to genuine human manipulation.

5.4.2 The Sync-Lock Safety Protocol and Intelligent Hysteresis

With the analog data successfully acquired and sanitized, the firmware required an arbitration logic to determine "who" possesses control over the robot: the PC Digital Twin or the manual HMI. Rather than utilizing a physical, binary toggle switch on the control panel to swap between "Manual" and "Serial" modes, a dynamic software-defined Sync-Lock protocol was implemented.

The core philosophy of this protocol is that the high-level Digital Twin acts as the default master, but the physical HMI holds absolute priority only when active human intent is detected. To detect human intent, the software actively monitors the temporal derivative of the smoothed slider positions. The system stores a baseline state (`adc_last_stable[i]`) and continuously calculates the absolute differential (Δ):

$$\Delta_i = |S_{t,i} - adc_last_stable_i| \tag{5.8}$$

To distinguish between residual electrical drift and an actual human finger moving the slider, a strict hardcoded noise threshold (`ADC_NOISE_THRESHOLD=30.0f`) is enforced.

The Override Event:
If $\Delta_i > 30$, the firmware positively identifies deliberate manual actuation. Instantly, the state machine triggers the Sync-Lock. The system overrides the incoming JSON serial commands for that specific motor, geometrically maps the 12-bit ADC value to the corresponding joint angle (e.g., 0° to 180°) and forces the trajectory generator to follow the physical slider. Simultaneously, the system logs the exact timestamp of this interaction (`last_hmi_interaction_time=HAL_GetTick()`).

Intelligent Hysteresis and Timeout:
The primary challenge of operating a hybrid control architecture is that analog circuits are intrinsically noisy. Even when the physical HMI sliders are completely untouched by the operator, ambient mechanical vibrations transmitted through the workspace, coupled with standard electrical fluctuations across the carbon wipers, cause continuous minor variations in the raw analog signal.

If the system lacked a robust arbitration protocol, a random noise spike from an untouched potentiometer could inadvertently trigger the override threshold while the robot is actively executing an autonomous trajectory commanded by the Digital Twin. This false positive would instantly hijack the control loop, interrupting the ongoing serial communication and generating a spurious, unintended position input.

To prevent this instability, an intelligent timeout-based hysteresis was integrated into the state machine. The firmware strictly requires a definitive, large delta ($\Delta_i > 30$) to initially wrest control away from the serial port. Once manual intent is positively identified, the system logs the timestamp (`last_hmi_interaction_time`) and the HMI retains an authoritative lock on the kinematic target for 2000 ms after the last explicitly detected physical movement.

This 2000-millisecond buffer acts as a strict "lock-out" period. It ensures that once manual control is established, subsequent analog noise or minor vibrations will not cause the system to rapidly bounce back and forth between the serial and manual domains. Only when the 2-second timer completely elapses without any analog signal exceeding the noise threshold does the state machine conclude that the manual override session is definitively over. The system then cleanly unlocks, seamlessly reverting the control hierarchy back to listening to the asynchronous UART stream for new trajectories.

5.5 The Digital Twin and High-Level Middleware

The embedded C firmware detailed in the preceding sections guarantees the physical safety, determinism and hard real-time execution of the robotic arm's low-level kinematics. However, an isolated microcontroller lacks the computational bandwidth to provide complex spatial visualization or remote, high-level programmability. To transform the bare-metal hardware into a modern, accessible didactic platform, the control architecture transitions from the embedded STM32 domain to a high-level host PC domain.

This high-level ecosystem is designed to act as a Digital Twin: a real-time, bidirectional virtual representation of the physical mechatronic plant. The architecture is structurally decoupled into three primary components: a specialized telemetry protocol, an asynchronous Python middleware bridge and a WebGL-based browser visualization.

5.5.1 Twin-Mode Telemetry and Dual-Frequency Protocol

To maintain precise spatial synchronization between the physical robot and its digital counterpart, the STM32 must continuously broadcast its internal kinematic state back to the host PC via the USART interface. As established in section 5.2.1, the embedded firmware supports a verbose Debug Mode capable of outputting detailed mathematical states (position, velocity, acceleration, PWM) for a single targeted actuator. However, attempting to scale this human-readable debugging format to broadcast the states of all 12 actuators simultaneously would instantly saturate the 115200 bps serial bus, causing buffer overflows, severe latency and critical data loss.

To resolve this bandwidth bottleneck, when the system is explicitly commanded into its dedicated Twin Mode (`TELEMETRY_MODE_TWIN`), the outbound serial communication completely suppresses the standard debug strings. Instead, it shifts into a highly optimized, machine-readable Dual-Frequency Telemetry Protocol. This protocol guarantees the Digital Twin receives all necessary information without overwhelming the serial pipeline by segregating data streams based on their required refresh rates:

- **High-Speed Kinematic Loop (50Hz):** The fundamental requirement of the 3D visualizer is the immediate geometric pose of the robot. Therefore, 50 times per second, the STM32 transmits a highly compressed string (identified by a `T:` prefix) containing a simple, comma-separated array of the instantaneous physical angles of all 12 actuators. This 50Hz refresh rate was mathematically

selected as the optimal compromise: it is dense enough to provide seamless, stutter-free interpolation for the graphical rendering in the browser, yet sparse enough to consume only a fraction of the available serial bandwidth.

- **Low-Frequency Parameter Loop (1Hz):** Secondary system parameters (i.e., current set maximum velocities, acceleration limits, absolute physical hard-stops) are critical for configuring the Digital Twin's UI sliders, but they do not alter dynamically during standard motion. To conserve bus capacity, these static or slow-changing kinematic limits are broadcasted independently at a frequency of just 1Hz. This dual-loop architecture ensures the high-level software remains accurately informed of the robot's physical constraints without polluting or delaying the critical high-speed positional stream.

5.5.2 Python Serial-to-WebSocket Bridge

While the STM32 successfully transmits its optimized telemetry via USB-to-Serial, relying on a direct, hardwired serial connection to the browser (e.g., via the Web Serial API) would strictly tether a single operator's PC to the physical robot. To maximize the platform's utility and enable flexible, remote interaction, a middleware layer was introduced to structurally decouple the hardware interface from the graphical front-end.

This routing is achieved via `serial_bridge.py`, a custom, multi-threaded middleware application developed in Python. This bridge acts as the central data broker between the physical mechatronic plant and the graphical user interface.

To manage the continuous, bidirectional flow of data without blocking, the middleware employs an asynchronous architecture (`asyncio`) operating in tandem with the websockets library. A dedicated background daemon thread is spawned solely to interface with the pyserial buffer. It auto-detects the active microcontroller port, reads the incoming `T`: telemetry frames, handles required offset calculations and parses the raw strings into structured JavaScript Object Notation (JSON) payloads.

