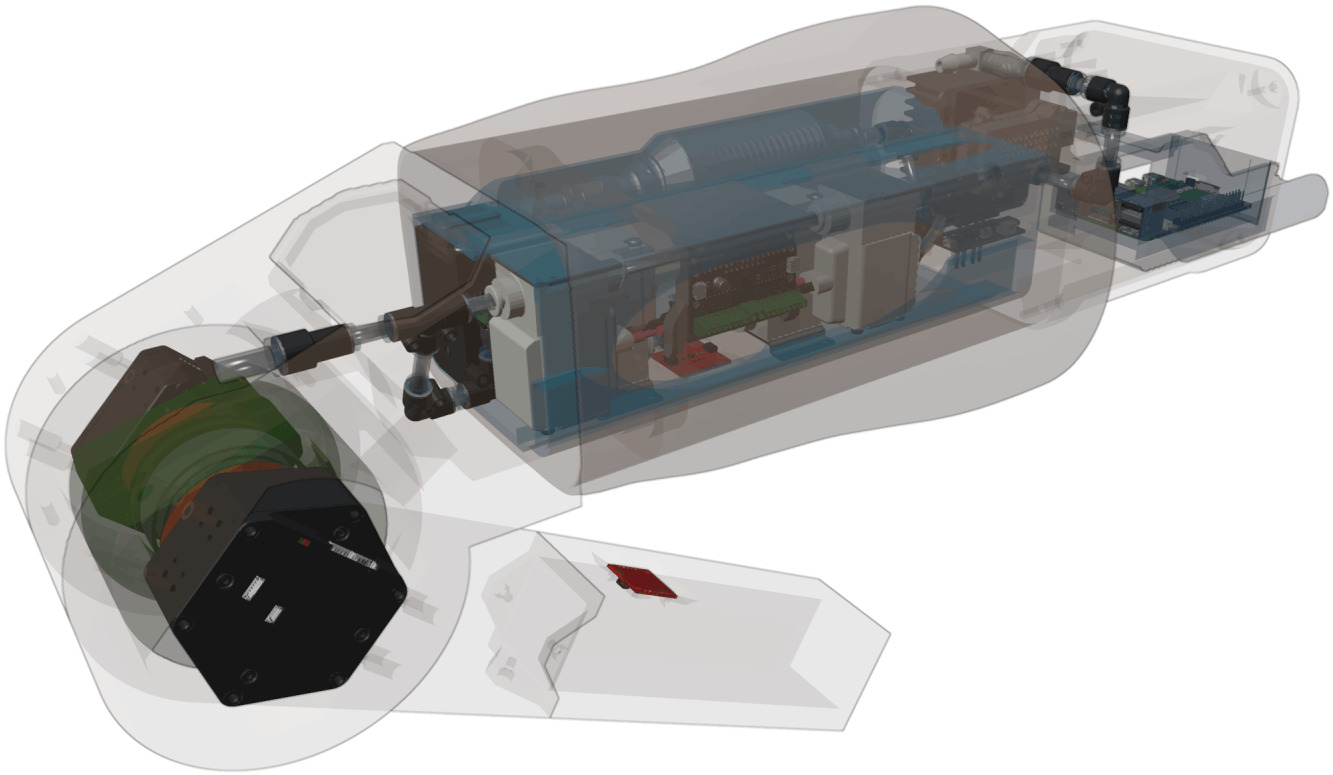
CHALMERS
UNIVERSITY OF TECHNOLOGY



Robotic Phantom Knee and Digital Twin Platform for Early-Stage Exoskeleton Validation

Nonlinear Stiffness and Tuneable Soft-Tissue Compliance

Master's thesis in Biomedical Engineering

WAJIH HABRAH

Department of Electrical Engineering

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2026
www.chalmers.se

Master's thesis 2026

Robotic Phantom Knee and Digital Twin Platform for Early-Stage Exoskeleton Validation

Nonlinear Stiffness and Tuneable Soft-Tissue Compliance

WAJIH HABRAH

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
Gothenburg, Sweden 2026

Robotic Phantom Knee and Digital Twin Platform for Early-Stage Exoskeleton Validation
Nonlinear Stiffness and Tuneable Soft-Tissue Compliance
Wajih Habrah

© Wajih Habrah, 2026.

Supervisor: Emmanuel Dean, Chalmers University of Technology
Examiner: Emmanuel Dean, Chalmers University of Technology

Master's Thesis 2026
Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2026

Abstract

Testing wearable exoskeletons and rehabilitation devices directly on human subjects is often expensive, time-consuming, and ethically challenging, especially during early-stage development when designs and control strategies are frequently updated. In addition, human trials introduce variability that complicates systematic benchmarking and controller verification. This thesis presents the design and development of a robotic phantom knee (RPK) testbed and a real-time Unity-based digital twin (DT) intended to provide a controlled bench-top platform for evaluating exoskeleton controllers.

The developed system combines a single-degree-of-freedom knee hinge with antagonistic actuation using two Koala BEAR linear actuators coupled through an elastic linkage to shape the joint's passive resistance. To emulate tuneable soft-tissue and interface compliance, the platform integrates inflatable chambers whose stiffness is adjusted via closed-loop pressure regulation using solenoid valves, a pump and reservoir, and pressure sensing. A modular ROS 2 (Jazzy) software stack on a Raspberry Pi 5 handles hardware communication, pressure control, and safety functions (including E-STOP and release-air). A Unity-based DT connects via ROS–TCP to visualize system state in real time and support interactive parameter tuning and experiment execution.

Subsystem validation demonstrated stable ROS–TCP streaming, safety handling (E-STOP and release-air), and closed-loop pressure tracking suitable for bench-top trials. The platform supports systematic controller development and validation before human testing by enabling controlled experiments under configurable compliance conditions.

Keywords: exoskeleton, robotic phantom knee, variable stiffness, antagonistic actuation, pneumatics, soft-tissue compliance, ROS 2, digital twin, Unity.

Acknowledgements

I would like to express my sincere gratitude to Emmanuel Dean, my supervisor and examiner, for his guidance, feedback, and support throughout this thesis work. His expertise and encouragement were invaluable in shaping both the technical direction and the final outcome of the project.
I also thank the Department of Electrical Engineering at Chalmers University of Technology for providing the resources and environment necessary to develop and test the Robotic Phantom Knee platform and its digital twin. Finally, I am grateful to everyone who contributed with discussions, practical support, and encouragement during the work.

Dedication

To my beloved mother, from whom I learned resilience in the face of hardship. Without her—after God's guidance—I would not have found this path, nor would I have had the strength to continue. I dedicate this work to her as a modest expression of my gratitude and love, knowing how deeply she values my success and academic achievements. She has been my greatest source of support throughout my academic journey. My caring mother, a wellspring of compassion and an endless source of giving.

To my brother, who has been a constant source of support throughout the different stages of my life.

To my sister, whose dedication, patience, and perseverance have been a source of inspiration to me.

Wajih Habrah, Gothenburg, January 2026

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

ABP	Amplified Basic Pressure (Honeywell pressure sensor series)
AOT	Ahead-of-Time (compilation)
CAD	Computer-Aided Design
DC	Direct Current
DoF	Degree of Freedom
DT	Digital Twin
E-STOP	Emergency Stop
ESP32	Espressif 32-bit microcontroller family
I^2C	Inter-Integrated Circuit
IMU	Inertial Measurement Unit
JSON	JavaScript Object Notation
KB	Koala BEAR (secondary actuator)
KBMB	Koala BEAR Muscle Build (primary actuator)
LOD	Level of Detail
LTS	Long-Term Support
NURBS	Non-Uniform Rational B-Splines
PID	Proportional–Integral–Derivative
PSU	Power Supply Unit
PWM	Pulse-Width Modulation
ROS	Robot Operating System
ROS–TCP	ROS over Transmission Control Protocol
ROS 2	Robot Operating System 2 (ROS version used in this work)
RPK	Robotic Phantom Knee
SEA	Series Elastic Actuator
SDK	Software Development Kit
SPI	Serial Peripheral Interface
UI	User Interface
USB	Universal Serial Bus
USB2BEAR	Westwood Robotics USB-to-BEAR host interface dongle
VIA	Variable Impedance Actuator
VSA	Variable Stiffness Actuator

Nomenclature

Below is the nomenclature of indices, sets, parameters, and variables that have been used throughout this thesis.

Parameters

r	Effective support radius of the spring mechanism
k_t	Torsional stiffness of the spring (linearized over operating range)
k	Equivalent linear stiffness at radius r
Δ	Antagonist band half-width (coordination constraint parameter)
p	Pressure (reservoir or chamber, depending on context)
$F_{\max}$	Approximate maximum end force at the spring radius

Variables

θ	Knee joint angle
$\dot{\theta}$	Knee joint angular velocity
τ	Knee joint torque (measured or estimated)
$\theta_{\text{KB,cmd}}$	Commanded position of the antagonist actuator (KB)
$\theta_{\text{KBMB,present}}$	Measured/present position of the master actuator (KBMB)
$\theta_{\min}, \theta_{\max}$	Lower/upper moving clamp limits for the antagonist command
p_{res}	Reservoir pressure
p_{thigh}	Thigh chamber pressure
p_{shin}	Shin chamber pressure

Contents

List of Figures

List of Tables

1
Introduction

Early-stage evaluation of wearable exoskeleton controllers is difficult to perform systematically because human testing is constrained in cost, logistics, and safety, and because subject-to-subject variability makes comparative benchmarking challenging. These constraints motivate a controlled bench-top platform that can reproduce key interaction mechanisms under defined conditions while supporting rapid iteration of mechanical design and control software.
This thesis addresses this need by developing a robotic anatomical phantom limb testbed, with a focus on the knee joint, intended for bench-top validation of wearable exoskeleton controllers. The platform is designed to reproduce interaction effects relevant to human–robot coupling, including non-linear joint impedance and interface compliance at the human–exoskeleton attachment interface. To support rapid prototyping and analysis, the testbed is paired with a real-time digital twin (DT) implemented in Unity, connected to the physical system through a Robot Operating System 2 (ROS 2) middleware architecture.

1.1 Problem formulation

Exoskeleton performance is strongly influenced by the mechanical impedance of the human limb, which is inherently non-linear and state-dependent due to muscle activation, co-contraction, passive tissue properties, and soft-tissue deformation. When an exoskeleton applies assistive torques, part of the mechanical energy can be absorbed by soft tissues, straps, and interface compliance rather than producing the intended joint motion. This reduces effective assistance and complicates controller tuning. A controlled phantom platform that captures key impedance characteristics can therefore improve controller development by enabling comparative bench-top experiments and by reducing reliance on early human trials.

1.2 Aim and objectives

Aim: To design and implement a robotic phantom knee testbed and DT capable of generating configurable non-linear joint stiffness and soft-tissue compliance for early-stage evaluation of exoskeleton controllers.
Objectives:

- Design and build a mechanical knee joint platform with antagonistic actuation [1, 2] capable of generating controlled joint torques and motion.

- Implement a spring-based variable-stiffness joint mechanism that generates a non-linear knee torque–angle response inspired by passive musculoskeletal tissues (e.g., ligament and tendon stiffness behavior).
- Integrate a controllable muscle-mimicking interface-compliance module using pneumatically regulated inflatable chambers, including pressure sensing and solenoid valve actuation for adjustable soft-tissue/strap compliance at the exoskeleton attachment interface.
- Develop a ROS 2-based control and data acquisition architecture that supports typed messaging, safety handling, parameter tuning, and optional inertial sensing for observing limb segment orientation during experiments.
- Create a Unity-based DT that visualizes joint motion, sensor signals, and system state in real time, enabling debugging and controller prototyping.
- Characterize the platform through experimental subsystem tests and demonstrate use cases relevant to exoskeleton evaluation.

1.3 Scope and limitations

This work focuses on a single-joint knee mechanism intended for bench-top testing rather than full-body gait replication. The system is designed to emulate selected aspects of human biomechanics that are most relevant to wearable joint assistance, especially non-linear stiffness behavior and soft-tissue compliance at the interface. Active muscle control and neuromuscular feedback are not reproduced; instead, the phantom provides a hardware-in-the-loop environment where stiffness and compliance can be tuned under controlled bench-top conditions. The DT is used for visualization and parameter interaction, not as a high-fidelity finite element model of tissues.

1.4 Contributions

The main contributions of this thesis are:

- A robotic phantom knee platform with antagonistic actuation and a spring-based variable stiffness mechanism to generate non-linear joint stiffness.
- A pneumatic soft-tissue compliance subsystem with pressure sensing and controllable actuation to emulate tunable interaction compliance.
- A ROS 2 middleware architecture enabling real-time parameter control, data logging, and safety features for experimental operation.
- A Unity DT connected via ROS–TCP that visualizes the platform state and supports interactive experimentation.

1.5 Thesis outline

Chapter 2 reviews related work on robotic anatomical phantoms, variable stiffness actuation, and soft-tissue interaction in exoskeleton testing.

Chapter 3 describes the methodology used for system design, implementation, and evaluation.
Chapter 4 presents the hardware and software system design, including the actuation, pneumatics, sensing, and control architecture.
Chapter 5 reports experimental results and characterization of the platform.
Chapter 6 discusses the findings, limitations, and implications for exoskeleton research.
Finally, Chapter 7 summarizes the work and outlines directions for future development.

2
Background

This chapter reviews related work on anatomical phantom platforms for exoskeleton evaluation and summarizes concepts used throughout the thesis: biomechanical impedance, variable stiffness and antagonistic actuation, and interface compliance. It also introduces the role of digital twins and ROS-based software architectures for real-time experimentation and system integration.

2.1 Robotic anatomical phantoms for exoskeleton evaluation

Several research efforts have proposed mechanical replicas of human limbs for evaluating assistive devices under standardized conditions.
Filius et al. presented the *Dummy Arm*, a modular and cost-effective platform for verification of arm exoskeletons, emphasizing configurable limb dimensions, adjustable masses, and passive joint stiffness for iterative development [3]. Their work demonstrates how physical limb surrogates can accelerate verification, but the platform primarily represents static or quasi-static properties and does not capture dynamic pathological behaviors such as spasticity or fatigue-related changes.
Barrutia et al. developed a lower-limb mechanical phantom designed to study knee kinematics and the influence of soft-tissue deformation on exoskeleton assistance [4]. Their findings highlight that interface compliance and tissue deformation can significantly reduce effective assistance, underscoring the importance of including tissue-like behavior in test platforms. However, their approach relied on static materials (e.g., gel analogs) and focused on a specific anatomical region.
Dezman et al. introduced a mechatronic leg replica for benchmarking physical interaction between humans and exoskeleton devices [5]. Their system supported force and torque transfer measurements under controlled conditions but did not include tunable soft-tissue compliance or mechanisms intended to emulate muscle-like impedance modulation.
Overall, prior phantom platforms demonstrate clear experimental value, yet they often remain limited in four areas: (i) capturing non-linear, state-dependent joint impedance, (ii) providing tunable soft-tissue compliance under controlled conditions, (iii) enabling rapid software–hardware iteration for controller development, and (iv) maintaining well-defined human–exoskeleton interface conditions such as axis alignment, attachment positioning, and load transfer. These gaps motivate a platform that combines controllable joint impedance, tunable interface compliance, and a workflow that supports rapid software–hardware iteration under well-defined

experimental conditions.

2.2 Biomechanical impedance and non-linear stiffness

Human joint mechanics are commonly described through *impedance*, which characterizes the dynamic relationship between motion (position, velocity, and acceleration) and interaction forces/torques. Joint impedance arises from both passive tissues (ligaments, joint capsule, tendons, skin) and active neuromuscular contributions. A key feature is non-linearity and state dependence: passive stiffness typically increases near end ranges, and muscle co-contraction can substantially raise effective stiffness and damping even at the same joint configuration.
In wearable robotics, impedance matters because it determines how applied exoskeleton torques translate into joint motion and interface loads. If the combined human–exoskeleton system exhibits substantial compliance at the attachment interface, part of the commanded torque is absorbed by soft-tissue deformation and viscoelastic losses rather than producing the intended joint rotation. This reduces apparent control bandwidth, complicates controller tuning, and can increase discomfort or lead to unintended load concentrations.
A phantom platform intended for exoskeleton evaluation should therefore reproduce the dominant impedance features relevant to the target use case. For knee assistance, these include a well-defined kinematic hinge, non-linear stiffness over the operating range, and adjustable interface compliance representing soft-tissue effects near attachment points, ideally with defined and configurable boundary conditions.