Once formatted, the asynchronous event loop instantaneously broadcasts these JSON payloads over a local WebSocket (`ws://`) connection to the browser. Crucially, this server-client architecture supports multi-node connectivity. A primary PC can host the physical serial connection while simultaneously allowing multiple students to access the 3D visualizer and control interface from their individual smartphones or laptops via the local Wi-Fi network, significantly expanding the platform's didactic utility. Furthermore, the bridge operates bi-directionally; it listens for incoming JSON command payloads generated by the browser UI and perfectly routes them back down the serial pipeline to the STM32's DMA ring buffers.

5.5.3 WebGL Visualization and URDF Parsing

The terminal node of the software architecture, depicted in figure 5.2, is the graphical front-end (`visualizer.html`). Built entirely in HTML5, CSS and JavaScript, this web application abstracts the underlying mathematical matrices and serial protocols into a highly intuitive, interactive 3D representation of the robotic arm.

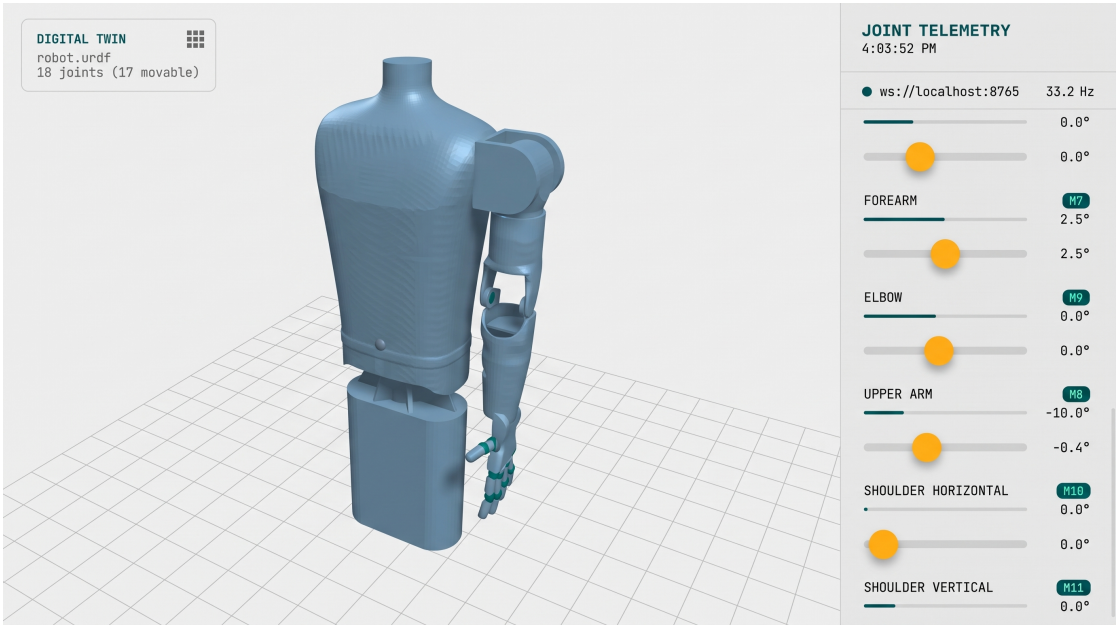


Figure 5.2: Web-based Three.js Digital Twin interface for real-time 3D kinematic visualization and telemetry. Mesh geometry adapted from *PIB – Printable Intelligent Bot: Open-Source 3D-Printable Humanoid Robot.*

To achieve hardware-accelerated 3D graphics directly within a standard web browser, the front-end leverages the `Three.js` WebGL library[6]. However, merely loading generic 3D meshes into a scene is insufficient for robotic simulation; the digital models must accurately reflect the complex parent-child mathematical relationships of the physical joints.

To bridge the gap between Computer-Aided Design (CAD) and software simulation, the specific kinematic geometry of the robotic arm was exported from Onshape into a Universal Robot Description Format (URDF) file (`robot.urdf`). The URDF acts as the mathematical skeleton of the Digital Twin. An URDF parser within the JavaScript front-end reads this XML-based file to automatically construct the kinematic tree, defining exactly where the origin of the wrist rotates relative to the forearm and linking the visual STLs (the 3D-printed PETG meshes) to their respective rotational axes.

During operation, the visualization logic is driven by a continuous WebGL rendering loop. When the WebSocket listener intercepts a 50 Hz telemetry payload from the Python bridge, it extracts the new physical angles and applies them directly to the transformation matrices of the corresponding URDF nodes. This results in perfect, real-time spatial synchronization: as the physical servo motors rotate, the digital meshes precisely track their movement on screen.

Finally, the front-end provides the user with an array of dynamic DOM-based (Document Object Model) touch sliders. When an operator manipulates a digital slider, the JavaScript fires a corresponding target JSON payload through the WebSocket bridge to the STM32. Because of the sophisticated Sync-Lock and noise-hysteresis protocols established in section 5.4.2, the embedded firmware seamlessly arbitrates between these digital browser commands and the physical analog potentiometers, finalizing a highly robust, transparent and bidirectional hybrid-control ecosystem.

Chapter 6

Conclusions and Future Developments

The preceding chapters described the mechanical redesign, fabrication, electronic architecture and software development required to transform the original PIB platform into a customized robotic manipulator. The resulting system integrates modified actuation, custom power and control electronics, embedded real-time control and a WebGL-based Digital Twin within a single experimental platform.

This chapter summarizes the main achievements of the developed system and evaluates its performance against the objectives of the project. Particular attention is given to the improvements introduced in the actuation and control architecture, the resulting dual-mode operation and the practical integration of the mechanical, electrical and software subsystems.

The chapter also discusses the main limitations identified during fabrication and experimental commissioning. These limitations provide the basis for a set of proposed improvements concerning the power architecture, distal transmission system, scalability and future integration of higher-level control and autonomous manipulation capabilities. Finally, the broader educational and open-source contributions of the project are considered, highlighting the practical engineering experience gained through the development and validation of the platform.

6.1 Achievements and System Performance

The primary objective of this project was to re-engineer the open-source PIB robotic manipulator into a robust, deterministic and transparent mechatronic platform. The finalized system successfully integrates custom power electronics, bare-metal embedded control and a real-time 3D Digital Twin, resolving the mechanical fragility, torque deficits and software opacity present in the baseline architecture.

The arm operates reliably across two control schemes: as an autonomous benchtop demonstrator driven by physical potentiometers and as an externally managed system synchronized with a WebGL simulation over serial communication. All primary electromechanical goals were achieved, resulting in a stable, high-resolution platform well-suited for academic demonstration and research.

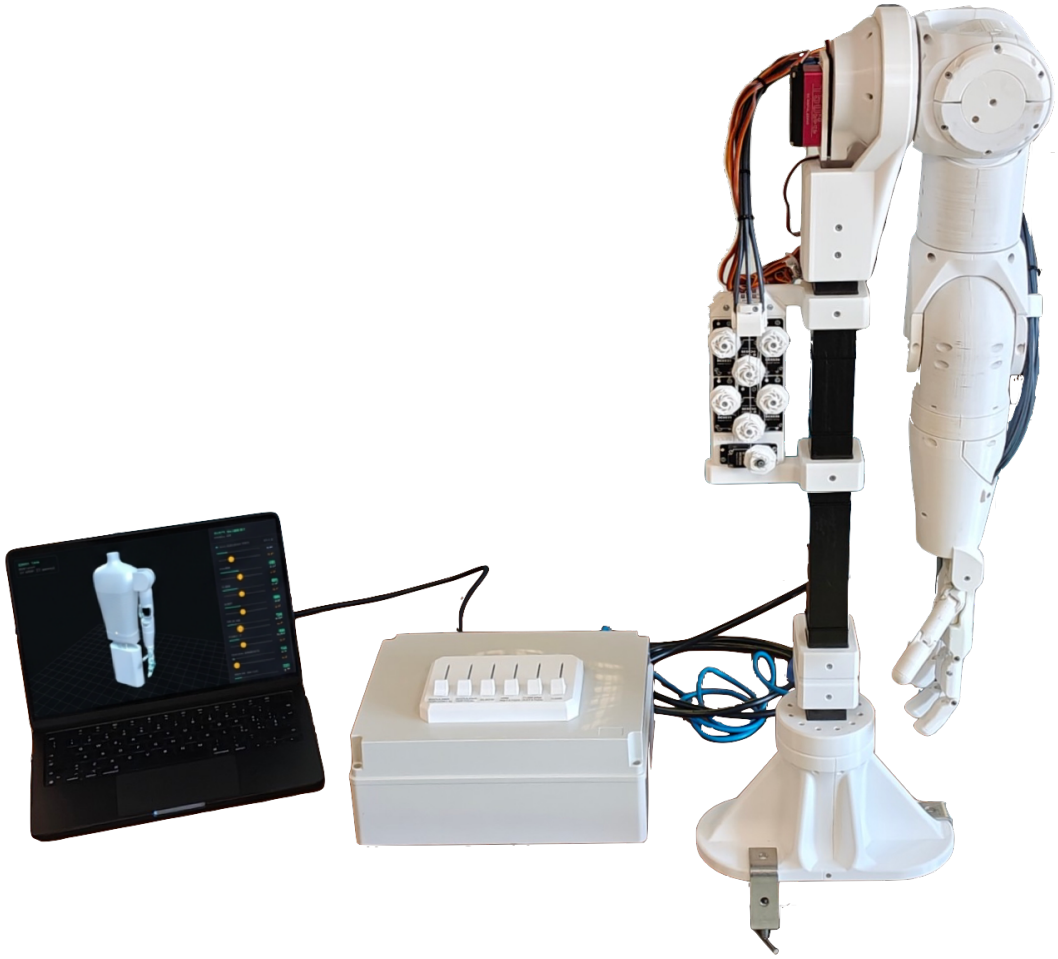


Figure 6.1: Complete experimental setup integrating the Digital Twin GUI, control enclosure, analog HMI and physical manipulator.

6.1.1 Full Mechatronic Integration and Dual-Mode Operation

The physical manipulator features 12 active degrees of freedom (DOF) utilizing high-torque digital servomotors:

- **Primary Articulations:** Driven by DSSERVO DS51150[21] actuators on the shoulder horizontal axis, shoulder vertical axis and elbow. These motors operate at up to 12.6 V, delivering a peak stall torque of 173 kg · cm across an extended 270° physical sweep.

- **Distal and Tendon Articulations:** Driven by DFRobot DC5535[22] coreless actuators for forearm rotation, wrist flexion and the underactuated fingers. These servos provide 35 kg · cm of stall torque at 7.0 V across an operating range of 180°.

A core technical improvement over the baseline system is the optimization of the control frequency. Commercial RC systems typically govern actuators using an abstracted 50 Hz refresh rate ($T = 20.0$ ms). In this project, the control loop was experimentally elevated to 300 Hz ($T \approx 3.33$ ms).

For standard pulse-width commands ranging from 0.5 ms to 2.5 ms, this frequency increase significantly expands the operational duty cycle window ΔD:

$$\Delta D_{50\text{Hz}} = \frac{2.5\text{ ms} - 0.5\text{ ms}}{20.0\text{ ms}} = 10.0\% \quad \longrightarrow \quad \Delta D_{300\text{Hz}} = \frac{2.5\text{ ms} - 0.5\text{ ms}}{3.33\text{ ms}} = 60.0\% \tag{6.1}$$

When mapped across the 12-bit hardware timer of the PCA9685 driver ($2^{12} = 4096$ discrete steps), the usable control range increases from 410 steps (4096×0.10) to 2458 steps (4096×0.60). For the 270° primary joints, this yields a measurable increase in positional resolution:

$$\text{Res}_{50\text{Hz}} = \frac{270^\circ}{410\text{ steps}} \approx 0.66^\circ/\text{step} \quad \longrightarrow \quad \text{Res}_{300\text{Hz}} = \frac{270^\circ}{2458\text{ steps}} \approx 0.11^\circ/\text{step} \tag{6.2}$$

This 600% enhancement in signal discretization eliminates low-speed stepping jitter, producing smooth mechanical trajectories and responsive control. System commissioning and everyday operation follow a consistent, safe operational workflow:

1. **Pre-Power Alignment (Manual Homing):** Commercial hobby servomotors lack absolute proprioceptive feedback and possess no holding torque when unpowered. Because the firmware initializes assuming all joints reside at calibrated neutral positions ($\theta_{\text{current}} = \theta_{\text{home}}$), the operator must place the unpowered arm in its gravity-assisted Home posture before powering the system. In this neutral pose, the arm hangs vertically downward, parallel to the carbon support mast, with the palm facing inward.
2. **Power Deployment:** The arm connects to the control enclosure via the SP21 power umbilical and the shielded RJ45 communication tether. Toggling the illuminated AC switch energizes the internal dual-rail power supplies.
3. **Standalone Manual Mode:** The system boots directly into an operational state governed by the six linear potentiometers on the enclosure lid. To prevent mechanical damage during manual testing, joint dynamics are automatically clamped to conservative limits.
4. **PC-Linked Mode and Digital Twin:** Connecting a computer to the external Mini-USB port establishes a 115200 bps serial bridge with the STM32 microcontroller. Streaming structured JSON commands allows the operator to unlock full kinematic limits and launch the Three.js WebGL Digital Twin, synchronizing the physical arm with its 3D virtual counterpart in real time.