2.3 Variable stiffness and antagonistic actuation

Variable stiffness actuators (VSAs) and variable impedance actuators (VIAs) provide a means to modulate stiffness independently of position, enabling safe physical interaction and human-like behavior [6, 7]. A widely used concept is *antagonistic actuation*, inspired by biological muscle pairs (e.g., quadriceps–hamstrings), where two actuators apply opposing torques through elastic elements. Co-activation increases stiffness through increased elastic pretension, while differential activation produces net torque and motion.
In mechanical implementations, an elastic element such as a spring stores energy and shapes the torque–angle relationship. With appropriate geometry and preloading, the resulting stiffness can be designed to be non-linear, increasing with deflection in a manner similar to passive tissues. Antagonistic architectures can also improve safety by limiting peak interaction forces through elastic compliance.
The platform developed in this thesis uses two linear actuators arranged in an antagonistic configuration and coupled through a spring-based linkage. This arrangement is selected to (i) enable controllable torque generation, (ii) produce a non-linear stiffness profile through the elastic element, (iii) support controlled and parameterizable experiments relevant to exoskeleton control, and (iv) allow rapid, low-latency mod-

ulation of torque and stiffness through direct mechanical coupling.

2.4 Soft-tissue compliance and exoskeleton interface effects

Beyond joint mechanics, the interaction between an exoskeleton and the human limb is dominated by soft tissues and attachment structures. Even if an exoskeleton applies a desired joint torque, soft-tissue deformation and strap compliance can lead to relative motion (slippage), delayed torque transmission, and energy losses. These effects are commonly observed as reduced assistance efficacy and increased discomfort.

Soft-tissue behavior is difficult to reproduce with static materials alone because effective compliance varies with load, strap tension, geometry, and time-dependent material properties. Tunable compliance mechanisms can improve phantom realism by allowing controlled modulation of stiffness and damping. Pneumatic elements, such as inflatable silicone chambers, offer a practical approach: pressure can be regulated to adjust stiffness, and the system can be instrumented to measure pressure and infer interaction conditions.

In this thesis, pneumatic chambers are integrated as an adjustable compliance module, with pressure sensing and valve control to generate adjustable stiffness settings. This enables experiments that explicitly investigate how interface compliance impacts control performance and torque transfer.

2.5 Digital twins and ROS-based architectures

A DT refers to a virtual representation of a physical system that is updated in real time using sensor data. In robotics development, digital twins support visualization, debugging, safety monitoring, and rapid prototyping of control logic. When paired with a physical phantom, a DT can also provide a consistent experimental interface and facilitate dataset generation [8, 9].

ROS 2 has become a widely adopted middleware for robotic systems, offering structured message passing, hardware abstraction, and integration with a broad ecosystem of tools for logging, visualization, and modular control. For a research platform intended to be iterated upon and extended, ROS 2 provides a flexible architecture for integrating actuators, sensors, safety logic, and user interfaces [10].

The present work combines a ROS 2-based control framework with a Unity-based DT. This supports real-time visualization of joint motion and sensor signals while enabling typed, structured communication between the physical system and the software environment, supporting rapid controller iteration and structured experiments [11].

2.6 Summary

Prior work demonstrates the value of anatomical phantom platforms for controlled exoskeleton testing, but highlights limitations in representing non-linear joint impedance, tunable interface compliance, and flexible software integration. These observations motivate the design choices and evaluation methods presented in the following chapters.

3

Methodology

This chapter describes the methodology used to design, implement, and evaluate the robotic phantom knee (RPK) testbed and its DT. The work follows an iterative engineering workflow: requirements were derived from the target use case of early-stage exoskeleton controller evaluation, a system architecture was defined, subsystems were implemented and integrated, and the platform was characterized through experimental tests. The methodology is structured to support traceability from design decisions to measurable outcomes.

3.1 Design requirements

The primary goal of the platform is to provide a bench-top test environment that enables systematic experimentation with exoskeleton controllers while capturing the dominant mechanical behaviors that influence human–exoskeleton interaction. The main requirements are summarized below.

R1: Well-defined joint kinematics. The mechanical structure shall provide a well-defined knee hinge axis with a stable range of motion and minimal unwanted compliance in the rigid structure.

R2: Controllable torque and motion. The actuation system shall generate bidirectional joint motion and allow controlled torque production across the operating range relevant to exoskeleton experiments.

R3: Non-linear stiffness behavior. The platform shall generate a non-linear torque–angle relationship representative of passive tissue behavior, with stiffness that can increase with joint deflection.

R4: Tunable soft-tissue compliance. The platform shall include a mechanism to emulate interface compliance (e.g., soft tissue and straps) in a controllable and adjustable way.

R5: Instrumentation and data access. The platform shall provide sensor feedback sufficient for monitoring and analysis, including joint motion and soft-tissue state.

R6: Real-time integration and usability. The system shall support real-time monitoring, parameter adjustments, and experiment execution through a software framework that is practical for iterative development.

R7: Safety. The platform shall include safety mechanisms that reduce the risk of hardware damage and operational hazards, and enable controlled shutdown of actuation and pneumatic subsystems.

3.2 System-level architecture and integration strategy

The platform is composed of four tightly-coupled subsystems:

1. **Mechanical knee structure** (hinge, linkages, mounting interfaces).
2. **Actuation subsystem** (antagonistic actuators and elastic linkage).
3. **Pneumatic compliance subsystem** (inflatable chambers, valves, pump, pressure sensors).
4. **Software and DT subsystem** (ROS 2 nodes, communication bridge, Unity visualization and user interaction).

A modular integration strategy was used:

- Each subsystem was first characterized in isolation using dedicated tests (e.g., actuator response tests, pressure regulation tests).
- Subsystems were then integrated incrementally, starting from hardware communications (USB, SPI, I^2C), followed by closed-loop control, then visualization and user interaction in the DT.
- Typed ROS 2 topics and services were used for all data and commands to reduce ambiguity, improve robustness, and simplify debugging.

3.3 Mechanical and actuation design methodology

The mechanical structure was designed using a CAD workflow to define geometry, clearances, and mounting interfaces. The knee joint was modeled as a single-degree-of-freedom hinge. To emulate non-linear joint stiffness, an elastic element (spring-based linkage) was integrated into an antagonistic actuation layout.

The actuation mechanism was selected to meet requirements R2 and R3:

- Two actuators generate opposing actions about the knee hinge, enabling both net motion and controlled preloading of the elastic element.
- The spring-based linkage shapes the torque–angle relationship, and the geometry defines the effective stiffness profile.
- The antagonistic configuration enables tuning of the effective joint stiffness by adjusting the relative actuator setpoints, thereby increasing or decreasing spring preload.

3.3.1 Antagonistic layout and elastic coupling

The two actuators are mounted facing each other, such that their shafts oppose in direction while actuating the same joint through a shared elastic linkage. This mirrored layout supports both bidirectional joint motion and controlled spring preloading, which is used to shape the effective stiffness profile of the knee.

A simplified representation of the coupling concept is:

$$[\text{Motor 1}] \leftrightarrow (\text{Elastic Element}) \leftrightarrow [\text{Motor 2}]$$

where the mirrored attachments enforce a consistent mechanical configuration and reduce the risk of assembly-dependent sign changes in the effective joint behavior.

3.3.2 Adaptive spring housing for systematic experiments

To support systematic testing of different stiffness behaviors with minimal redesign, an adaptive spring housing was developed. The initial configuration was designed to accommodate a single spiral (clock) spring, while an alternative configuration was designed to accommodate four compression springs within the same outer housing concept. This modular approach enables future studies to compare springs with different stiffness curves and compliance characteristics without changing the overall mechanical layout or motor mounting interfaces.

Mechanically, the housing is implemented as a two-part assembly: (i) a reaction component that bears against the second hinge link (shin link) and defines the load path to the joint, and (ii) an internal actuated component that applies displacement to the spring set. The resulting spring forces are transmitted from the actuated component to the reaction component and into the hinge link, thereby generating the desired joint torque.

The housing is connected to both motors through two dedicated flanges, designed with a keyed geometry such that mounting is only possible at a predetermined angle. This ensures defined alignment of the spring linkage and reduces assembly-dependent variability.

Key design decisions were evaluated by:

- checking range-of-motion feasibility under mechanical constraints,
- verifying expected stiffness scaling with deflection based on linkage geometry,
- assessing actuator limits (speed/force) against target operating conditions.

The physical realization of the adaptive spring housing and its alternative configurations is presented in Chapter 4 (see Figures 4.5 and 4.6).

3.4 Pneumatic soft-tissue compliance methodology

To emulate soft-tissue compliance (R4), inflatable chambers were integrated into the mechanical structure at locations relevant to exoskeleton attachment. The pneumatic system was designed around:

- a pump and reservoir providing pressure supply,
- solenoid valves for supply and vent control per chamber,
- pressure sensors measuring reservoir and chamber pressures,
- Flexible polyurethane and silicone tubing, and a check valve to enforce a stable flow direction.

A closed-loop pressure regulation approach was implemented:

- Pressure setpoints are defined directly per chamber and are adjustable at runtime (e.g., via the DT interface) to achieve the desired level of interface compliance.
- The controller never opens the supply and vent valves at the same time. If the measured pressure is below the setpoint, only the supply valve is activated; if

the pressure is above the setpoint, only the vent valve is activated; otherwise both valves remain closed to hold pressure. This avoids fighting flows and improves stability.
- Because the pump and solenoid valves are binary (on/off) actuators, the controller uses a discrete switching strategy. In particular, pulse-width modulation (PWM) is applied to the solenoid valve outputs to time-proportion their on-time [12]. The duty cycle acts as an effective flow command, enabling smoother pressure regulation than pure bang–bang switching. A minimum duty and small dither are applied to overcome valve dead zones and stiction at low actuation levels.
- To avoid rapid on/off switching when the pressure is close to the setpoint, the measured pressure is low-pass filtered and a small deadband (hysteresis) is used. This reduces valve "chattering" and helps overcome valve dead zones (ranges where a short activation produces little or no flow).

The pneumatic subsystem was characterized through step-response experiments and steady-state tracking performance, demonstrating controlled pressure regulation under typical operating conditions.
The pneumatic topology and its physical integration are detailed in Chapter 4 (see Figures 4.8, 4.9, and 4.10).

3.5 Sensing and data acquisition methodology

To support monitoring, control, and evaluation (R5), the following measurements are used:
- **Joint state:** joint position and velocity derived from actuator state feedback and/or DT kinematics.
- **Pneumatic state:** reservoir pressure and chamber pressures from pressure sensors.
- **Inertial sensing:** IMU measurements for thigh and shin segments, used when orientation information is required.

All measurements are published as typed ROS 2 messages with defined update rates and change-based throttling where appropriate (publishing only when the change exceeds a defined threshold, otherwise holding the last value) to reduce bandwidth, while preserving signal fidelity.

3.6 Software methodology: ROS 2 and DT

A ROS 2-based architecture was chosen to satisfy R6 and support modularity. The software stack was implemented as a set of nodes:
- **Actuators node:** handles device communication, actuator control modes, safety limits, and antagonistic coordination logic. In the experiments reported in this thesis, the actuators are primarily operated in position control, while torque/current limits are used to bound interaction forces and ensure safe operation.

- **Pneumatics node:** handles pressure sensing, pump (compressor) and solenoid-valve actuation, closed-loop pressure regulation, and safety services (e.g., E-STOP and release-air).
- **Feathers node (IMU/temperature bridge):** reads sensor data from the FeatherS3 devices and publishes typed ROS 2 topics.
- **Play/orchestrator node:** coordinates launch and runtime state of subsystems.

Unity is used as a DT environment for real-time visualization and interaction. ROS–TCP bridging enables bidirectional communication:

- Unity subscribes to sensor/state topics for visualization.
- Unity publishes command and parameter topics for interactive tuning and actuation commands.

The DT interface was designed to support:

- live visualization of joint angle and subsystem states,
- interactive parameter tuning (e.g., controller gains and setpoints),
- user-initiated actuation and safety actions such as actuator torque disable and pneumatic release.

3.6.1 DT asset pipeline: CAD-to-Unity workflow

Engineering CAD tools typically represent geometry as parametric solids (e.g., NURBS [13]), which must be tessellated into triangle meshes for real-time rendering in Unity. Direct CAD-to-mesh export often results in overly dense models that are inefficient for interactive visualization. Blender was therefore used as the primary modeling tool to support a real-time asset pipeline: geometry could be simplified (decimation/retopology), LOD and collision meshes could be generated [14], and pivots and materials could be prepared in a Unity-friendly way. In addition, Blender supported the fabrication workflow used in this project (3D-printed components) by enabling dimensionally consistent models with verified scale and fit, while maintaining separate render-optimized versions for the DT.

In practice, the workflow was:

- import or reconstruct geometry from mechanical design references,
- create fabrication-ready parts with real scale and verified clearances,
- generate simplified meshes (decimation/retopology) for Unity rendering and collision,
- export Unity assets with consistent coordinate frames and units.

3.7 Evaluation methodology

The platform is evaluated through experimental characterization aligned with the design requirements:

- **Mechanical range:** verify the achievable range of motion and joint positioning behavior under defined commands.
- **ROS–Unity integration performance:** verify stable bidirectional ROS–TCP communication, including tolerance to Unity client restarts, and quantify effective ROS topic publish rates during active operation.

- **Safety and orchestration validation:** validate Play subsystem start/stop services and safety workflows, including latched E-STOP and release-air behavior.
- **Antagonistic actuation validation:** confirm stable actuator communication, controlled joint motion, and bounded antagonist behavior under the coordination constraint (band clamp), including edge-case stress tests.
- **Pneumatic compliance characterization:** evaluate pressure regulation stability, setpoint tracking performance, and safety venting behavior during release-air.
- **Feather sensor streaming validation:** validate continuity of IMU and temperature topic publishing during normal operation and assess reconnect behavior as an observed limitation.

Quantitative metrics include steady-state error, rise time, and overshoot for pressure control. Qualitative evaluation includes usability of the DT interface and robustness during experiment execution.

3.8 Summary

The methodology follows an iterative system-engineering workflow with modular subsystem tests and requirements-driven evaluation. By combining antagonistic actuation, a spring-based stiffness mechanism, and pneumatic compliance within a ROS 2 and Unity DT framework, the platform supports controlled and extensible exoskeleton controller evaluation.

4

System Design

This chapter presents the design of the RPK platform, including mechanics, actuation, pneumatic soft-tissue compliance, sensing, and the software/digital twin architecture. The system is designed as a bench-top testbed for exoskeleton controller evaluation, with hardware-in-the-loop operation and real-time visualization through a Unity DT.

4.1 System overview

At a high level, the platform consists of:

- a single-degree-of-freedom knee hinge connecting thigh and shin segments,
- two antagonistic actuators coupled through a spring-based linkage to produce controllable joint motion and non-linear stiffness,
- a pneumatic subsystem that regulates inflatable air chambers to emulate soft-tissue compliance,
- sensor feedback (pressure, inertial orientation, temperature) for monitoring and analysis,
- a ROS 2 middleware layer on a Raspberry Pi 5 that interfaces with hardware and communicates with Unity through ROS–TCP.

4.2 Mechanical structure and knee joint

The mechanical structure is organized into two rigid segments (thigh and shin) connected through a hinged joint representing the knee. The hinge constrains motion to flexion and extension about a single axis, providing a controlled and measurable joint angle. Mounting interfaces and structural components are designed to support bench-top operation and stable attachment of the actuation and pneumatic subsystems.
The mechanical design emphasizes:

- a well-defined hinge axis and stable range of motion,
- rigid load paths for actuator forces to reduce unintended compliance,
- accessible mounting points for chambers, valves, sensors, and cabling.

4.2.1 Custom 3D-printed structural components

Figure 4.1 shows the full set of 3D-printed components used in the RPK platform. These parts were designed to support modular assembly and compact integration of

actuation, pneumatics, and embedded electronics. The numbering in the figure is used consistently in the part descriptions below.

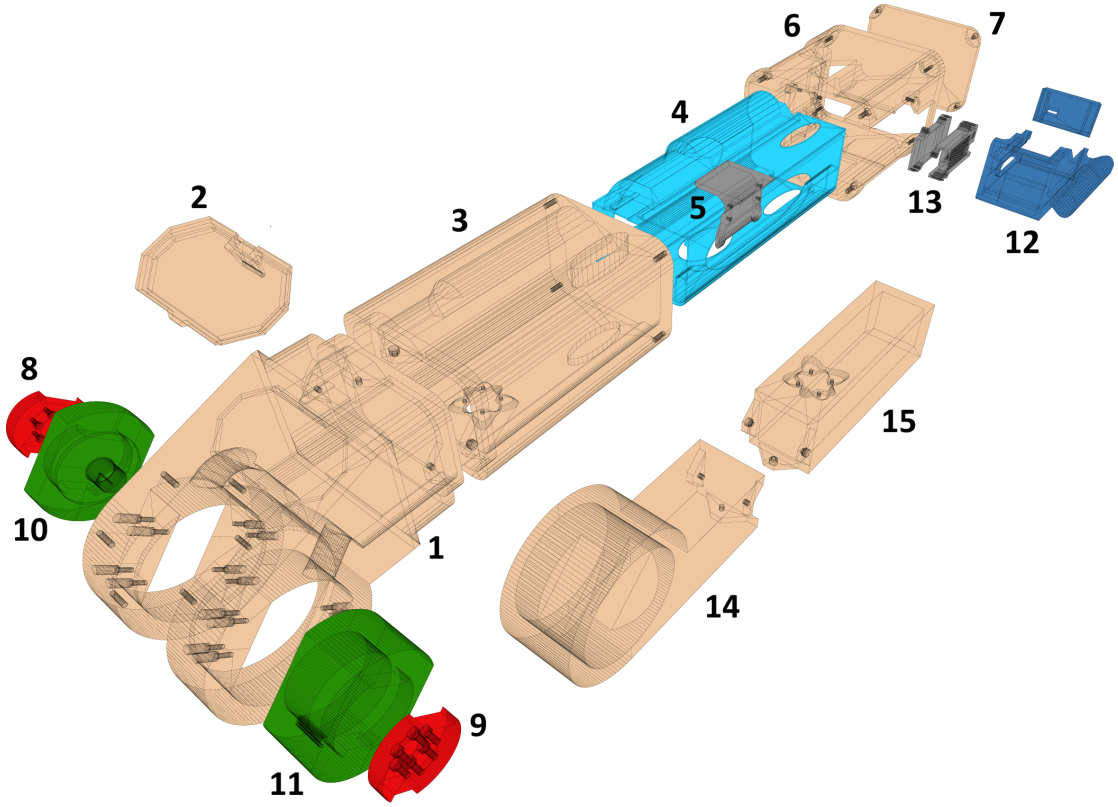


Figure 4.1: Overview of the custom 3D-printed parts used in the RPK platform. Numbers correspond to the part descriptions in Section 4.2.1.

1. **Thigh Hinge** — Structural hinge component defining the knee joint axis on the thigh side and ensuring rigid alignment of the flexion–extension axis. It also provides mounting interfaces for both actuators and integrates routing channels and placement features for internal wiring and components to improve service accessibility.
2. **Access Cover** — A snap-lock access cover that protects internal wiring and components while allowing tool-free access for maintenance and some adjustments.
3. **Thigh Body** — Main load-bearing structural component of the thigh segment, integrating mounting interfaces for the hinge, pneumatic hardware, and electronic components, and supporting the air chambers via prototype pump wedges (used during iterative testing) and final chamber sleeves up to 25 cm in length.
4. **Pneumatics Container** — Protective enclosure that houses the pneumatic subsystem components, including the pump/compressor, check valve, reservoir, solenoid valve manifold (supply/vent valves), and tube fittings, while providing structured routing for both tubing and electrical wiring (solenoid driver connections and sensor cabling). The enclosure also incorporates pass-through routing channels for cables that continue to other subsystems, including ac-

tuator wiring and the Feather sensor connections (IMU and temperature) for both the thigh and shin. The assembly uses a rail-guided sliding interface with tight tolerances to ensure defined positioning, stable mounting, and simplified maintenance.

5. **Solenoid Driver Holder** — Dedicated mount for the solenoid driver board, providing mechanical protection, electrical isolation, and service accessibility.
6. **Rear Container** — Rear enclosure used for cable management and electronics integration. It houses the Raspberry Pi 5 drawer/holder, two DC–DC converters (one supplying the pneumatics subsystem and one supplying the Raspberry Pi 5), and one Unexpected Maker FeatherS3 (ESP32-S3), enabling a compact and serviceable layout.
7. **Thigh Rear Cover** — Protective cover closing the rear container while allowing easy access for maintenance and inspection.
8. **KB Flange** — Output-shaft-mounted interface flange that couples the KB (antagonist) actuator to the elastic linkage assembly. The flange incorporates a keyed geometry that constrains the Spiral Spring Hub to a single mounting orientation, ensuring safe, defined angular alignment during assembly.
9. **KBMB Flange** — Output-shaft-mounted interface flange that couples the KBMB (primary) actuator to the elastic linkage assembly. It is mirrored relative to the KB flange to match the facing-actuator layout and includes the same keyed indexing geometry, ensuring defined angular alignment during assembly.
10. **Spiral Spring Hub** — Central hub interfacing the flat spiral spring. It is primarily driven by the KB-side flange, while the resulting spring torque is transmitted through the elastic linkage and shared between both actuator-side flanges (KB and KBMB).
11. **Spiral Spring Housing** — Housing that encloses the flat spiral spring and transmits the resulting torque to the shin-side link arm through its outer geometry.
12. **RP5 Drawer/Holder** — Sliding drawer used to mount the Raspberry Pi 5 for convenient removal during debugging, updates, or replacement, and to hold a dedicated breakout/interface board for the pressure sensors.
13. **RP5 Buck Converter Enclosure** — Protective enclosure for the DC–DC step-down converter supplying regulated power to the Raspberry Pi 5, improving electrical safety by reducing the risk of accidental short circuits.
14. **Shin Hinge** — Structural hinge component on the shin side, forming the complementary half of the single-degree-of-freedom knee joint. Further mechanical optimization is intended, but was out of scope within the available project timeframe.
15. **Shin Body** — Main structural body of the shin segment, providing mounting points for the hinge and routing for pneumatic and sensor integration. Additional design improvements are planned; however, the current implementation was prioritized to meet the thesis time constraints while enabling the reported experiments.

4.2.1.0.1 As-built prototype status at time of writing. Figure 4.2 shows the RPK platform as implemented at the time of writing. The photograph is included to document the physical integration state used during iterative testing. In this configuration, temporary pump wedges are visible in place of the final air-chamber sleeves, and the complete shin-side 3D-printed components are not shown in the pictured assembly. These elements were iterated in parallel while prioritizing functional verification under the project time constraints.

Figure 4.2: As-built RPK prototype at the time of writing, documenting the physical integration state used during iterative testing. The image shows temporary pump wedges used during prototyping in place of the final air-chamber sleeves; the complete shin-side 3D-printed components are not shown in this view.

4.3 Antagonistic actuation subsystem

4.3.1 Actuator configuration

Joint actuation is implemented using two Westwood Robotics Koala BEAR linear actuators in an antagonistic configuration [15]. Both actuators are connected on a shared BEAR bus and addressed as separate devices through the BEAR SDK [16]. Communication is provided by a USB2BEAR dongle and a daisy-chained BEAR signal-port connection (Figures 4.3 and 4.4).

- Koala BEAR Muscle Build **(KBMB)** as the primary actuator (device ID: 1),
- Koala BEAR **(KB)** as the secondary/antagonist actuator (device ID: 2),

- **USB2BEAR** dongle for host communication and daisy-chained BEAR signal ports.

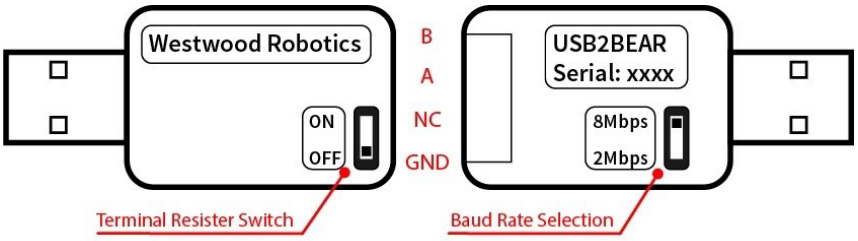


Figure 4.3: USB2BEAR dongle used as the host interface to the BEAR actuator bus. [16]

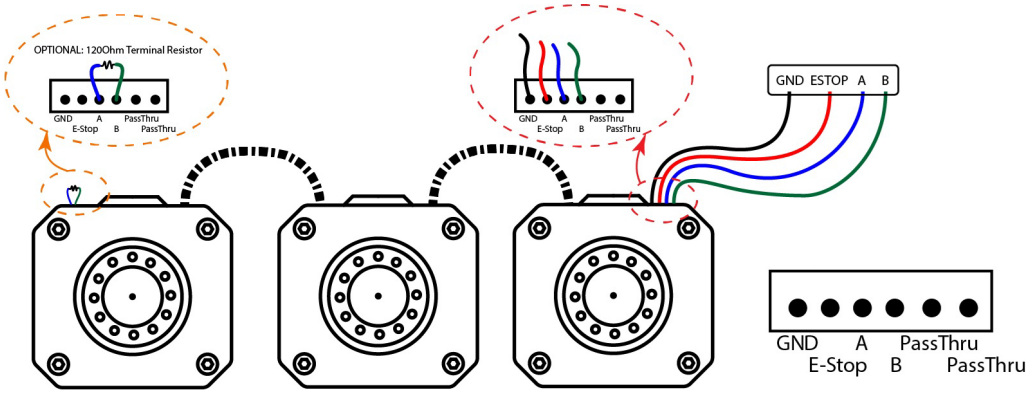


Figure 4.4: Daisy-chained BEAR signal-port connection used to communicate with both actuators on a shared bus. [16]

The actuators are mounted facing each other with mirrored mechanical attachments across the knee linkage. This mirrored layout introduces an inherent sign difference: a positive command in one actuator's local coordinates can correspond to the opposite mechanical effect in the joint frame for the other actuator. To avoid ambiguity in experiments and user interaction, the control software maintains a consistent joint (world-frame) sign convention for commands (e.g., flexion/extension) and maps the sign appropriately for each actuator.

These actuators were selected because they provide a well-supported control and feedback stack that is practical for iterative bench-top experimentation. Using two actuators from the same vendor yields a unified configuration and control workflow (common SDK, consistent registers/telemetry, and one integration pathway), which reduces software complexity and integration risk. Through the BEAR SDK, the system can command standard operating modes and read back state variables required for monitoring and analysis [16]. In this work, low-level control and monitoring are implemented via the PyBEAR Python interface, vendored in the ROS 2 workspace [17], which exposes device configuration, command registers, and feedback signals used by the actuator node.

4.3.1.1 Actuator torque requirement (simple estimate)

This section provides a simplified estimate of the peak knee torque required to move and hold the lower leg against gravity. The goal is not to model detailed human biomechanics, but to obtain a conservative torque target that can be compared directly to the actuator capabilities and used to motivate the selected actuator pair.

4.3.1.1.1 Assumptions. Standard body-segment parameter tables indicate that one lower limb (thigh+shin+foot) is approximately 16–18% of body mass, with the thigh about ∼10% and the shin+foot about ∼6% [18]. For a representative adult, $m_{\text{body}} = 70$ kg is assumed, giving

$$m_{\text{shin}} \approx 0.06\, m_{\text{body}} \approx 4 \text{ kg},$$

where the shin and foot are lumped into the *shin* link. The shin link length is assumed to be $L_{\text{shin}} = 0.40$ m. For simplicity, the link is modeled as a uniform rod, so its center of mass is located at

$$r_{\text{shin}} = \frac{L_{\text{shin}}}{2} = 0.20 \text{ m}$$

from the knee joint axis. A sizing value of $g \approx 10$ m/s^2 is used.

4.3.1.1.2 Knee torque from the shin link. The gravitational knee torque is maximal when the shin is horizontal:

$$\tau_{\text{knee,grav,max}} = m_{\text{shin}} g r_{\text{shin}} \approx (4)(10)(0.20) = 8 \text{ N} \cdot \text{m}.$$

4.3.1.1.3 Design margin and target torque. To cover acceleration effects, friction, and modeling uncertainty, a conservative margin $S = 2$ is applied:

$$\tau_{\text{req,peak}} \approx S\, \tau_{\text{knee,grav,max}} \approx 16 \text{ N} \cdot \text{m}.$$

4.3.1.1.4 Connection to actuator selection. The estimated peak knee torque requirement of approximately 16 N·m motivates the selection of an actuator pair with sufficient torque margin. Table 4.1 summarizes vendor-specified peak torque capabilities and gear ratios for the selected BEAR-series actuators.
In the antagonistic layout, the KBMB motor is assigned as the primary actuator due to its higher torque capability, while the KB motor is used as the secondary (antagonist) actuator to support coordinated motion, stiffness shaping through elastic preload, and bounded antagonistic operation. This asymmetric pairing provides adequate torque headroom for the intended operating range while avoiding unnecessary oversizing of both actuators.

Table 4.1: Vendor-specified actuator specifications for the BEAR-series actuators used in the antagonistic actuation system (from [16]).

Actuator	**Peak torque (15s)**	**Peak torque (1.5s)**	**Gear ratio**
KB	4.2 N·m	10.5 N·m	9:1
KBMB	8.0 N·m	20.0 N·m	20:1

Beyond torque capacity, the actuators were also evaluated with respect to backdrivability, speed, and mechanical integration. Their gear ratios allow partial backdrivability under external loading, which is desirable in a phantom joint intended to emulate compliant human interaction rather than rigid industrial positioning. In the present system, this complements the elastic linkage by allowing externally applied torques to be absorbed through a combination of spring deformation and drivetrain compliance.
The actuators further provide motion speeds suitable for the intended exoskeleton test scenarios, where quasi-static and moderately dynamic knee motions are of primary interest, enabling smooth flexion–extension cycles.

Table 4.2: Vendor-specified speed constants for the selected actuators (from [16]).

Actuator	**Speed constant**
KB	27.3 RPM/V
KBMB	9.0 RPM/V

From a mechanical integration perspective, the actuators provide a compact form factor (small external dimensions) relative to their torque capability. Mounting interfaces on multiple sides of the actuator housing provide multiple mounting options, enabling the facing, mirrored installation required by the antagonistic layout.

4.3.2 Spring-based linkage and non-linear stiffness

A flat spiral (clock) spring is used as the primary elastic element shaping the torque–angle relationship of the knee mechanism. The selected spring is a **Lesjöfors flat spiral spring** SCS 1.5×15×5 (code 925), manufactured from stainless steel (EN 1.4310) and characterized by a strip thickness of $t = 1.5$ mm, width $b = 15$ mm, and a support radius of $r = 33$ mm [19].
Within the reference operating range, catalog data provide representative torque values (e.g., $\tau \approx 8.10$ N·m at 336°). For control-oriented modeling and interpretation, the torsional stiffness is approximated by a linearized slope over the catalog reference range. The catalog spring rate is specified as 24.1 N·mm/° [19], which corresponds to

$$k_t \approx 1.38 \text{ N} \cdot \text{m/rad}$$

after unit conversion.
Using this linearized stiffness, the corresponding torque at the default coordination limit of 90° is estimated as

$$\tau(90^\circ) \approx k_t \left(\frac{\pi}{2}\right) \approx 2.17 \text{ N} \cdot \text{m}.$$

At the catalog reference point of 336°, the peak torque is $\tau(336^\circ) \approx 8.10$ N·m.

4.3.2.0.1 Adaptive spring housing concepts (spiral vs. compression). To investigate how the elastic linkage can shape the knee's passive response while remaining mechanically safe under antagonistic actuation, two adaptive housing concepts were developed: (i) a spiral-spring housing intended for large bidirectional

relative rotation (Figure 4.5), and (ii) a compression-spring housing intended for a compact design with a smaller allowable relative motion (Figure 4.6).