6.1.2 Embedded Architecture and Real-Time Low-Level Control

The proprietary Tinkerforge Servo Bricklet[20] was replaced with a custom embedded controller based on an STM32 microcontroller[24] using the STMicroelectronics Hardware Abstraction Layer (HAL). Implementing the firmware directly in bare-metal C provided determinism, low-latency execution and hardware-level control over the physical plant:

- **Deterministic 300 Hz Kinematic Execution:** A 32-bit hardware timer generates periodic interrupts every $dt = 3.33$ ms, executing the trajectory update routine for all 12 joints simultaneously. This mathematically overrides the internal step-response PID of the hobby servos. The algorithm computes the required stopping distance d_{stop} at each step:

$$d_{\text{stop}} = \frac{v_{\text{current}}^2}{2a_{\max}} \tag{6.3}$$

  State variables are integrated via discrete Euler equations ($v_{k+1} = v_k + a_{\max} \cdot dt$, $\theta_{k+1} = \theta_k + v_k \cdot dt$), producing continuous trapezoidal velocity profiles that protect the 3D-printed linkages from high inertial shocks.

- **Suppression of Asymptotic Limit-Cycling:** Discrete-time trajectory generators can oscillate near zero-velocity targets, causing high-frequency limit-cycling, acoustic buzzing and continuous current draw. The firmware suppresses this by applying dynamic deadbands ($\epsilon_p = 0.3°$, $\epsilon_v = 5.0°/\text{s}$):

$$\text{If } |\theta_{\text{target}} - \theta_{\text{curr}}| < \epsilon_p \quad \text{AND} \quad |v_{\text{curr}}| < \epsilon_v \implies \theta_{\text{curr}} = \theta_{\text{target}}, \quad v_{\text{curr}} = 0$$

  This clamping silences the servomotors once at the setpoint, preventing unnecessary Joule heating and thermal saturation.

- **Non-Blocking Analog Sensing (ADC + DMA):** Polling six 12-bit ADC channels at 300 Hz would consume thousands of clock cycles and disrupt control loop timing. Instead, the internal ADC operates in Continuous Scan Mode paired with Direct Memory Access (DMA). The DMA controller transfers conversions directly into an SRAM ring buffer (`adc_raw_buffer[6]`) without CPU overhead. Wiper noise and rail ripple are mitigated via an Exponential Moving Average (EMA) filter ($\alpha = 0.1$):

$$S_{t,i} = \alpha R_{t,i} + (1 - \alpha) S_{t-1,i} \tag{6.4}$$

- **Asynchronous Communications and Zero-Allocation Parsing:** Inbound serial strings are transferred into an SRAM ring buffer via UART DMA. Upon receiving a newline character, the firmware decodes payloads using `jsmn`[28], a lightweight, zero-allocation JSON parser. This eliminates dynamic memory allocations (`malloc`), preventing heap fragmentation and control stalls.

- **Hardware Controlled Homing:** Because hobby servomotors lack internal holding brakes, cutting DC power during an emergency would cause the arm to collapse under gravity, risking structural fractures. The external emergency pushbutton is wired as an active-low hardware interrupt (EXTI). When triggered, power remains active while the firmware commands a controlled, low-velocity retreat to the calibrated Home position (`ROBOT_HOMING`). The onboard Nucleo button (B1) provides a redundant fallback trigger.

- **Sync-Lock Arbitration:** The firmware dynamically arbitrates between manual potentiometers and serial commands without requiring a physical toggle switch. When manual slider velocity exceeds an empirical threshold ($\Delta > 30$ counts), the system grants exclusive control to the analog input and enforces a 2000 ms hysteresis timeout before allowing serial commands to resume.

6.1.3 Modular Infrastructure, Portability and Didactic Observability

The hardware layout was engineered to decouple power generation from the mechanical arm, providing electrical reliability, quick deployment and open structural observability:

- **Isolated High-Current Power Infrastructure:** High-voltage AC mains wiring is fully enclosed within an IP56 junction box protected by an IEC C14 fused inlet (6 A). Stepping down to DC is handled by two Mean Well SMPS[26] units: a 12.0 V @ 30 A supply for primary joints and a 7.5 V @ 20 A supply for tendon servos. Both units share a common reference ground and incorporate secondary overcurrent protection via 25 A and 20 A automotive blade fuses. Exhaustive testing confirmed the power system operates without thermal distress, handling high transient currents without voltage sag.
- **Split-Rail Decentralized Distribution Node:** To handle theoretical stall currents exceeding 50 A without the cost of a custom PCB, a commercial PCA9685[25] board was physically modified. Factory V+ traces were severed and dual 1.5 mm^2 copper busbars were soldered across the headers, establishing dedicated 7.5 V (Channels 0–11) and 12.0 V (Channels 12–15) power domains. Mounting this node on the mast behind the shoulder keeps PWM signal runs short, eliminating the EMI and inductive crosstalk that would result from routing low-voltage square waves through an umbilical line.
- **Modular Interconnects and Telemetry Redundancy:** The arm detaches completely from the base enclosure using two industrial panel connectors:
  - **Power:** An industrial 3-pole SP21 aviation connector[27] rated at 30 A transmits DC power over a 2.0-meter 3x2.5 mm^2 multi-conductor cable, reducing resistive heating (I^2R_{wire}).
  - **Data:** A shielded RJ45 connector carries the 400 kHz Fast-mode I2C telemetry. Its eight conductors are wired in redundant parallel pairs (2x GND, 2x +3.3 V, 2x SCL, 2x SDA), doubling cross-sectional area and providing hardware fail-over against contact wear.
- **Didactic Transparency and High-Level Extensibility:** Omitting the original closed torso shell leaves all mechatronic subsystems fully exposed. The platform serves as an open visual demonstrator where students can inspect Bowden cable behavior, Capstan friction, split-rail busbars and real-time microcontroller states. Furthermore, because the low-level firmware is dedicated strictly to deterministic hardware abstraction and safety interlocks, high-level control remains completely decoupled. External computational

platforms, such as a Raspberry Pi running ROS2 or an edge-AI computer vision model, can command the arm simply by streaming standard JSON payloads over the serial interface.

6.2 Criticalities and Experimental Limitations

While the upgraded manipulator achieved its core functional requirements, several mechanical, tribological and electrical limitations emerged during fabrication, assembly and dynamic commissioning. These bottlenecks highlight the engineering trade-offs inherent in modifying an open-source, polymer-based robotic framework with high-torque commercial hardware.

6.2.1 Procurement Hurdles and Additive Manufacturing Realities

The transition from the digital design to physical hardware encountered immediate supply chain and fabrication constraints:

- **BOM Extraction and Institutional Sourcing:** Sourcing the complete proprietary hardware kit from pib.rocks was economically unfeasible under the thesis budget. Consequently, a comprehensive Bill of Materials (BOM) was compiled from raw CAD assemblies. University procurement regulations prohibited ordering from consumer-oriented or non-EU platforms (such as AliExpress or Amazon), where motor packs and fastener kits were available at low cost. Procurement was restricted to certified distributors, primarily RS Components for mechanical hardware and filament and DigiKey for semiconductors and actuators. Administrative approval cycles and staggered distributor lead times resulted in asynchronous deliveries, intermittently stalling physical assembly.
- **Polymer Additive Manufacturing (PETG vs. PLA):** White Verbatim PETG[23] was selected over standard PLA to exploit its higher ductility, impact resistance (8.1 kJ/m^2) and thermal stability ($T_g \approx 80$°C). However, PETG proved significantly more demanding to process. Due to its hygroscopic nature, strict dehydration protocols were mandatory to prevent hydrolysis-induced delamination and nozzle boiling. Achieving acceptable layer adhesion and dimensional tolerance (± 0.05 mm) required running the FDM platforms at conservative speeds (30 mm/s for exterior perimeters) and optimizing retraction dynamics to minimize stringing. Despite these manufacturing penalties, PETG remained the correct structural choice to withstand the dynamic shear forces exerted by the upgraded servomotors.

- **Mechanical Post-Processing and CAD Redesign:** Extensive manual post-processing—including reaming shaft bores, deburring mating surfaces and thermal heat-setting of M3 nuts—was necessary to resolve spatial interferences. Furthermore, several CAD components were redesigned: non-standard mounting brackets were developed to fit the 40x40 mm square carbon fiber support mast and internal cavities were modified to house the larger form factors of the DS51150 and DC5535 servomotors.

6.2.2 Distal Transmission Nonlinearities and Joint Flexibility

The remote tendon transmission system governing the wrist and the underactuated hand introduced kinematic nonlinearities that degrade distal repeatability:

- **The Underactuated Hand Dilemma:** Distal digit extension relies entirely on passive return springs rather than active antagonistic cabling. In the initial build, stiff return springs were installed to enforce sharp extension. However, in extreme arm postures where the cumulative sheath curvature was large, the resulting friction increase caused the DC5535 servomotors to stall before completing finger flexion. Replacing them with softer springs eliminated the stall current and restored full flexion, but introduced dynamic underdamping: during rapid arm accelerations, the fingers exhibit visible structural flutter and kinematic floppiness.
- **Wrist Articulation Impedance:** Wrist flexion is similarly underactuated and spring-loaded. Six rigid PTFE Bowden sheaths (one per finger, two for the thumb) traverse the articulating wrist joint. The mechanical stiffness of this parallel tube bundle exerts high passive resistance against the joint. Although high-torque coreless servos were installed, the wrist's physical range of motion remains severely restricted. When the actuator pulls at peak torque, the internal nylon tendon elongates elastically under the combined loads of Capstan sheath friction and tube bending resistance. Because wrist movement is heavily constrained, this axis was omitted from the primary six-slider manual HMI panel.

6.2.3 Dynamic Deceleration and Inductive Back-EMF Overvoltage

The physical manipulator possesses considerable cantilevered mass and rotational inertia, concentrated predominantly at the proximal shoulder yaw and pitch axes. During aggressive trajectory reversals or sudden braking maneuvers, the high-torque DSSERVO DS51150 actuators undergo regenerative braking, transitioning

from electrical loads into generators. The generated back-electromotive force (back-EMF), governed by the electromechanical relation $e(t) = K_e \dot{\theta}(t)$, pumps substantial reverse transient currents back through the internal H-bridge diodes toward the DC distribution bus.

The power delivery stage relies on Mean Well switching mode power supplies (SMPS) designed exclusively for unidirectional current sourcing. Because these industrial converters lack internal dynamic braking circuitry or active reverse-current dissipation stages, they cannot absorb this regenerative energy. Consequently, the unabsorbed reverse current rapidly charges the output filter capacitors, causing the DC bus potential to surge past 14.0 V on the nominal 12.0 V rail. This voltage spike instantly trips the internal Over-Voltage Protection (OVP) comparator of the Mean Well SMPS, triggering a complete hardware shutdown. The resulting power cutout eliminates all holding torque, allowing the manipulator to drop under gravity and risking structural fracture.

To stabilize the system during dynamic operations, this issue was temporarily mitigated within the embedded firmware. By enforcing conservative software caps on the maximum operational velocity ($v_{\max}$) and acceleration ($a_{\max}$) parameters inside the trapezoidal trajectory generator, the magnitude of joint deceleration is strictly bounded. This constraint prevents the back-EMF spikes from reaching the OVP activation threshold, ensuring electrical stability during maneuvering at the expense of peak dynamic speed.

6.3 Proposed Solutions and Future Improvements

The experimental characterization of the manipulator highlighted several mechanical, electrical and control bottlenecks that constrain dynamic agility and distal dexterity. To advance the platform from an open-loop didactic demonstrator to a fully capable autonomous manipulation testbed, a structured roadmap of hardware and software improvements is established.

6.3.1 Active Power Regeneration and Electrical Protection

The over-voltage protection (OVP) cutouts triggered by regenerative back-EMF during aggressive joint braking currently require conservative software clamping of maximum velocities ($v_{\max}$) and accelerations ($a_{\max}$) within the trajectory generator. Because industrial switching mode power supplies (SMPS) are unidirectional converters that cannot sink reverse current, a dedicated hardware-level energy dissipation circuit is required to operate the arm at its full dynamic capability.

The proposed solution is an active dynamic braking circuit (brake chopper) integrated directly across the 12.0 V bus inside the IP56 enclosure. In this topology, an aluminum-housed power shunt resistor (R_{brake}) is placed in series with a low-side, logic-level N-channel power MOSFET between the 12.0 V DC bus and ground. When the DC bus voltage surges during joint deceleration, an analog comparator switches its output high, driving the MOSFET into saturation. This immediately shunts the excess reverse current from the bus through R_{brake} to ground, dissipating the kinetic energy of the decelerating arm as heat before it can charge the SMPS output capacitors.

The braking resistor is sized at $R_{\text{brake}} = 1.0\ \Omega$ with a continuous power rating of 100 W. Under peak transient conditions at 13.0 V, the instantaneous electrical parameters are:

$$I_{\text{peak}} = \frac{V_{\text{bus}}}{R_{\text{brake}}} = \frac{13.0\ \text{V}}{1.0\ \Omega} = 13.0\ \text{A} \tag{6.5}$$

$$P_{\text{peak}} = \frac{V_{\text{bus}}^2}{R_{\text{brake}}} = \frac{(13.0\ \text{V})^2}{1.0\ \Omega} = 169.0\ \text{W} \tag{6.6}$$

Although the peak instantaneous power of 169.0 W exceeds the continuous 100 W rating of the resistor, braking events in this manipulator occur in short transient bursts lasting tens to hundreds of milliseconds. The thermal mass of a wirewound, aluminum-clad power resistor easily absorbs these short thermal pulses without exceeding safe operating temperatures.