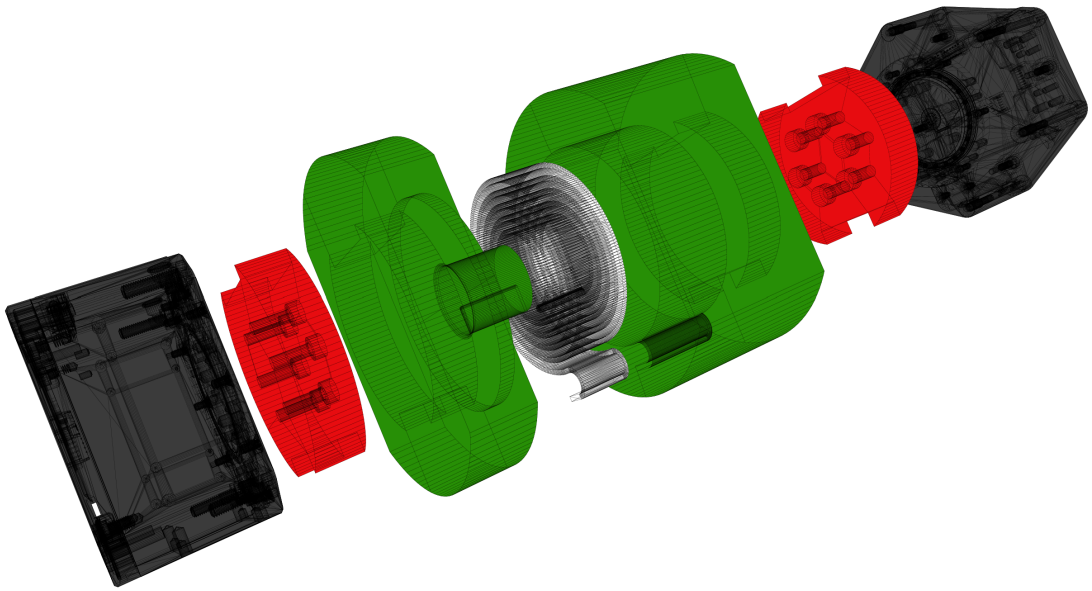


Figure 4.5: Adaptive spiral-spring housing concept (primary elastic element), designed for large bidirectional relative rotation ($\Delta = \pm 90°$).

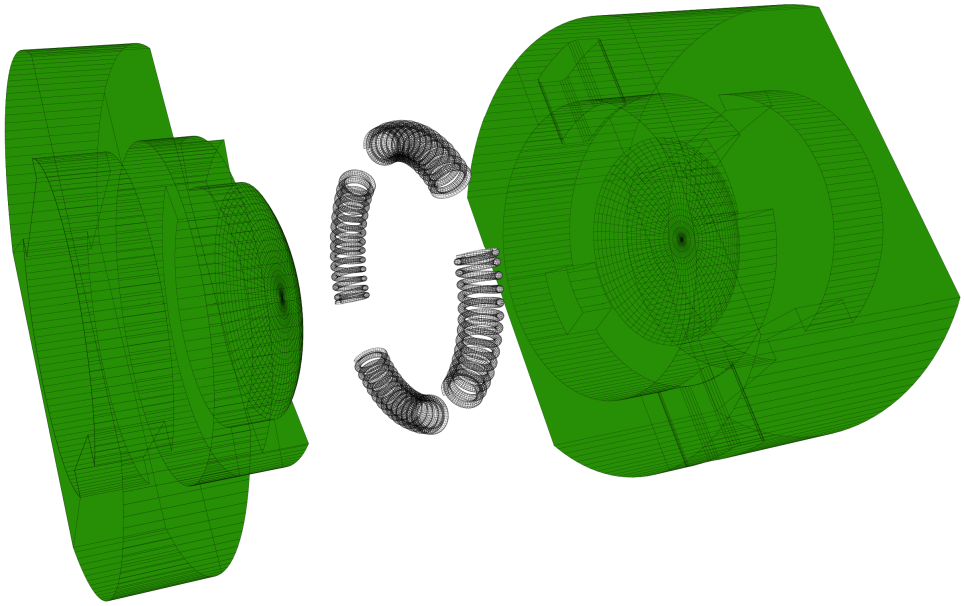


Figure 4.6: Adaptive compression-spring housing concept (compact alternative), designed for small relative rotation ($\Delta = \pm 9°$).

Based on these trade-offs, the spiral-spring housing concept was selected for the RPK platform. The as-built spring placement inside the adaptive housing is shown in Figure 4.7.

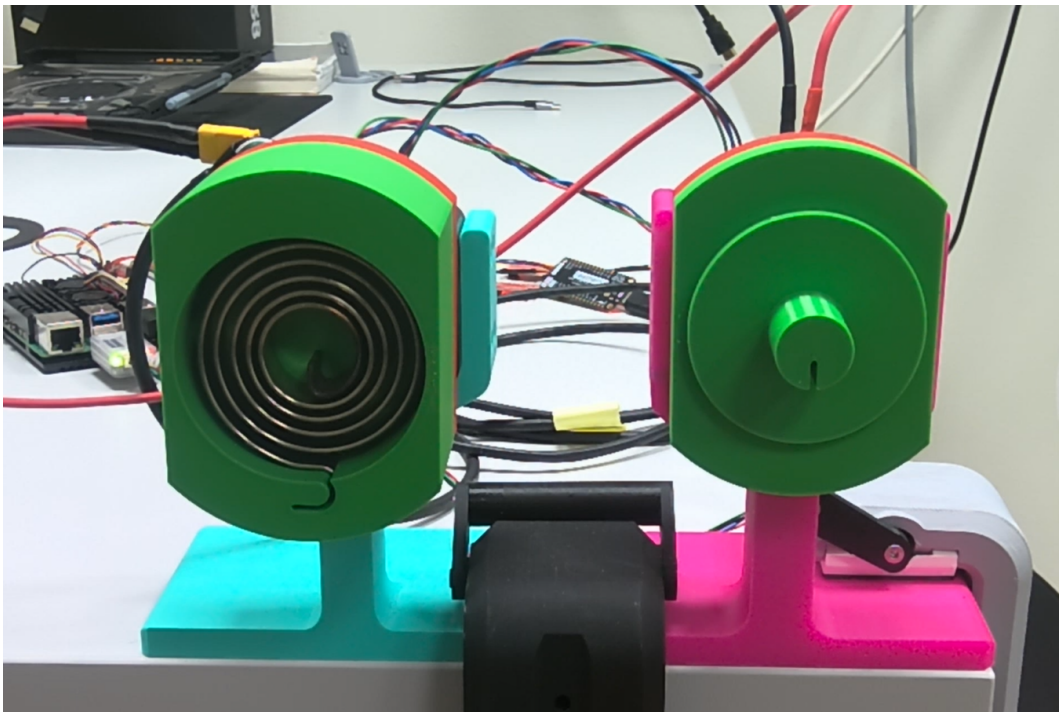


Figure 4.7: Real-life placement of the flat spiral spring inside the adaptive housing.

The two concepts illustrate trade-offs between allowable relative motion (Δ margin), packaging complexity, and the achievable stiffness profile. The spiral spring concept supports a wide Δ band suitable for bidirectional operation around the neutral position, while the compression-spring concept prioritizes compactness but requires a tighter synchronization margin.

These parameters are used as baseline values for interpreting the system's passive compliance and for relating observed joint behavior to a simplified stiffness model.

Combined with antagonistic actuation, the elastic linkage enables two key behaviors:

- **Net joint motion** generated by differential actuator action about the knee,
- **Effective stiffness shaping** through elastic loading and linkage geometry, resulting in an angle-dependent torque response.

4.3.3 Actuator safety and coordination logic

To support safe experimental operation, the actuator control layer includes:

- torque enable/disable control,
- command limiting (position/velocity/current bounds),
- an antagonist coordination constraint (band clamp) that restricts the antagonist actuator (KB) to remain within a bounded range of the primary actuator (KBMB) position, reducing the risk of mechanical conflict and excessive preload.

4.4 Pneumatic soft-tissue compliance subsystem

4.4.1 Hardware architecture

Soft-tissue compliance in the RPK is emulated using inflatable air chambers controlled by a dedicated pneumatic circuit. The hardware provides independent pressure regulation for the thigh and shin chambers, supporting controlled inflation and deflation, while a reservoir buffers the supply to improve pressure stability.

The pneumatic subsystem follows a centralized supply topology with two independently regulated branches, one for the thigh and one for the shin, as illustrated in Figure 4.8. To complement the functional schematic, Figure 4.9 shows a CAD-based alignment view of the same assembly, clarifying physical placement and routing, while Figure 4.10 shows the corresponding real hardware implementation.

The main components of the system are:

- a pump used as a compressor (SparkFun ROB-10398),
- a check valve and a reservoir (Festo CRVZS-0.1) to stabilize the supply pressure,
- four solenoid valves (SMC VX210JZ1E), arranged as supply and vent valves for each chamber,
- three pressure sensors (Honeywell ABPDANV030PGSA3) measuring reservoir, thigh chamber, and shin chamber pressures,
- an I^2C solenoid driver (Adafruit 6318) used to actuate the pump and solenoid valves.

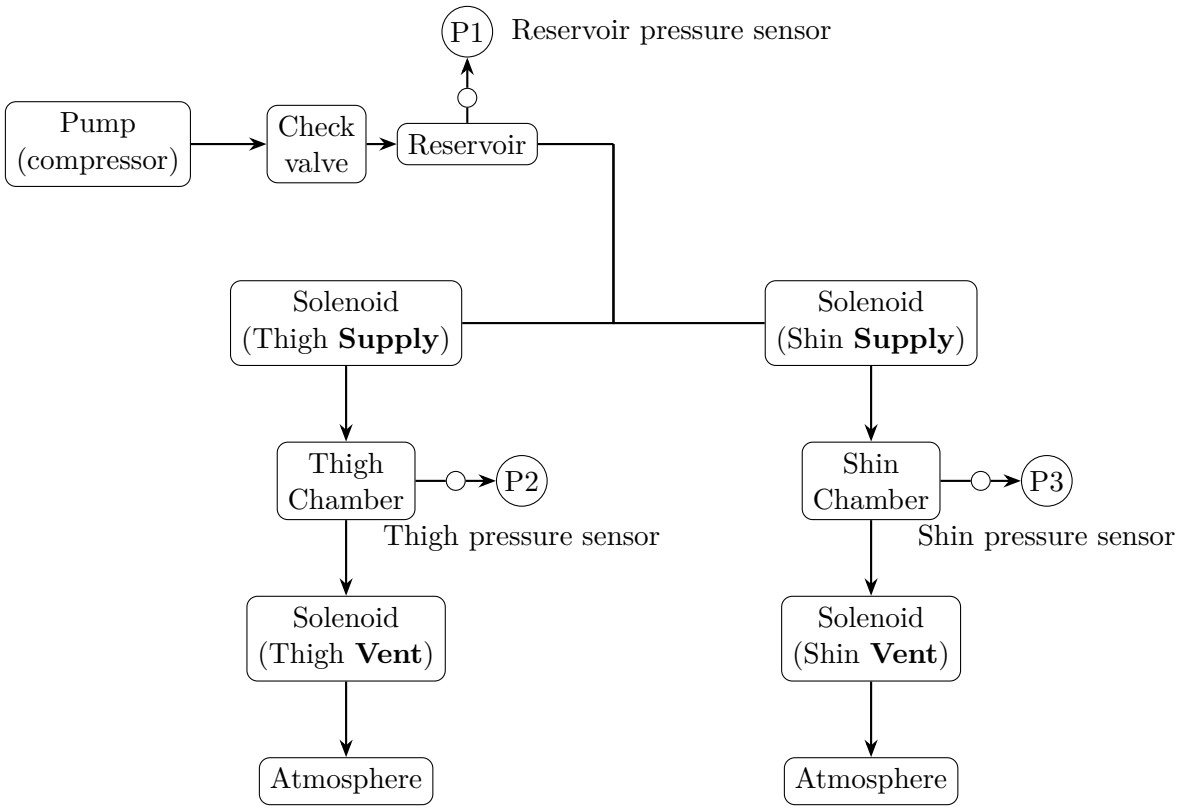


Figure 4.8: Schematic of the pneumatic system topology. The pump supplies the reservoir through a check valve, while two identical branches (thigh and shin) regulate chamber pressure using separate supply and vent solenoid valves and local pressure sensing.

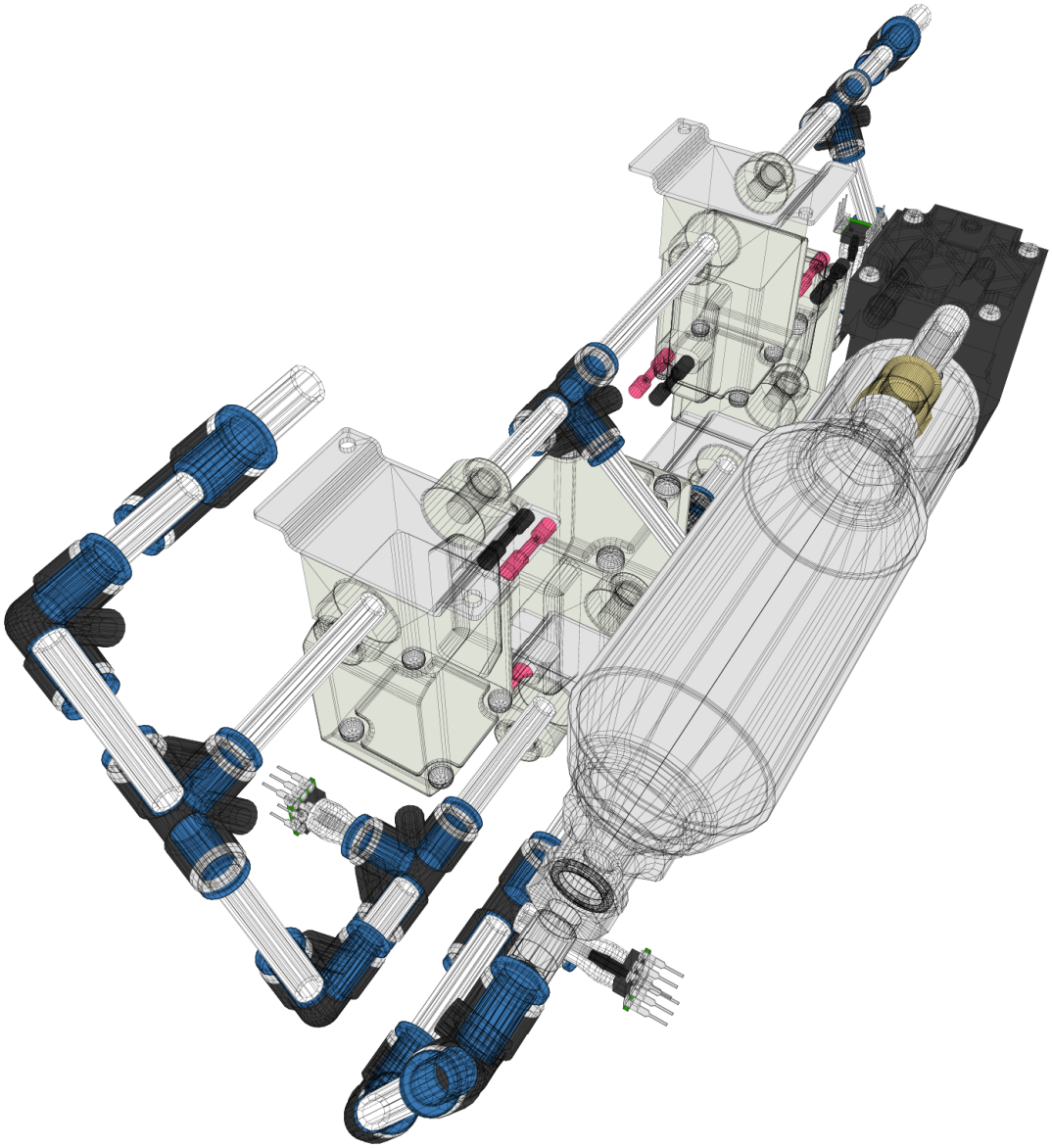


Figure 4.9: CAD alignment view of the pneumatic assembly in its assembled configuration, showing the relative placement and alignment of almost all the components, despite the air chambers. The figure is intended to clarify physical integration and routing; functional flow and control logic are described by the schematic in Figure 4.8.