To govern the gate drive cleanly and eliminate switching chatter around the tripping boundary, the comparator is wired in a non-simmetrical inverting Schmitt-trigger configuration with positive feedback[7] as shown in figure 6.2. The circuit enforces an upper threshold of $V_{\text{high}} = 13.0$ V and a lower threshold of $V_{\text{low}} = 12.5$ V, establishing a hysteresis band of $\Delta V = 0.5$ V. When back-EMF elevates the bus potential to 13.0 V, the MOSFET conducts, pulling the bus down until it crosses 12.5 V, at which point the gate drive turns off. This bounds the transient overvoltage strictly between 12.5 V and 13.0 V, safely below the ≈ 14.0 V cutoff of the Mean Well SMPS and eliminating false OVP shutdowns.

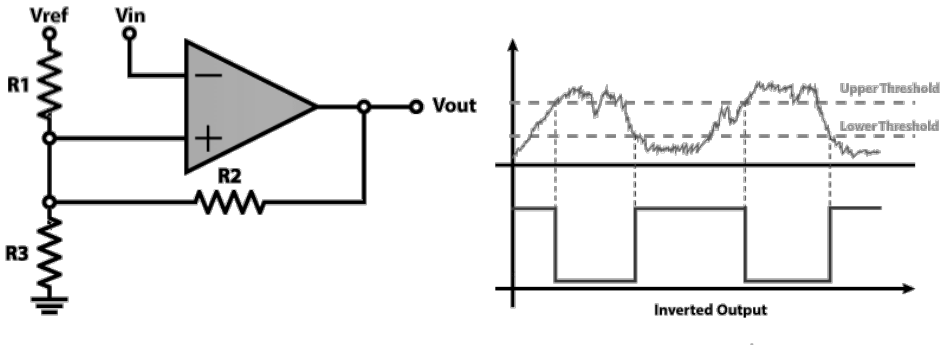


Figure 6.2: Asymmetric inverting Schmitt trigger circuit and simulated voltage response. Adapted from the book *Microelectronic Circuits*.

Because the Schmitt-trigger comparator intrinsically requires a stable reference voltage at its input to establish its upper and lower switching thresholds, a reference potential must be provided. To minimize component count, board footprint, and circuit complexity, the design avoids adding dedicated external voltage reference ICs, precision Zener diodes or auxiliary step-down regulators. Instead, it directly leverages the pre-existing regulated DC rails within the enclosure. The circuit can utilize the secondary 7.5 V Mean Well supply or, more advantageously, the regulated +5.0 V rail provided by the onboard linear regulator of the STM32 Nucleo. Connecting an external electrolytic capacitor across this +5.0 V line buffers it against transient fluctuations. Because the Nucleo is powered by the 7.5 V SMPS, the input voltage to the microcontroller board would have to drop below approximately 6.0 V before its internal regulator drops out. This provides a stable reference potential for the comparator divider network with minimal external components.

An alternative approach is replacing or supplementing the benchtop power supply with a hybrid battery-buffered architecture. Paralleling a 4S Lithium Iron Phosphate ($LiFePO_4$) pack (nominal 12.8 V) across the primary DC bus allows reverse back-EMF currents to be absorbed directly through charging. Small-format lithium batteries exhibit high charge and discharge C-ratings in compact volumes, handling dynamic stall currents and regenerative spikes that exceed the continuous capacity of standard power bricks. In addition, this configuration provides tether-free operation for mobile laboratory demonstrations, requiring only a compact desktop supply or low-current trickle charger to maintain battery state of charge.

6.3.2 Mechanical Redesign of Distal Articulations

The mechanical compliance and friction losses observed in the distal kinematics require material and routing modifications to improve motion fidelity and range of motion:

- **Low-Compliance Tendon Materials:** The flexible nylon monofilament currently installed exhibits significant elastic elongation under sustained tensile loads. According to Hooke's Law:

$$\Delta L = \frac{F_{\text{in}} \cdot L_0}{E \cdot A} \tag{6.7}$$

  Where E is the Young's modulus and A is the cross-sectional area. While nylon has an elastic modulus of $E \approx 2$ to 4 GPa, braided Ultra-High-Molecular-Weight Polyethylene (UHMWPE/Dyneema) achieves $E \approx 100$ to 120 GPa. Replacing the nylon lines with UHMWPE increases longitudinal stiffness by a factor of thirty, directly transferring servo displacement to the joints without lost motion from tendon stretch.

- **Decoupled Wrist Routing:** The six rigid PTFE Bowden sheaths routed through the wrist joint exert significant mechanical resistance, severely restricting its range of motion. To eliminate this bending resistance, future iterations should terminate the outer PTFE sheaths at the distal end of the forearm link. From that boundary, the bare UHMWPE tendons can traverse the articulating wrist through low-friction polished ceramic or stainless-steel eyelet guides. Decoupling the rigid outer sheaths from the joint pivot eliminates sheath bending stiffness across the articulation.

- **Visco-Elastic Damping for Phalange Underactuation:** The trade-off between return-spring stiffness and structural flutter must be addressed. Low-stiffness springs resolve servo stall caused by Capstan friction but result in underdamped finger flutter during arm movements. The finger joints should incorporate miniature visco-elastic elastomer inserts or silicone torsion dampers placed in parallel with progressive-rate return springs. This setup provides velocity-dependent damping to eliminate dynamic wobbling without imposing high static resistance at full flexion.

6.3.3 Scalability: Bilateral Expansion and Multi-Driver Support

The engineering foundation developed for the single-arm prototype provides a validated framework for scaling the platform into a dual-arm manipulator or a complete upper-torso humanoid.

The logistical and manufacturing setup is fully established. The complete Bill of Materials (BOM), mechanical fastener schedules, FDM printing profiles and procurement channels (RS Components and DigiKey) are finalized and documented. Constructing the contralateral (right) arm avoids the exploratory iterations required during the initial build. Furthermore, the physical left arm provides a reference model for resolving assembly tolerances, bearing press-fits and cable routing.

From an electrical perspective, multi-limb scaling is supported by the I2C bus topology. The PCA9685 PWM controller[25] includes address selection solder pads (A_0 through A_5), allowing up to 62 unique modules to coexist on a single bus. By bridging the A_0 address pad on a second shoulder-mounted PCA9685, its hardware address shifts from 0x40 to 0x41. Both boards can then share the existing Fast-mode 400 kHz I2C telemetry lines routed through the shielded CAT6 RJ45 connection.

The embedded processing hardware is estimated to be fully capable of driving all 24 actuators simultaneously. With the core control loop running deterministically at 300 Hz ($dt = 3.33$ ms), the computational overhead of the trajectory generation and safety routines is low. Scaling to 24 actuators requires minor firmware modifications to duplicate joint definitions, state tracking variables and limit checks. Because UART reception, JSON tokenization and ADC conversions execute via DMA channels and hardware interrupts without CPU blocking, I estimate that the STM32 microcontroller can update 24 independent actuators well within the 3.33 ms control period.