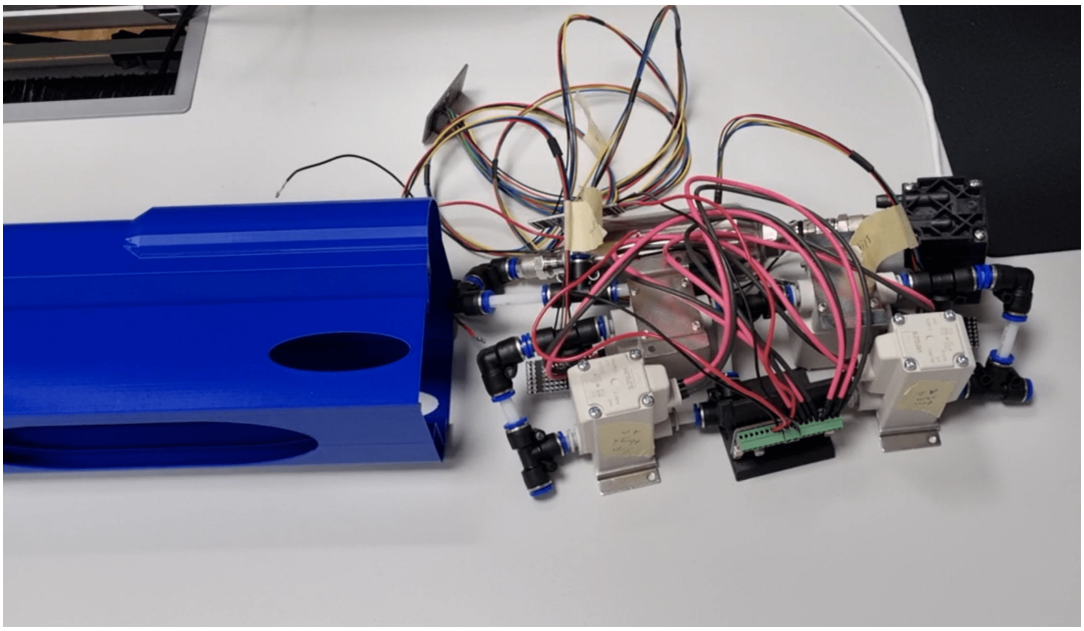


Figure 4.10: Photograph of the implemented pneumatic subsystem on the RPK platform, corresponding to the architecture in Figure 4.8 and the CAD alignment view in Figure 4.9.

Figure 4.11 shows the same pneumatic subsystem after final packaging inside the Pneumatics Container.

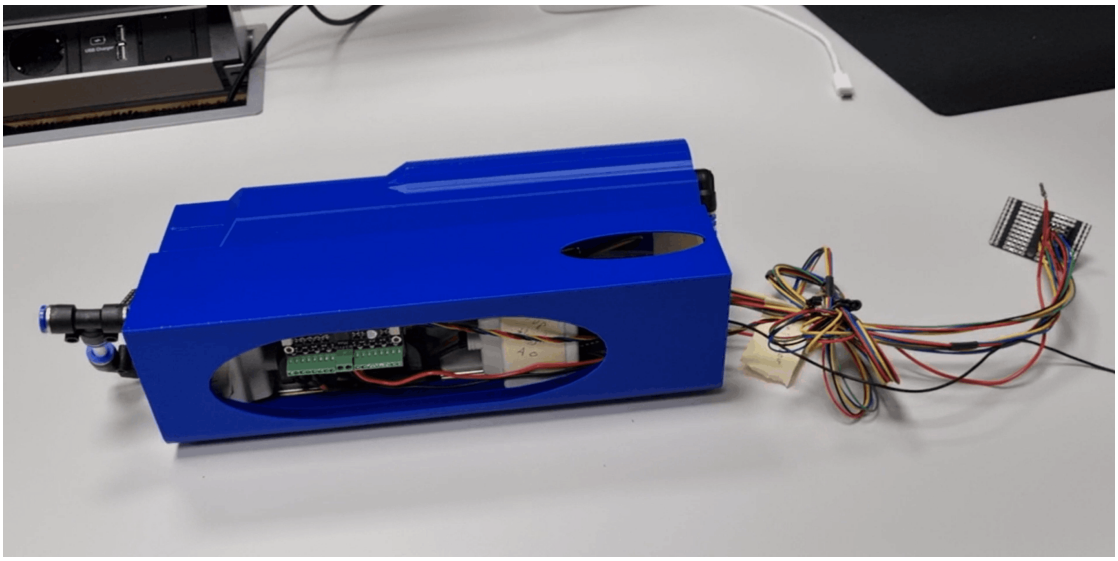


Figure 4.11: Real-life implementation of the pneumatic subsystem after final packaging inside the Pneumatics Container.

Compressed air is routed from the pump through a check valve into a shared reservoir, which acts as a buffer to reduce pressure fluctuations and improve pressure stability. Reservoir pressure is monitored by a dedicated pressure sensor connected via a T-connector, providing feedback on available pressure headroom for downstream control.

From the reservoir, two identical downstream branches originate, corresponding to the thigh and shin chambers. Each branch consists of a supply solenoid valve that regulates inflation of the chamber and a vent solenoid valve that releases air to atmosphere. A pressure sensor connected via a T-connector measures the local chamber pressure, enabling closed-loop pressure regulation for each limb segment independently.

This architecture decouples the pressure dynamics of the thigh and shin chambers while maintaining a compact and modular layout. In the present implementation, each branch supports independent pressure setpoint control for the thigh and shin chambers. This provides a foundation for future extensions in which pressure setpoints are modulated as a function of joint angle and/or loading to emulate angle-dependent and load-dependent soft-tissue compliance.

4.4.1.0.1 Deflation concepts: atmospheric venting vs. exhaust recirculation. An alternative deflation concept was evaluated in which the chamber exhaust was routed back to the pump inlet (exhaust recirculation), effectively using the pump as the exhaust sink. In practical testing, this configuration resulted in noticeably slower pressure decay and was therefore not adopted.

This behavior is consistent with the component flow capacities. The selected pump (SparkFun ROB-10398) operates with a working flow range of approximately 10–15 L/min, which inherently limits the achievable exhaust rate when used as the discharge path. In contrast, the SMC VX210-series vent solenoid valves provide substantially higher exhaust capacity; conservatively, their discharge flow can be treated as on the order of $\sim$ 100 L/min under typical operating conditions. As a result, direct venting to atmosphere provides a significantly faster deflation in practice.

The final design therefore retains atmospheric venting as the primary deflation mechanism. The pump-return routing evaluated during development is shown in Figure 4.12.

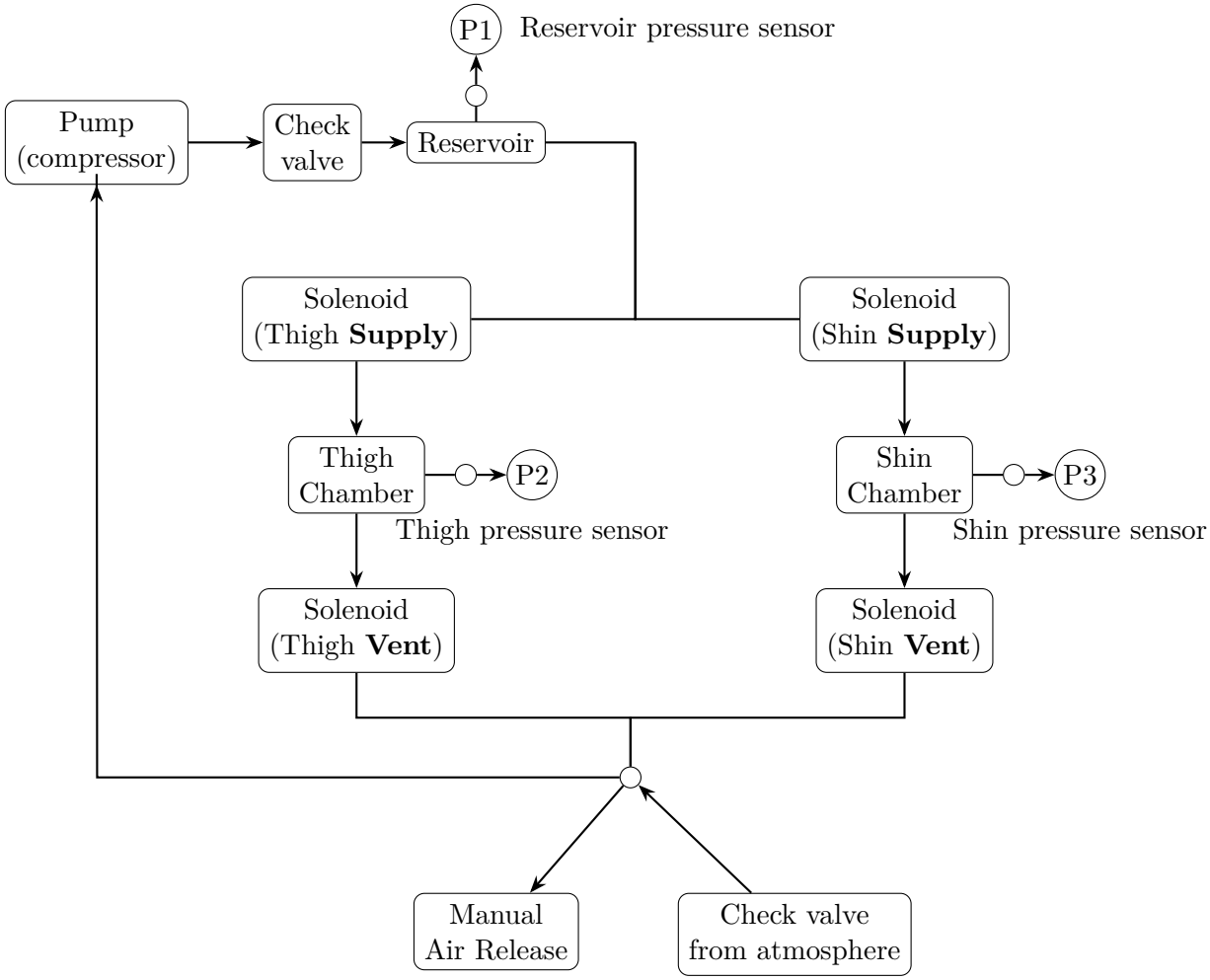


Figure 4.12: Pneumatic topology used to evaluate pump-return deflation. The thigh and shin vent outlets are merged and routed back to the pump inlet as an alternative discharge sink. Practical tests showed slower pressure decay with this routing, so the final design retained direct atmospheric venting.

4.4.2 Control approach

Chamber pressure is regulated using a closed-loop controller. The controller is designed for robustness with solenoid valves by:

- using a split-range strategy (either supply or vent is active at a time),
- applying filtering and hysteresis to reduce chatter,
- supporting predefined compliance "profiles" that map to pressure setpoints.

4.5 Sensing and embedded peripherals

4.5.1 Pressure sensing

The three pressure sensors (reservoir, thigh, shin) provide the primary feedback for the pneumatic subsystem. Reservoir pressure indicates available headroom for

supply control, while chamber pressures represent the effective compliance state of the thigh and shin soft-tissue (air chambers) elements.

4.5.2 IMU and temperature sensing

Two sensor bridge microcontrollers (Unexpected Maker FeatherS3, ESP32-S3) are used to forward I^2C sensor data over USB-C to the main computer:

- two 9-DoF IMUs (SparkFun SEN-19895) for thigh and shin orientation,
- two temperature sensors (SparkFun SEN-15805).

These sensors provide additional context during dynamic experiments and assist in monitoring thermal conditions during extended operation.

4.6 Computing and software architecture

4.6.1 Compute platform and middleware

The main control computer is a Raspberry Pi 5 running Ubuntu Server 24.04 LTS and ROS 2 Jazzy. ROS 2 provides the message-passing and modular structure required to integrate actuation, pneumatics, and sensors, and to interface with the Unity DT.

4.6.2 ROS 2 node structure

The software is structured as a set of ROS 2 nodes that separate hardware I/O, control, and orchestration:

- **Actuators node:** communicates with the USB2BEAR bus, reads actuator state, applies limits and coordination logic, and exposes command topics for torque, position, velocity, and current control.
- **Pneumatics node:** reads pressure sensors, drives solenoid outputs, performs closed-loop pressure regulation, and exposes safety services (E-STOP, release air).
- **Feathers node:** receives IMU and temperature data from the FeatherS3 devices and publishes typed ROS 2 topics.
- **Play/orchestrator node:** manages process startup/health checks and provides convenience services to start/stop subsystems as a group.

All interfaces are implemented using typed ROS 2 messages and services (no JSON payloads), improving robustness and making the system easier to validate and extend.

4.6.3 Unity DT integration

A Unity-based DT is used for real-time visualization, debugging, and interactive parameter tuning. Communication between ROS 2 and Unity is provided through ROS–TCP:

- ROS 2 publishes system state (joint variables, pressures, valve states, safety flags),

- Unity publishes commands and parameter updates (targets, mode toggles, gains).

The DT supports:

- live visualization of joint motion and limb orientation,
- inspection of pneumatic state (reservoir and chamber pressures),
- interactive control panels for experiment execution and safety actions.

4.7 Safety workflow

Safety and operation are organized around the *Play* subsystem, which serves as the operator-facing entry point for starting, stopping, and placing the platform in a safe state.

In addition, a single, latched E-STOP action is implemented as the primary safety mechanism. Triggering E-STOP forces a safe state by:

- disabling actuator torque (soft-stop behavior by commanding zero velocity and zero current),
- triggering the pneumatic *release-air* routine to vent the reservoir and both air chambers for a fixed duration,
- locking the operator interface by disabling relevant controls in the Unity UI.

To reduce the risk of restarting with stale pneumatic commands, the pressure setpoints in the Unity UI are forced to 0 psi when E-STOP is triggered and the corresponding zero setpoints are transmitted to the pneumatic controller. The latched E-STOP state is released via a hold-to-reset gesture (validated in Chapter 5); after reset, the system remains depressurized until the operator explicitly increases the setpoints again.

4.8 Summary

This chapter presented the RPK system design, combining a single-axis knee hinge with antagonistic actuation through a spring-based linkage to generate controlled joint motion and a non-linear. Tunable soft-tissue/interface compliance is implemented using a reservoir-buffered pneumatic circuit that regulates inflatable chambers in the thigh and shin via supply/vent solenoid valves with local pressure sensing and closed-loop control. Pressure, IMU, and temperature sensing provide monitoring and experimental context. A modular ROS 2 software architecture on a Raspberry Pi 5 integrates the hardware subsystems and connects to a Unity DT via ROS–TCP for real-time control, visualization, and parameter tuning. Safety is enforced through a latched E-STOP workflow coordinating actuator soft-stop behavior, pneumatic venting, and UI lockout.

5

Results

This chapter reports experimental and system-level results obtained from the RPK platform under the default runtime workflow: the ROS 2 stack on a Raspberry Pi 5 connected to a Unity DT via ROS–TCP. The results are organized by subsystem: (i) ROS–TCP integration, (ii) Play orchestration and safety, (iii) antagonistic actuation, (iv) pneumatic compliance control, and (v) Feather sensor streaming.

5.1 Experimental setup and data collection

All tests were performed on the integrated platform with ROS 2 running on the Raspberry Pi 5 and Unity connected as a ROS–TCP client. Evidence was collected using: (i) time-stamped ROS topic streams (recorded to bag files when applicable), (ii) ROS CLI verification (`ros2 topic hz`, `ros2 topic echo`, `ros2 service call`), and (iii) Unity UI screenshots during specific test stages (e.g., E-STOP).

Table 5.1: Default runtime configuration used during results collection. Nominal loop/poll rates are listed; observed ROS topic publish rates are reported separately in Fig. 5.2.

Parameter	**Default setting**
ROS–TCP endpoint	`0.0.0.0:10000`
Actuator command loop	$\sim$ 60 Hz
Actuator state streams	fast $\sim$ 50 Hz, slow $\sim$ 10 Hz
Pressure sensor polling	$\sim$ 200 Hz
Pneumatic control loop	$\sim$ 50 Hz
Feather poll timer	$\sim$ 200 Hz
Reservoir keeper	enabled ($\sim$ 25–26 psi)
UI joint-angle convention	0°=full flexion, 180°=full extension

Robustness was verified by restarting the Unity client during runtime: the ROS–TCP endpoint remained active, and Unity successfully reconnected and resumed streaming after reopening.

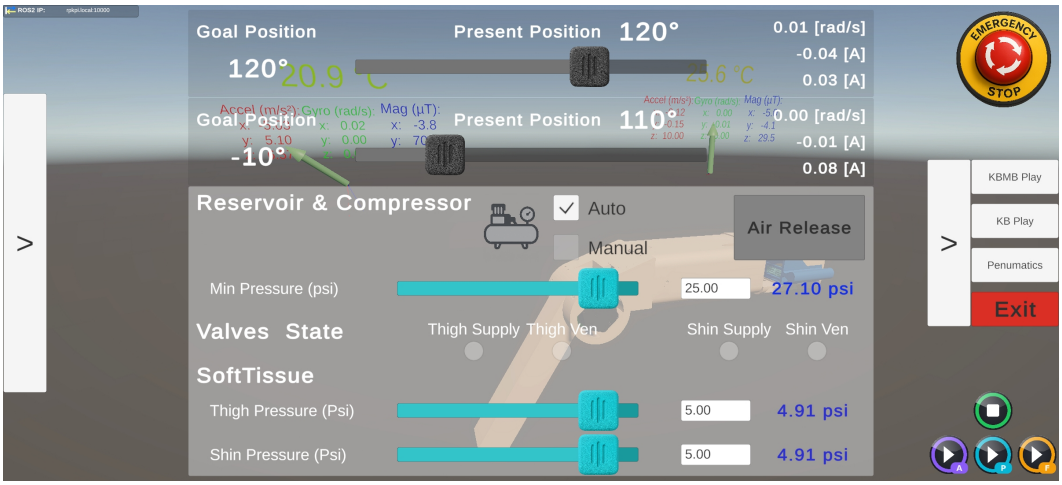


Figure 5.1: Example of synchronized visualization in the Unity DT during hardware operation.

5.1.1 Observed ROS topic publish rates

Update behavior was verified using `ros2 topic hz` on representative streams during *active operation* (i.e., joint motion and/or changing pressures). The measured values in Figure 5.2 reflect the effective ROS publish rates observed at runtime. Some streams are published with on-change logic and filtering; therefore, the effective rate depends on signal dynamics and may be lower when the system is stationary.

```
ROS topic publish-rate evidence (ros2 topic hz)

1) /kbmb/state/present_position
average rate: 38.155

2) /pressure/reservoir
average rate: 38.133

3) /imu_thigh
average rate: 52.643
```

Figure 5.2: Observed ROS topic publish rates measured using `ros2 topic hz` during active operation. Representative values were ≈ 38 Hz for actuator position and reservoir pressure topics, and ≈ 53 Hz for the thigh IMU stream.

5.2 Play subsystem results (orchestration and safety)

5.2.1 Service-based orchestration

The Play subsystem was validated as the operator-facing entry point for starting and stopping the platform. Functional tests confirmed that: (i) starting the system consistently brought the platform online (endpoint active, nodes running, data streaming), and (ii) stopping the system reliably shut down managed processes and

returned the platform to an idle state. Per-subsystem start/stop services were also verified, enabling subsystem isolation during debugging.

5.2.2 E-STOP behavior (latched trip + hold-to-reset)

The E-STOP workflow was validated using the single-button design implemented in Unity:

- **Trip action:** pressing the E-STOP button triggers a latched safety state.
- **Reset action:** holding the E-STOP button for 2.0 s and releasing outside the button region resets the latched state.

When E-STOP was triggered during motion, the Play subsystem consistently forced a safe state by disabling actuator torque (soft-stop by commanding zero velocity and zero current), triggering the pneumatic release-air routine to vent the reservoir and both air chambers, and locking the operator interface by disabling relevant UI controls.

At the same time, the pressure-command sliders in the Unity UI were immediately forced to a safe state: the reservoir, thigh, and shin pressure setpoints were set to 0 psi and a corresponding zero command was transmitted to the pneumatic controller, preventing any unintended pressurization while the system remained in the latched E-STOP state (Figure 5.3).

After the reset gesture, the UI returned to an operable state; however, the pressure setpoints remained at 0 psi until manually readjusted (Figure 5.4). This design enforces an explicit operator action before re-pressurization, reducing the risk of restarting with stale setpoints from a previous trial and ensuring recovery begins from a known depressurized baseline.

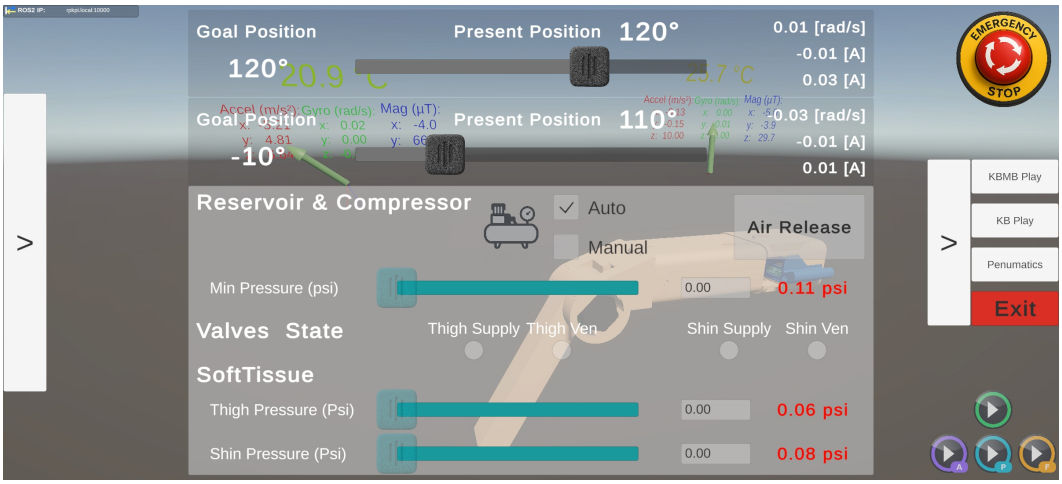


Figure 5.3: Unity operator interface immediately after pressing E-STOP (latched trip). Controls are disabled/greyed out, pneumatic release-air is triggered, and pressure setpoints are forced to 0 psi and transmitted to the pneumatic controller.

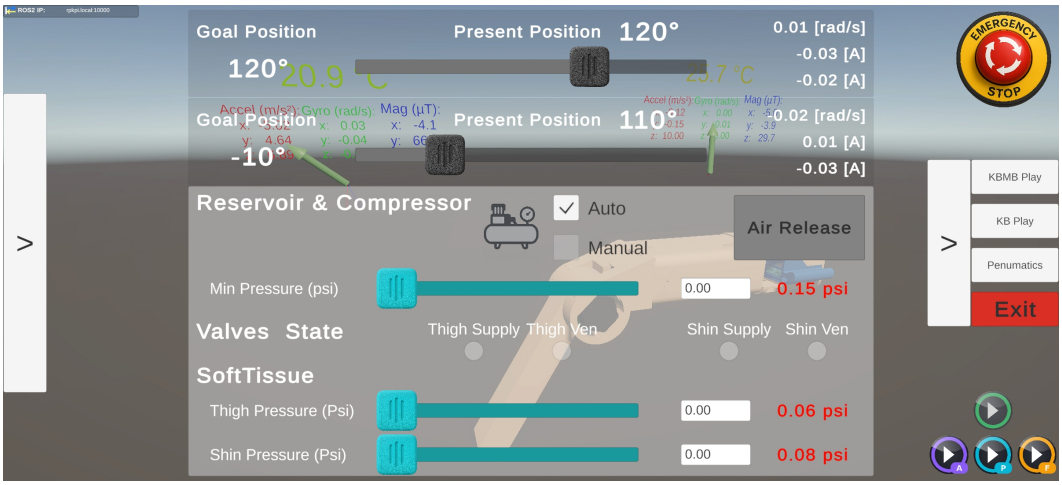


Figure 5.4: Unity operator interface after the hold-to-reset gesture (E-STOP released). Controls are re-enabled, while pressure setpoints remain at 0 psi until manually adjusted, enforcing an explicit operator action before re-pressurization.

5.3 Actuators subsystem results (KBMB + KB)

5.3.1 Runtime connectivity and controlled motion

The antagonistic actuation subsystem was validated under the default runtime configuration with two BEAR actuators (KBMB ID 1 as the primary and KB ID 2 as the antagonist). During runtime tests, actuator state streaming remained stable and the Unity DT reflected present position in real time, enabling direct visual verification of motion.

5.3.2 Antagonistic coordination constraint (band clamp)

During antagonistic operation, the antagonist actuator (KB) was commanded under a moving band constraint defined relative to the master actuator (KBMB) present position:

$$\theta_{\mathrm{KB,cmd}} \in [\theta_{\mathrm{KBMB,present}} - \Delta,\ \theta_{\mathrm{KBMB,present}} + \Delta]\,, \tag{5.1}$$

where Δ is a configurable band half-width. Under the default configuration, $\Delta = 20°$ (≈ 0.35 rad). The corresponding band limits ($\theta_{\min}$, $\theta_{\max}$) therefore shift as KBMB moves.

5.3.2.0.1 Observed clamp behavior. In trials with representative trajectories and deliberately out-of-range commands, the computed band limits moved with the master (KBMB) present position, and the antagonist (KB) position remained within the allowed band. Figure 5.5 shows an example where the requested KB goal command exceeds the band limits, while the measured KB position stays bounded within the moving range.

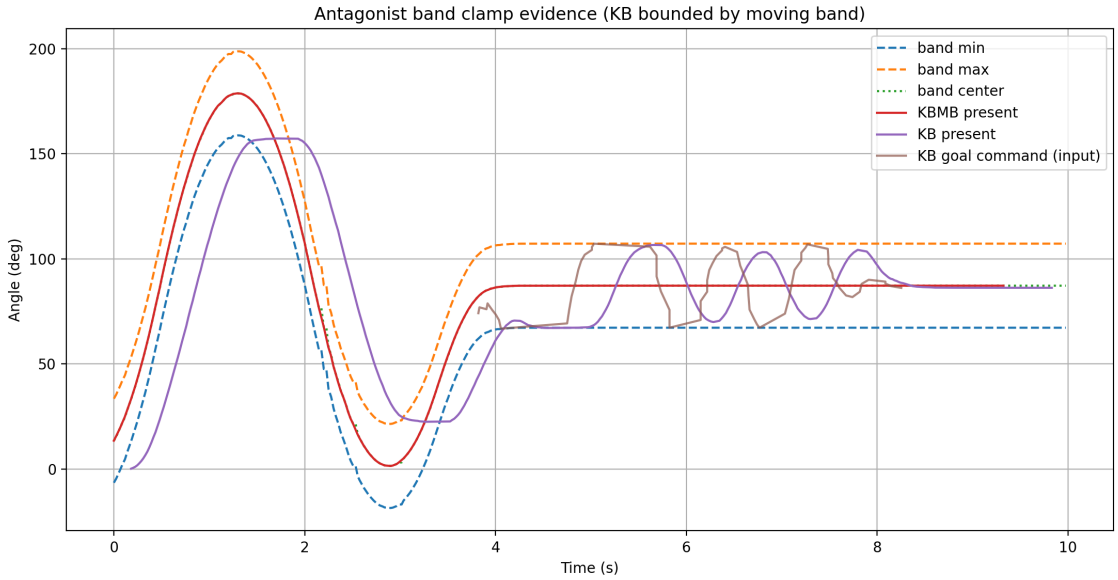


Figure 5.5: Evidence of the antagonistic band clamp. The allowable KB command band ($\theta_{\min}$, $\theta_{\max}$) is centered on the KBMB present position and shifts as KBMB moves. The requested KB goal command exceeds the band, while KB remains bounded within the limits.

5.3.2.0.2 Edge case: $\Delta = 0$ (synchronization limit). An edge-case stress test was performed with $\Delta = 0$, forcing the antagonist to track the master without an allowed band. Figure 5.6 shows that KB followed KBMB closely; however, during rapid changes, a small follower lag was observable while overall tracking remained stable. This lag is consistent with the one-dongle daisy-chain topology, where command and state transactions for two actuators are serialized over a shared communication path rather than executed simultaneously.

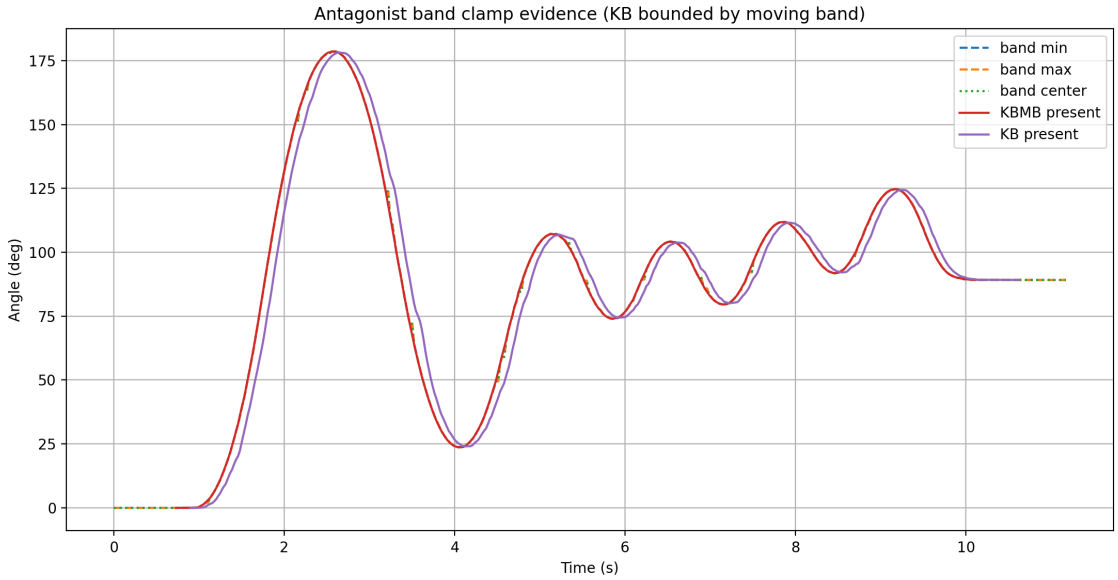


Figure 5.6: Edge-case synchronization test with $\Delta = 0$. The antagonist (KB) follows the master (KBMB) without an allowed band. A small lag is observable during rapid motion, while overall tracking remains stable.

5.4 Pneumatics subsystem results (soft-tissue compliance)

5.4.1 Setpoint tracking and step response

To provide an evidence of closed-loop pressure regulation, the chamber setpoints were commanded as a square-wave sequence between 0 and 5 psi for three consecutive cycles (inflate → hold → deflate → hold). Measured chamber pressures and the corresponding supply/vent valve commands were logged and plotted.

Figures 5.7 and 5.8 show that both chambers track the commanded setpoint pattern consistently across cycles: pressures rise toward the 5 psi plateau during inflation phases and return toward 0 psi during deflation phases, with valve command activity aligned to the expected phases (higher supply activity during inflation and higher vent activity during deflation). Isolated transient spikes were occasionally observed in the pressure trace; these were non-persistent and did not change the overall tracking behavior across the cycles.