6.3.4 High-Level Control, Edge AI and Autonomous Manipulation

The low-level STM32 firmware is intentionally limited to hardware abstraction, physical protection and real-time motion generation. Its core role is to prevent mechanical damage by calculating smooth trapezoidal acceleration profiles, arresting excessive jerk that could fracture 3D-printed linkages and enforcing hard-coded joint limits to prevent mechanical collisions. By executing these safety functions on bare-metal silicon, the firmware completely replaces the closed Tinkerforge Servo Bricklet while keeping high-level computational tasks decoupled from real-time motor driving.

Because external communication relies on an asynchronous JSON-over-serial protocol, any high-level computing environment can control the manipulator simply by streaming formatted strings over USB serial. This structural decoupling allows

external controllers like standard desktop workstations and single-board computers (such as a Raspberry Pi 5) or dedicated edge accelerators (such as an NVIDIA Jetson Orin Nano) to act as high-level masters without requiring firmware modifications.

This architecture provides a scalable path for advanced robotics research:

- **Robotic Middleware and Motion Planning:** The system can be directly integrated into the ROS2 (Robot Operating System) ecosystem. Kinematic solvers within frameworks such as MoveIt can compute numerical Inverse Kinematics (IK), Cartesian toolpath generation and multi-joint collision avoidance. The resulting joint-space trajectories can be published directly to a ROS2 serial bridge node that translates them into standard JSON strings (`{"m":ID, "p":Deg, "v":V, "a":A}`) and sends them to the STM32 DMA ring buffer at runtime.

- **Computer Vision and Local AI Pipelines:** The addition of an RGB-D camera (such as an Intel RealSense or Luxonis OAK-D) enables vision-guided manipulation. Lightweight vision models (e.g., YOLOv8 for object localization and classification or lightweight convolutional networks for 6-DOF grasp-pose detection) can execute locally on edge AI hardware. The vision node detects target coordinates, computes the required spatial grasp and commands the arm through the serial interface.

- **Human-Robot Interaction (HRI):** Edge AI platforms can process optical flow, facial tracking or skeletal pose estimation (e.g., using MediaPipe) to implement real-time gesture following and collaborative tasks. Because low-level joint acceleration and velocity limits are enforced on the STM32, erratic or noisy high-level vision predictions cannot drive the arm into self-destructive motions or violent snap transitions.

- **Reinforcement Learning and Behavioral Cloning:** The combination of the physical arm and the WebGL Digital Twin provides an experimental framework for sim-to-real transfer. Policies trained in simulation can be deployed directly to the hardware, using the embedded Dynamic Deadbands and trapezoidal limits as a baseline safety envelope.

This division of responsibilities highlights the didactic and experimental value of the platform. Mechanical, electrical and mechatronic students can study the exposed physical hardware, analyzing Bowden cable Capstan friction, split-rail busbar power distribution and bare-metal HAL peripherals. Concurrently, computer science and automation students can deploy advanced AI and control pipelines at the high-level layer without needing to manage low-level embedded hardware or risk damaging the mechanical structure.

6.4 Educational Reflections and Professional Takeaways

Developing this robotic manipulator served as a comprehensive mechatronic capstone, bridging theoretical control theory, mechanical engineering, power electronics and embedded software into a single physical system. Beyond producing a functional 12-DOF manipulator, the project provided valuable experience in troubleshooting real-world engineering constraints, managing university laboratory resources and contributing directly to the global robotics community.

6.4.1 Comprehensive Mechatronic Competencies

The project required a 360-degree mechatronic workflow spanning three core engineering disciplines:

- **Mechanical Engineering and Additive Manufacturing:** Sourcing constraints required extracting a bottom-up Bill of Materials (BOM) directly from raw CAD models and adapting parts to institutional suppliers (RS Components and DigiKey). Managing the manufacturing cycle involved tuning an FDM platform, enforce material dehydration protocols to prevent hydrolysis and fine-tuning of the toolpaths using Gyroid TPMS infill for isotropic strength. Mechanical commissioning developed practical competencies in dimensional tolerancing, manual post-processing, bearing press-fitting and embedding captive M3 fasteners via thermal insertion into the polymer matrix.
- **Electrical and Power Systems Engineering:** Delivering safe, high-current DC power to heterogeneous digital actuators required designing a dedicated power architecture. Practical skills included sizing industrial switching power supplies (Mean Well 12.0 V @ 30 A and 7.5 V @ 20 A), integrating multi-stage AC and DC circuit protections, assembling high-current aviation umbilicals (SP21) and configuring noise-shielded, dual-redundant I2C lines over CAT6 Ethernet. In addition, physically modifying commercial PCA9685 boards with soldered 1.5 mm^2 solid copper busbars provided practical experience in mitigating trace Joule heating and resolving spatial Electromagnetic Interference (EMI).
- **Embedded Firmware and Software Architecture:** Replacing commercial black-box controllers required designing a bare-metal embedded system using the STM32 Hardware Abstraction Layer (HAL) in C. Developing the firmware provided experience in configuring hardware timers for deterministic 300 Hz interrupt loops, setting up non-blocking Direct Memory Access (DMA) for six-channel Continuous Scan ADC acquisition and handling asynchronous UART

reception. Kinematic programming reinforced control concepts through the implementation of real-time Euler integration for trapezoidal velocity curves, dynamic deadbands and priority state machines (Sync-Lock). At the high-level layer, developing the Python WebSocket bridge and the Three.js WebGL Digital Twin provided hands-on experience in full-stack robotic teleoperation and URDF kinematics.

- **Laboratory Instrumentation and Hardware-in-the-Loop Testing:** The project provided extensive hands-on experience with professional laboratory equipment. A digital storage oscilloscope was used to evaluate signal integrity, characterize PWM edge rise times and identify inductive noise spikes across the power bus. An independent laboratory function generator was utilized to empirically sweep and calibrate actuator angular limits prior to firmware integration, while benchtop supplies and soldering stations were employed to assemble custom busbars and sensor harnesses safely.

6.4.2 Contributions to the Open-Source Humanoid Community

A significant outcome of this thesis was the active collaboration established with the international open-source development team behind the PIB humanoid platform.

Because the baseline PIB CAD models were originally designed around the assumption of a full humanoid torso and standard commercial servos, fabricating an isolated single arm revealed several unaddressed mechanical interferences and structural weaknesses. Throughout the assembly and testing phases, detailed technical feedback—including CAD revisions for structural arm brackets, fastener clearance optimizations and reports on Bowden tube routing friction—was documented and communicated to the core development team.

Several of these design recommendations and structural revisions were accepted and directly integrated into the newly released upstream version of the PIB robot. Contributing physical validation data back to an international open-source robotics project demonstrated how student research can directly impact broader engineering communities.