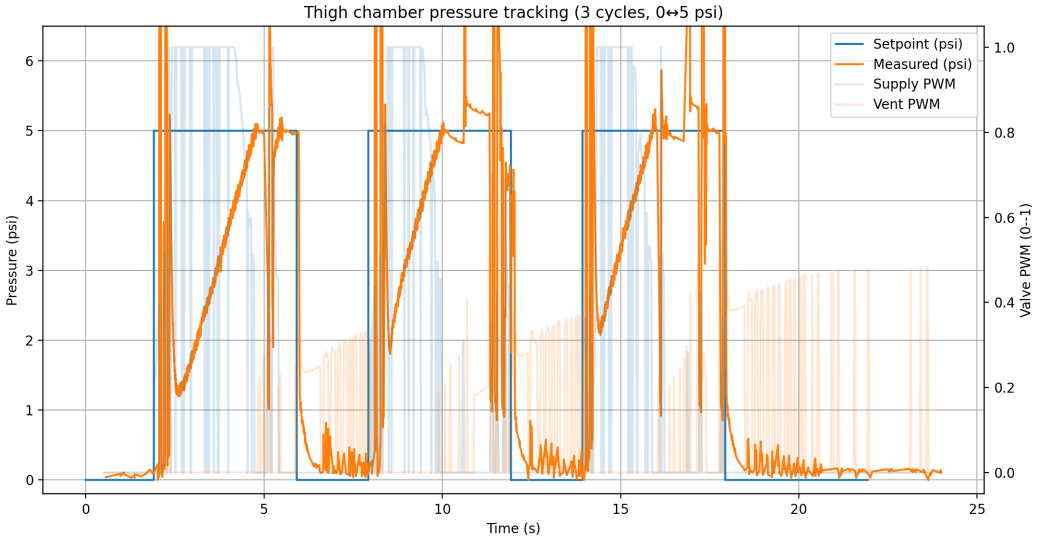


Figure 5.7: Thigh chamber pressure tracking over three inflation/deflation cycles (0↔5 psi), with supply/vent valve command activity.

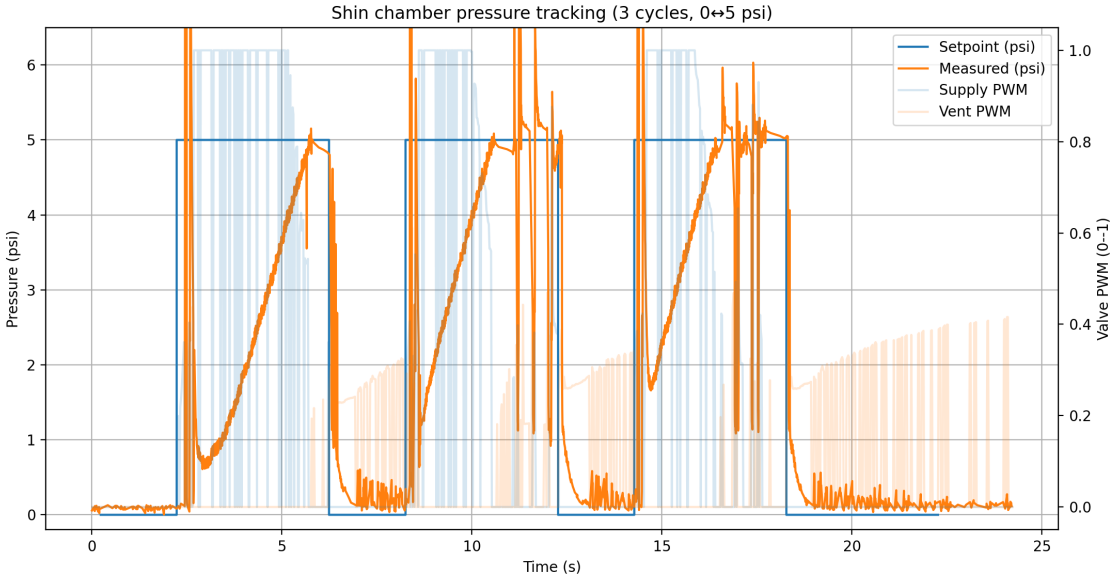


Figure 5.8: Shin chamber pressure tracking over three inflation/deflation cycles (0↔5 psi), with supply/vent valve command activity.

5.4.2 Compliance profile execution

To demonstrate execution of time-varying compliance commands, a predefined pressure profile was applied simultaneously to both thigh and shin chambers. The commanded sequence was $0 \rightarrow 2 \rightarrow 4 \rightarrow 6 \rightarrow 4 \rightarrow 2 \rightarrow 0$ psi. Three consecutive cycles were executed using different plateau durations per level: 3 s, 2 s, and 1 s, respectively. Chamber pressures and valve commands were logged during the run. Figure 5.9 shows that both chambers follow the commanded profile pattern across

the three timing regimes. Measured pressures rise toward each plateau during inflation segments and return toward lower levels during deflation segments. Valve command activity is visible throughout the transitions, consistent with closed-loop pressure regulation during profile execution.

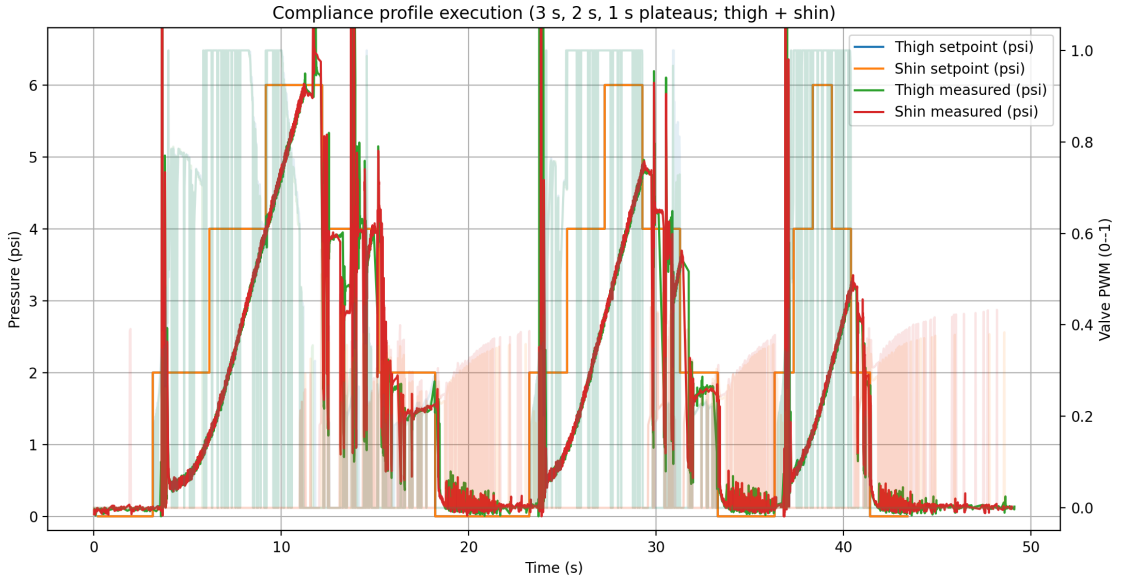


Figure 5.9: Compliance profile execution for thigh and shin commanded simultaneously: $0 \rightarrow 2 \rightarrow 4 \rightarrow 6 \rightarrow 4 \rightarrow 2 \rightarrow 0$ psi, performed for three cycles using 3 s, 2 s, and 1 s plateaus per level. Faint lines on the secondary axis show valve PWM activity.

5.4.3 Release-air and safety behavior

The release-air function was validated as the primary pneumatic safety mechanism and was triggered from a representative pressurized operating state. During each trial, pressures (reservoir, thigh, shin) and the corresponding valve commands were logged while a single release event was initiated. The release command forces both supply and vent valves to an open state for a fixed time window (10 seconds), providing rapid venting and removal of stored pneumatic energy.

Figure 5.10 shows a representative release event. All channels decayed toward 0 psi within the release window, while valve PWM commands saturated during the release phase, confirming that the venting sequence was executed as intended.

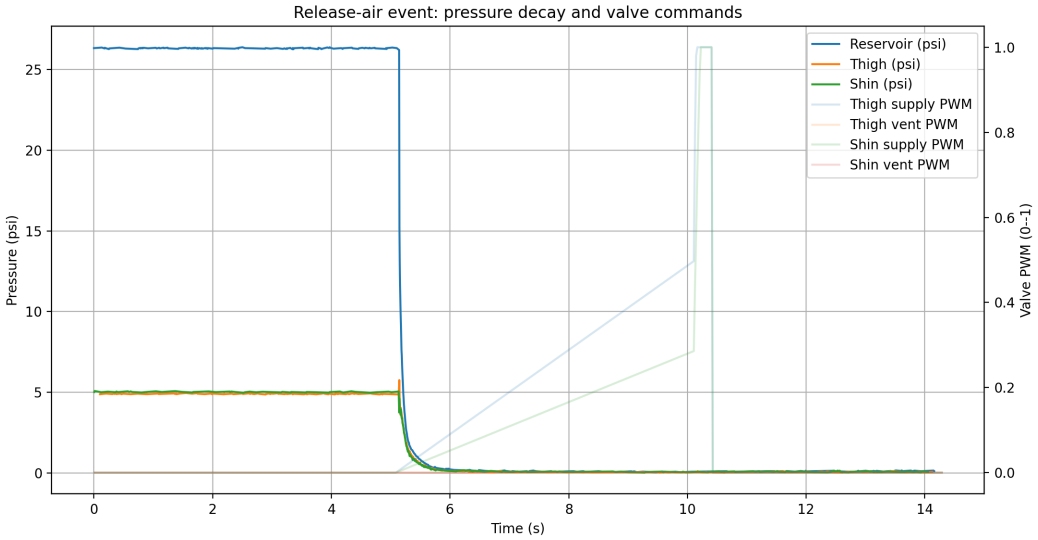


Figure 5.10: Pressure decay during a release-air event used for pneumatic safety shutdown, with supply/vent valve command activity shown on the secondary axis.

5.5 Feathers subsystem results (IMU + temperature streaming)

5.5.1 Topic publication and signal continuity

The Feather subsystem provides two independent sensor streams for the thigh and shin. During normal operation, IMU and temperature topics were published continuously, and data remained available to the Unity DT without requiring manual resynchronization. No cascading failures were observed during transient communication disturbances, and sensor streaming did not disrupt the overall ROS–Unity workflow.

5.5.2 Orientation flip in Unity visualization at horizontal pose (observed limitation)

During IMU rotation tests, an orientation jump/flip was observed in the real-time 3D representation in Unity when the sensor was rotated into a near-horizontal pose, corresponding to approximately 90° in roll or pitch under the chosen reference frame. This behavior is consistent with known singularities and discontinuities that can occur when orientation is represented or interpreted using Euler angles (roll–pitch–yaw), where the mapping becomes ill-conditioned at ±90°. This effect is a representation/visualization limitation rather than a sensor hardware fault; quaternion-based handling avoids these discontinuities [20].

5.5.3 Hot-plug and reconnect behavior (observed limitation)

A disconnect/reconnect stress test was performed by unplugging one Feather device during runtime while logging IMU topics. When the shin Feather was disconnected, the corresponding topic stream stopped immediately as expected. However, reconnecting the device did not reliably restore publishing within the same runtime session. In repeated trials, the stream only recovered after a full Raspberry Pi power cycle, indicating that hot-plug auto-reconnect is not yet robust in the current prototype configuration.
As a result, while transient sensor loss does not interrupt the overall ROS–Unity workflow, automatic recovery from USB disconnects is not considered validated. In practice, the platform is operated with both Feather devices connected before launch, and reconnect events are treated as faults requiring a manual restart procedure.

5.6 Summary

Under the default runtime configuration, the results demonstrate that:

- ROS–TCP provides stable bidirectional communication between ROS 2 and the Unity DT and tolerates Unity client restarts.
- The Play subsystem reliably orchestrates start/stop and enforces a latched E-STOP workflow with a safe reset gesture.
- The antagonistic actuator subsystem achieves controlled motion and stable clamp coordination under the default band constraint.
- The pneumatic subsystem provides tunable compliance via closed-loop pressure regulation, with reliable release-air behavior for safety shutdown.
- The Feather subsystem streams IMU and temperature topics continuously during normal operation; however, an orientation flip was observed in the Unity 3D visualization near a horizontal IMU pose, and hot-plug auto-reconnect was not validated and may require a manual restart.

6
Discussion

This chapter discusses the results presented in Chapter 5 in relation to the thesis aims and the broader context of exoskeleton verification. The discussion focuses on the platform's ability to provide a controllable experimental workflow; the utility of tunable interface compliance through pneumatic regulation; and the practical implications of the ROS 2 + Unity DT architecture for experimentation and debugging. Limitations and future improvements are also identified, with emphasis on measurements that can be defended with direct evidence.

6.1 Interpretation of results relative to objectives

The overall aim of this thesis is to develop a RPK testbed and digital twin that can be used for controlled bench-top evaluation of exoskeleton controllers under configurable interaction conditions. The results indicate that the platform meets core requirements for experimentation and safety: subsystems can be started and stopped consistently, Unity can visualize the system state in real time via ROS–TCP, pneumatic pressures can be regulated, and safety behaviors (E-STOP and release-air) behave predictably.
At the same time, the results in Chapter 5 prioritize *verifiable* subsystem-level evidence (logs, plots, and UI states). As a consequence, biomechanical claims about "human-like" joint impedance are treated conservatively: the prototype includes a spring-based elastic mechanism intended to shape resistance with deflection, but the present work does not provide direct joint torque ground truth or an experimentally validated stiffness curve.

6.1.1 Experimental control

A primary advantage of the developed testbed is the ability to apply defined input conditions during bench-top experiments. In practice, this thesis demonstrates system-level operation: start/stop sequences bring the platform to an operational state, state streams remain continuous during typical operation, and safety events return the system to a known safe baseline.
For controller development, this controlled workflow is valuable because it reduces dependence on human subject variability and enables controlled comparisons, for example when testing different assistance profiles or parameter sets under the same commanded trajectories and compliance settings.
However, workflow repeatability is not equivalent to biomechanical realism. Even

if the platform behaves consistently, the extent to which it reproduces human joint impedance depends on the fidelity and validation of the mechanical elastic mechanism and the soft-tissue compliance module. Therefore, the platform should be interpreted as an engineering test rig that captures key interaction mechanisms in a controlled setting, rather than a complete biomechanical replica of the human knee.

6.1.2 Elastic mechanism and stiffness: intent and validation limits

The mechanical design includes an elastic element (spring-based linkage) intended to introduce passive resistance during joint motion. This is a practical design choice for a research testbed: passive elasticity stores energy physically and can reduce reliance on high-bandwidth active control, which can improve safety and reduce the risk of unstable software-only impedance emulation.

In the present prototype, the elastic mechanism is treated as a *hardware feature* that contributes qualitative resistance, but it is not claimed as a validated "non-linear stiffness characterization" result. Two limitations motivated this decision:

- **No joint torque ground truth:** the platform does not currently include an inline torque sensor or load cell at the knee joint, which prevents direct validation of torque–angle behavior.
- **Model-only estimates are insufficient for claims:** stiffness curves derived from spring catalogs and geometry models can be useful for design intuition, but they are not a substitute for experimental identification when the objective is to defend biomechanical realism.

As a result, Chapter 5 focuses on experimentally evidenced outcomes (communication, orchestration, safety, pressure regulation, and streaming robustness). Quantitative joint impedance validation is left as future work once torque sensing is available.

6.1.3 Soft-tissue compliance and pneumatic control

The pneumatic compliance subsystem provides a tunable representation of interface compliance that is difficult to achieve with rigid structures or static soft materials. By controlling chamber pressures, the system can apply configurable compliance conditions and switch between profiles during experiments. This directly supports the use case of exoskeleton controller verification, where interface compliance can strongly affect torque transfer and the apparent stability of the human–robot system.

From the results, the pressure regulation performance (step response plots and tracking metrics) is suitable for bench-top experiments, particularly when experiments focus on quasi-static or moderate-speed interaction. Nevertheless, pneumatic systems have inherent bandwidth limits due to valve dynamics, flow restrictions, and compressibility. This can constrain realism during fast, highly dynamic interactions, and it may limit the ability to reproduce rapid time-varying tissue behavior.