Naturally, as with any complex mechatronic system operating at the intersection of polymer additive manufacturing, tendon transmissions and commercial off-the-shelf actuators, certain physical trade-offs and transmission non-linearities remain inherent to the platform's mechanical reality. Completely eliminating challenges

such as Bowden Capstan friction, distal joint compliance or regenerative back-EMF would require cost-prohibitive industrial hardware—such as custom cycloidal drives, machined alloys and expensive multi-quadrant motor controllers—which fundamentally diverges from the core objective of this research.

When evaluated within the framework of its strict budgetary constraints, the reliance on rapid desktop FDM fabrication and the condensed academic development timeline, the level of performance, responsiveness and deterministic control achieved is remarkably compelling. The finalized manipulator is not intended to be a commercial product, but rather a tangible, highly transparent and accessible mechatronic testbed. It successfully bridges theoretical kinematic models with physical execution, demonstrating that intelligent embedded architecture and targeted mechanical redesign can elevate an open-source educational project into a robust, observable platform for future students and researchers.

Bibliography

Foundational Textbooks and Monographs

[1] J. L. Meriam, L. G. Kraige, and J. N. Bolton. *Engineering Mechanics: Statics*. Ninth. Hoboken, NJ: John Wiley & Sons, 2018.

[2] I. Gibson, D. Rosen, B. Stucker, and M. Khorasani. *Additive Manufacturing Technologies*. Third. Cham, Switzerland: Springer, 2021.

[3] C. Noviello. *Mastering STM32*. Second. Leanpub, 2022.

[4] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: Modelling, Planning and Control*. London, UK: Springer-Verlag, 2009.

[5] W. Bolton. *Mechatronics: Electronic Control Systems in Mechanical and Electrical Engineering*. Seventh. Harlow, UK: Pearson, 2019.

[6] J. Dirksen. *Learn Three.js: Programming 3D Animations and Visualizations for the Web with HTML5 and WebGL*. Fourth. Birmingham, UK: Packt Publishing, 2023.

[7] A. S. Sedra, K. C. Smith, T. C. Carusone, and V. Gaudet. *Microelectronic Circuits*. Eighth. New York, NY: Oxford University Press, 2020.

Peer-Reviewed Journal and Conference Papers

[8] S. Saeedvand, H. S. Aghdasi, and M. Baltaci. «A Comprehensive Survey on Humanoid Robot Development». In: *Robotica* 38.10 (Oct. 2020), pp. 1756–1785.

[9] M. Lapeyre, P. Rouanet, J. Grizou, S. Nguyen, F. Depraetre, A. Le Falher, and P.-Y. Oudeyer. «POPPY: Open-source 3D Printed Humanoid Robot Platform». In: *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Chicago, IL, Sept. 2014, pp. 2194–2200.

[10] G. Palli, C. Melchiorri, and T. Vassura. «A Miniature Tendon-Sheath Transmission System for Robotic Applications: Modeling and Control». In: *IEEE/ASME Trans. Mechatronics* 16.5 (Oct. 2011), pp. 814–825.

Open-Source Robotic Platforms and Repositories

[11] BCN3D Technologies. *BCN3D Moveo: Open-Source 3D-Printed Robotic Arm*. 2016. URL: https://github.com/bcn3d/bcn3d-moveo.

[12] Á. L. M. Palmero. *Thor: 3D-Printable Open-Source 6-DOF Robotic Arm*. 2016. URL: https://hackaday.io/project/12989-thor.

[13] *OpenManipulator-X Modular Robotic Arm Documentation*. ROBOTIS Co., Ltd. Seoul, South Korea, 2020.

[14] G. Langevin. *InMoov: Open-Source Life-Size 3D Printed Humanoid Robot*. 2024. URL: https://inmoov.fr/.

[15] D. Nedelkovski. *How Cycloidal Drives Work and Designing 3D-Printed Reducers*. 2021. URL: https://howtomechatronics.com/.

[16] isento GmbH. *PIB – Printable Intelligent Bot: Open-Source 3D-Printable Humanoid Robot*. 2023. URL: https://pib.rocks/.

Technical Datasheets, Hardware Manuals and Libraries

[17] *MG996R High Torque Metal Gear Dual Ball Bearing Servo Specification Sheet*. TowerPro Inc. 2019.

[18] *DS3225 25kg High Torque Metal Gear Digital Waterproof Servo Datasheet*. DSSERVO (Dongguan Desheng Intelligent Tech Co., Ltd.) 2020.

[19] *DS5180 SSG Large Format High Torque Digital Servo Specification Sheet*. DSSERVO (Dongguan Desheng Intelligent Tech Co., Ltd.) 2020.

[20] *Servo Bricklet 2.0 Hardware Documentation and Technical Specifications*. Tinkerforge GmbH. Schloß Holte-Stukenbrock, Germany, 2021.

[21] *DS51150-12V 150kg/173kg Large Metal Gear Digital Servo Motor Technical Datasheet*. DSSERVO (Dongguan Desheng Intelligent Tech Co., Ltd.) 2022.

[22] *35KG High Torque Metal Waterproof Servo (SER0060 / DC5535) Technical Datasheet*. DFRobot. 2022.

[23] *Verbatim White PETG 1.75mm Technical Data Sheet (TDS)*. Verbatim GmbH. Eschborn, Germany, 2022.

[24] *STM32 Nucleo-64 Boards (UM1724)*. STMicroelectronics. Geneva, Switzerland, 2020.

[25] *PCA9685: 16-channel, 12-bit PWM Fm+ I2C-bus LED Controller*. NXP Semiconductors. Apr. 2015.

[26] *LRS-350 and LRS-150 Series Enclosed Switching Power Supply Datasheets*. Mean Well Enterprises Co., Ltd. New Taipei City, Taiwan, 2021.

[27] *SP21 Series IP68/IP56 Circular Aviation Connector Technical Specifications*. Weipu Electric Appliance Co., Ltd. Guangdong, China, 2021.

[28] S. Banković. *jsmn: A Minimalist JSON Parser in C*. 2021. URL: https://github.com/zserge/jsmn.

Additive Manufacturing Calibration and Guides

[29] Prusa Research. *Material Guide: Polyethylene Terephthalate Glycol (PETG) and Moisture Dehydration Protocols*. 2023. URL: https://help.prusa3d.com/.

[30] *Anycubic Mega S 3D Printer User Manual and Firmware Calibration Guide*. Shenzhen Anycubic Technology Co., Ltd. Shenzhen, China, 2020.

[31] *Raise3D Pro3 Series Technical Specifications and Operation Manual*. Raise3D Technologies, Inc. Irvine, CA, 2021.

[32] Prusa Research. *Printables Open-Source 3D Model Repository*. 2023. URL: https://www.printables.com/.

Declaration on the Use of Generative AI

During the preparation of this thesis, the author used generative AI models exclusively for language editing, grammatical correction and improvement of the readability and clarity of the text. The technical content presented in this work, including designs, hardware implementations, experimental testing, kinematic models and analytical conclusions, is based on the author's own work and on prior work documented and appropriately referenced in the bibliography.