6.1.3.1 Design trade-off: exhaust recirculation versus atmospheric venting

During development, an alternative exhaust topology was evaluated in which chamber deflation flow was routed to a common manifold and fed back to the compressor inlet (exhaust recirculation). In testing, this concept produced *slower* pressure decay than direct venting to atmosphere. The behavior can be explained by the limiting flow element in the exhaust path: when venting to atmosphere, discharge is primarily limited by the vent solenoid valves and downstream tubing, whereas with exhaust recirculation the effective discharge becomes limited by the pump intake characteristics.
A first-order deflation time scale can be expressed as

$$t_{\text{deflate}} \approx \frac{V_{\text{ch}}}{\dot{V}_{\text{eff}}}, \tag{6.1}$$

where V_{ch} is the chamber volume and $\dot{V}_{\text{eff}}$ is the effective exhaust volumetric flow rate. For compact pneumatic solenoid valves and short tubing runs, a conservative order-of-magnitude exhaust capacity in atmospheric venting mode is $\dot{V}_{\text{vent}} \sim 100$ L/min, while small diaphragm pumps of the type used in this platform have free-flow intake rates on the order of $\dot{V}_{\text{pump}} \sim 10$–$15$ L/min. Therefore, if the pump inlet becomes the limiting sink, a corresponding slow-down factor is expected:

$$\frac{t_{\text{pump-return}}}{t_{\text{atmosphere}}} \approx \frac{\dot{V}_{\text{vent}}}{\dot{V}_{\text{pump}}} \sim 7\text{–}10. \tag{6.2}$$

This order-of-magnitude comparison matches the observed outcome that inlet recirculation reduced deflation speed. Consequently, the final design prioritizes predictable and fast deflation by retaining atmospheric venting behavior through a common exhaust manifold.
The release-air behavior is an important practical safety contribution: stored pneumatic energy can be vented in a controlled manner, supporting controlled shutdown and reducing risk during fault conditions. The Unity UI behavior that forces pressure setpoints to 0 psi on E-STOP further strengthens this safety stance by enforcing an explicit operator action before re-pressurization. This is consistent with the design choice to prioritize a low-impedance exhaust path for rapid depressurization.

6.2 Implications for exoskeleton controller validation

The developed platform supports several controller-relevant test scenarios that can be defended with the current evidence base:

- **Compliance sensitivity studies:** By varying pneumatic pressure setpoints while keeping joint commands constant, the system can test how interface compliance influences controller behavior and apparent torque transfer.
- **Robustness under safety constraints:** Controllers can be exercised while the platform enforces safety logic (E-STOP, release-air, UI lockout), enabling

safer early-stage debugging than tests on humans.
- **Structured debugging and visualization:** Unity provides immediate visual feedback of state streams and commanded variables, allowing faster iteration and clearer documentation of experimental conditions.

These scenarios align with the broader motivation of reducing early dependence on human trials and improving the evidence base before clinical evaluation. The testbed can also support dataset generation for controller development in a controlled setting.

6.3 DT and ROS 2 architecture: benefits and trade-offs

A central design choice in this thesis is the combination of a ROS 2-based hardware control stack (device I/O, safety enforcement, and closed-loop control) with a Unity DT for real-time visualization and operator interaction over ROS–TCP. This separation supports structured experiments in which actuator commands and pneumatic compliance setpoints can be adjusted while system state is observed live.

6.3.1 Benefits

The main benefits observed are:
- **Modularity and separation of concerns:** Splitting actuators, pneumatics, sensing, and orchestration into distinct ROS 2 nodes improves maintainability and supports incremental development and debugging.
- **Typed, inspectable interfaces:** Using typed ROS 2 messages and services enables direct verification with standard ROS tools (`ros2 topic echo/hz`, `ros2 service call`) and reduces ambiguity compared to ad hoc string/JSON payloads.
- **Operator-centric visualization and interaction:** Unity provides a convenient front end for monitoring state streams, tuning parameters, and documenting experiments, while tolerating client restarts without requiring a full system restart.

6.3.2 Trade-offs

The main trade-offs include:
- **Latency and non-determinism across the bridge:** ROS–TCP is suitable for visualization, parameter tuning, and supervisory control, but it is not a hard real-time link. Time-critical control loops and safety actions (e.g., E-STOP and venting) should remain on the ROS 2 side.
- **Integration complexity:** A multi-process workflow increases the number of components that must be configured, launched, and monitored. Clear orchestration procedures and reproducible documentation are therefore essential for reliable operation.

Overall, the architecture provides a practical balance between interactive experimentation in Unity and robust hardware control in ROS 2.

6.4 Limitations

Several limitations affect the current platform and the scope of defensible conclusions:

- **Biomechanical fidelity and interface realism:** The phantom captures selected interaction mechanisms, but it does not reproduce neuromuscular dynamics (reflexes, active muscle recruitment). In addition, the pneumatic chambers provide tuneable compliance but do not replicate the anisotropic, viscoelastic, and geometry-dependent behavior of real soft tissue.
- **No joint torque ground truth:** Without direct torque sensing at the knee, joint impedance and stiffness cannot be experimentally validated, limiting claims about "human-like" stiffness behavior.
- **Pneumatic bandwidth:** Pressure regulation has limited speed and may not reproduce fast tissue dynamics during highly dynamic interactions.
- **Unity orientation visualization discontinuity:** An orientation jump/flip was observed in the real-time 3D representation in Unity when the IMU passed through a near-horizontal pose (corresponding to $\approx 90^\circ$ in roll or pitch under the chosen frame). This behavior is consistent with Euler-angle singularities/discontinuities in roll–pitch–yaw representations; quaternion-based handling avoids such artifacts [20].
- **Single-joint scope:** The platform focuses on knee flexion/extension. Multi-joint coupling and gait dynamics are not represented.
- **Knee kinematic simplification (hinge approximation):** The human knee exhibits rolling–sliding motion with a migrating instantaneous center of rotation [21]. The prototype models the knee as a single-axis hinge with a fixed center of rotation, which improves robustness and mechanical simplicity but reduces kinematic fidelity and may contribute to joint–exoskeleton misalignment. Polycentric/variable-center knee mechanisms reported in the literature [22] were not implemented in the present platform.

6.5 Future work

Several directions can extend and strengthen the platform:

- **Direct torque sensing:** Integrate a torque sensor or load cell to obtain joint torque ground truth and enable experimentally validated impedance and stiffness characterization.
- **System identification:** Perform identification experiments to map actuator commands and pneumatic pressures to effective joint response, enabling higher-fidelity benchmarking and stronger quantitative claims.
- **Enhanced tissue models:** Extend the compliance subsystem with multi-chamber or region-specific elements and incorporate viscoelastic effects to better emulate soft-tissue behavior.

- **Standardized experimental protocols:** Develop standardized benchmarking scripts (e.g., disturbance tests, compliance sweeps, and safety-event tests) with automated logging and figure generation.
- **Improved sensor robustness:** Increase resilience to USB disconnects by addressing devices using persistent identifiers (e.g., `/dev/serial/by-id` or`/dev/serial/by-path`) rather than `/dev/ttyACM*`, and extend reconnect logic to handle port re-enumeration. In addition, remove the Unity visualization flip near horizontal IMU poses by using quaternion-based orientation handling end-to-end and restricting Euler-angle conversion to UI display only [20].
- **Extended DT tooling:** Add automated experiment logging, playback, and parameter-sweep tools in Unity to support systematic studies.
- **Polycentric knee mechanism:** Incorporate a polycentric knee joint (e.g., cam-based or multi-link) to better approximate the migrating instantaneous center of rotation observed in the human knee while preserving the controlled operation required for controller verification.

6.6 Summary

The RPK platform provides a controlled experimental environment based on a ROS 2 hardware stack and a Unity DT for real-time visualization and interaction. The system integrates antagonistic actuation, tuneable pneumatic compliance, and a safety workflow that coordinates actuator soft-stop behavior, pneumatic venting, and operator-interface lockout. The results support evidence-based conclusions about system integration, safety, and pressure-based compliance control. Quantitative validation of joint impedance and stiffness is intentionally deferred until direct torque sensing and experimental identification are available, providing a clear path to strengthen biomechanical fidelity in future work.

7
Conclusion

This thesis presented the design and development of a RPK platform and a Unity-based DT intended to support controlled evaluation of wearable exoskeleton controllers under configurable interaction conditions. The motivation was to reduce reliance on early-stage human testing by providing a controlled experimental environment with antagonistic actuation, passive elasticity, and tuneable interface compliance.

Summary of work

The developed system integrates:

- a mechanically constrained single-axis knee hinge for controlled flexion/extension,
- antagonistic actuation using two Koala BEAR linear actuators coupled through a spring-based linkage that introduces passive resistance and enables stiffness shaping via preload,
- a pneumatic compliance subsystem based on inflatable chambers, supply/vent solenoid valves, a pump and reservoir, and pressure sensing for closed-loop regulation,
- a modular ROS 2 (Jazzy) software stack on a Raspberry Pi 5, providing typed interfaces, subsystem separation, and safety handling,
- a Unity DT connected via ROS–TCP for real-time visualization, operator interaction, and parameter tuning.

Main contributions

The main contributions of this work are:

- A hardware test platform that combines a rigid hinge, antagonistic actuation, and a spring-based elastic element to provide controllable motion with passive resistance relevant to wearable interaction studies.
- A tuneable pneumatic compliance module with closed-loop pressure regulation and a safety-oriented release-air function for controlled depressurization.
- A ROS 2-based integration architecture with typed messages and services that enables structured subsystem control, monitoring, and safety workflows.
- A Unity DT that provides real-time visualization and an operator-facing interface for running experiments, triggering safety actions (E-STOP and venting), and documenting runtime states.

Concluding remarks

The results show that the integrated platform supports a controlled experimental workflow. ROS–TCP communication between ROS 2 and Unity was stable under runtime operation and tolerated Unity client restarts. The Play subsystem provided consistent start/stop behavior and enforced a latched E-STOP workflow that coordinated actuator soft-stop behavior, pneumatic venting, and operator-interface lockout. The antagonistic actuation subsystem demonstrated controlled motion under the implemented coordination constraint, and the pneumatic subsystem tracked commanded pressure sequences and executed rapid release-air shutdowns.
The platform is therefore best interpreted as an engineering test rig that enables controlled experiments on interaction-relevant mechanisms (antagonistic actuation, passive elasticity, and tuneable interface compliance), rather than as a fully validated biomechanical replica of the human knee. In particular, quantitative claims about joint impedance and "human-like" stiffness are intentionally treated conservatively because direct joint torque ground truth was not available in the present prototype.

Future work

Future development could strengthen fidelity and broaden applicability by:

- integrating direct joint torque sensing (torque sensor or load cell) to enable experimentally validated torque–angle and impedance characterization,
- performing system identification to map actuator commands and pneumatic pressures to effective joint response under defined test conditions,
- expanding the soft-tissue module with multi-chamber or region-specific compliance and viscoelastic effects,
- standardizing controller benchmarking protocols with automated logging, plotting, and parameter sweeps,
- extending the DT with experiment playback and automated reporting, and ensuring robust quaternion-based orientation handling to avoid visualization discontinuities near Euler singularities.

Bibliography

[1] N. Hogan, "Adaptive control of mechanical impedance by coactivation of antagonist muscles," *IEEE Transactions on Automatic Control*, vol. 29, no. 8, pp. 681–690, 1984. [Online]. Available: https://doi.org/10.1109/TAC.1984.1103644

[2] B. Vanderborght, A. Albu-Schäeffer, A. Bicchi *et al.*, "Variable impedance actuators: A review," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1601–1614, 2013. [Online]. Available: https://doi.org/10.1016/j.robot.2013.06.009

[3] S. J. Filius, B. J. van der Burgh, and J. Harlaar, "The design of the dummy arm: A verification tool for arm exoskeleton development," *Biomimetics*, vol. 9, no. 10, p. 579, 2024. [Online]. Available: https://doi.org/10.3390/biomimetics9100579

[4] W. S. Barrutia, J. Bratt, and D. P. Ferris, "A human lower limb mechanical phantom for the testing of knee exoskeletons," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2023. [Online]. Available: https://doi.org/10.1109/TNSRE.2023.3276424

[5] M. Dezman, S. Massardi, D. P. Fernández, and D. Torricelli, "A mechatronic leg replica to benchmark human-exoskeleton physical interactions," *Bioinspiration & Biomimetics*, vol. 18, no. 3, p. 036009, 2023. [Online]. Available: https://doi.org/10.1088/1748-3190/accda8

[6] Z. Li, X. Chu, X. Hu, Z. Zhang, N. Li, and J. Li, "Variable stiffness methods for robots: a review," *Smart Materials and Structures*, vol. 33, no. 6, p. 063002, may 2024. [Online]. Available: https://doi.org/10.1088/1361-665X/ad0753

[7] B. Vanderborght, A. Albu-Schaeffer, A. Bicchi, E. Burdet, D. Caldwell, R. Carloni, M. Catalano, O. Eiberger, W. Friedl, G. Ganesh, M. Garabini, M. Grebenstein, G. Grioli, S. Haddadin, H. Hoppner, A. Jafari, M. Laffranchi, D. Lefeber, F. Petit, S. Stramigioli, N. Tsagarakis, M. Van Damme, R. Van Ham, L. Visser, and S. Wolf, "Variable impedance actuators: A review," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1601–1614, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889013001188

[8] M. Grieves and J. Vickers, "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems," in *Transdisciplinary Perspectives on Complex Systems*, F.-J. Kahlen, S. Flumerfelt, and A. Alves, Eds. Cham: Springer International Publishing, 2017, pp. 85–113. [Online]. Available: https://doi.org/10.1007/978-3-319-38756-7_4

[9] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital twin in manufacturing: A categorical literature review and classification," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, 2018, 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896318316021

[10] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, and W. Woodall, "Robot operating system 2: Design, architecture, and uses in the wild," *Science Robotics*, vol. 7, no. 66, p. eabm6074, 2022. [Online]. Available: https://doi.org/10.1126/scirobotics.abm6074

[11] M. Singh, J. Kapukotuwa, E. L. S. Gouveia, E. Fuenmayor, Y. Qiao, N. Murry, and D. Devine, "Unity and ros as a digital and communication layer for digital twin application: Case study of robotic arm in a smart manufacturing cell," *Sensors*, vol. 24, no. 17, p. 5680, 2024. [Online]. Available: https://doi.org/10.3390/s24175680

[12] R. W. Erickson and D. Maksimović, *Fundamentals of Power Electronics*, 3rd ed. Cham: Springer, 2020.

[13] L. Piegl and W. Tiller, *The NURBS Book*, 2nd ed. Berlin, Heidelberg: Springer, 1997.

[14] Unity Technologies, "Unity Manual: Level of Detail (LOD) Group," Online documentation, 2025, accessed: 2026-01-09. [Online]. Available: https://docs.unity3d.com/Manual/class-LODGroup.html

[15] Westwood Robotics, "Bear series product page," Web page, 2025, accessed: 2025-12-28. [Online]. Available: https://www.westwoodrobotics.io/bearseries/

[16] ——, *BEAR SDK Manual (English) v1.1.0*, Westwood Robotics, Aug. 2025, accessed: 2025-12-28. [Online]. Available: https://www.westwoodrobotics.io/wp-content/uploads/2025/08/BEAR_SDK_Manual_EN_v1.1.0.pdf

[17] ——, "Pybear (version 0.1.3)," GitHub repository, 2025, accessed: 2025-12-28. [Online]. Available: https://github.com/Westwood-Robotics/PyBEAR/tree/0.1.3

[18] D. A. Winter, *Biomechanics and Motor Control of Human Movement*, 4th ed. John Wiley & Sons, 2009.

[19] Lesjöfors, "Flat spiral spring scs 1.5×15×5 (code 925)," Product catalog page, 2025. [Online]. Available: https://shop.lesjofors.com/se/en-US/all-products/flat-springs/scs-15x15x5-925/

[20] J. Diebel, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," Stanford University, Tech. Rep., Oct. 2006, technical report, dated 20 October 2006. [Online]. Available: https://www.astro.rug.nl/software/kapteyn-beta/_downloads/attitude.pdf

[21] S. Koo and T. P. Andriacchi, "The knee joint center of rotation is predominantly on the lateral side during normal walking," *Journal of Biomechanics*, vol. 41, no. 6, pp. 1269–1273, 2008. [Online]. Available: https://doi.org/10.1016/j.jbiomech.2008.01.013

[22] K. Liu *et al.*, "Design and optimization of an adaptive knee joint orthosis with a variable rotation center for biomimetic motion rehabilitation

assistance," *Biomimetics*, vol. 9, no. 2, p. 98, 2024. [Online]. Available: https://doi.org/10.3390/biomimetics9020098

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY