



**HAL**  
open science

# Control Architecture for Electro-Hydraulic Humanoid Robot

Subhi Jleilaty

► **To cite this version:**

Subhi Jleilaty. Control Architecture for Electro-Hydraulic Humanoid Robot. Automatic. Université Paris-Saclay, 2024. English. NNT : 2024UPAST010 . tel-04524748

**HAL Id: tel-04524748**

**<https://theses.hal.science/tel-04524748v1>**

Submitted on 28 Mar 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Control Architecture for Electro-Hydraulic Humanoid Robot (HYDROïD)

*Architecture de contrôle pour un robot humanoïde électro-hydraulique  
(HYDROïD)*

## Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 580, sciences et technologies de l'information et de la  
communication (STIC)  
Spécialité de doctorat : Robotique  
Graduate School : Informatique et sciences du numérique. Référent : Université d'Évry Val  
d'Essonne

Thèse préparée dans l'unité de recherche **IBISC (Université Paris-Saclay, Univ Evry)**,  
sous la direction de **Lydie NOUVELIERE, MCF-HDR** et, la co-direction de **Samer  
ALFAYAD, Professeur**.

Thèse soutenue à Paris-Saclay, le 20 février 2024, par

**Subhi JLEILATY**

## Composition du Jury

Membres du jury avec voix délibérative

<b>Eric MONACELLI</b> Prof., Versailles Saint-Quentin Yvelines	Président
<b>Mohamad KHALIL</b> Prof., Lebanese University	Rapporteur & Examineur
<b>Thibaut RAHARIJAONA</b> Prof., Engineering School ENIM Metz	Rapporteur & Examineur
<b>Lama ALBASSIT</b> Dr., French National Institution	Examineur
<b>Antoine DEQUIDT</b> MCF, Polytechnic University	Examineur

**Titre :** Architecture de contrôle pour un robot humanoïde électro-hydraulique (HYDROiD)

**Mots clés :** architecture de contrôle, robot humanoïde, électro-hydraulique

**Résumé :** Malgré les progrès significatifs réalisés jusqu'à présent dans le contrôle des robots humanoïdes, ceux-ci sont encore loin de présenter de manière fiable des comportements semblables à ceux des humains. Les différents éléments qui composent les humanoïdes contribuent à l'amélioration de leurs comportements. Alors que l'augmentation de la limitation physique contribue aux mouvements dynamiques, l'architecture de contrôle est la clé pour gérer ces mouvements et déterminer les capacités du robot afin d'améliorer ses comportements. Ce travail vise à développer une architecture de contrôle émulant la fonctionnalité du système nerveux humain. Les architectures classiques traitaient des cycles sensorimoteurs mais sans distribution de l'intelligence. Qu'elles soient centralisées,

où tous les composants sont connectés à une unité centrale, ou décentralisées, où les unités distribuées servent d'interface entre les E/S et le contrôleur principal sans pouvoir prendre de décision. La solution proposée est une architecture de contrôle en temps réel distribuée avec ROS. Les contrôleurs conjoints ont l'intelligence de prendre des décisions, de dominer leurs actionneurs et de publier leur état. Les validations expérimentales ont été effectuées sur notre robot humanoïde électro-hydraulique (HYDROiD). Les résultats démontrent des avancées de 50 % dans le taux de mise à jour par rapport à d'autres humanoïdes et de 30 % dans la latence du processeur principal et des tâches de contrôle. L'architecture proposée permet de créer et de tester des systèmes à intelligence artificielle distribuée.

**Title :** Control Architecture for Electro-Hydraulic Humanoid Robot (HYDROiD)

**Keywords :** Control Architecture, Humanoid Robot, Electro-Hydraulic

**Abstract :** E Despite the significant improvements achieved until now in controlling humanoid robots, they are still a long way from reliably exhibiting human-like behaviors. Various components that formulate humanoids contribute to improving their behaviors. While increasing the physical limitation contributes to dynamic motions, the control architecture is the key to managing those motions and determining robot capabilities to improve its behaviors. This work aims to develop a control architecture emulating the functionality of the human nervous system. Classical architectures dealt with sensorimotor cycles but without intelligence distribution. Whether centralized, where all components are connected to a central unit, or decentralized, where the distributed units are used as an interface between the I/Os and the master controller with no ability to make a decision.

The proposed solution is a distributed real-time control architecture with ROS. The joint controllers have the intelligence to make decisions, dominate their actuators, and publish their state. The real-time capabilities are ensured in the master controller by using a Preempt-RT kernel beside OROCOS middleware to operate the real-time tasks. And in the customized joint controllers by FreeRTOS firmware. The experimental validations were performed on our electro-hydraulic humanoid robot (HYDROiD). The results demonstrate 50% advancements in the update rate compared to other humanoids and 30% in the latency of the master processor and the control tasks. The proposed architecture gives the possibility to create and test the systems with distributed artificial intelligence.





## Résumé

Alors que le monde de la robotique continue de sortir des produits avec des capacités de type humain, les robots humanoïdes ont été témoins d'une propagation remarquable dans des applications réelles et des environnements industriels. Les robots humanoïdes sont ciblés pour interagir avec les humains, naviguer et travailler dans des environnements adaptés aux humains. Par conséquent, ils doivent imiter non seulement l'anthropomorphisme, la locomotion, les manipulations et les comportements des humains, mais aussi la façon dont les humains les gèrent. Malgré les améliorations significatives réalisées jusqu'à présent dans le contrôle de ces robots, ils sont encore loin de montrer de manière fiable des comportements de type humain. Divers composants qui constituent les humanoïdes contribuent à améliorer leurs comportements. Le système d'actionnement imite les muscles humains; ces actionneurs devraient pouvoir fournir des forces élevées dans de petits volumes. Les systèmes d'actionnement électrique et hydraulique sont à la pointe de l'actionnement de ces robots. Au cours des cinq dernières années, le marché des robots électriques s'est rapidement développé en raison du processus de mise en œuvre rapide et de la facilité de contrôle de ces robots tels que Optimus-2022, Cyber-one-2022, Apollo-2023, GR-1-2023 de Fourier Intelligence et Unitree H1-2023. Ces robots sont encore limités à effectuer des activités lourdes avec des mouvements dynamiques.

D'autre part, il y a très peu de recherche sur le système d'actionnement hydraulique dans le monde, comme Boston Dynamics, Kalysta et Sarcos Robotics. Les actionneurs hydrauliques ont augmenté les limites physiques du robot en fournissant des mouvements dynamiques élevés tout en effectuant des activités lourdes comme Atlas-2017. Ces actionneurs présentent la capacité unique de maintenir efficacement les charges à l'état statique des articulations et de minimiser la consommation d'énergie. Selon ces considérations, notre recherche a choisi des actionneurs hydrauliques pour développer des robots humanoïdes. Même si le système hydraulique ordinaire utilise un groupe hydraulique centralisé pour entraîner les articulations actives uniformément, négliger le fait que tous les articulations n'exigent

pas la même force entraîne une perte de puissance importante. En outre, d'autres défis rencontrés dans le développement de robots humanoïdes avec des actionneurs hydrauliques sont représentés par i) la difficulté de contrôle de ces actionneurs, ii) l'encombrement en raison de l'utilisation des tuyaux hydrauliques pour livrer le fluide aux articulations, iii) la complexité de la mise en œuvre.

Développer un actionneur hybride, surmonter les inconvénients et combiner les avantages des technologies électriques et hydrauliques est la clé pour relever certains défis fondamentaux dans le domaine. Le servomoteur électro-hydraulique (SEHA) est un servomoteur « Plug and Play » complet. Il offre les meilleures performances du marché en termes de compacité (rapports Force/Volume et Force/Poids). L'utilisation des technologies hydrauliques et électriques mélangées dans un dispositif compact permet de fournir en même temps les performances de la technologie hydraulique et les performances de la technologie électrique (prix, facilité d'utilisation et de contrôle, et entretien). Il comprend i) un module de production d'énergie nommé Mini Group Hydraulic (MGH), ii) un actionneur hydraulique hautement instrumenté nommé Servo Intelligent Cylinder (SIC), iii) des blocs de circuits hydrauliques, iv) un retour hydraulique interne pour un fonctionnement sûr, v) les capteurs requis pour un contrôle raisonnable, et vi) les cartes électroniques pour créer une unité autonome efficace.

Alors que SEHA facilite la conception mécanique et les systèmes d'actionnement en remplaçant la pompe hydraulique centrale par des actionneurs électro-hydrauliques locaux qui intègrent les tubes hydrauliques et contribuent aux mouvements dynamiques du robot, l'architecture de contrôle est la clé pour gérer ces mouvements et déterminer les capacités du robot pour améliorer ses comportements.

Cette thèse se concentre sur le développement d'une architecture de contrôle humanoïde basée sur SEHA qui émule la fonctionnalité du système nerveux humain. Les architectures classiques traitent des cycles sensorimoteurs mais pas de la distribution de l'intelligence. Qu'ils soient centralisés, où tous les composants sont connectés à une unité centrale, ou décentralisés, où les unités distribuées sont utilisées comme interface entre les E/S et le contrôleur maître sans pouvoir

prendre de décision. La solution proposée est une architecture de contrôle temps réel distribuée. Les contrôleurs interarmées ont l'intelligence de prendre une décision, de piloter les actionneurs interarmées et d'informer le contrôleur principal de l'état de l'articulation. La boucle de commande sensorimotrice pourrait être déployée à la fois dans le contrôleur commun et maître. Les capacités en temps réel sont assurées dans le contrôleur maître en utilisant un noyau Preempt-RT à côté du middleware OROCOS pour exécuter les tâches en temps réel, accompagné de quatre contrôleurs conjoints personnalisés exploités par le firmware FreeRTOS. Les validations expérimentales ont été réalisées sur le IBISC de robot humanoïde électro-hydraulique (HYDROïD). Les résultats démontrent une progression de 50% du taux de mise à jour par rapport aux autres humanoïdes et de 30% de la latence du processeur maître et des tâches de contrôle. L'architecture proposée donne la possibilité de créer et de tester les systèmes avec l'intelligence artificielle distribuée.

Les principales contributions de cette thèse sont:

- Architecture de contrôle humanoïde basée sur SEHA: émulation du système nerveux humain dans la distribution de l'information. Il simplifie la réplique en utilisant des données biomédicales mesurées des mouvements humains dans son algorithme de mouvement. Il améliore le système de contrôle global du robot en contrôlant les mécanismes hybrides localement. Il permet de créer et de tester des systèmes à intelligence artificielle distribuée.
- Amélioration du contrôle SEHA: identification du système pour trouver le modèle de système basé sur l'architecture de contrôle développée.
- Conception mécatronique: recherche préalable et spécifications imposées en termes de i) largeur de bande de communication, ii) contrôle principal. la capacité de le faire s'intégrer de manière transparente avec d'autres sous-systèmes, et iii) le temps de cycle, ont permis de mettre au point un système flexible et très fiable.
- Prototypes de contrôleurs de fabrication pour SEHA actionneur et ses modules séparément. Conception d'une interface utilisateur graphique pour à

des fins de test. Un test de latence à long terme a été exécuté pour le contrôle principal Composants d'architecture (contrôleur maître et réseau). Un système de contrôle Les bancs d'essai sont conçus et exécutés pour valider le module SEHA basé sur Normes ISO ; ISO 4409:2019, pour le Mini Groupe Hydraulique.

- Démonstration d'une expérience de marche sur le bas du corps du HYDROiD à quatre DDL présente du contrôle du mécanisme de la hanche.

## Abstract

As the robotics world continues to bring out products with human-like abilities, humanoid robots have witnessed a remarkable spread in real-world applications and industrial environments. Humanoid robots are targeted to interact with humans, navigate, and work in environments adapted to humans. Therefore, they must imitate not only humans' anthropomorphism, locomotion, manipulations, and behaviors but also how humans manage those to achieve the highest degree of human-like emulation. Despite the significant improvements made so far in controlling these robots, they are still a long way from reliably exhibiting human-like behaviors. Various components that formulate humanoids contribute to improving their behaviors. The actuation system emulates human muscles; the actuators should be able to provide high forces in small volumes. The electric and hydraulic actuation systems are leading in actuating these robots. In the last five years, the electric-based robots market has been rapidly growing due to the fast implementation process and the ease of control of these robots like Optimus-2022, Cyper-one-2022, Appolo-2023, Fourier Intelligence's GR1-2023, and Unitree H1-2023. However, these robots are still limited to performing heavy-duty activities with dynamic motions. On the other hand, there is very limited research on hydraulic actuation systems in the world, like Boston Dynamics, Kalysta, IIT, and Sarcos Robotics. Hydraulic actuators increase the robot's physical limitations by providing high dynamic motions while performing heavy-duty activities. These actuators exhibit a unique ability to hold loads efficiently during the static state of joints and minimize power consumption. According to these considerations, our research uses hydraulic actuators to develop humanoid robots. The ordinary hydraulic system employs a centralized hydraulic power unit to drive active joints uniformly, neglecting that not all joints require the same force, resulting in a substantial power loss. Furthermore, some other challenges faced in developing humanoid robots with hydraulic actuators are represented by i) the complexity of controlling these actuators, ii) the bulkiness because of using the hydraulic pipes to deliver the fluid to the joints, and iii) the implementation complexity.

In order to avoid the disadvantage of centralized hydraulic actuation, a decentralized electro-hydraulic actuator is required to be developed. Our research team patented (WO2020173933A1) and developed a Servo Electro-Hydraulic Actuator (SEHA). This hybrid actuator combines the advantages of electric and hydraulic technologies, attempting to address some fundamental challenges in the field. SEHA is a full-featured “Plug and Play” actuator. It offers the highest market performance in compactness (Force/Volume and Force/weight ratios). Using the hydraulic and electric technologies mingled in one compact device delivers the performance of hydraulic and electric technologies (price, ease of use and control, and maintenance). It is a modular actuator that consists of i) a power generation module named Main Group Hydraulic (MGH), ii) a highly instrumented hydraulic actuator named Servo Instrumented Cylinder (SIC), iii) hydraulic circuit blocks, iv) internal hydraulic feedback for safe operation, v) required sensors for reasonable control, and vi) electronic cards to create an effective standalone unit.

While SEHA facilitates the mechanical design and actuation systems by replacing the central hydraulic pump with local electro-hydraulic actuators that integrate the hydraulic tubes and contribute to the robot’s dynamic motions, the control architecture is the key to managing those motions and determining robot capabilities to improve its behaviors.

This thesis focuses on developing SEHA-based humanoid control architecture that emulates the functionality of the human nervous system. The classical architectures deal with the sensorimotor cycles but not with the distribution of intelligence. Whether centralized, where all components are connected to a central unit, or decentralized with very limited coordination between the control units, or even the distributed control system that uses the units as an interface between the I/Os and the master controller with no ability to make a decision. The proposed solution is a distributed real-time control architecture. The joint controllers have the intelligence to make a decision, dominate the joint actuators, and inform the master controller about the joint’s state. The sensorimotor loops could be deployed in both the joint and master controller. The real-time capabilities are ensured in the master controller by using a Real-Time Operating System

(RTOS) beside middleware operating a hard real-time communication system, accompanied by customized joint controllers operated by RTOS for embedded systems. The experimental validations were performed on our electro-hydraulic humanoid robot (HYDROiD). The results demonstrated 50 % advancement in the update rate compared to other humanoids and 30 % in the latency of the master processor and the control tasks. The developed architecture allows creating and testing of systems with distributed artificial intelligence.

The main contributions of this thesis are:

- **SEHA-based humanoid control architecture:** emulating human's nervous system in the intelligence distribution. It simplifies replicating human motions by employing measured biomedical data of human motions in its motion algorithms. It improves the overall robot's control system by locally controlling the hybrid mechanisms. It allows the creation and testing of systems with distributed artificial intelligence.
- **Control architecture mechatronic design:** The research identified and described specifications and constraints of the key software and hardware control architecture components: i) master controller real-time capability and ability to seamlessly integrate with other subsystems, ii) joint controllers distribution and capability to execute adaptive control strategies, iii) communication interface, and iv) cycle time, have collectively led to the development of a flexible and highly dependable system.
- **Experimental validation:** Manufacturing controllers' prototypes for SEHA actuator and its modules separately. Designing a graphic user interface for testing purposes. A long-term latency test was executed for the main control architecture parts (Master controller and network). A control system for the test benches is designed and executed to validate SEHA module based on ISO standards; ISO 4409:2019, for the Main Group Hydraulic.

- **SEHA control Enhancement:** System identification was conducted to find the system model based on the developed control architecture.
- **Demonstration:** for a walking experiment of the HYDROiD's lower body with four DoFs presents controlling the hip mechanism.



# Contents

<b>Introduction</b>	<b>1</b>
<b>1 Biomedical Study</b>	<b>7</b>
1.1 Introduction . . . . .	8
1.2 Research Questions and Objectives . . . . .	9
1.3 Human Motions Control Exploration . . . . .	10
1.4 Data Collection and Methods . . . . .	15
1.5 Biomedical Study Outcomes and Results . . . . .	21
1.5.1 Human Control System . . . . .	21
1.5.2 Gait Data analysis . . . . .	21
1.6 Outcomes Discussion . . . . .	23
1.6.1 Implications of Research Findings in the Context of Research Questions: . . . . .	23
1.6.2 Significance of Distributed Decision-Making Centers in the Human Body: . . . . .	24
1.6.3 Challenges and Implications of Replicating Human Joint Motions on Humanoid Robots: . . . . .	24
1.7 Conclusion . . . . .	25

<b>2</b>	<b>State of the Art</b>	<b>27</b>
2.1	Introduction . . . . .	27
2.1.1	Importance of Bipedal Robots . . . . .	28
2.1.2	Humanoid robot challenges . . . . .	29
2.2	Historical evolution of humanoid robots . . . . .	30
2.2.1	Humanoid Control Architectures . . . . .	44
2.2.2	Communication Technologies . . . . .	47
2.2.3	Software and Middleware . . . . .	48
2.2.4	Joint Controllers . . . . .	50
2.2.5	Human-Like Anthropomorphism in Humanoids . . . . .	52
2.2.6	Actuation Technologies . . . . .	53
2.2.7	Applications And Control Architecture Variations . . . . .	57
2.3	Conclusion . . . . .	59
<b>3</b>	<b>Servo Electro-Hydraulic Actuator (SEHA)</b>	<b>61</b>
3.1	Introduction . . . . .	62
3.2	SEHA Development . . . . .	63
3.2.1	SEHA Overview . . . . .	63
3.2.2	SEHA Characteristics . . . . .	63
3.2.3	SEHA Functional Structure . . . . .	64
3.3	SEHA Modules . . . . .	67
3.3.1	Hydraulic Power Generation Module . . . . .	68
3.3.2	Servo Speed Control Module . . . . .	70
3.3.3	Servo Force Compensation Module . . . . .	72
3.3.4	New efficient hydraulic circuit . . . . .	73

3.3.5	Output hydraulic cylinder . . . . .	75
3.4	SEHA Functioning Principle . . . . .	78
3.5	Conclusion . . . . .	80
<b>4</b>	<b>SEHA-Based HYDROïD Control Architecture - Approach and Development</b>	<b>83</b>
4.1	Introduction . . . . .	84
4.2	Problematic . . . . .	85
4.3	Approach . . . . .	85
4.3.1	HYDROïD Robot State . . . . .	87
4.3.2	Performance Requirements . . . . .	89
4.3.3	Control Architecture' Levels . . . . .	91
4.4	Control Architecture Hardware Development . . . . .	92
4.4.1	Master Controller . . . . .	93
4.4.2	Communication Protocol . . . . .	94
4.4.3	Distributed joint controllers and their components . . . . .	96
4.5	Control Architecture Software Development . . . . .	99
4.5.1	Master Controller Software . . . . .	100
4.5.2	Joint Controller Software . . . . .	101
4.6	Conclusion . . . . .	102
<b>5</b>	<b>SEHA-Based HYDROïD Control Architecture Realization and Validation</b>	<b>103</b>
5.1	Introduction . . . . .	104
5.2	Main Hydraulic Group (MGH) Tests and Validation . . . . .	104
5.2.1	Board Development . . . . .	105

5.2.2	Testing SEHA Controller . . . . .	107
5.2.3	ISO Test Control System . . . . .	115
5.3	Servo-Instrumented Cylinder Tests and Validation . . . . .	119
5.3.1	Board Development . . . . .	119
5.3.2	SIC Controller Servo Valve Tests . . . . .	120
5.3.3	SIC Controller Pressure Sensors Tests . . . . .	124
5.3.4	SIC Controller Position Sensor Tests . . . . .	125
5.4	System Identification For Servo Instrumented Cylinder . . . . .	127
5.4.1	System Essential Dynamics . . . . .	130
5.4.2	Experiment Setup and Data Collection . . . . .	133
5.4.3	Model Structure Selection . . . . .	139
5.4.4	Model Identification . . . . .	139
5.4.5	Model Validation . . . . .	141
5.5	SEHA Control Architecture tests and results . . . . .	143
5.5.1	Master controller performance test . . . . .	143
5.5.2	Network performance test . . . . .	145
5.5.3	Walking gait Experiment . . . . .	146
5.6	Conclusion . . . . .	147
<b>6</b>	<b>Conclusions and Perspectives</b>	<b>151</b>
6.1	Conclusions . . . . .	151
6.2	Perspectives . . . . .	154
	<b>Publications</b>	<b>155</b>
	<b>Bibliographie</b>	<b>156</b>

<b>A SEHA Control System Design Documents</b>	<b>175</b>
A.1 Sequence of SEHA Operations . . . . .	175
A.2 SEHA Control Mode Selection . . . . .	179
A.2.1 Control Mode Description . . . . .	179
A.3 GUI Performance and Requirements . . . . .	184
<b>B SIC Control System Design Documents</b>	<b>193</b>
B.1 SIC Control Mode Selection . . . . .	193



# List of Figures

1.1	Simplest reflex establishes direct connections between the sensory routes and motor pathway [6] . . . . .	11
1.2	Summary of the somatosensory and vestibular sensory pathways and their integration into the brain and spinal cord. [9] . . . . .	14
1.3	Hips motion angles in degrees during four steps of a healthy person with 63 cm step length at 100 cm/s velocity . . . . .	16
1.4	Knees motion angles in degrees during four steps of a healthy person with 63 cm step length at 100 cm/s velocity . . . . .	16
1.5	Ankles motion angles in degrees during four steps of a healthy person with 63 cm step length at 100 cm/s velocity . . . . .	17
1.6	Arms motion angles in degrees during four steps of a healthy person with 63 cm step length at 100 cm/s velocity . . . . .	17
1.7	Hips motion forces during four steps of a healthy person with 63 cm step length at 100 cm/s velocity . . . . .	18
1.8	Knees motion forces during four steps of a healthy person with 63 cm step length at 100 cm/s velocity . . . . .	18
1.9	Ankles motion forces during four steps of a healthy person with 63 cm step length at 100 cm/s velocity . . . . .	19
1.10	Hips motion moments during four steps of a healthy person with 63 cm step length at 100 cm/s velocity . . . . .	19

1.11	Knees motion moments during four steps of a healthy person with 63 cm step length at 100 cm/s velocity . . . . .	19
1.12	Ankles motion moments during four steps of a healthy person with 63 cm step length at 100 cm/s velocity . . . . .	20
1.13	Hips-Knees-Ankles Motion Powers during four steps of a healthy person with 63 cm step length at 100 cm/s velocity . . . . .	20
2.1	The figure illustrates the developed humanoid robots starting from the WABOT project to the ASIMO robot and some of their characteristics . . . . .	32
2.2	Humanoid Robot Project HRP series robots since the primary versions with their characteristics and the employed technological solutions . . . . .	34
2.3	The Korea Advanced Institute of Science and Technology (KAIST) Project's series robots since the primary versions with their characteristics and the employed technological solutions . . . . .	36
2.4	The iCub. The figure shows a photograph of the iCub robot [34] . . . . .	37
2.5	Some of the advanced robots that electrically actuated are developed by various projects . . . . .	40
2.6	Hydraulically Actuated Robots . . . . .	43
2.7	Commercial Robots . . . . .	44
3.1	SEHA cross-section; the figure illustrates the actuator modules structure and their placements. . . . .	64
3.2	Functional structure of SEHA- 1- The power generation module: hydraulic pump and BLDC motor, 2- Servo-speed control module: V2 a 4/3 directional valve and C4/C5 a servo-hydraulic cylinder, M2 a linear motor, 3- Servo-force compensation module: V3 a 10/2 directional valve, M3 a liner motor, K-M3 a spring, 4- Hydraulic circuit, 5- Output cylinder [114] . . . . .	66



3.3	SEHA power generation module as a stand-alone Mini Group Hydraulic (MGH) . . . . .	68
3.4	SEHA-MGH parts: BLDC motor and micro axial piston pump . . . . .	69
3.5	SEHA-MGH manufactured controller: BLDC motor driver, stepper motor driver for the servo-speed control, voice-coil motor driver for the servo-force compensation, positions, pressures, and temperature sensors' inputs and EtherCAT interface ports. . . . .	70
3.6	Servo speed control module connected to the MGH swashplate $C_0$ , a directional valve $V_2$ acting on a servo-hydraulic cylinder $C_4/C_5$ with mechanical feedback and controlled by linear motor $M_2$ . . . . .	71
3.7	SEHA-Servo-force compensation module in the deactivated state. . . . .	72
3.8	SEHA-Servo-force compensation module in activation mode, the valve $V_3$ is actuated to reset the positive swashplate's angle $\alpha$ via the valve $V_1$ . . . . .	73
3.9	SEHA-Servo-force compensation module in activation mode, the valve $V_3$ is actuated to reset the negative swashplate's angle $\alpha$ via the valve $V_1$ . . . . .	74
3.10	SEHA-SIC module: Servo Instrumented Cylinder with the integrated servo-valve and the module controller . . . . .	76
3.11	SEHA-SIC module: Integrated position sensor . . . . .	77
3.12	SEHA-SIC module: Integrated full-bridge strain gauge sensor placed on the cylinder rod's end [125] . . . . .	77
3.13	SEHA-SIC module: Integrated pressures (1,2) and oil temperature (3) sensors . . . . .	78
3.14	The main stages for SEHA's mathematical modeling . . . . .	79
4.1	Hydroid mechanisms and corresponding DOFs . . . . .	86
4.2	Control Architecture Levels . . . . .	92

4.3	Software architecture defines the relation between different software.	93
4.4	Hardware architecture showing central PC, decentralized nodes and communication bus. . . . .	94
4.5	EtherCAT frame structure based on the Ethernet . . . . .	95
4.6	Node of the type 1-ch in three combined boards: computational unit, EtherCAT interface and peripheral interface. . . . .	97
4.7	Hardware Control Architecture, Joint Controllers' Distribution, ROS Discovery Server. . . . .	98
4.8	Sensorimotor programs selection. . . . .	99
5.1	SEHA general controller combining the functionalities of the actuator modules. . . . .	106
5.2	SEHA-Controller boards and corresponding functional blocks . . . .	107
5.3	Position Sensor Test Bench: The figure shows the position sensor installed on the CNC machine's nozzle and the magnetic strip installed on the movable plate. . . . .	109
5.4	M2 test bench: The figure shows the voice coil motor AVM12-13-C48 installed on the motion axis with the position sensor circuit AMS AS5311. . . . .	111
5.5	M3 test bench: The figure shows the stepper motor 20DAMXXD2B installed on the motion axis with the position sensor circuit AMS AS5311. . . . .	113
5.6	ISO hydraulic circuit diagram . . . . .	115
5.7	ISO-4409 Implemented Hydraulic circuit . . . . .	117
5.8	ISO-test bench Beckhoff controller and Parker A10 motor driver . .	117
5.9	The effective flow rates at different rotational speeds and pressures.	118
5.10	SIC-Controller functional blocks distributed on three boards . . . .	119

5.11	SIC Manufactured controller boards. . . . .	120
5.12	The SIC installed on development kit with both position sensors . .	126
5.13	System Identification Workflow. . . . .	130
5.14	Moog30 Flow-Load Characteristics . . . . .	131
5.15	Moog.30 Servo Valve Internal Dynamics . . . . .	132
5.16	Hydraulic and Electrical Test Circuit. . . . .	134
5.17	Measured Velocity and Position with a Step Input 10 mA at 10ms Sampling Time . . . . .	136
5.18	Measured Velocity and Position with a Step Input 10 to -10 mA at 10ms Sampling Time . . . . .	137
5.19	Measured Velocity and Position with a Step Input -10 mA at 10ms Sampling Time . . . . .	138
5.20	Measured Velocity and Position with a Step Input 10 mA at 3ms Sampling Time . . . . .	138
5.21	Measured Velocity and Position with a Step Input -10 mA at 3ms Sampling Time . . . . .	139
5.22	Current Input Step with Flow Output Response . . . . .	140
5.23	Real Data with Model Response Comparison . . . . .	142
5.24	Real Data with Model Response Comparison . . . . .	142
5.25	Real-Time Kernel Latency Experiment . . . . .	144
5.26	Non Real-Time Kernel Latency Experiment . . . . .	145
5.27	Latency Network Test . . . . .	145
5.28	Network latency measurements . . . . .	146
5.29	Distributed Sensorimotor Program with gait walking . . . . .	147
5.30	Position Sensors AMS AS5045B Installed on the Robot Hip Axis . .	147
5.31	Hip-Pitch position tracing measurements during the gait walking . .	148
5.32	HYDROiD Implementing Gait Walking . . . . .	149



# List of Tables

4.1	HYDROiD: Degrees of Freedom, Motion Angle, and Speed for involved joints in the walking gait [126] . . . . .	88
4.2	Sensors and actuation elements of HYDROiD robot . . . . .	89
4.3	Cycle Time Elements . . . . .	95
4.4	Functional distribution of nodes . . . . .	97
5.1	Power supply 3.3 V and 5V Test . . . . .	108
5.2	Pressure sensor input test results, the figure illustrates the captured signals with 16-bit resolution converted to an integer number and the corresponding error percentage. . . . .	109
5.3	Position sensor test results, the figure illustrates the captured displacement of a CNC machine, sensor measurement reading, and the corresponding error in ( $\mu\text{m}$ ). . . . .	110
5.4	M2 Voice coil motor test results, the figure illustrates the position setpoint, sensor measurement reading, and the corresponding error in (mm). . . . .	112
5.5	M3 Stepper motor test results, the figure illustrates the motor position setpoint, position sensor measurement reading, and the corresponding error in (mm). . . . .	113
5.6	M1 BLDC motor test results, the figure illustrates the motor speed setpoint, Tachometer measurement reading, controller and motor temperature, and the corresponding error in (RPM) and (%). . . . .	114

5.7	Permissible variation for classes of measurement accuracy: The permissible variations listed in this table concern deviation of the indicated instrument reading and do not refer to limits of error of the instrument reading. These variations are used as an indicator of steady state and are also used where graphical results are presented for a parameter of fixed values. The actual indicated value should be used in any subsequent calculation of power, efficiency or power losses. $a \text{ Pa} = 1 \text{ N/m}^2$ . [3] . . . . .	116
5.8	Test bench components: the table presents the used sensors and components in the ISO test bench with their quantities and characteristics. . . . .	117
5.9	SIC Controller: Servo Valve Output Current Measurements and Static Error Results . . . . .	121
5.10	SIC Controller: Servo Valve Output Settling Time and Current Overshoot Measurements Results. . . . .	122
5.11	SIC Controller: Servo Valve Cut-Off Frequency Test Results. . . . .	123
5.12	SIC Controller: Servo Valve Cut-Off Flow Test Results. . . . .	124
5.13	SIC Controller: Pressure Sensors Test Results. . . . .	125
5.14	SIC Controller: Position Sensor Test Results. . . . .	127
5.15	MGH Speeds with Corresponding Flow Rates . . . . .	134
5.16	System Parameters at step input 5 and 10 mA . . . . .	140

# Acknowledgment

I would like to express my sincere gratitude to the PAUSE program [1](Programme d'aide à l'Accueil en Urgence des Scientifiques en Exil) for providing me with a life-changing scholarship.

I extend my heartfelt thanks to the startup KALYSTA ACTUATION and its CEO, Mr. Jean Claude Rassou, for their invaluable support in conducting my research. Their assistance has been instrumental in the successful completion of my work.

I thank Evry-Paris Saclay University for creating a conducive and comfortable environment with its excellent facilities. I also thank the IBISC (Informatique, Bio-Informatique, Systèmes Complexes) lab and its dedicated team for their unwavering logistic support. I am thankful for the administrative support the STIC (Sciences et Technologies de l'Information et de la Communication) doctoral school provides, which has played a crucial role in my academic journey.

I am indebted to my supervisors, Prof. Lydie Nouveliere and Prof. Samer AlFayad, for their exceptional guidance, encouragement, and unwavering support throughout the thesis. Their mentorship has been invaluable. I am also grateful to my colleagues and the multidisciplinary team I had the privilege to work with. Their cooperation and team spirit have significantly contributed to the breadth of my knowledge.

Lastly, my deepest thanks go to my family. Their unwavering belief, motivation from my daughter, and the constant support and encouragement from my wife have been my pillars of strength throughout this journey.





# Introduction

Understanding the phenomena of locomotion and manipulation of humans is one of the main interests of most scientific research for decades. Emulating the human body by replicating it with humanoid robots has been one of the methods used to improve this understanding. This emulation requires a reliable hardware design replicating complex human mechanisms with a powerful actuation system and a robust control system achieving dynamic motions like humans.

The control system architecture is a critical bottleneck for producing humanoid robots that maintain stability while performing different activities (avoiding obstacles, localization, mapping, and others); Developing this architecture still faces several challenges. i) Since one of the aims of humanoid robots is to emulate the human body, their control systems should perform like how humans control their muscles during the stability phase and the postural transitions. A set of sensorimotor programs in humans [2] represents postural coordination; the transitions are behavioral consequences of changes among programs operating at the central nervous system level. ii) Humanoid robots should be able to collaborate with humans safely, the physical human-robot interaction (pHRI) should be critically safe despite the challenging hardware and software constraints. The control architecture for achieving pHRI should be operated in real-time, with deterministic latency. iii) The distribution of numerous actuated degrees of freedom (DoFs) and associated sensors make it essential to seek an efficient control system with enormous processing capability that considers limitations of the joint control level update rates and the network bandwidth. iv) The socially accepted appearance is challenging while developing humanoid robots, where connecting all the actuators

and sensors requires installing an abundant number of cables, making the robot bulky and having an unpleasant appearance.

This thesis discusses the proposed control architecture of the robot HYDROïD. It is an under-developed, full-size humanoid robot that is hydraulically actuated. It is designed to be able to achieve hard and useful tasks and replace humans in disaster environments. The desired bio-loyal behavior concerns the capacity of transmission of force and motion and joint deflections while ensuring an anthropomorphic appearance of the produced devices. It has a central hydraulic pump and many integrated hydraulic pipes into parallel mechanisms like shoulders, hips, ankles, and torso that assist to replicate the complex mechanisms of humans. HYDROïD's mechanisms have been the subject of two international patents (WO2009144320 and WO2009135694) and several publications in journals and international conferences. The fact of having one central pump constrain integrating all the hydraulic pipes and deliver the same force to every joints which affect the robot's power consumption.

In 2019, an intensive collaboration was carried out with the Servo-Electric-Hydraulic Actuator (SEHA) project. SEHA was conducted under a startup KALYSTA, a spin-off Paris-Saclay University, funded by the SATT Paris-Saclay <sup>1</sup> SEHA is a modular actuator consists of i) a power generation module named Mini Group Hydraulic (MGH), ii) a highly instrumented hydraulic actuator named Servo instrumented Cylinder (SIC), iii) hydraulic circuit blocks, iv) internal hydraulic feedback for safe operation, v) required sensors for reasonable control, and vi) electronic cards. Using SEHA in HYDROïD aims to replace the central pump with local electro-hydraulic actuators that deliver the exact required flow to each join and integrate the hydraulic tubes contributing to the robot's morphology and dynamic motions. SEHA's main modules, like the MGH and SIC, could work separately or combined as one actuator, enabling hybrid mechanisms to be built into the hardware design and replicating human-like motions.

This work aims to develop a control system that imitates humans' nervous system regarding the multitude of decision-making centers and their distribution to

---

<sup>1</sup>SATT is the Technology Transfer Accelerator Office of the Paris-Saclay Cluster.

reach the highest level of human emulation. The development methodology measures the human motions, increases the robot's physical capabilities, and develops a control architecture to employ the measured motions. Much research is needed to develop parallel mechanisms to achieve some complex joint motions based on hydraulic linear actuators to increase the robot's physical capabilities. However, there is still a need to interpret these motions to fit the measured data, which is an angular motion. Replacing the ordinary control system of using one controller for each joint with controllers for a group of joints that work together into parallel or hybrid mechanisms. That enables the execution of the Inverse Geometrics Models (IGM) locally and divides the processing into the joint controllers.

The proposed architecture is able to use human motions' measured data and control the robot's actuators to execute these motions. This work concentrates on the joint-level controller and the software architecture modulated to the humanoid robot's whole-body joints. The proposed solution assists in defining the best ways to reproduce the sensorimotor loops by enabling them to be implemented in low and high control levels. It guarantees task execution in a determined time, ensuring safe interaction, and provides a software architecture with multitasking and prioritizing capability. In addition, thanks to the controllers' distribution, the processing capacity will be divided into joint controllers, and the used cable for peripherals will be minimized.

## Main contributions

The main contributions of this thesis are:

- **SEHA-based humanoid control architecture:** emulating human's nervous system in the intelligence distribution. It simplifies replicating human motions by employing measured biomedical data of human motions in its motion algorithms. It improves the overall robot's control system by locally controlling the hybrid mechanisms. It allows the creation and testing of systems with distributed artificial intelligence.

- **SEHA control Enhancement:** System identification was conducted to find the system model based on the developed control architecture.
- **Control architecture mechatronic design:** The research identified and described specifications and constraints of the key software and hardware control architecture components: i) master controller real-time capability and ability to seamlessly integrate with other subsystems, ii) joint controllers distribution and capability to execute adaptive control strategies, iii) communication interface, and iv) cycle time, have collectively led to the development of a flexible and highly dependable system.
- **Experimental validation:** Manufacturing controllers' prototypes for SEHA actuator and its modules separately. Designing a graphic user interface for testing purposes. A long-term latency test was executed for the main control architecture parts (Master controller and network). A control system for the test benches is designed and executed to validate SEHA module based on ISO standards; ISO 4409:2019 [3], for the Mini Group Hydraulic.
- **Demonstration:** for a walking experiment of the HYDROiD's lower body with four DoFs presents controlling the hip mechanism.

## Thesis plan

This thesis report presents a bio-medical study in Chapter.1 that introduces the human mechanisms for muscle control, explores the sensorimotor loops, and provides an overview of the tasks, functions, and responsibilities associated with each part of the human nervous system. Chapter.2 investigates the development of humanoid control architecture, including its degrees of freedom, actuation types, joint controllers, software, middleware, and communication technology. Chapter.3 introduces the features, structure, and operations of the developed hybrid actuator SEHA. Chapter.4 introduces in detail the development methodology of a control architecture for the robot HYDROiD using SEHA actuators and explains the capabilities of the proposed system to implement the sensorimotor loops in different

control levels. Chapter.5 provides an overview of the SEHA modules' controllers development, where SIC and MGH boards are designed to meet the required functionalities, manufactured, and tested on development kit prototypes. A bench test was designed and executed according to ISO standard (4409:2019 [3]). The other axe of this chapter examines the control architecture elements and explores the validation processes of the system's capabilities and performance. The future works, and perspectives are presented in the conclusion Chapter.6.



# Chapter 1

## Biomedical Study

### Contents

---

<b>1.1</b>	<b>Introduction</b>	<b>8</b>
<b>1.2</b>	<b>Research Questions and Objectives</b>	<b>9</b>
<b>1.3</b>	<b>Human Motions Control Exploration</b>	<b>10</b>
<b>1.4</b>	<b>Data Collection and Methods</b>	<b>15</b>
<b>1.5</b>	<b>Biomedical Study Outcomes and Results</b>	<b>21</b>
1.5.1	Human Control System	21
1.5.2	Gait Data analysis	21
<b>1.6</b>	<b>Outcomes Discussion</b>	<b>23</b>
1.6.1	Implications of Research Findings in the Context of Research Questions:	23
1.6.2	Significance of Distributed Decision-Making Centers in the Human Body:	24
1.6.3	Challenges and Implications of Replicating Human Joint Motions on Humanoid Robots:	24
<b>1.7</b>	<b>Conclusion</b>	<b>25</b>

---

## 1.1 Introduction

In the pursuit of developing advanced humanoid robots that seamlessly replicate human motions and capabilities, understanding the complex mechanisms of human muscle control becomes crucial. This chapter explores a biomedical study to investigate the mysteries of how humans control their muscular systems. Through a simple exploration of sensorimotor loops and the distribution of intelligence within the human body, this chapter sheds light on the inner processes of our physiological machinery.

It aims to explore some of the main questions that form the basis of the research endeavors. How do humans control their muscles, and how does their control system function? Are there distinct decision-making centers distributed throughout the body, and how do they collaborate in producing complex motions? Furthermore, do muscles possess an intelligence, enabling them to adapt and respond to diverse stimulants? These questions serve as the guiding compass as we navigate the depths of human biomechanics.

The importance of understanding the complexities of human muscle control lies at the core of replicating humanoid robots, particularly our SEHA-based HYDROiD, with the flexibility and precision of human-like motions. A deep awareness of these physiological mechanisms unlocks new horizons in robotics and promises applications across healthcare, rehabilitation, and various industries where human-robot collaboration is on the rise.

An investigation of sensorimotor programs controlling stability phases and postural transitions was conducted, drawing insights from the postural coordination changes. Additionally, the incorporation of real-world biomedical data was collected during human locomotion as a foundational data set for our exploration. This data, capturing the joints' pitch, yaw, and roll angles, reveals the complexity underneath seemingly simple movements.

This chapter outlines the key findings and explains the difficulties discovered during the study. In addition, it presents the implications of our findings and their relevance to the development of humanoid robots like HYDROiD.



As a conclusion to the introduction, the insights acquired from this biomedical study will deepen our understanding and serve as the starting point to build an analysis of humanoid robots in the next chapter, "State of the Art." This interplay between human and robotic elements brings us closer to realizing a future characterized by the integration of humans and machines.

## 1.2 Research Questions and Objectives

This section outlines the key research questions that guide our exploration into human muscle control mechanisms. Each question is followed by its objective, providing a comprehensive understanding of our investigative approach and goals.

### Research Questions and Objectives

- **Question-1:** How do humans control their muscles during various phases of motions and locomotion, including stability and postural transitions?

Objective: To investigate the human nervous system's sensorimotor programs and postural coordination mechanisms to regulate muscle activity during different movement phases. Understanding these mechanisms is crucial for replicating human-like movements in humanoid robots and enhancing their stability and adaptability.

- **Question-2:** Are the distinct decision-making centers distributed throughout the human body responsible for coordinating complex movements?

Objective: To explore decision-making centers' anatomical and functional distribution and their role in directing coordinated muscle actions. Identifying these centers improves our understanding of the distributed intelligence that supports human motor control, potentially guiding the design of more human-emulation-capable robotic systems.

- **Question-3:** Does human musculature exhibit adaptive responses close to a form of intrinsic intelligence?

Objective: To examine whether muscles demonstrate adaptive behaviors and responsiveness to various stimulants, indicating a level of inherent intelligence. Insights into muscle adaptability can inspire innovative approaches to robot control and human-robot interaction.

- **Question-4:** How can insights from the study of human muscle control mechanisms be applied to enhance the development of humanoid robots?

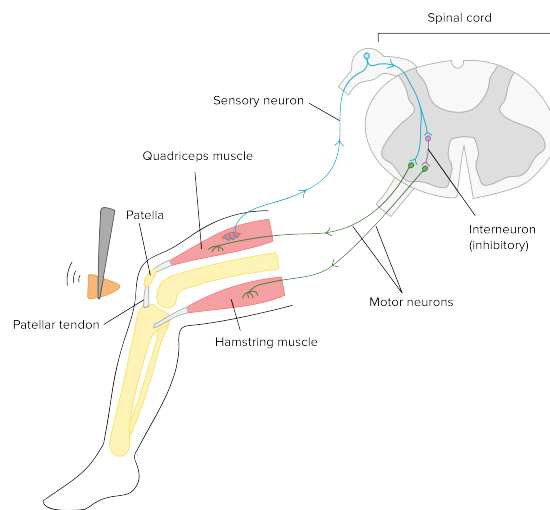
Objective: To draw practical implications from our findings to inform humanoid robot design and control architecture, focusing on our SEHA-based HYDROiD platform. That will approach between human and robotic control mechanisms, advance the field of robotics, and open new possibilities for real-world applications.

### 1.3 Human Motions Control Exploration

The main human motions are represented by locomotion and manipulation. Locomotion is a form of motion that leads to movement from one place to another by walking, running, jumping, or hopping [4]. The legs are the main limbs responsible for locomotion, whereas the arms are used for more accurate balancing. On the contrary, human manipulation, like grabbing an object or using a screwdriver, occurs mainly by the arms, whether during locomotion or standing, setting, or laying postures. The human manipulation for avoiding obstacles is achieved by sequential locomotions, and the manipulation to open a door is sequential postural transitions. Therefore, two groups could classify these motions: the first includes motions for achieving locomotion, and the second includes the postures and the transitions between them. The discussion in this section is mainly about these two kinds of motions, but before going deeper into those, an overview of the nervous system and some of the used expressions could clarify the context of this section.

Humans control their movements through complex processes involving the various components of the nervous system. The human nervous system is functionally segmented into two main components: the central nervous system, which consists

of the spinal cord and brain, and the peripheral nervous system, represented by all neurons outside the central nervous system. The processes are known as sensorimotor programs executed through sensory routes and motor pathways. It is a common feature of humans that sensory routes connect with motor pathways at various levels of the nervous system. Each level contributes to different reflexes and motor control aspects, ranging from simple reflexes to more complex and coordinated movements and postures. The nervous system levels are defined by the involved parts of the nervous system to achieve sensorimotor continuity. The simplest reflex in humans is established through the continuity of only sensory and motor cells, where sensory cells directly send their process signals to motor cells. These reflexes are segmental, where afferent and efferent neurons belong to the same segment like reflexes in the spinal cord and the brain stem [5]. Figure 1.1 illustrates the classically described spinal reflex arc and shows the direct connection between the sensory routes and the motor pathways [6]. This reflex is considered as an involuntary motion that occurs without the human conscience.



**Figure 1.1:** Simplest reflex establishes direct connections between the sensory routes and motor pathway [6]

The next complexity level of controlling motions corresponds to the second motions group of postural control. The standing posture may be considered one of the simplest human motion controls. One of the studies presents the complexity of

the sensorimotor control system in maintaining balance during quiet standing [7]. The study explores human postural control and sensorimotor integration during an upright stance and examines how the nervous system adjusts to maintain balance in various conditions. The sensorimotor continuity at this level includes connections through the medial reticular formation of the brain stem. Sensory fibers connect with interneurons in the spinal cord, which then communicate with large cells in the medial reticular formation. The medial reticular formation is also influenced by cranial nerves, enhancing its connectivity [5]. It stabilizes whole-body postures and movements, particularly maintaining the upright posture, aided by vestibular reflexes.

In [8], the author emphasizes the instability of the human body during the standing posture. Consequently, there is a need for continuous muscular control with effective coordination between the segments and joints of the human body, even in the simplest postures. A set of motor programs represents these postural coordinations, and the transitions between postures are explained in [2] as behavioral consequences of changes between programs operating at the central nervous system's level. The study presents the mechanical restrictions of the postural states as a potential reason for the alteration between various postural states. These mechanical constraints can be defined in several ways, including limitations on the forces and torques exerted at the support surface or joints, constraints on the intrinsic frequency of postural patterns, or restrictions on the amplitude of body movements that the patterns can accommodate. Following this explanation, a task that demands body movements beyond these mechanical limits could alternate from one pattern to another.

For example, ankle rotations make the body sway by applying torque at the support surface, and these rotations are most effective on extended and rigid surfaces that resist this torque. The trunk rotations linked to the legs could be the alternative to the ankle rotations. In this pattern, rotations at the hips make the body sway by applying a shear force to the support surface. Consequently, moving to a surface with low shear resistance would switch from hip to ankle rotation.

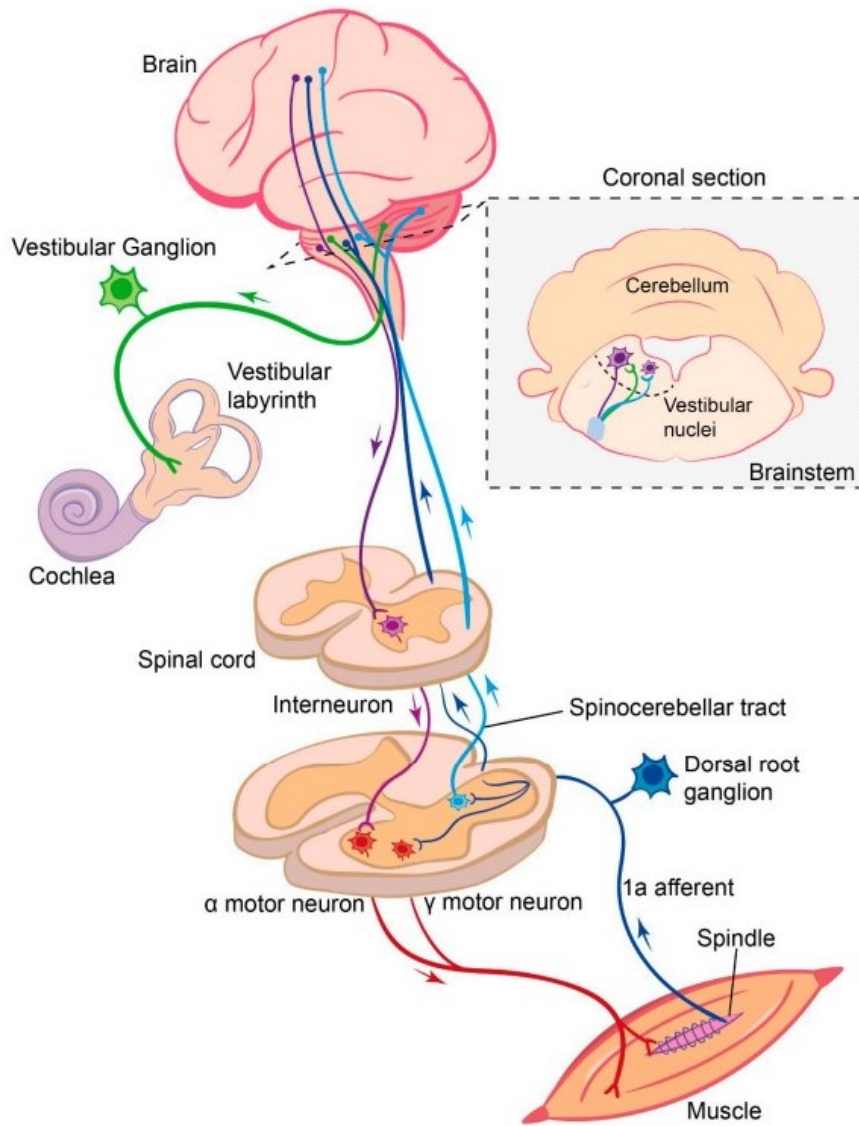
Taking the first step following the standing posture means the motion con-

control will switch from postural control to locomotion control, related to the first motion group mentioned above. This sensorimotor continuity involves various subsystems in the nervous system, like the somatosensory and vestibular systems. These systems constantly provide segmental coordination and body stabilization information. Avoiding obstacles involves the visual system to provide information about the surrounding environment. A study presents how leg movements during locomotion result from a collaborative interaction between interconnected neurons in the spinal cord, known as the central pattern generator, and sensory feedback from the segmental somatosensory system and higher-level centers such as the vestibular system [9]. The study outlines the roles played by proprioceptive and vestibular feedback depending on the speed of locomotion. Vestibular feedback becomes crucial at slower speeds, while somatosensory feedback is essential at higher velocities. Figure 1.2 shows the interconnection in the central nervous system between the two sensory systems and their sensory routes alongside the motor pathways.

The sensorimotor continuity levels represent increasingly complex and interconnected systems in the nervous system, allowing for more sophisticated control and coordination of sensory and motor functions. Each level builds upon the previous ones, adding layers of complexity and adaptability to human responses to its environment [5].

Another study investigates the role of locomotion sub-functions in stabilizing walking at different speeds [10]. It draws inspiration from biological locomotory systems and template models that simplify understanding locomotion dynamics and control. The three basic sub-functions in locomotion are elastic axial leg function (stance), leg swinging, and body alignment (balancing/posture control). It reveals that different control strategies stabilize walking at different speeds. Specifically, balancing and leg swinging controllers mainly change when transitioning from slow to moderate speeds, while the stance leg controller contributes consistently across various speeds.

Since this research aims to develop control architecture for humanoid robots, and after the motion control exploration, there is a need to analyze these motions. The



**Figure 1.2:** Summary of the somatosensory and vestibular sensory pathways and their integration into the brain and spinal cord. [9]

following section investigates human walking by analyzing collected bio-medical data of healthy humans while walking.

## 1.4 Data Collection and Methods

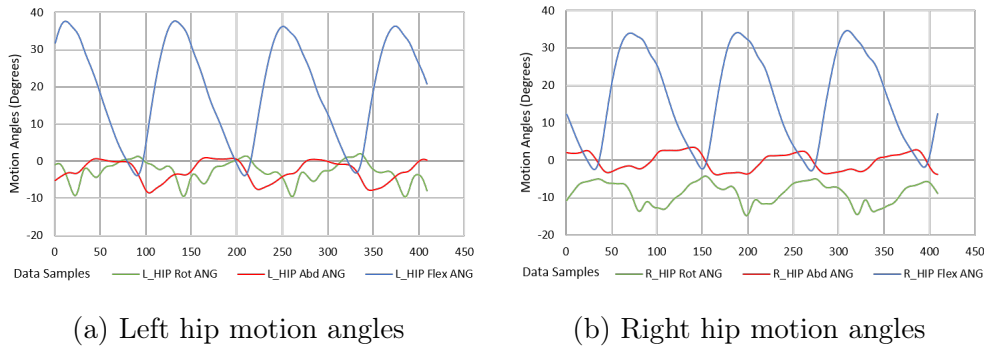
This analysis was conducted based on recorded biomedical data at Raymond-Poincaré Hospital, gait analysis laboratory. The Conventional Gait Model (CGM) approach was chosen for this data collection [11], [12] to capture and understand human motion during walking. The Conventional Gait Model is an approach that deconstructs the complex walking motion into measurable components and uses mathematical modeling to analyze and understand human gait. The CGM provides i) joints kinematic data (angles, positions) throughout the gait cycle, which provides insights into how joints move during the walking, ii) kinetic data (forces and moments acting on the body), which helps in understanding the forces that drive and stabilize the motion, iii) temporal data (timing and duration of different phases of the gait cycle).

The joint kinematics are measured using motion capture systems that employ multiple cameras to track the movement of reflective markers placed on specific anatomical body landmarks or segments. The kinetics are obtained through ground reaction force plates that measure the forces exerted by the feet on the ground during walking. These plates contain sensors that detect the magnitude and direction of forces applied by each foot as it contacts and pushes off from the ground.

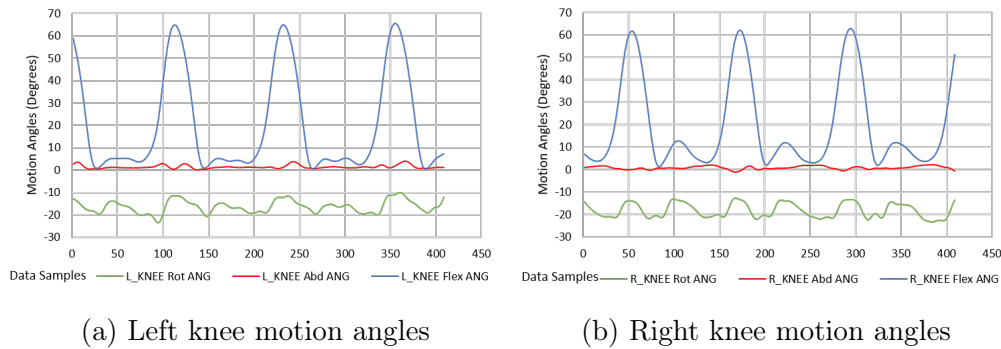
A few persons participated in the data collection process; each performed three walking tests at different speeds (slow, regular, and fast) and recorded multiple steps. The person's walk is described with the parameters of Robinson and Smidt 1981 [13]. The data set provides walk information, such as walking velocity (cm/s) and step length (cm). The moment is calculated using geometry and the body's mass distribution.

Since the control architecture development is targeted for the HYDROïD robot, the chosen data set is for a person with similar anthropomorphic characteristics to

the HYDROiD robot. HYDROiD has the same morphology and mass distribution as the human body [14]; therefore, it is enough to choose a human subject with the same height and weight as the robot. One of the data sets meets the robot's highest (170 cm) and weight (75 kg). Figures 1.3, 1.4, 1.5, and 1.6 show some of the noticeable joints' kinematics like hips, knees, ankles, and arms during locomotion. The joint motion is presented by three angles, flexion, rotation, and abduction angle, measured in degrees during four steps of a healthy person with a 63 cm step length at 100 cm/s velocity. The data set provides additional kinematics related to the head, trunk, pelvis, and foot.



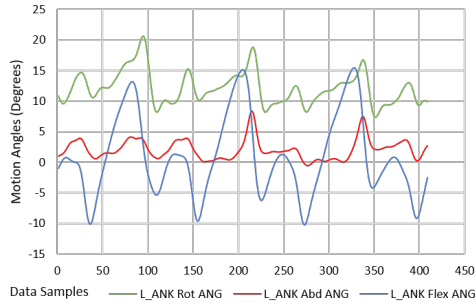
**Figure 1.3:** Hips motion angles in degrees during four steps of a healthy person with 63 cm step length at 100 cm/s velocity



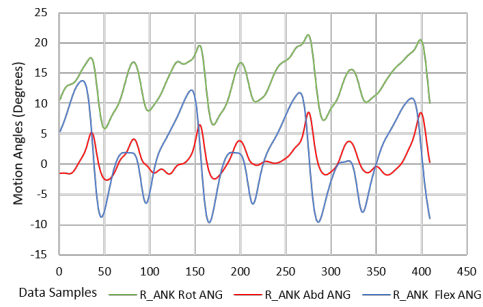
**Figure 1.4:** Knees motion angles in degrees during four steps of a healthy person with 63 cm step length at 100 cm/s velocity

The joint motion forces, moments, and power of the flexion, rotation, and abduction angles represent the measured kinetic data. Figures 1.7, 1.8, and 1.9



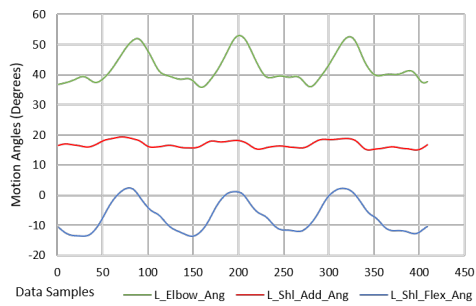


(a) Left ankle motion angles

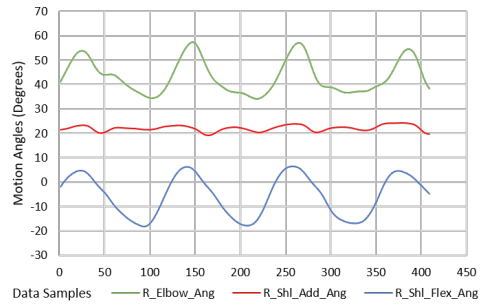


(b) Right ankle motion angles

**Figure 1.5:** Ankles motion angles in degrees during four steps of a healthy person with 63 cm step length at 100 cm/s velocity



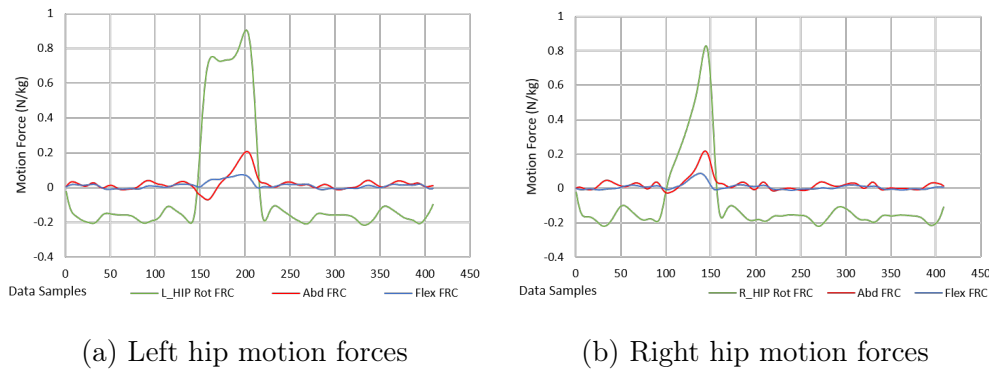
(a) Left arm motion angles



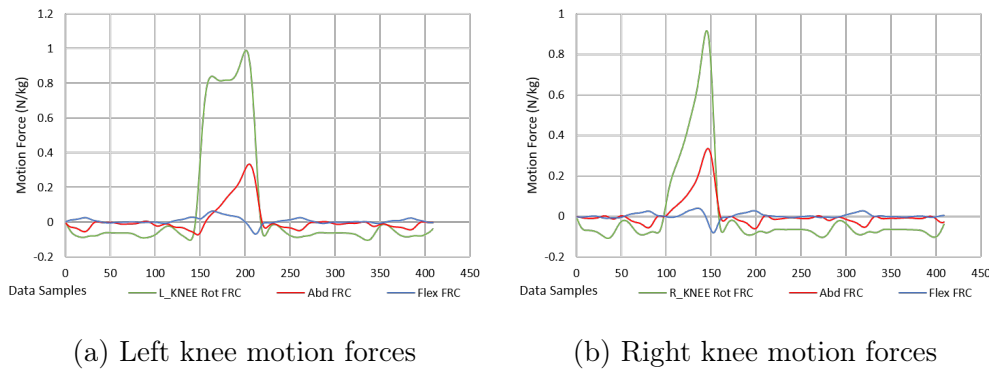
(b) Right arm motion angles

**Figure 1.6:** Arms motion angles in degrees during four steps of a healthy person with 63 cm step length at 100 cm/s velocity

show the captured motions' forces (N/kg) of the hips, knees, and ankles joints. The presented data is taken from the same walking experiment ( four steps of a healthy person with 75 kg weight, 175 cm highest, 63 cm step length, at 100 cm/s velocity).



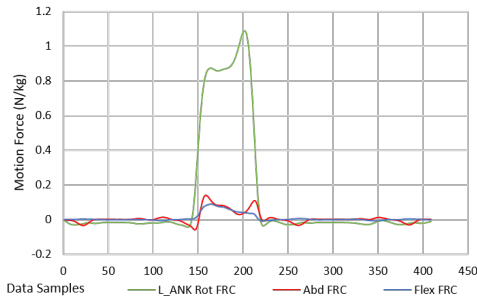
**Figure 1.7:** Hips motion forces during four steps of a healthy person with 63 cm step length at 100 cm/s velocity



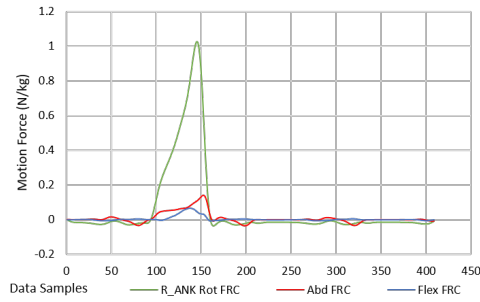
**Figure 1.8:** Knees motion forces during four steps of a healthy person with 63 cm step length at 100 cm/s velocity

Figures 1.10, 1.11, and 1.12 show the captured motions' moments (Nm/kg).

Figures 1.13a and 1.13b show the calculated powers of the studied joints: hips, knees, and ankles measured in (watts/kg)

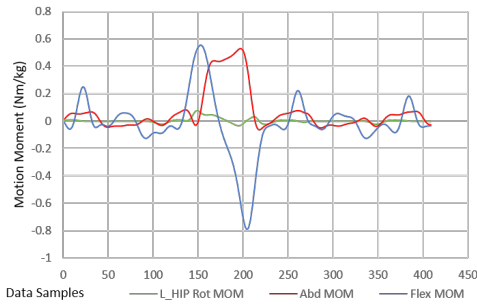


(a) Left ankle motion forces

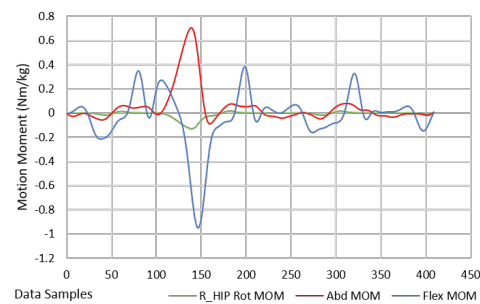


(b) Right ankle motion forces

**Figure 1.9:** Ankles motion forces during four steps of a healthy person with 63 cm step length at 100 cm/s velocity

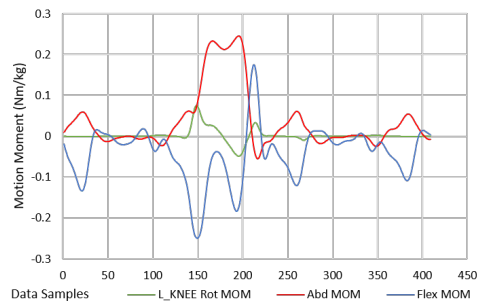


(a) Left hip motion moments

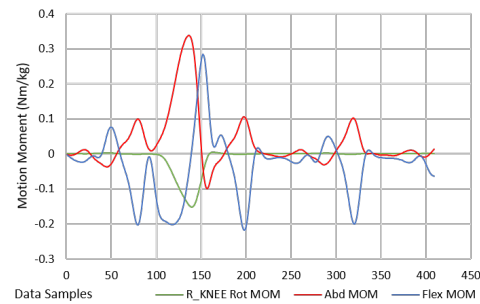


(b) Right hip motion moments

**Figure 1.10:** Hips motion moments during four steps of a healthy person with 63 cm step length at 100 cm/s velocity

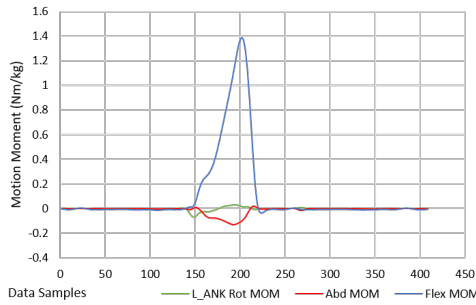


(a) Left knee motion moments

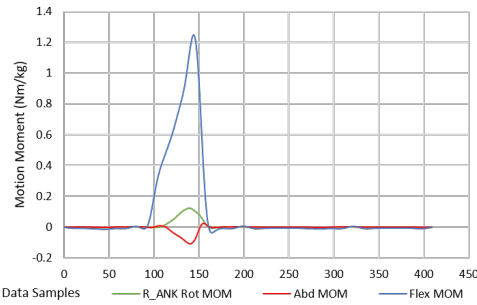


(b) Right knee motion moments

**Figure 1.11:** Knees motion moments during four steps of a healthy person with 63 cm step length at 100 cm/s velocity

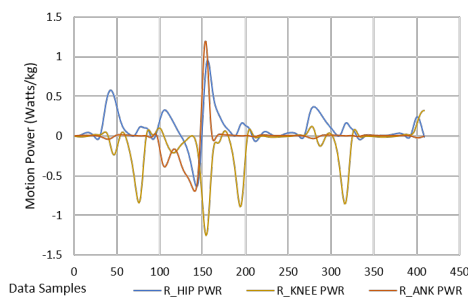


(a) Left ankle motion moments

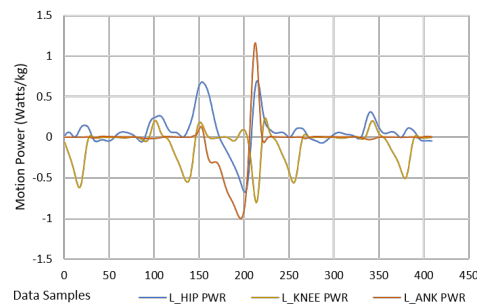


(b) Right ankle motion moments

**Figure 1.12:** Ankles motion moments during four steps of a healthy person with 63 cm step length at 100 cm/s velocity



(a) Left Hip-Knee-Ankle Powers



(b) Right Hip-Knee-Ankle Powers

**Figure 1.13:** Hips-Knees-Ankles Motion Powers during four steps of a healthy person with 63 cm step length at 100 cm/s velocity

## 1.5 Biomedical Study Outcomes and Results

The key outcomes of the bio-medical study were divided into two parts. The first handles the human control system and its intelligence centers distribution, and the second discusses and analyzes the captured data of human walk.

### 1.5.1 Human Control System

According to the human motions control exploration, section 1.3, the complexity of the motion control system corresponds to the complexity of the desired motion, and there is a switching mechanism between these control strategies stimulated by the motion type or the mechanical limitations. The simplest motions are the involuntary reactions, then the postural control, and finally, the locomotion at dynamic speeds. The control system for the simplest motions starts with direct connections between the sensory and motor pathways. Then, posture control included connections through the medial reticular formation in the brain stem and spinal cord. And finally, for the locomotion, the control system includes some subsystems.

### 1.5.2 Gait Data analysis

The findings were separated into two parts related to the data analysis sources: Kinematic data findings from the captured physical motion and Kinetic data findings from the captured motion forces, moments, and power.

#### **Kinematic Data**

- **Asymmetric motion angles:** The study's observation revealed differences in the amplitude of measured joint angles between the right and left sides of the body. For instance, the right hip abduction angle and the left hip abduction angle exhibited distinct variations, as shown in Figures 1.3a, 1.3b. These differences are attributed to many factors, including asymmetry in body form and variations in muscle tissue magnitude since human reliance

on one side more than the other, mainly during activities like lifting weights, contributes to these differences.

- **Joint motion complexity:** The joint kinematics analysis highlighted the complexity of human joint movements. While it was expected that the hip and ankle mechanisms would be represented by three motion angles (Pitch, Roll, and Yaw) as shown in Figures 1.3 and 1.5, it was remarkable that the knee joint also exhibited three distinct motion angles as shown in Figure 1.4. This observation reflects the intricate nature of human motion, where even joints connecting two seemingly static segments can generate complex three-dimensional motions.
- **Arm Motions:** The captured data highlights that arm motions were the most notable of the upper body motions during the walking gait as shown in Figure 1.6. The arm's contribution to overall motion dynamics emphasizes its significance in maintaining balance and coordination during walking.

## Kinetic Data

- **Joint Forces and Moments:** The captured data revealed that joint force was mainly displayed in the joints' rotation motions as shown in the hips, knees, and ankles' forces in Figures 1.7, 1.8, and 1.9. In contrast, the joint motion moments were most prominent in flexion and abduction movements, which presented in the hips, knees, and ankles' moments Figures 1.10, 1.11, and 1.12. This phenomenon represents a compensatory mechanism, where joint moments in flexion and abduction are increased to offset the force encountered during rotational motion. This finding highlights that the joint moment shows independence from the faced force in the same joint motion.
- **Joint Power Peaks:** A direct correlation between joint power and joint moments has been noted. Joint power peaked in close alignment with the moments at the respective joints as Figures 1.13b, 1.13a emphasizing the

efficiency of the musculoskeletal system during the steps without encountered moments.

- **Kinetic-Kinematic Independence:** One of the most fascinating discoveries was the independence of kinetic and kinematic data. Despite significant changes in kinetic readings, the results indicated remarkable stability in kinematic motion patterns. This finding suggests the presence of robust compensatory mechanisms that enable the body to maintain relatively consistent motion despite variations in kinetic factors.

These findings illuminate the intricate interplay between kinematic and kinetic aspects of human motion, underscoring the multifaceted nature of biomechanics. This study gives valuable insights into the complexities of joint movements, offering a deeper understanding of human motions and locomotion dynamics and their implications for fields ranging from sports science to physical therapy.

## 1.6 Outcomes Discussion

### 1.6.1 Implications of Research Findings in the Context of Research

#### Questions:

The research findings provide valuable insights into human muscle control mechanisms and their implications for replicating human-like motions in humanoid robots. By addressing the first research question regarding how humans control their muscles during various phases of locomotion, we have discovered essential sensorimotor programs and postural coordination mechanisms. Understanding these mechanisms is pivotal for enhancing the stability and adaptability of humanoid robots when mimicking human-like motions. The complexity of the motion control system, following the desired motion, highlights the need for adaptive control strategies. The existence of involuntary reactions, postural control, and locomotion at dynamic speeds demonstrates the hierarchical nature of human motor control. These findings reinforce the significance of studying the human nervous system's sensorimotor programs in developing advanced robotic control architectures.

### **1.6.2 Significance of Distributed Decision-Making Centers in the Human Body:**

The investigation into the distribution of decision-making centers throughout the human body has far-reaching implications for understanding human motor control. The distributed intelligence responsible for coordinating complex movements is a testament to the multifaceted nature of the human nervous system. Recognizing these decision-making centers' anatomical and functional distribution provides critical insights into how the body manages diverse muscle actions and motor functions. This understanding is crucial not only for advancing the field of robotics but also for gaining a deeper comprehension of the intricate interplay between the human brain and the musculoskeletal system. These distributed centers open new possibilities for developing more human-emulation-capable robotic systems that can adapt to real-world scenarios.

### **1.6.3 Challenges and Implications of Replicating Human Joint Motions on Humanoid Robots:**

Replicating human joint motions on humanoid robots presents several challenges and implications. The research findings highlight the complexity of joint motions, including the presence of three-dimensional motions in seemingly static segments, such as the knee joint. These complexities underscore the intricacies of human motion and pose challenges for robotic emulation. The variations in joint angles between the right and left sides of the body can be avoided in the humanoid robot design. However, there is a need for adaptive control strategies to account for individual differences and maintain balance and coordination during robotic locomotion. The observed independence of kinetic and kinematic data is a challenging insight, suggesting the development of compensatory mechanisms for humanoids to maintain consistent motion patterns despite kinetic variations. However, the challenge remains in implementing these mechanisms effectively in robotic control systems. Overall, these findings emphasize the need for innovative approaches to robot control and human-robot interaction, ultimately bridging the gap between



human and robotic control mechanisms and advancing the field of robotics for a wide range of real-world applications.

## 1.7 Conclusion

The biomedical study has provided profound insights into human muscle control mechanisms, shedding light on the intricacies of sensorimotor programs, postural coordination, and adaptive control strategies. We have explored the distributed decision-making centers throughout the human body, revealing the multifaceted nature of human motor control. These findings have far-reaching implications for the field of robotics, offering a deeper understanding of human-like motions and control mechanisms that can be employed to enhance humanoid robots' stability and adaptability. These key takeaways from the biomedical study are invaluable as we launch on developing the SEHA-based humanoid robot, HYDROiD, as explained in the chapter 4. By understanding the complexities of human joint motions, the existence of distributed decision-making centers, and the potential for adaptive control strategies, we are better equipped to bridge the gap between human and robotic control mechanisms. We aim to create a humanoid robot capable of emulating human motions and interacting seamlessly in real-world scenarios. As a transition to the next chapter, "State of the Art," we carry the insights gained from the biomedical study. These insights will serve as a foundation for examining humanoid robots in the state of the art. We will explore how the principles of distributed control, adaptive strategies, and human-like movements are integrated into the latest advancements in humanoid robotics. This interplay between our biomedical study and the state of the art brings us one step closer to realizing the vision of seamless integration between humans and machines. This vision has the potential to revolutionize various fields and open new possibilities for human-robot collaboration and interaction.



# Chapter 2

## State of the Art

### Contents

---

<b>2.1 Introduction</b> . . . . .	<b>27</b>
2.1.1 Importance of Bipedal Robots . . . . .	28
2.1.2 Humanoid robot challenges . . . . .	29
<b>2.2 Historical evolution of humanoid robots</b> . . . . .	<b>30</b>
2.2.1 Humanoid Control Architectures . . . . .	44
2.2.2 Communication Technologies . . . . .	47
2.2.3 Software and Middleware . . . . .	48
2.2.4 Joint Controllers . . . . .	50
2.2.5 Human-Like Anthropomorphism in Humanoids . . . . .	52
2.2.6 Actuation Technologies . . . . .	53
2.2.7 Applications And Control Architecture Variations . . . . .	57
<b>2.3 Conclusion</b> . . . . .	<b>59</b>

---

### 2.1 Introduction

The previous chapter explores human muscle control mechanisms during various motion patterns. An intensive collaboration and accurate synchronization between

the muscular and nervous systems achieve these motions. We found that switching mechanisms in the control system enables adaptable control strategies ranging from the nervous system levels to the switching between the involved subsystems that achieve more sophisticated control for these motions. In addition, it explores the muscular system's impact on controlling human motions. The muscular system is mirrored by the actuation system in robotics alongside the robot's physical limitations, including the used joints' number and degrees of freedom. The control architecture emulates the functionality of the human nervous system. It comprises the central controller, joint controllers, software, middleware, and communication system.

This chapter investigates prior research and the evolutionary trajectory of those elements that mainly affect the performance of achieving human-like motions and behaviors. It aims to comprehensively understand how these elements have evolved, shaping the field and paving the way for future developments. The exploration core will revolve around the state of the art in bipedal robotics, offering insight into the historical evolution and contemporary challenges encountered in the pursuit of humanoid control architecture and actuation systems.

### **2.1.1 Importance of Bipedal Robots**

Bipedal robots are critical in scenarios where human access is unsafe or impossible, launching into places like malfunctioning nuclear power plants or navigating the intricate staircases and corridors of burning buildings [15]. These environments are inherently designed for human mobility, posing challenges or hard-to-manipulate obstacles for other types of robots, whether tracked or wheeled. Beyond their remarkable utility in disaster response and exploration, the domain of bipedal robotics is evolving diverse applications, extending to the design and control of assistive devices. Robotic prosthetic legs and exoskeletons can enable patients who have lost the ability to walk or run or even provide superhuman performance, such as lifting and transporting exceedingly heavy loads [16]. Bipedal robotics, which refers to robots with two legs that can walk like humans, is an essential

area of research in robotics. Bipedal robotics applications offer several significant benefits:

- **Increased mobility:** Bipedal robots are designed to move around in environments not easily accessible by wheeled or tracked robots, such as stairs, uneven terrain, or narrow spaces.
- **Versatility:** Biped robots are designed to perform various tasks, such as search and rescue, manufacturing, entertainment, and healthcare.
- **Human-like interaction:** Bipedal robots can interact with humans more naturally, which makes them suitable for tasks such as caring for the elderly or helping people with disabilities.
- **Research and development:** Bipedal robots are used as a platform for research and development in areas such as artificial intelligence, machine learning, and control systems.
- **Future potential:** As technology advances, bipedal robotics is expected to become increasingly important for applications such as space exploration.

Overall, the importance of biped robotics applications lies in their potential to increase mobility, versatility, and human-like interaction while simultaneously serving as a platform for research development endeavors across a spectrum of fields.

### **2.1.2 Humanoid robot challenges**

Common challenges of humanoid robots include:

- **Stability:** Maintaining stability is one of the critical challenges in the advancement of humanoid robots. Unlike wheeled or tracked robots, humanoid robots must constantly balance themselves to avoid falling over. Addressing this challenge requires advanced algorithms and sensor systems for precise real-time monitoring of the robot's position and motion adjustments.

- **Control complexity:** Humanoid robots are inherently more complex to control than wheeled or tracked robots, as they have a greater number of degrees of freedom and must be able to balance, walk, and interact with their environment similarly to humans. Addressing this challenge requires a flexible and adaptable control system to ensure the robot moves smoothly and efficiently.
- **Terrain adaptability:** Humanoid robots may have difficulty navigating uneven or unstable terrain, such as stairs, rocks, or soft ground. It requires advanced sensors and control algorithms to adapt to different types of terrain and maintain stability.
- **Power consumption:** Humanoid robots walking is energy-intensive; they require significant power to locomote and maintain their balance. This challenge can be a constraint in applications where energy efficiency is essential. For instance, in battery-powered robots.

## 2.2 Historical evolution of humanoid robots

This section explores the history of full-scaled anthropomorphic humanoid robots and their capabilities, features, and formulating elements like their actuation systems, DoFs, control systems, software, middleware, and communication systems classified by projects. Then, it discusses the revolution of the technological solutions used in the abovementioned elements.

### **WABOT Project 1967-2006**

In 1967, Waseda University initiated the WABOT project, and in 1973, completed the WABOT-1 robot, the world's first full-scale anthropomorphic humanoid robot. It was the first android able to walk, communicate with a person with an artificial mouth, measure distances and directions to objects using external receptors (artificial ears and eyes), and grip and transport objects with hands [17]. In 1980, the project developed a WABOT-2 keyboard-playing robot. With a focus on its

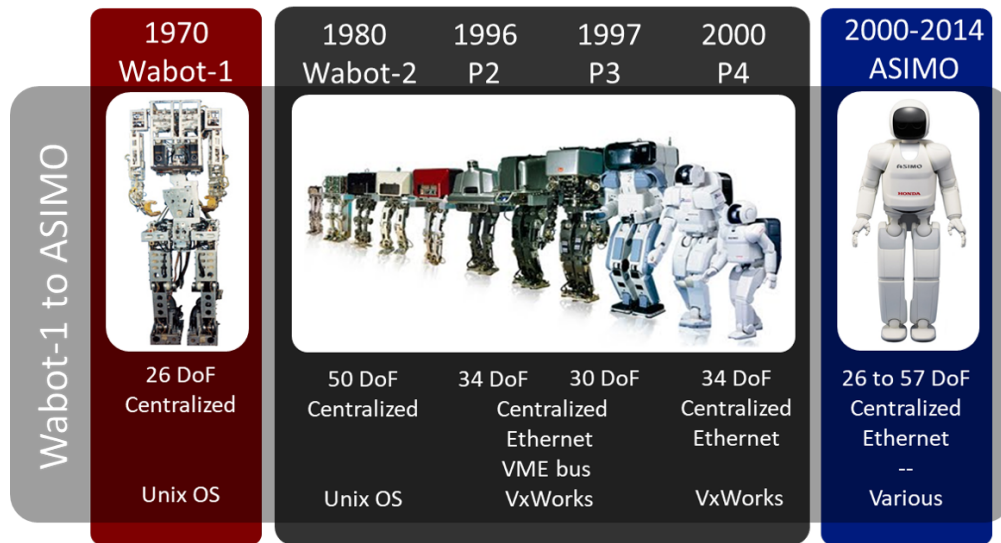
arm-and-hand mechanisms, hierarchically structured control computer system, information processing method, and finger-arm coordination control [18]. The primary aim of this development was to enable the robot to possess soft functions, such as dexterity, speediness, and intelligence, which go beyond traditional hard functions.

Then, the project overlapped with the E Series project that started in 1986 with the E0 robot until 1993 with the E6 robot. These robots began as lower-body experimental prototypes but later transformed into the Honda P series. Over time, Honda accumulated the necessary knowledge and experience to create their advanced humanoid robot, ASIMO.

### **Honda P Series and ASIMO Development 1993-2014**

In 1993, Honda initiated the development of "Prototype" models known as the "P" series, where they connected legs to a torso equipped with arms capable of executing fundamental tasks. In the first version of the Honda P1 robot, the study concentrated on realizing the coordinated movement of legs and arms. Therefore, the computers for image processing and action plans, and electric power supply were not installed on the first robot. The second prototype in this series, known as P2, made its debut in 1996 and gained recognition for being the initial self-sustaining, two-legged walking robot [19]. P2 had 34 DoFs, weighed 210 kg, and stood at an impressive height of 183 cm. Subsequently, in 1997, Honda unveiled the P3 model, marking a significant milestone as the first entirely self-sufficient bipedal humanoid walking robot. P3 boasted a height of 162 cm and weighed 130 kg. The major difference between this and the previous prototype was its reduced size and weight due to the change in the link material from aluminum alloy to Magnesium alloy and the decentralized control that has been adopted. In addition, the existing motors were replaced with brushless DC motors to help with the robot's reliability [20].

From the year 2000 until 2014, Honda Unveiled the ASIMO robot and many of its advancements [21]; for a long time, it remained the most advanced humanoid



**Figure 2.1:** The figure illustrates the developed humanoid robots starting from the WABOT project to the ASIMO robot and some of their characteristics

robot in the world that was able to walk, climb stairs, perform complex motions that demanded high dynamics, and interact with humans using natural language. Its versions have a range of DoFs from 29 to 57, weighing 50 to 54 kg, and a height of 120 to 130 cm, [20]. The primary version of the intelligence system is composed of two PCs for image processing and speech recognition and synthesis, a control and planning processor, and a controller for a radio communication network that is used to communicate with the external system for map management and navigation. The common feature of all the mentioned robots up to now is the centralized control system, where the joint drivers are controlled directly by one centralized controller. In contrast, the other PCs are responsible for dominating subsystems such as the vision control and others.

Figure 2.1 illustrates the developed humanoid robots from the WABOT project to the ASIMO robot and some of their characteristics.



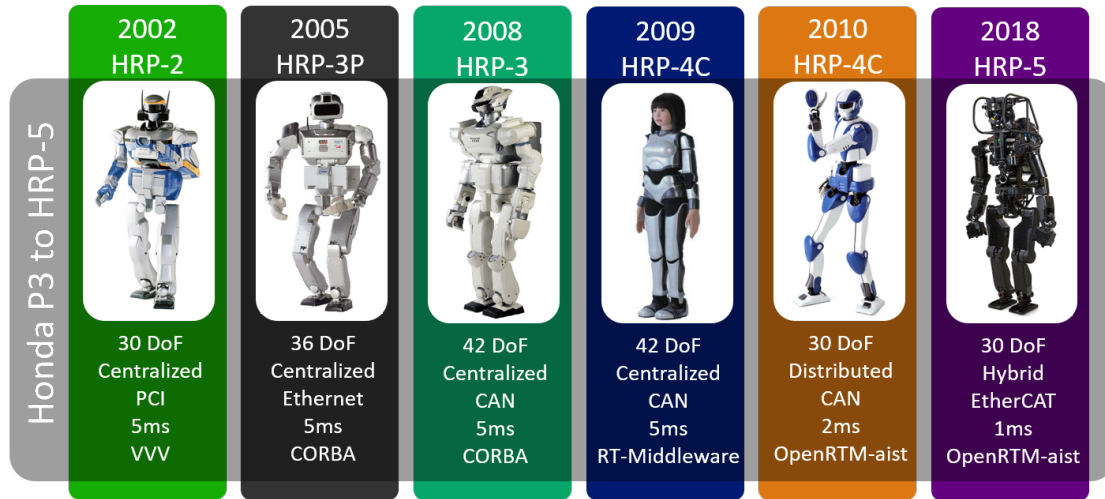
### **Humanoid Robotics Project (HRP) 1997-2019**

Humanoid Robotics Project spearheaded by Kawada Industries in collaboration with the National Institute of Advanced Industrial Science and Technology (AIST) in Japan, HRP was a series of humanoid robots designed to create versatile home assistant robots. The project was initiated with three Honda P3 robots. Then, they were developed to be integrated with the company communication system as the first version of the project HRP-1 and kept some of their features, such as the telecommand system [22]. In 2002, the HRP-1S was developed by modifying the control hardware of HRP-1 and implementing a custom control software. The motion controller of HRP-1S conforms to the developed plug-in, like the real-time walking pattern generator (KWALK plug-in) and a reflex controller (STABILIZER plug-in) to maintain the dynamic balance. The whole-body motions controller operates on one CPU board alongside an I/O board that handles input from all types of sensors used in the robot and sends output to the distributed motor drivers [23].

The biped locomotion capabilities have been enhanced in HRP-2 to enable it to navigate uneven surfaces, walk at almost human walking speed, and traverse narrow paths [24]. Additionally, HRP-2's whole-body motion capabilities have been improved, allowing it to self-right if it tips over safely. The HRP-2 control system consists of many I/O boards connected through a PCI bus to two CPU boards within its body. One is dedicated to real-time control of whole-body motion, while the other is employed for non-real-time control purposes, including the VVV software and sound systems [25]. It has 30 DoFs, 154 cm height, and 58 Kg weight.

In 2005, the robot HRP-3P was developed with brushless DC motors and harmonic drive for 42 DoFs, 160 cm height, 65 kg weight. The control system was an Ethernet-based distributed control system with several types of node controllers. In 2007, the HRP-3 robot was developed, and the communication system internal network adopted the Controller Area Network CAN to improve the system reliability and maintenance [26]. One notable feature of HRP-3 is its design, which prioritizes the prevention of dust or spray infiltration into its primary mechani-

cal and structural components. Additionally, HRP-3 boasts newly enhanced wrist and hand designs to optimize manipulation. The software tailored for a humanoid robot operating in real-world conditions has also improved. This information encompasses the mechanical attributes of HRP-3 and the newly developed hand.



**Figure 2.2:** Humanoid Robot Project HRP series robots since the primary versions with their characteristics and the employed technological solutions

The advancement in the next project's versions concentrates on developing lightweight robots (e.g., HRP-4 with 39 kg weight [27]) to increase the robot's dynamics and decrease the robot power. It focuses on the used software and middleware to enhance the robot's reliability and adaptability, the RT middleware was used in the robot HRP-4C [28], and then replaced with the OpenRTM-aist middleware in the robot HRP-4 [27] and HRP-5 the last version of this series that designed for construction and other manual labor tasks [29]. OpenRTM-aist middleware is a software technology that facilitates the integration of diverse functional robotic components through a communication network. In the last version, HRP-5, the control system was described as a hybrid system, where the whole body motion controller communicates with the distributed joint controllers through EtherCAT bus for real-time operations, and the non-real-time computer communicates with the peripheral devices through the PC-104 bus and USB serial. The EtherCAT communication technology was used for the first time in developing

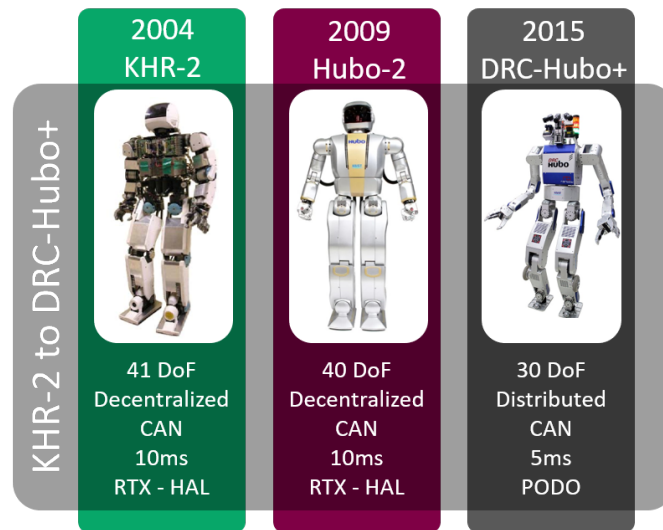
this robot series to ensure real-time performance on the soft and hard aspects and enhance the global control cycle time.

Figure 2.2 shows the Humanoid Robot Project HRP series robots since the primary versions with their characteristics and the employed technological solutions. It illustrates the variety of used DoFs in this series changing due to the targeted application of each robot. It shows how the control system was developed in the primary robots as centralized, then replaced with a distributed, and the last version was hybrid. The communication protocols started in this series with a non-real-time communication bus, then replaced with a soft real-time system, and then the hard and soft real-time communication system; changing the communication bus alongside the used software and middleware resulted in the global control cycle time.

### **KAIST Project 2003-2015**

The Korea Advanced Institute of Science and Technology (KAIST) developed KHR-1, the first humanoid robot platform in the HUBO laboratory, then KHR-2, KHR-3, and Alpert HUBO until 2009 when the robot HUBO-2 was developed, a humanoid robot designed for research and development purposes, especially in advanced robotics and human-robot interaction. It has 40 DoFs actuated by brushless DC motors and Harmonic Drive technology for actuation [30]. The robot's control system has been the same since the development of KHR-2, as they announced in [30]. It is distributed and managed by two PC104-embedded computers [31]. It runs on a Windows XP operating system with RTX (Real-Time Extension) for Windows and utilizes CAN communication technology [32]. Figure 2.3 illustrates some of the developed humanoid robots in the HUBO laboratory since the robot that all the subsequent robots have the same control architecture. The last robot was significantly improved in design, actuation system, and control architecture.

In 2015, KAIST developed DRC-Hubo+ for the Defense Advanced Research Projects Agency Robotics Challenge (DRC), a humanoid robot explicitly tailored for disaster response scenarios. In this updated version, the degrees of freedom were



**Figure 2.3:** The Korea Advanced Institute of Science and Technology (KAIST) Project’s series robots since the primary versions with their characteristics and the employed technological solutions

reduced to 32 DoFs with the same actuation system. For seamless control, DRC-Hubo+ adopts a distributed control system. Its control hardware involves an Intel NUC i5 with an SSD PC for head sensing and motion control. The operational system runs on a Linux system (Ubuntu 12.04), with software and middleware based on PODO, a custom motion control framework developed at KAIST [33]. PODO, a hierarchical software architecture, consists of five layers enabling effective communication and control in a robot. The first layer contains the robot simulator and end devices, such as motor controllers and sensor boards. The second layer is managed by a process called Daemon, which is responsible for real-time control. Shared memory, the third layer, facilitates interprocess communication and holds hardware abstraction data and user declaration data. The fourth layer comprises user processes (ALs) that independently read sensor data and generate joint references. Lastly, the fifth layer accommodates external processes like user-designed GUIs. PODO’s key components include ALs, shared memory, and Daemon, allowing multiple users to develop their ALs simultaneously and efficiently manage robot behaviors and motion algorithms without extensive knowledge of the entire system. The motion owner property minimizes conflicts among ALs, promoting

hassle-free programming and development.

### RobotCub Consortium

In 2004, the RobotCub Consortium introduced iCub, shown in Figure 2.4, a child-sized humanoid robot designed to develop cognitive and social skills through human interactions. iCub has 53 DoFs actuated using brushless DC motors and Harmonic Drive technology. Its control system was centralized, with bidirectional communication facilitated by a PC104 card to an external control station. iCub's operational system was compatible with Linux, Windows, and MacOS. Communication technology was based on PC104-CAN, and it utilized the YARP middleware [34]. The design of iCub follows a roadmap inspired by human development, em-



**Figure 2.4:** The iCub. The figure shows a photograph of the iCub robot [34]

phasizing the significance of prediction in skillful motion control. This perspective witnesses development as the progressive improvement of predictive abilities. iCub adopts a sensorimotor control and development, where the core element of cognitive behaviors is 'action' purposeful movements driven by a motivated agent with a predictive nature [35].

## NASA

In 2010, NASA and General Motors developed Robonaut2, a humanoid robot explicitly tailored for space exploration shown in Figure 2.5. Its actuation system employs brushless DC motors paired with Harmonic drive reduction gears for precise control. The control system for R2 is distributed to the low-level joint controllers consisting of an FPGA-based controller with an embedded PowerPC processor and PLC coupled to the motors. While detailed information about the software is not provided, Robonaut2 represents a significant advancement in humanoid robotics technology for applications in the challenging space exploration environment [36]. Its capability has been decentralized and distributed to the low-level joint controllers integrated within each of the arm joints. These highly compact controllers, known as "Superdrivers," play a pivotal role in the upper arm's motion control strategy. They comprise an FPGA-based controller housing an embedded PowerPC processor alongside programmable logic, all complemented by a 3-phase brushless DC inverter bridge. The Superdriver assumes responsibilities ranging from motor commutation and current regulation to serialization and deserialization of joint data and commands, sensor data processing, and control over the arm's series elastic actuators.

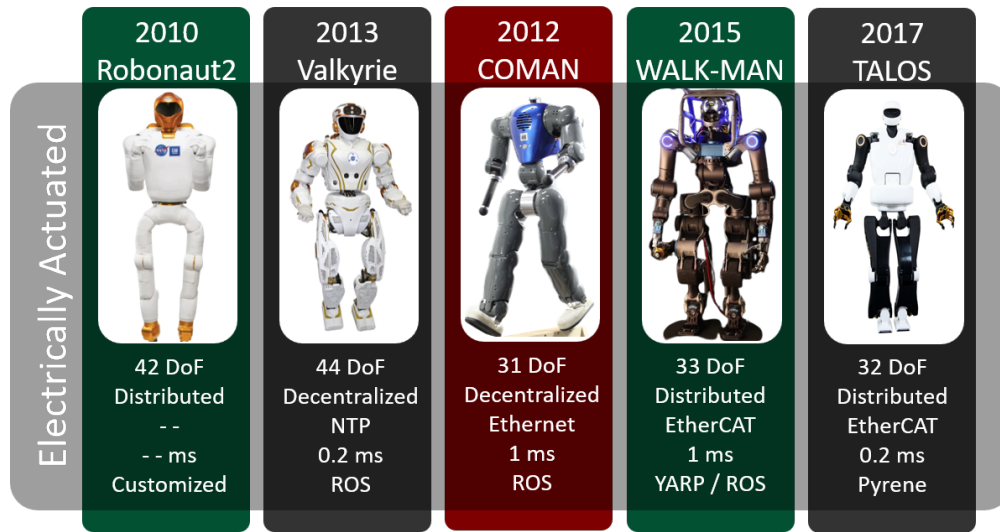
In 2013, NASA's Johnson Space Center introduced Valkyrie (R5), designed to operate in various challenging environments such as space exploration and disaster response, shown in Figure 2.5. Valkyrie has 44 DoFs, driven by a combination of Brushless DC Motors, Harmonic drives, and Serial Elastic Actuators (SEAs). Its control system follows a decentralized approach for enhanced versatility [37]. The robot is equipped with three Intel Core i7 COM Express CPUs, each endowed with 8 GB of DDR3 memory. Additionally, a CARMA development kit, comprising a Tegra3 ARM A9 processor and an NVIDIA Quadro 1000M GPU, facilitates the implementation of parallel sensor interpretation algorithms. Data communication is established through a wired Ethernet connection and WiFi connectivity. A rigid body model at the high level is assumed, while low-level actuators are represented as ideal torque sources. This decentralized control architecture's effectiveness heavily relies on joint-level torque controllers' capabilities, aiming to

emulate an ideal torque source model closely [38]. A noteworthy advantage of the decentralized approach in this robot is the ability to tune and validate the hierarchical controller at the individual joint level. This implies that each joint in a multi-degree-of-freedom (multi-DOF) system can be tested independently before integration into the complete robot. This approach expedites the development process and simplifies testing higher-level functionalities once the robot is fully assembled. For achieving this degree of isolation and modularity, the torque control of each actuator must be genuinely decoupled from the whole system.

### **Italian Institute of Technology (IIT)**

In 2012, the Italian Institute of Technology introduced the COMpliant huMANoid COMAN [39], designed for research and development, specifically focusing on bipedal locomotion shown in Figure 2.5. The robot employs brushless DC motors featuring 14 custom-series elastic elements for improved compliance in its actuation system. COMAN has 31 DoFs. The control system of COMAN follows a decentralized approach, with custom-made boards based on the ARM Cortex A8 architecture for low-level control tasks, along with an additional board dedicated to high-level control functions. The operational system is Linux, enhanced by the real-time framework Xenomai. The robot relies on Ethernet for interactions with external devices and Wi-Fi for remote control. Software and middleware utilized include YARP [40].

In 2015, the Italian Institute of Technology (IIT) introduced WALK-MAN, a humanoid robot engineered for search and rescue operations in disaster-stricken environments shown in Figure 2.5. The robot has 33 DoFs actuated by Electric Serial Elastic Actuators (SEAs) [41], which endow the robot with advanced flexibility and compliance[42]. The WALK-MAN is controlled through a distributed system architecture, controlled by two vision and control PCs with a quad-core i7, which is prepared to operate the control software and is configured as the EtherCAT master connected through wireless to an external pilot PC. The operation system is based on Ubuntu Linux with an RT-preempt kernel for real-time functionality. WALK-MAN's communication capabilities are supported by EtherCAT,



**Figure 2.5:** Some of the advanced robots that electrically actuated are developed by various projects

USB, Ethernet, and TCP/UDP protocols, enabling effective data exchange. The robot's software is custom-designed, leveraging middleware such as YARP and ROS.

## PAL ROBOTICS

In 2017, TALOS was introduced by PAL ROBOTICS as an advanced humanoid robot engineered for intricate tasks within research and industrial environments shown in Figure 2.5. TALOS utilizes brushless DC motors in combination with Harmonic Drive technology to actuate its limbs, resulting in precise and robust movements. TALOS has 32 DoFs, allowing for versatile and coordinated motion [43]. It relies on a control system powered by two computers with dual i7 CPUs operated by Ubuntu 14.04 LTS operation system for an open-source platform with an RT-PREEMPT kernel to manage its complex operations efficiently. TALOS benefits from a software and middleware stack that includes ROS and ros-control, facilitating various control and communication tasks. TALOS employs the EtherCAT network for seamless data exchange and communication, enhancing its overall efficiency and performance in challenging research and industrial scenarios.



### **SARCOS Robotics 2006**

In 2006, the Computational Brain CB robot was unveiled by SARCOS Robotics as a hydraulically powered, force-controlled humanoid robot. It has a mass of approximately 95 kg and stands at a height of around 1.7 meters. The main actuation method relies on a 3000 psi hydraulic system powered by an external pump and lots of hydraulic pipes [44]. The robot has a total of 34 hydraulically actuated joints controlled by distributed real-time network-based architecture that aims to explore various levels of human-like processing. It consists of on-board low-level computing and higher-level perceptual processing [45] based on ART-Linux operating system and Ethernet communication bus.

### **HYDROiD 2009**

HYDROiD is a hydraulically-actuated full size humanoid robot, created in 2009 [46] with parallel mechanisms [47], [48]. A patented for a spherical hinge with coupled actuator was conducted [49]. The robot has 36 DoF with servo valve-based hydraulic actuation. During the last decade, many axes of research are grown up through this project. Research on decreasing the weight of the robot [50], [51] are ongoing. The emotional head is studied and developed [52]. Mechanical design is under enhancement [53]. Artificial skin is under development [54]. In addition, some advanced actuation-based research like the Integrated Electro-Hydraulic Actuator (IEHA) and several versions of IEHA have been developed [55], [56]. And the patented Servo Electro-Hydraulic Actuator (SEHA) [57], but they are not yet implemented on the robot.

### **BOSTON DYNAMICS**

In 2009, Boston Dynamics introduced PETMAN, a biped robot developed for testing chemical protective clothing. It was actuated using hydraulic cylinders or compact hydraulic rotary actuators and equipped with various sensors, including position and torque sensors at the joints, force/torque sensors, and an IMU. With

a total of 29 DOFs, PETMAN utilized a distributed control system, ran on a custom PC, and employed a CANbus communication network. Its software was custom-developed [58].

In 2012, Boston Dynamics was chosen by the Defense Advanced Research Projects Agency (DARPA) to construct the Atlas humanoid robot as a part of the DARPA Robotics Challenge (DRC). In 2013, the robot Atlas-DRC was introduced [59], [60].

In 2015, Boston Dynamics and DARPA introduced a new version of Atlas that no longer required tethers and operated on batteries, Atlas-Unplugged, a versatile humanoid robot capable of walking, climbing stairs, and opening doors. This robot was actuated by hydraulic servo-valves and had 30 DOFs. It featured sensors such as Lidar, stereo vision, and position/force sensors at the joints [61], [62]. The control system was centralized and operated by two quad-core i7 computers. The control computer directly connects to the robot handling balancing and motion control algorithms, operating the low-level joint control in a responsive 1 kHz loop. The vision computer manages camera drivers and computer vision algorithms. The third computer manages communication with the operator through the wireless network and oversees control of the two Robotiq hands connected directly to two dedicated network ports [63]. This computer handles all high-level task control, as it doesn't require time-critical or processor-intensive functions. These PCs ran on an Ubuntu Linux platform with an RT-preempt kernel. Communication technologies included EtherCAT, USB, Ethernet, TCP/UDP. Custom-designed software and middleware such as ROS and Orocos were used for control.

In 2017, Boston Dynamics unveiled an updated version of Atlas [64], showcasing its agility with complex parkour movements. Similar to its predecessor, this robot was actuated using hydraulic servo-valves and had 28 DOFs. Its sensor suite included Lidar, stereo vision, and position/force sensors at the joints. The control system remained centralized, powered by dual i7 quad-core processor COM Express PCs, operating on Ubuntu Linux with RT-preempt kernel. Communication technologies continued to be EtherCAT, USB, Ethernet, and TCP/UDP, and the software and middleware were custom-designed, featuring ROS and Orocos.

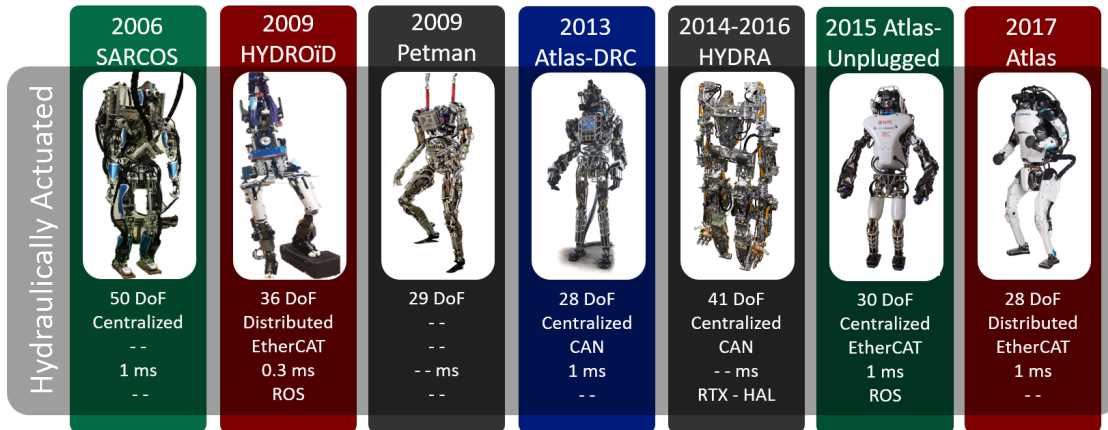


Figure 2.6: Hydraulically Actuated Robots

## UNIVERSITY OF TOKYO

In 2019, researchers at the University of Tokyo and Sony Corporation collaborated to create Hydra [65], an electro-hydraulically driven humanoid robot focused on advanced mobility and agility. Hydra boasts 41 degrees of freedom. The robot's actuation involves backdrivable electro-hydraulic actuators for all joints except the neck joint. These actuators are coupled with hydrostatic transmission driven by servomotors. Its sensor array includes force/torque sensors, IMUs, Lidar, stereo vision cameras, and joint position/torque sensors [66]. The robot is controlled through a centralized system, running Ubuntu Linux with an RT-preempt kernel for operational stability. Communication between microcontroller units (MCUs) and the real-time PC occurs via EtherCAT technology. Custom-designed software, known as ROBO-BODY, governs the robot's behavior and control.

## Commercial Robots

The electric and hydraulic actuation systems are leading in actuating these robots. In the last five years, the electric-based robots market has been rapidly growing due to the fast implementation process and the ease of control of these robots like Optimus-2022, Cyber-one-2022, Apollo-2023, Fourier Intelligence's GR1-2023, and Unitree H1-2023 as shown in Figure 2.7. However, these robots are still limited

to performing heavy-duty activities with dynamic motions. On the other hand, there is very limited research on the hydraulic actuation system in the world, like Boston Dynamics, Kalysta, and Sarcos Robotics. Hydraulic actuators have increased the robot's physical limitations by providing high dynamic motions while performing heavy-duty activities like the robot Atlas in 2017.

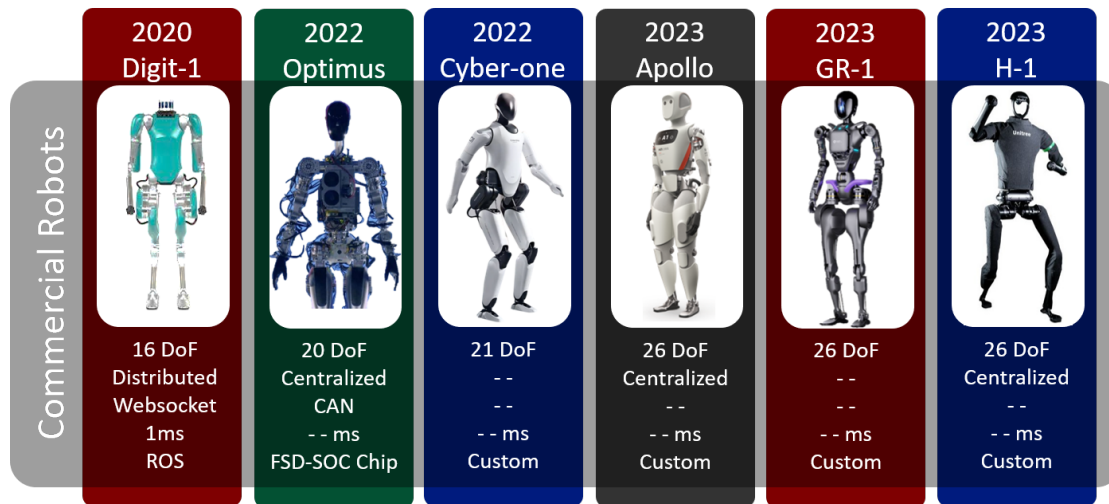


Figure 2.7: Commercial Robots

## AGILITY ROBOTICS

In 2019, the robot Digit-1 was developed by Agility Robotics; Digit is a bipedal robot designed for commercial applications, such as package delivery and warehouse logistics. It had a similar design to the Cassie robot, which considered the lower body of the Digit-1 robot but with additional features, such as arms and hands, that enabled it to manipulate objects and perform tasks.

### 2.2.1 Humanoid Control Architectures

The state of the art illustrates that the main control architecture utilized in humanoid robots are four: the centralized, decentralized, distributed, and hybrid control architecture. This section discusses the history of these methods in the

humanoid robot and mentions some of their main advantages, disadvantages, and challenges.

### **Centralized control architecture**

Early humanoid robots, such as WABOT-1 (1972) [17], ASIMO (2000) [20], and HRP-2 (2002) [23], [24] used centralized control architecture, where a central computer controlled all aspects of the robot's motion. This approach offers the benefit of fast communication, low task latency, substantial capacity, and simple implementation but limits the robot's capabilities. Humanoid robots have numerous degrees of freedom, the centralized approach limiting the control strategies that require substantial computational resources to solve optimization problems. The control strategies are limited in their ability to adapt to changing environments or unexpected disturbances, as they rely on a single controller to generate control signals for all robot joints. For example, in the case of ASIMO [20], its motions were only sometimes as smooth or fluid as a human's, and it had difficulty walking on uneven terrain or in situations where it had to make quick decisions. Another limiting factor of the centralized control system is that it requires excessive wires to be connected to the central computer, which makes the robot bulky and heavy, limiting its mobility. In addition, the long wires are prone to electromagnetic interference.

### **Decentralized control architecture**

The decentralized control architecture approach is used in humanoids to handle the complexity of their motions. This approach facilitates the distribution of control tasks across multiple controllers that operate relatively independently and make local decisions based on their inputs and setpoints. This method has the advantage of optimizing the control of the robot's movements and expediting the processing of sensor data, enhancing the robot's ability to respond rapidly. Robots such as COMAN (2012) [40] utilized this architecture enabling to achieve impressive stability and agility.

However, decentralized control faces coordination challenges, as different controllers may have conflicting goals or generate control signals inconsistent with the robot's overall behavior. These strategies may also be sensitive to sensor noise or measurement errors, as they rely on sensor feedback to adjust control signals. For example, one of the main limitations of the decentralized control architecture used in COMAN was the difficulty of coordinating the behavior of multiple modules, mainly when dealing with complex tasks or unexpected events. The robot's control system had to constantly communicate and synchronize with each module, which is time-consuming and requires significant computational resources.

### **Hybrid control architecture**

The hybrid control architectures combine different control techniques to achieve more robust and versatile control of humanoid robots. This approach takes advantage of the centralized and decentralized architectures and improves stability and efficiency, as exemplified by the HRP-5 [29] developed by Kawada Industries in Japan. Implementing hybrid control strategies requires complex algorithms and hardware, integrating multiple control modules or techniques. It requires high expertise and computational resources to ensure that all sub-control systems work together seamlessly and synchronously. The other limitation is that due to the complexity and amount of computational resources needed to implement the robot's sub-control systems, it may need a significant amount of energy to operate, which limits its mobility and endurance.

### **Distributed control architecture**

The distributed control concept was employed for bipedal robots in the 2000s in bipedal robots like the HRP-3P robot [67], [68] and by upgrading the HRP-3 robot that inherited the HRP-2 mechatronics [69]. This architecture involves distributed control across multiple independent controllers that communicate with the master controller and each other peer-to-peer, enabling them to collaborate and exchange information to achieve coordinated behavior. It provides scalability,

fault tolerance, and adaptability. The control system in the robot BHR-2 adopted the distributed method [70]. The robot control system consists of three computers: one for the vision and hearing processes, one for the real-time motion control that coordinates distributed joint controllers, and one for teleoperations. This method substitutes ordinary centralized control architectures, parallelizes the control processes to provide scalable computing power at low energy, and contributes to reliable operations using its network instead of long sensor wires.

The evolution of control architecture topology for humanoid robots, in general, has been accompanied by a progression in the capability of these robots. It is leading the control methods to manage advanced techniques that can handle the complexity of humanoid motions. However, coordination and synchronization among distributed or decentralized controllers are complex and require robust communication mechanisms. The following section discusses the communication technologies that enhance the control architecture performance.

### 2.2.2 Communication Technologies

The humanoid robot control architectures rely heavily on communication buses, which play a crucial role. Since the high control levels are restricted by the update rate of the lower control levels, Employing communication protocols with a high bandwidth and deterministic transmission is strongly recommended. Addressing these constraints depends on the data type involved. For example, meeting deterministic deadlines is essential in motion control systems, whereas achieving high bandwidth is crucial in vision systems. Due to these considerations, the main characteristics that should be considered are the transmission bandwidth, deterministic transmission, robustness, and reliability. The interest in employing advanced communication technologies addressed in enhancing the update rate over time in the humanoid robots' projects like in the HRP project shown in Figure 2.2, where the first predecessor started with 5ms cycle time, then it was enhanced by using other communication protocols to achieve 2ms in the robot HRP-4C, and then it achieved 1ms in the HRP-5, the last robot in the series. The enhancement has

been accomplished by changing the communication bus or only changing the configuration. In the HRP project 2.2 the first communication bus is an ordinary PCI bus in HRP-2 robot controlled by a centralized control system, then replaced by an Ethernet bus in HRP-3P, then CAN protocol is used in HRP-3, HRP-4C, and HRP-4 with updated configuration which reduces the cycle time from 5 to 2 ms, and finally the EtherCAT protocol was used in HRP-5 achieving shorter cycle time.

In some robotics applications, customized communication protocol has been developed, like in HRP-3P [68], which is an Ethernet-based communication bus, Herbert [71], and MAHRU [72] robots. Recently, humanoid robots were developed based on industrial communication protocols, such as the CAN Bus, which is familiar for use in the automotive industry. CAN Bus is used in HRP-3 [73], iCub [34], and Hubo [74]. However, most advanced robots using an EtherCAT communication protocol, such as Atlas [75], Lola [76], Talos [77], and Walk-Man[78]. EtherCAT communication protocol is advantageous because it serves both hard and soft real-time communication, enabling an advanced control system [79]. Moreover, EtherCAT provides high bandwidth (100 Mbit/s) compared to other communication protocols like CAN protocol with (1Mbit/s) that is used in HRP3 [73], HRP4 [80], and other humanoid robots [81]. Multiple CAN networks are used in parallel for some robots but still have a low bandwidth range; therefore, the update rate was still determined between 200 and 500Hz. The Sercos-II with only bandwidth (16Mbit/s) is used for the robot TORO [82], reaching to 1kHz update rate. Sercos-III (100Mbit/s) is used with LOLA [83] robot and achieves a 1kHz update rate. Similarly, two teams used the EtherCAT communication protocol in their control architecture; EtherCAT bus was used with RoboSimian [84], where the update rate was up to 1kHz, and with LOLA [76] robot again this time, the maximum achievable update rate was 2kHz.

### 2.2.3 Software and Middleware

Robotics' software and middleware are affected significantly by the control architecture topology and the used communication bus. They provide essential tools,



frameworks, and libraries for various functionalities suitable to robot control and communication systems to develop and operate humanoid robots.

In the centralized control system, the software is executed by the central computational unit, whether a computer or controller board. A bare-metal approach or an operating system approach can program computational units. The software in the bare-metal approach runs directly on the hardware. It encompasses codes for reading sensors, controlling actuators, executing tasks, and performing robot behaviors. While the software runs on top of an operating system that deals with the peripheral input and output devices and their drivers, it interprets the devices' signals to data that the software understands without needing to interact directly with the hardware where the operating system abstracts the hardware details. The hardware abstraction layer is usually managed by middleware that bridges a humanoid robot's hardware and software components. It facilitates communication, data exchange, and coordination among modules or subsystems.

Choosing between a bare-metal or operating system approach is problematic in distributed or decentralized control systems with multiple computational units. An operating system-based approach can provide more flexibility because of the ability to run multiple algorithms in parallel. Still, it faces latency, which refers to the time the system takes to respond to an event or request from a user or a hardware device. Deterministic performance is therefore required, which is difficult to achieve due to managing the operating system's interrupts, shared resources, cache, and software complexity. Familiar general-purpose operating systems like Windows, Mac, and Linux are configured to provide optimal overall performance but are unsuitable for deterministic requirements. Linux is an open-source operating system, and it can be patched in order to respect real-time criteria. Preempt-RT, RTAI, and Xenomai [85] are commonly-used patches. Other commonly used real-time operating systems are VxWorks, WinCE, FreeRTOS, ThreadX, and QNX [86]. The operating-system approach is applied in almost all humanoid robots for whole-body control in the central computational unit. However, the bare-metal approach for the whole-body controller is applied in the Herbert robot [71] when the bare-metal approach is generally used at the joint level controllers.

Using robotics middleware assists in managing the software architecture. Various robotics middlewares are available; the most popularized middleware is the Robot Operating System (ROS) [87], but the first version lacks real-time capability. Among the middlewares that focus on deterministic and real-time performance, there is YARP [88], OROCOS [89], [90], Mauve [91], and OpenRTM [92]. Some new middlewares are under development through humanoid projects, such as XbotCore middleware for the robot Walk-Man [93] and RTRobMultiAxisControl middleware for the robotic project ORHRO [94], [95]. A unique standardized real-time middleware is needed in the robotics community for real-time applications. Therefore, the development of ROS.2 is still actively in progress nowadays. It offers a range of capabilities for robotics development. One of its key advantages is interoperability. ROS.2 is designed to work seamlessly with various hardware and software platforms. This allows for greater flexibility and scalability in the development of robotics applications. With the ability to easily integrate with other systems and tools, ROS2 enables developers to create innovative and sophisticated robotics solutions.

### 2.2.4 Joint Controllers

The joint controllers differ according to the adapted control topology. These differences are represented by their capability, responsibility, and intelligence capacity to make decisions. In centralized systems, joint drivers consider joint controllers. The driver receives the control signal digitally through a communication bus or physically through driver inputs and interprets these signals to generate the required electrical power to move the joint motor. This type of joint controller is used in the robot HRP-2 [96], where the motor drivers and encoders are connected to I/O boards physically, and then the boards are connected to the central computer by PCI bus.

In contrast, the HUBO-2 and BHR-2 robots have a distributed control system, and their joint motor controllers receive the desired position transmitted by the master controller by CAN network [31], [70]. The position feedback is connected

to joint controllers, enabling them to perform adequate position controls at a low control level with a higher update rate than the higher control level. These controllers can operate in different modes, such as torque, position, and force control, each suited for specific tasks and applications.

The decentralized control systems enable joint controllers to make decisions and dominate the joint, demonstrating their intelligence ability. This joint controller type is used in the robot COMAN [40], the compliant humanoid robot. It performs an impedance control technique to plan and realize a task-appropriate impedance profile by establishing stable and compliant contact with objects while generating sufficient rotational torques. Its joint controllers play a significant role in managing the interplay between achieving impedance control and delivering the desired torques.

The advanced joint controller can generally perform position, torque, and force control. Robots like TALOS exemplify position control [43], HRP-5P [29], Valkyrie [37], iCub [97], and WALK-MAN [41], where they use position feedback to achieve precise joint positioning, enabling them to navigate challenging terrains and interact with the environment. Robots like HUBO-2 exemplify torque control [98], Toro [99], TALOS [100], Valkyrie [101], and WALK-MAN [102] leverage advanced motor control algorithms to generate and regulate joint torques accurately. This enables them to maintain balance, perform agile movements, and adapt to changing environments.

Force control finds application in robots like iCub robot [103] and Boston Dynamics' Atlas. Atlas is designed to perform complex manipulation tasks in dynamic environments, and its force-controlled joints allow it to exert controlled forces for tasks like tool usage or object manipulation.

This variety in joint controllers and their respective control strategies highlights the adaptability and flexibility of humanoid robots in managing diverse tasks and environments. The following section will explore the crucial components of robot morphology and actuation systems that enable these impressive control capabilities.

### 2.2.5 Human-Like Anthropomorphism in Humanoids

Humanoid robots earn their name due to their remarkable similarity to human anthropomorphism. Despite the difference in the degree of similarity among the humanoid robots developed so far, they all have standard anthropomorphic parts: four limbs, a torso, and a head, regardless of their varying sizes and the number of joints involved in creating human-like motions. Some robots possess only the necessary joints for locomotion, precisely in their legs and arms, without having any DoFs in their hands or heads, like Atlas [64] with 28 DoFs, COMAN [40] with 31 DoFs, and lots of commercial robots like Cyber-one by Xiaomi [104] with 21 DoFs, and H-1 by Unitree [105] with 26 DoFs. These robots have a range of DoFs between 20 to 35 DoFs and can perform locomotion, but they are unsuitable for applications that require manipulation like grabbing an object.

In comparison, robots with additional DoFs in their hands and heads can perform complicated tasks requiring perception and manipulation. The movable robot head assists the installed sensors in the robot head to cover more surrounding areas for better perception, and the hands' motion enables the robot to perform manipulation tasks. These robots have a range of DoFs between 40 to 60 DoFs, like Hubo-2 [30] with 40 DoFs, Valkyrie [37] with 44 DoFs, and iCub [34] with 53 DoFs.

The additional DoFs are distributed in their arms and hands, exemplified in [106] as a compact and powerful humanoid-applicable anthropomorphic 7-DoF arm with 8-DoF hand. The design closely resembles human form and motions while posing no physical threat to humans, allowing the robot to be compact enough to work with existing narrating-model humanoids while still being powerful and flexible enough to perform a wide range of tasks, including dexterous hand motions.

Other DoFs could be added to the robot body, like creating an anthropomorphic pelvis inspired by the human skeleton to improve the robot's walking characteristics, including knee stretch, heel-contact, and toe-off [107]. The use of the anthropomorphic pelvis allows the robot to adjust its center of gravity through pelvic tilt, reducing the torque required at the ankle joint and minimizing velocity

variations in human-like walking.

According to human-like anthropomorphic humanoids exploration, pursuing more significant human-like anthropomorphism by increasing the DoFs in humanoid robots leads to the development of increasingly capable machines, enhancing their proficiency in executing intricate and precise tasks, but comes with challenges, such as increased control complexity.

### 2.2.6 Actuation Technologies

#### Electric Actuators

Humanoid robots, in their initial stages, used conventional DC motors in the robot WABOT-1 [17], WABOT-2 [18], and Honda P2. The existing motors were replaced in the Honda P3 with brushless DC motors to increase the robot's reliability, efficiency, lifespan, and flexibility to perform more dynamic motions [20]; the replacement assisted in reducing noise levels for more precise control, and reducing the robot's weight from 210 kg in the P2 version to 130 kg in the P3. During the evolution of the HRP project and while developing the robot HRP-2 and enhancing the reliability of the computer system, the back-plane board for HRP-2P was replaced due to occasional hang-ups caused by noise, including PWM noise generated by servo motor drivers [96].

The brushless DC motors coupled with harmonic drive gear reductions were used in many advanced humanoid robots; they were used in the HRP-3P robot [26], HUBO-2 [32], Robonaut2 [36]. Combining brushless DC motors with harmonic drive matches some humanoid robots' requirements; harmonic drives exhibit minimal backlash, reduced weight and volume, offer superior efficiency, and produce less noise than the conventional gearbox.

Still, the electric actuation in humanoid robots faces challenges like i) the necessity for a braking mechanism to enhance the robot's efficiency during static postures. In robotic scenarios, active locking systems can control the duration and timing of the locking process. The appropriate system for a given application is assessed based on several attributes, including compactness, weight, power

consumption, locking torque, and switching speed [108]. ii) The physically safe interaction is lacking in the conventional electric actuation systems, which motivated many researchers to develop methods to overcome this challenge like the following:

**Series Elastic Actuation** Series elastic actuators have been developed for force-controlled robots, mainly for compliance robots. These actuators offer enhanced shock tolerance, enable low output impedance, and enhance stability in force control. The linear motion of the actuator, combined with the presence of an elastic element, generates small forces. When high gains are applied, this ensures precise control without undesirable oscillations [109]. Series elastic actuators were used in the Valkyrie robot [37] for conducting compliant control and protecting the robot and interacted objects from unexpected collisions.

**Passive Compliance Actuators** based on the series elastic actuation principle (SEA), it was used in the robot COMAN [39] and introduced a novel and systematic approach to optimize the tuning of joint elasticity in multi-degree-of-freedom SEA robots. This method is based on resonance analysis and maximizing energy storage.

**Variable Impedance Actuators** The variable impedance actuators are suitable for interactions with dynamic environments, including humans. They are classified based on the principles used to achieve variable stiffness and damping into four groups: active impedance by control, inherent compliance and damping actuators, inertial actuators, and combinations of the previous groups [110].

The electric actuators are the most used actuators in humanoid robots, but still limit the robots' ability to perform heavy-duty tasks with dynamic motions; for example, the robot Cassie [111], which is the lower body of the Digit-1 robot [112] has established the Guinness world record for the fastest one hundred meters by a bipedal robot. Still, it couldn't jump for more than 15 to 20 cm.

## Hydraulic Actuators

The early hydraulically actuated humanoid robots used an external hydraulic pump and lots of pipes to deliver the required fluid to robot joints, made very bulky and heavy like the Computational Brain CB robot in 2006 [44], the first version of the robot HYDROiD in 2009 [14], the Petmen robot in 2009[58], and the robot Atlas-DRC in 2012 [60]. However, as technologies advanced, the limitations of hydraulic actuators have been enhanced by reducing the pump size and their integration into the robot body, as is the case for ATLAS-Unplugged [62], and the last version of Atlas in 2017 [64]. The actuation system in these robots is powered by a central hydraulic pump with several pressure-limiting valves to regulate the pressure delivered to the joint since not all the joints require the same pressure. This results in a significant power loss and actuation system efficiency decrease. In addition, another enhancement concentrates on integrating the hydraulic pipes into the robot parts to increase the robot's mobility and flexibility to achieve more complex and precise motions like the HYDROiD [48].

Another approach to avoid robot bulkiness is the Electro-Hydrostatic Actuator (EHA) designed for robotics. The EHA is an answer to both performance and size issues, which is exemplified in the HYDRA robot [65] where multiple small-size hydraulic pumps were installed in the robot body. This technology aims to achieve both backdrivability and high control bandwidth for robots operating in uncertain environments. EHA revolutionized the Hydra's whole-body control performance, including bipedal locomotion, marking the first instance of bipedal locomotion by an EHA-driven humanoid robot. The robot can achieve position feedback control with sufficient stiffness to enable position-based locomotion. The joint backdrivability plays a role in mitigating the impact of disturbances applied to the robot's distal part on the overall motion of the entire body, resulting in realizing torque control-based locomotion that combines proper Center of Mass (COM) stabilization and null space compliance [113].

A hydrostatic transmission actuator was developed to be installed in the robot HYDROiD [51], which is designed to optimize power consumption, controlled by

displacement and has the capacity for energy storage. The actuator achieves a high power-to-mass ratio and produces substantial power at low speeds within a compact space.

In addition, An advanced Integrated Electro-Hydraulic Actuator (IEHA) has been developed with a focus on delivering efficient and compliant actuation [55]. This novel actuator is specifically engineered to power each joint of a robotic system independently and is designed to be positioned as close as possible to the joint itself. The design aims to enhance performance while minimizing the typical limitations associated with conventional hydraulic actuation. The innovative IEHA incorporates an integrated micro-pump with a floating barrel, enabling flow direction reversal without altering the rotation of the input electric motor. Additionally, integrating a micro-pump and a rotary hydraulic distributor ensures the compactness of this proposed solution [56].

In conclusion, the choice of actuation technology in humanoid robotics significantly impacts a robot's capabilities, performance, and applicability in various tasks and environments. Electric actuators have historically been the standard choice, offering reliability, dynamics, low cost, and easily achieving precise control. However, they come with challenges, such as the need for additional braking mechanisms to improve their efficiency and complex development to achieve physically safe interactions with the environment.

On the other hand, hydraulic actuators, initially characterized by their bulkiness and reliance on external pumps, have evolved to become more integrated and mobile. The Electro-Hydrostatic Actuator (EHA) has revolutionized hydraulic actuation, allowing for backdrivability and high control bandwidth, making them suitable for uncertain environments and complex tasks. The hydraulic transmission actuator and the Integrated Electro-Hydraulic Actuator (IEHA) have further enhanced the power-to-mass ratio and energy efficiency of hydraulic actuation.

The choice between hydraulic and electric actuation should be made based on the specific needs and requirements of the robotic application, considering factors like force, dynamics, ease of control, power consumption, and physical safety. Hybrid solutions may offer the best of both worlds in the near future, enhancing the



capabilities of humanoid robots across a wide range of tasks and environments.

### 2.2.7 Applications And Control Architecture Variations

Each of the factors listed, including control structure, degrees of freedom, actuation types, joint controllers, software, middleware, and communication technology, collectively exert a pivotal influence on the performance of humanoid robots. This implies that when selecting these factors, careful consideration must be given to the ultimate purpose of building the robot, which includes the intended tasks, operational environment, and various other variables.

#### Locomotion Applications

A robot aimed at achieving basic locomotion tasks in a well-known environment could be constructed with a reduced number of DoFs. Such a robot might employ a centralized control system, where the central computational unit connects to peripheral devices like cameras and LiDAR sensors for navigation, as well as joint drivers through physical I/O terminals, eliminating the need for a communication bus for locomotion execution. The actuation system would then be tailored to the robot's payload requirements. Some of the commercial robots are the closest examples of those robots, like the robot Digit-1 by Agility robotics [112], Cyber-one by Xiaomi [104].

#### Rescue Applications

A robot designed for rescue operations demands advanced locomotion and manipulation tasks to navigate uneven terrains and execute highly precise motions in uncertain environments. It requires an increased number of DoFs in its hands and precise actuation types controlled by more complex control architecture, such as the distributed control system, to deal with the high number of advanced joint controllers that provide efficient and accurate control. Additionally, the software and middleware adopted for this robot must be robust and fault-tolerant, allowing

it to operate effectively in unpredictable and hazardous conditions. Finally, the communication technology used in the robot must be reliable and efficient, allowing it to transmit and receive data over long distances. The robot DRC-HUBO+ exemplifies this kind of robot [33].

### **Human Interaction Applications**

A robot designed to navigate and work in environments that are adapted to humans for physical interaction or collaboration with humans, requiring high physical safety and compliance of the whole robot and on the joint level. The control tasks should be distributed across multiple joint controllers that operate relatively independently and make local decisions based on their inputs and set points to achieve position and active impedance controls. Therefore, the Real-time decentralized control allows effective control of these compliant joints, enabling the robot to respond rapidly to external forces and interact with its environment safely. This approach allows each joint to adapt independently to the dynamic requirements of the task.

The robot COMAN [40] uses Series Elastic Actuators to provide passive compliance to absorb high bandwidth impacts and enhance its physical interaction performance. The Decentralized control of its compliant joints provides accessibility to impedance control, joint configuration, speed, torque measurements, and force-torque readings. This level of flexibility and accessibility is essential for research, development, and fine-tuning of the compliance control strategy. It allows local adjustments and coordination across the robot's multiple DOFs to achieve effective compliance control.

In summary, the choice of control architecture in robotics is profoundly influenced by the specific application for which a robot is designed. Whether for locomotion, rescue operations, or human interaction, the complexity of control systems, the number of Degrees of Freedom (DoFs), and the level of compliance and precision vary according to the intended use case. As technology advances, robotics continues to adapt, offering tailored control solutions to meet the unique

demands of diverse applications. Therefore, carefully considering the robot's intended purpose is crucial in determining the most suitable control architecture for optimal performance and effectiveness in various real-world scenarios.

## 2.3 Conclusion

In this comprehensive State-of-the-Art chapter, diverse facets of humanoid robotics were explored, addressing the fundamental topics outlined in the table of contents determined according to the key findings of the biomedical study chapter. The chapter highlights the significance of bipedal robots and the multifaceted challenges that humanoid robots encounter to emulate human-like capabilities. As the chapter delved into the historical evolution of humanoid robots, it uncovered the intricate control architectures that enable their motion and interaction.

Furthermore, the chapter delved into the pivotal role of communication technologies, software, middleware, joint controllers, and human-like anthropomorphism in shaping the landscape of humanoid robotics. These elements interplay together to empower humanoids with the ability to perceive and interact with their environment in increasingly sophisticated ways. In exploring actuation technologies, the discussion focused on the essential but challenging aspect of powering humanoid robots by examining the evolution of electric and hydraulic actuation systems and how they influence a robot's capabilities. The search for a harmonious blend of the advantages of both technologies hints at an exciting future for humanoid robots.

As we transition to the upcoming chapter on "Servo Electro-Hydraulic Actuators," our journey continues into the realm of hybrid actuation systems, where a deeper dive into this innovative actuation technology promises to unlock new horizons in humanoid robot performance.



# Chapter 3

## Servo Electro-Hydraulic Actuator (SEHA)

### Contents

---

<b>3.1</b>	<b>Introduction</b>	<b>62</b>
<b>3.2</b>	<b>SEHA Development</b>	<b>63</b>
3.2.1	SEHA Overview	63
3.2.2	SEHA Characteristics	63
3.2.3	SEHA Functional Structure	64
<b>3.3</b>	<b>SEHA Modules</b>	<b>67</b>
3.3.1	Hydraulic Power Generation Module	68
3.3.2	Servo Speed Control Module	70
3.3.3	Servo Force Compensation Module	72
3.3.4	New efficient hydraulic circuit	73
3.3.5	Output hydraulic cylinder	75
<b>3.4</b>	<b>SEHA Functioning Principle</b>	<b>78</b>
<b>3.5</b>	<b>Conclusion</b>	<b>80</b>

---

### 3.1 Introduction

In the previous chapter, we concluded that there is a need for hybrid actuators to cover the disadvantages of the actuation systems that are used in robotics. This chapter presents the Servo Electro-Hydraulic Actuator (SEHA) as a context of this work. The start-up KALYSTA team conducted the project, and this work contribution to the ongoing development of the SEHA is focused on developing the control architecture hardware and software elements, which will be explained in detail in the following chapter (SEHA-Based HYDROïD Control Architecture - Approach and Development 4). This chapter aims to give a comprehensive projection of the SEHA actuator to assist in designing a suitable control architecture and improve the performance of the actuation system.

Developing hybrid actuators, particularly in electro-hydraulic systems, will result in a pivotal advancement in robotics. The hybrid electric and hydraulic actuators take advantage of the synergistic benefits of both technologies. It seeks to address two fundamental challenges with a long history in the field: physical safety and efficiency. The physical safety in electric actuators is exemplified by elastic actuators driven electric motors; they have significantly improved the physical safety aspects of robotic systems. However, electric motors encounter limitations in holding loads during static states of joints. On the other hand, hydraulic actuators exhibit the unique ability to hold loads efficiently in static states, thus minimizing power consumption. Still, traditional hydraulic actuation systems employ a centralized hydraulic power unit to drive all active joints uniformly, neglecting the fact that not all joints require the same amount of force. This inefficiency results in a substantial power loss and a subsequent decrease in overall system efficiency. Furthermore, hydraulic actuators have historically lacked the physical safety features necessary to handle unexpected shock interactions, which are increasingly relevant in real-world applications. According to these considerations, integrating electric and hydraulic actuation systems addresses these key challenges and advances the safety and efficiency of robotic systems. This chapter focused on the characteristics, structure, development, and operation of the hybrid Servo Electro-Hydraulic

Actuator (SEHA). Using SEHA in robotics enables the replacement of the single hydraulic power unit, supplies the active joints with multiple power units and integrated actuators, and delivers the required pressure punctually at the joint for increased efficiency during the system's functioning.

## 3.2 SEHA Development

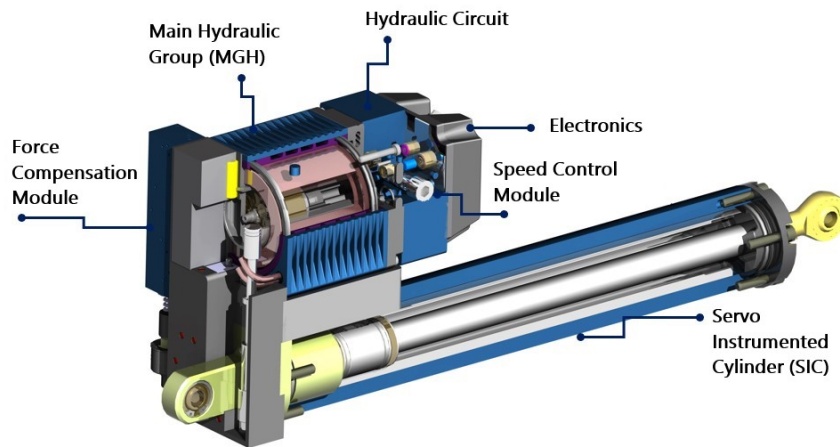
### 3.2.1 SEHA Overview

SEHA is a full-featured "Plug and Play" actuator. It offers the highest market performances in terms of compactness (Force/Volume and Force/weight ratios). SEHA enables new applications by simplifying the developer's system architecture, costs, and go-to-market life cycle. Using the hydraulic and electric technologies mingled in one compact device permits to deliver at the same time the performance of Hydraulic technology and the performance of Electric technology (price, ease of use and control, and maintenance). It comprises i) a power generation module, ii) a highly instrumented hydraulic actuator named Servo instrumented Cylinder (SIC), iii) hydraulic circuit blocks, iv) internal hydraulic feedback for safe operation, v) required sensors for reasonable control, and vi) electronic cards to create an effective standalone unit as shown in Figure 3.1.

### 3.2.2 SEHA Characteristics

The Servo Electro-Hydraulic Actuator exhibits the following characteristics:

- Improved power-to-weight and power-to-volume ratios: It can handle an estimated applied load of  $8.5kN$  while maintaining a linear speed of  $0.15cm/s$ , all within a compact form factor weighing approximately  $4kg$  and occupying a volume of  $130 \times 75 \times 200mm^3$ .
- Enhanced safety, particularly for human-robot interactions: The SEHA incorporates a force compensation block, automatically reducing the generated



**Figure 3.1:** SEHA cross-section; the figure illustrates the actuator modules structure and their placements.

pressure in case the output actuator unexpectedly hits an obstacle, ensuring safer interactions.

- Enhanced portability and autonomy: The SEHA features a hydraulic circuit centered around a 4/3 symmetric valve that allows for locking the output cylinder in place with minimal energy consumption. This design significantly extends the device's autonomy, preserving battery usage for the actual motion phases.
- Improved joint movement characteristics to mimic human-like motion: Similar to elastic actuators, the SEHA's control is adaptable, enabling adjustments to the output cylinder's speed either through the motor's rotational speed driving the pump or via the integrated block that acts on the pump's displacement mechanism.

### 3.2.3 SEHA Functional Structure

The development of Servo Electro-Hydraulic Actuator pivots around the following main functions;

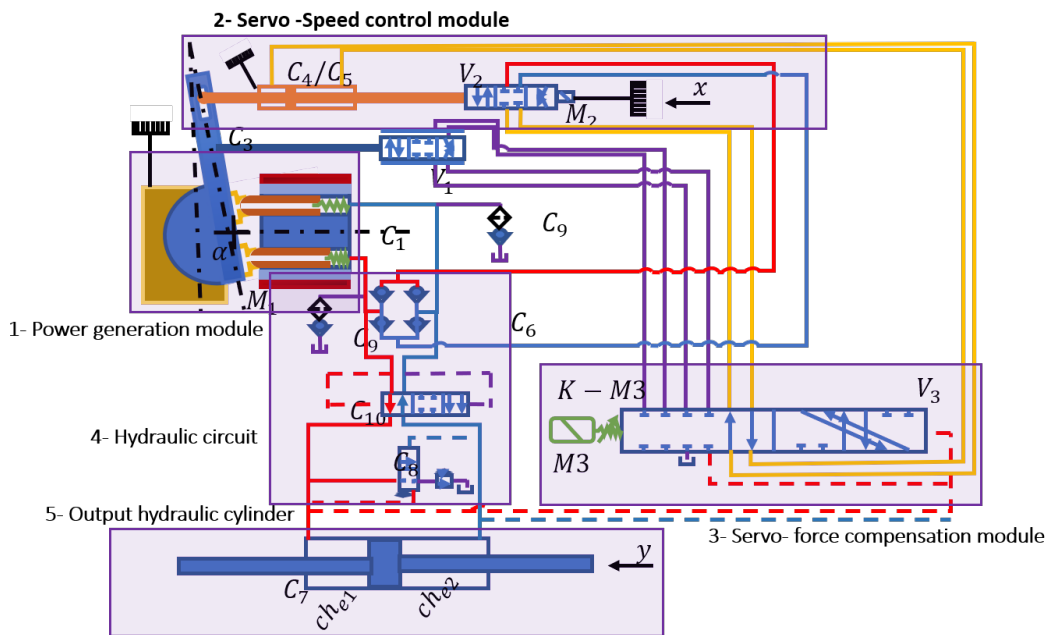


- Delivering the desired forced linear motion by the hydraulic cylinder (Extraction/retraction).
- Controlling the hydraulic cylinder speed.
- Ensuring the safe interaction with humans by protecting the hydraulic system and the personnel around the robot from the high pressure achieved by a servo force compensation module [57].

Figure 3.2 illustrates the functional structure of SEHA. At its core, a cylindrical hydraulic pump placed within the BLDC motor rotor formulates the power generation module. This pump generates a flow that controls the direction and speed of the hydraulic cylinder's output. To adjust this flow, two methods are employed: the first involves altering the speed of the electric motor. In contrast, the second utilizes a servo speed control module that impacts the pump's variable displacement mechanism. Operating an electric motor to drive the output hydraulic cylinder eliminates the need for additional directional valves, which are known to raise friction-related losses in the system. Furthermore, using a hydraulic circuit without directional valves ensures that the actuator is backdrivable, enhancing safety.

However, this approach has its drawbacks. It does not provide instant flow generation due to the gradual increase in rotational speed. Additionally, reversing the actuator's direction necessitates a dead speed zone for the electric motor. Consequently, a more practical approach for driving SEHA involves varying the pump's displacement. This method is achieved through a linear motor that operates a hydraulic valve responsible for controlling the flow variation mechanism, thus adjusting the pump's displacement. In the event of a pressure increase within the circuit, hydraulic feedback from the actuator activates a specialized servo force compensation valve to reduce the pump's flow and alleviate the system.

To gain a deeper understanding of SEHA, its development will be separated into modules based on their respective functions.



**Figure 3.2:** Functional structure of SEHA- 1- The power generation module: hydraulic pump and BLDC motor, 2- Servo-speed control module:  $V_2$  a 4/3 directional valve and  $C_4/C_5$  a servo-hydraulic cylinder,  $M_2$  a linear motor, 3- Servo-force compensation module:  $V_3$  a 10/2 directional valve,  $M_3$  a liner motor,  $K-M3$  a spring, 4- Hydraulic circuit, 5- Output cylinder [114]

### 3.3 SEHA Modules

Embracing a modular design strategy facilitates the total development of SEHA, resulting in reduced system complexity. This approach offers several advantages:

- **Maintenance Ease:** It enables the creation of functional modules that can be replaced for maintenance without replacing the entire actuator.
- **Commercialization Flexibility:** Modular design allows the marketing of individual modules independently rather than the entire actuator.
- **Upgradability:** It permits the enhancement of specific modules without requiring a comprehensive redevelopment of the whole structure.
- **Independent Testing:** The modular approach allows for independent testing of individual modules, ensuring their functionality before integration.
- **Streamlined Manufacturing:** By breaking down the system into modular blocks, manufacturing complexity is reduced, leading to optimization in terms of cost and time efficiency.

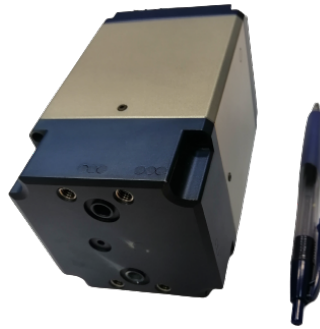
A successful modular design requires properly dividing the product into modules. Techniques for modularization in a step-by-step development process are presented in [115] [116], providing valuable tools for guiding module selection. Pahl et al. [117] outlined modular design techniques through well-defined phases. They started with task clarification, where various functions performed by the product were determined, such as generating high pressure, enabling linear motion of the output cylinder, ensuring safety, and managing motor cooling, among others. Once these functions were determined, a function structure was created to break them into smaller, fundamental, and auxiliary functions. This stage also involved a cost-effective analysis to reduce multiple sub-functions into a single function where achievable. Finally, solutions for module development were proposed, grouping sub-functions with similar operational principles into cohesive

modules. These solutions were evaluated based on economic and technical criteria to determine the most suitable modularization approach.

Following the same principle, SEHA is designed and developed as five separate function modules: i) The hydraulic power generation module (MGH), ii) the speed control module, iii) the force compensation module, iv) the hydraulic circuit module, and v) the output Servo instrumented module. Each module offer functional independence. Furthermore, this modular approach allows for the potential commercialization of three distinct blocks. The following sections will elaborate on the functioning of each module that comprises the SEHA.

### 3.3.1 Hydraulic Power Generation Module

The power generation module in SEHA actuator was manufactured as a standalone Mini Group Hydraulic (MGH), which is one of the smallest hydraulic groups in the world shown in Figure 3.3. It provides the hydraulic cylinders with in-demand pressure and flow in a compact, powerful, and lightweight power pack suitable for installation in the robot's body.

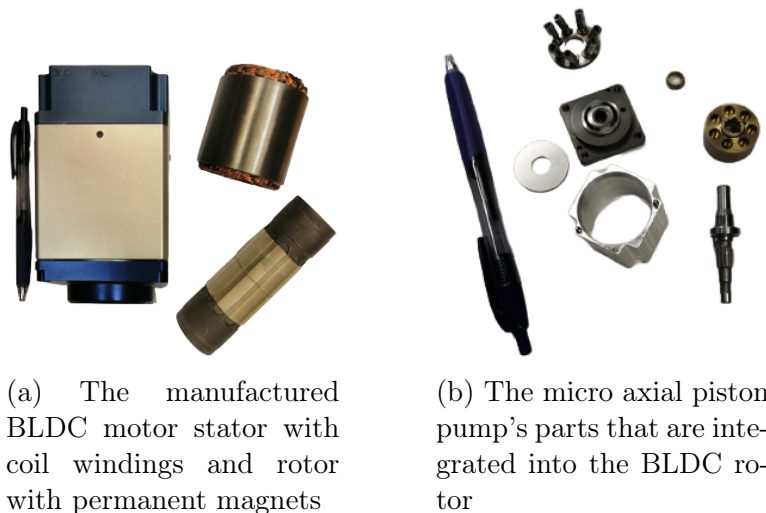


**Figure 3.3:** SEHA power generation module as a stand-alone Mini Group Hydraulic (MGH)

Conventional hydraulic pumps are driven by motors coupled to their shafts, where flexible couplings that allow for minor misalignment and limited relative movement between connected shafts are utilized.

Axial piston pumps have indicated superior efficiency compared to other pump types, like the Vane pumps, which have the lowest power-to-weight ratio due to

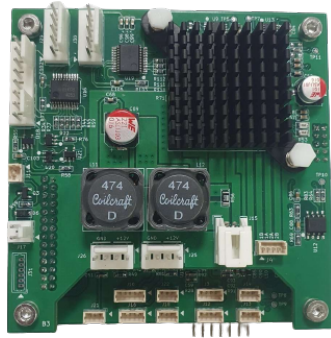
their lower pressure rating, as noted by [118]. In contrast, piston pumps offer higher efficiency and a better power-to-weight ratio, as reported by [119]. The axial piston pump, in particular, achieves an impressive 98% efficiency and operates effectively across a wide pressure range, typically between 140 and 800 bars, according to [118]. Furthermore, in radial pumps, boosting the flow rate will lead to an increase in the cylinder block diameter. In contrast, axial piston pumps can provide a higher flow by extending their length, which is a crucial feature as it enables a more optimal alignment with the human body shape. The proper motor to rotate this pump is limited to using the DC motors supplied by batteries suitable as a power source for humanoid robots. Brushless DC motors have become widely preferred due to their compact design and high efficiency. These motors feature a rotor with permanent magnets, while the stator comprises coil windings powered by direct current, resulting in a reduction in the number of friction surfaces.



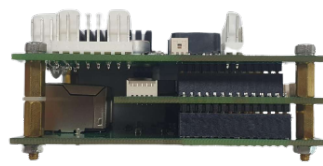
**Figure 3.4:** SEHA-MGH parts: BLDC motor and micro axial piston pump

Therefore, an axial piston pump shown in Figure 3.4b and a BLDC motor shown in 3.4a were manufactured for the MGH development. The axial pump was integrated into a brushless DC motor's rotor. The MGH integrates a customized electronic driver board shown in Figure 3.5, an oil temperature sensor, and three hall effect sensors to measure the motor speed. The customized MGH controller integrates the BLDC motor driver, stepper motor driver for the servo-speed control,

a voice-coil motor driver for the servo-force compensation, positions, pressures, and temperature sensors' inputs, and EtherCAT interface ports. The controller development and its functional blocks will be discussed in detail in the Realization and Validation chapter 5



(a) MGH- controller upper view 10\*10\*5 cm



(b) MGH- controller side view 10\*10\*5 cm

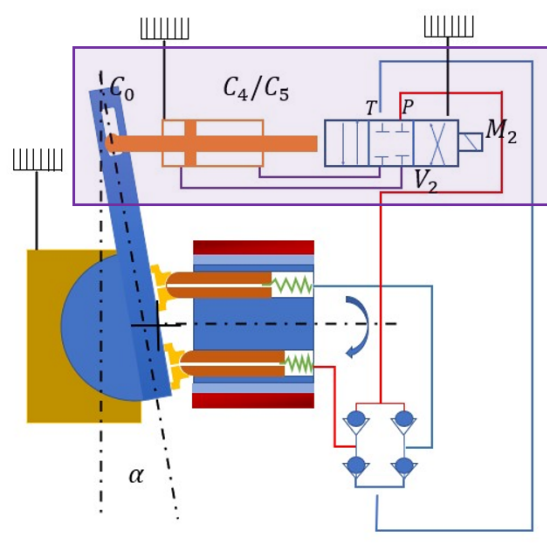
**Figure 3.5:** SEHA-MGH manufactured controller: BLDC motor driver, stepper motor driver for the servo-speed control, voice-coil motor driver for the servo-force compensation, positions, pressures, and temperature sensors' inputs and Ether-CAT interface ports.

### 3.3.2 Servo Speed Control Module

The speed control of SEHA's output cylinder can be conducted, as mentioned previously, either by controlling the motor's rotational speed or the swashplate displacement. The motor rotational speed controls the developed EHA by Nakamura et al., which provides high backdrivability and fewer components in the system, resulting in reduced friction losses [120]. However, this control method fails to generate constant power during the speed increasing from zero to the nominal speed to overcome the pump's inertia. On the other hand, some EHAs are controlled by controlling the swashplate angle displacement [121]. This control strategy demands additional components that make the system more complex while enabling the attainment of displacement variation.

Figure 3.6 illustrates a servo speed control module connected to the MGH module explained in section 3.3.1. It is necessary to upgrade the MGH module

to a variable displacement pump to be controllable by the servo speed control module. This module has a symmetric servo-hydraulic cylinder  $C_4/C_5$  that rotates the swashplate. The cylinder rod is mechanically fixed to the static part of a directional valve with mechanical feedback  $V_2$ . The linear motor  $M_2$  drives the dynamic part of the valve  $V_2$  and provides an offset at the initial operation state to deliver the fluid to the cylinder. The cylinder moves the swashplate's actuation shaft until closing the open valve  $V_2$ . The valve opening value specifies the linear displacement of the swashplate actuation shaft and defines its rotational angle  $\alpha$ .



**Figure 3.6:** Servo speed control module connected to the MGH swashplate  $C_0$ , a directional valve  $V_2$  acting on a servo-hydraulic cylinder  $C_4/C_5$  with mechanical feedback and controlled by linear motor  $M_2$ .

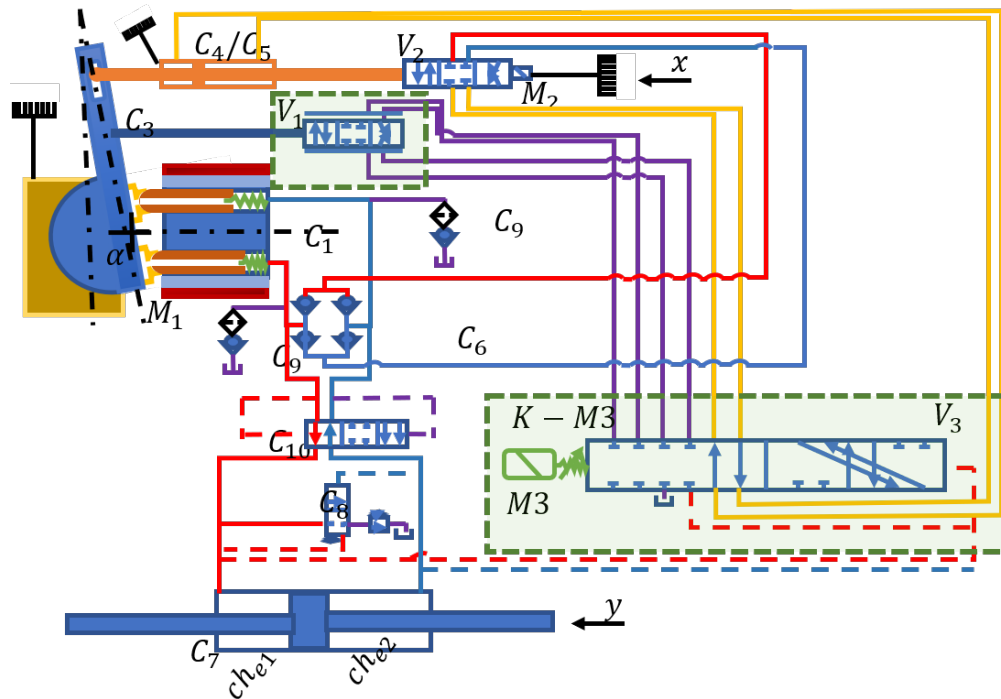
Building a directional valve with mechanical feedback on the active piston offers the benefit of improved integration of the spool within the module's body and reduces internal valve leakage. This setup eliminates the need for sensors since any offset occurring in  $V_2$  is automatically rectified at the  $C_4/C_5$  stage. Additionally, the position of the swashplate can be maintained without consuming any additional energy. Still, the position sensor to measure the swashplate's rotational angle is needed to control the output hydraulic cylinder speed.

The control loop of the servo-speed control module must involve the linear

motor M2, measure the swashplate angle by a rotational position sensor, and the output hydraulic cylinder speed by a linear position sensor. These I/O's are considered in the MGH module controller shown in Figure 3.5

### 3.3.3 Servo Force Compensation Module

The physical safety in electric actuators is achieved mainly by series elastic actuators. Their integration into humanoid robots enables torque or force feedback for improved control. They make a low-pass filter for shocks because the elastic elements are not transmitted to the actuator, and its load will pass via the same low-pass filter [122].



**Figure 3.7:** SEHA-Servo-force compensation module in the deactivated state.

Figure 3.7 shows the deactivated servo-force compensation module. This module is responsible for the physical safe interaction and the compliance of the actuator. It places the compliant element in the actuator's feedback. Limiting the force is achievable by controlling the liner motor M3's displacement to adjust the







rical and asymmetrical. Figure 3.7 shows an asymmetrical cylinder. In this type, there is a difference in volumes between the cylinder chambers due to the presence of the rod in one of them. During piston retraction, the oil volume entering chamber  $Ch_{e2}$ —necessary to move the cylinder—is smaller than the volume leaving chamber  $Ch_{e1}$ . This oil overflow from  $Ch_{e1}$  must be contained and returned to the reservoir. A back-pressure valve controls the oil overflow [124], which consists of a pressure-limiting valve and a 2/2 directional valve. Excess oil is directed back to the reservoir when the directional valve is activated. In the same manner, SEHA’s system ( $C_8$ ) achieves that in the retraction phase. In contrast, a larger oil volume is necessary to recompense the chambers’ volume difference and the internal leakages while the cylinder is extracting. SEHA’s valve ( $C_9$ ) compensates the oil in this phase by drawing from the reservoir.

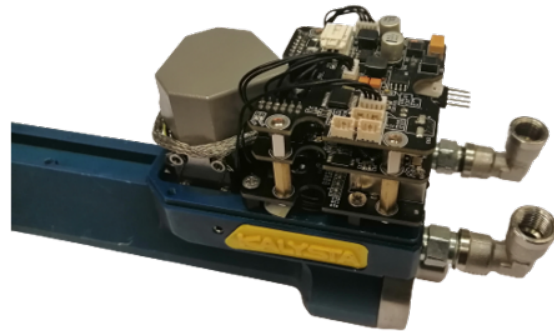
The new efficient hydraulic circuit in SEHA significantly enhances humanoid robot stability and power conservation during static postures. It secures the cylinder’s position and remains in place without additional energy consumption; everything is mechanically and automatically actuated without any control intervention. The control can predict the short and long pauses to decide to control the swashplate’s angle or switch off the motor when the hydraulic circuit securing the cylinder to effectively manage the power consumption.

### 3.3.5 Output hydraulic cylinder

The output hydraulic cylinder module in the SEHA actuator was manufactured as a standalone Instrumented Hydraulic Cylinder (SIC).  $C_7$  shown in the previous figures like 3.2 is the output cylinder that converts hydraulic energy into mechanical energy. Its full integration into SEHA eliminates the need for hydraulic pipes. The SIC module offers two potential applications: It can either be integrated into SEHA, driven by its hydraulic circuit outlined in section 3.3.4, or as a standalone product driven by a servovalve.

The SIC integrates a customized electronic driver shown in Figure 3.10, a servo-valve actuator, a linear position sensor, a full bridge strain gauge, an oil

temperature sensor, and two pressure sensors. The customized SIC controller integrates the servo-valve driver, position, pressures, strain gauge, and temperature sensors' inputs interfaced with EtherCAT ports. The controller development and its functional blocks will be discussed in detail in the Realization and Validation chapter 5



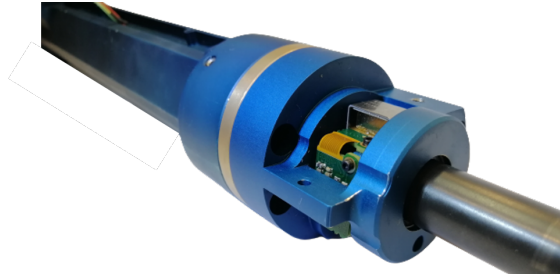
**Figure 3.10:** SEHA-SIC module: Servo Instrumented Cylinder with the integrated servo-valve and the module controller

The integrated SIC module's sensors are the following:

**Position Sensor:** Measuring the position accurately and at high update rates is critical in applications using linear hydraulic cylinders to measure the speed of those providing essential feedback for controlling joint motion. The integrated linear position sensor into the SIC module was chosen among many. A deep comprehensive comparison of the available solutions was carried out regarding their compactness, robustness, reliability, and stability [114].

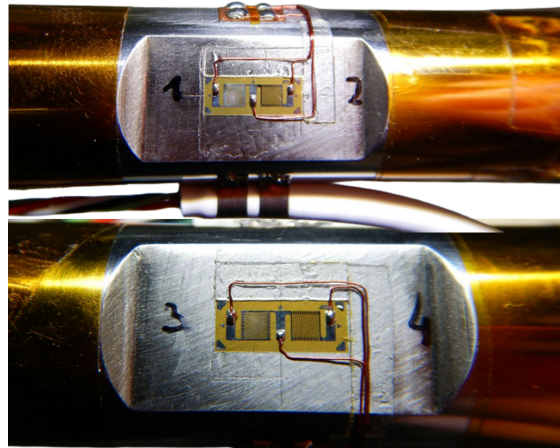
In the chosen solution, the magnet is situated within the rod, and the sensing element is positioned within the cylinder cap, shielded by seals, illustrated in Figure 3.11. This configuration offers the advantage of a fixed distance between the magnet and the sensing element, enhancing the sensor's lifetime and increasing the sensor reading's reliability.

**Strain Gauge Sensor:** Force control is advantageous in humanoid robots for safe interaction and collaboration with humans. The integrated force sensing into the SIC module was chosen after comparing the most popular force sensing



**Figure 3.11:** SEHA-SIC module: Integrated position sensor

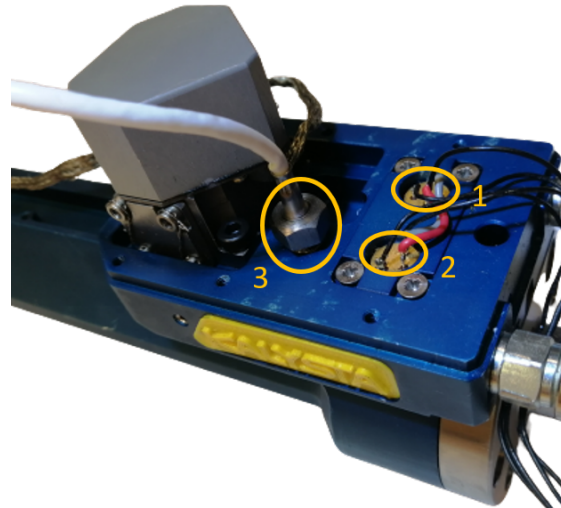
methods used in humanoid robots. The comparison included load cell, strain gauges, six-axis force/moment, optical force, and Piezoelectric force sensors [125].



**Figure 3.12:** SEHA-SIC module: Integrated full-bridge strain gauge sensor placed on the cylinder rod's end [125]

The strain gauge force sensor was chosen due to its superior qualities. Strain gauges provide high reading accuracy and sensitivity in a compact size and cost-effectiveness, offering a direct strain measurement for increasing small forces' sensitivity. They are smaller and lighter than load cells, making integration into compact robotic systems easier. Moreover, strain gauge sensors offer a wide sensing range, enabling accurate force measurement across a broad spectrum, suitable for various applications in robotics. Additionally, these sensors exhibit versatility and can measure normal and shear forces, making them ideal for tasks involving a wide range of robotic forces, such as grasping and manipulation.

**Pressure & Temperature Sensors:** The cylinder motion speed is performed by the flow rate entering and exiting the cylinder chambers. The flow rate is directly affected by the pressure of these chambers. Therefore, measuring the cham-



**Figure 3.13:** SEHA-SIC module: Integrated pressures (1,2) and oil temperature (3) sensors

bers' pressures provides essential parameters in the speed control loop. Meanwhile, the variation in oil temperature changes the oil viscosity, consequently affecting the flow rate. The oil temperature impacts defining the secondary parameters for better control and identification of the actuator. These sensors are integrated into a designed cap that permits combining the electronic board and the servovalve in the cap. As a result, the electrical wires are kept short, and the electronics take up a relatively small volume on the barrel.

Figure 3.13 illustrates the integrated pressure and temperature sensors under the module controller. The length of their cables is less than 15 cm, enough to be connected to the electronic controller.

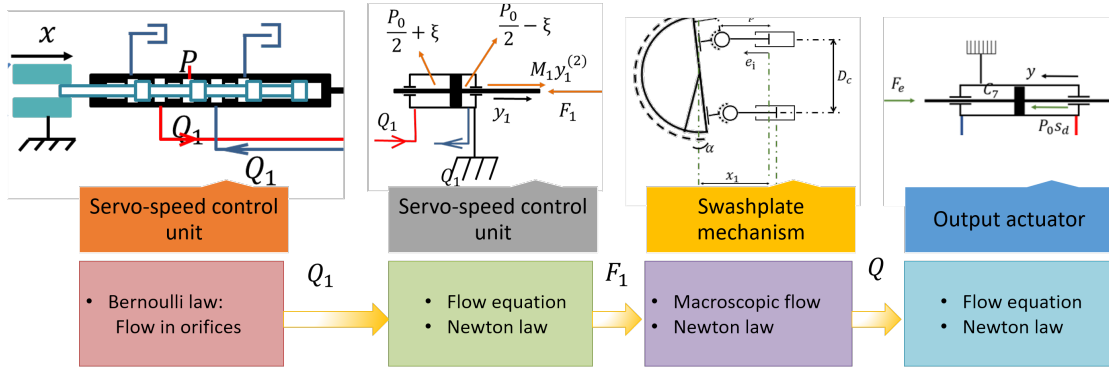
### 3.4 SEHA Functioning Principle

Referring to Figure 3.7, the normal functioning of SEHA can be described as follows: SEHA's input is the servo speed control module: its valve ( $V_2$ ) is actuated

by the voice coil motor ( $M_2$ ) that allows for a linear movement of the spool  $x$ . Applying Bernoulli's law on the valve's orifices the flow that crosses the valve  $Q_1$  is calculated (Figure 3.14)

$$Q_1 = f(d, \rho, P_o)$$

where  $d$  is  $V_2$ 's diameter,  $\rho$  oil density, and  $P_o$  the operational pressure.



**Figure 3.14:** The main stages for SEHA's mathematical modeling

The flow causes the movement of the servo-hydraulic cylinder ( $C_4/C_5$ ), which in turn rotates the swashplate and yields to its rotation. The provoked offset ( $x$ ) in valve ( $V_2$ ) naturally prompts the servo-hydraulic cylinder ( $C_4/C_5$ ) to move left if ( $x > 0$ ) and right if ( $x < 0$ ), continually striving to close the valve ( $V_2$ ) due to the mechanical link between the servo-hydraulic cylinder's piston ( $C_4/C_5$ ) and its body. By applying the flow equation movement on the  $C_4/C_5$  and Newton's law, the force  $F_1$  exerted on the swashplate mechanism can be calculated:

$$F_1 = f(M_1, y_1, x, P_o, \xi, \rho)$$

with  $M_1$  the weight of the  $C_4/C_5$  piston,  $y_1$  its linear movement, and  $\xi$  the pressure variation in the chambers.

The rotation of the swashplate translates into a variation in the flow of the

pump. This translates into the generation of the pump's flow  $Q$ .

$$Q = f(N, S_p, D_c, \omega, \alpha)$$

where  $N$  is the number of pistons in the pump,  $S_p$  their cross-section,  $D_c$  the pitch diameter,  $\omega$  the rotational speed, and  $\alpha$  the swashplate angle.

Concurrently, the pump's oil flows into the hydraulic output cylinder ( $C_7$ ), equipped with a hydraulic circuit module, to drive it in a linear movement  $y$ : the SEHA system output. Applying the flow equation and Newton's law on the output hydraulic actuator, the relationship between the input  $x$  and the output  $y$  can be generated. It was explained in detail in [114].

$$y = f(x, \dot{y}, y^{(2)}, y^{(3)}, y^{(4)}, y^{(5)}, y^{(6)}, \omega, \omega^{(2)}, \omega^{(3)})$$

In case of an unexpected surge in pressure resulting from a high-force incident, SEHA acts automatically to decrease the flow and bring the system to a safe operation point. In fact, SEHA is equipped with a force compensation module that comprises a stepper motor ( $M_3$ ), a spring  $K - M3$ , and a two-position hydraulic valve  $V_3$ .  $M_3$  pre-compresses the +spring  $K - M3$  to create a resisting force that acts on the spool of  $V_3$ . When the force at the other end of the spool, caused by the applied pressure, exceeds the force caused by the spring compression, the valve  $V_3$  is automatically activated. This will bypass the valve  $V_2$  to bring the swashplate toward the neutral position and relieve the system from increased pressures.

### 3.5 Conclusion

This chapter provided an overview of SEHA, a hybrid modular actuator, detailing its characteristics, structure, development, and operation. Controlling the actuation system in humanoids is a part of their control architecture, where the actuation system controllers are the control architecture joint controllers; therefore, understanding the used actuation system provides a clear vision of the system



requirements for developing controllers that satisfy the robot control architecture needs.

The modular approach employed in SEHA involves multiple interconnected modules working concurrently to accomplish the system's functionalities.

The main modules include i) the main hydraulic group (MGH), which integrates a BLDC motor, three hall effect sensors to measure the motor speed, and an oil temperature sensor, ii) a servo speed control module, controlling the MGH flow rate by integrating a linear motor M2 to actuate the valve ( $V_2$ ) to move servo hydraulic cylinder ( $C_4/C_5$ ) and a rotational position sensor to measure the swashplate angle, iii) a servo force compensation module with the valve ( $V_3$ ) that activated when the force at the output cylinder exceeds a preset value by the motor M3 pressing the spring on the valve spoon, iv) a hydraulic circuit module fixing the output cylinder position in its idle state without power consumption, v) a highly instrumented output cylinder integrates a servo-valve and two pressure, oil temperature, position, strain gauge sensors.

This chapter defined each module's monitoring and controlling points separately to perform the assigned functionalities in sufficient control strategies. The upcoming chapter will delve into the development of the actuation system controllers and the other elements of the HYDROiD control architecture.



# Chapter 4

## SEHA-Based HYDROiD Control Architecture - Approach and Development

### Contents

---

<b>4.1</b>	<b>Introduction</b>	<b>84</b>
<b>4.2</b>	<b>Problematic</b>	<b>85</b>
<b>4.3</b>	<b>Approach</b>	<b>85</b>
4.3.1	HYDROiD Robot State	87
4.3.2	Performance Requirements	89
4.3.3	Control Architecture' Levels	91
<b>4.4</b>	<b>Control Architecture Hardware Development</b>	<b>92</b>
4.4.1	Master Controller	93
4.4.2	Communication Protocol	94
4.4.3	Distributed joint controllers and their components	96
<b>4.5</b>	<b>Control Architecture Software Development</b>	<b>99</b>
4.5.1	Master Controller Software	100

4.5.2 Joint Controller Software . . . . .	101
<b>4.6 Conclusion . . . . .</b>	<b>102</b>

---

## 4.1 Introduction

The previous chapter investigated the functioning principle of the modular actuator SEHA and defined each module’s control and monitor I/O points. It laid the foundation for developing the SEHA-based humanoid control architecture and outlined the critical aspects of the servo speed control module, servo force compensation module, and hydraulic circuit, and emphasized the essential modules that formulate the simplest actuation system, the main hydraulic group (MGH) and the instrumented output cylinder (SIC).

This chapter aims to translate the mechanical functionalities of SEHA into an intelligent and adaptable control architecture. The development pushes to mimic the human control system regarding the distribution of decision-making centers and how it increases its complexity to respond to varying complexities of tasks, as discussed in the biomedical chapter. Integrating insights from the biomedical study chapter into the development methodology enhances the adaptability and efficiency of the humanoid control system.

Furthermore, this development is connected to the insights gained from the State-of-the-Art chapter. The influence of diverse control architectures on the performance of humanoid robots is explored, highlighting the adaptability of control strategies based on specific applications.

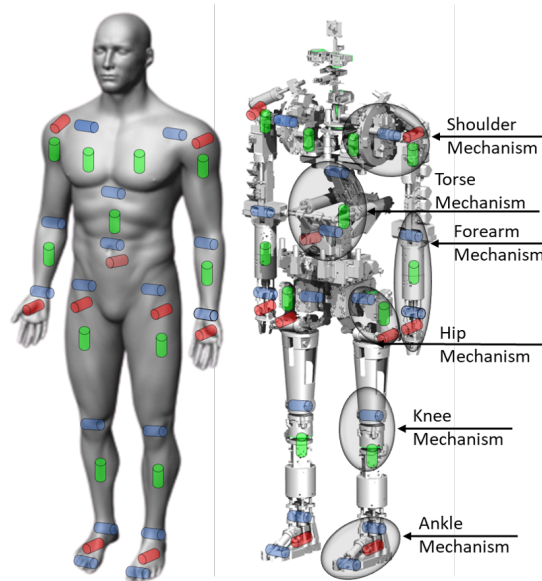
This chapter outlines the methodology and approach employed in the development process, encompassing various facets such as hardware and software development, system identification, and defining key control architecture levels.

## 4.2 Problematic

Humanoid robots navigate and work in environments adapted to humans; therefore, they must imitate human anthropomorphism, manipulation, and locomotion. Mimicking human manipulations and locomotion requires a reliable hardware design replicating complex human mechanisms, as shown in Figure 4.1, a powerful actuation system, and a robust control system delivering dynamic motions. The control architecture is a critical bottleneck for producing humanoid robots that maintain stability while performing different activities (avoiding obstacles, localization, mapping, and others). Developing this architecture still faces several challenges, mainly the following: i) Since humanoid robots aim to emulate the human body, their control systems should perform like how humans control their muscles during the stability phase and the postural transitions. ii) Humanoid robots should be able to collaborate with humans safely, and the physical human-robot interaction (pHRI) should be critically safe despite the challenging hardware and software constraints. The control architecture for achieving pHRI should be operated in real-time, with deterministic latency. iii) The distribution of numerous actuated degrees of freedom (DoFs) and associated sensors make seeking an efficient control system with enormous processing capability essential. iv) The socially accepted appearance is challenging while developing humanoid robots, where connecting all the actuators and sensors requires installing an abundant number of cables, making the robot bulky and having an unpleasant appearance.

## 4.3 Approach

In tackling the challenge of human emulation within the control architecture, the development approach focuses on creating a dynamic and adaptable system that mirrors the intricacies of the human control system, which draws inspiration from the key findings of the biomedical study chapter 1. The approach aims to imbue the humanoid control system with the ability to modify its architecture in response to assigned tasks. The first key insight from the biomedical study chapter



**Figure 4.1:** Hydroid mechanisms and corresponding DOFs

underscores the importance of the dynamics of sensorimotor programs in humans for achieving postural coordination [2]. Understanding that postural transitions are behavioral consequences of changes among programs operating at the central nervous system level provides a foundation for the development strategy. The joint controller will have pre-developed sensorimotor programs to translate these findings into the control architecture. These programs, inspired by human physiological responses, enable the humanoid robot to replicate the human control system and its sensorimotor programs while performing motions. For instance, the involuntary reaction control is decentralized, where the sensorimotor is processed locally. It is equivalent to a force-controlled sensorimotor program that makes a local decision and dominates the controller output regardless of what the higher control level requests.

Moreover, the state-of-the-art chapter's application and control architecture variation section classified the control architectures used in humanoid robots according to the robot's intended purpose. It outlines that versatile robots need an adaptable architecture influenced by their assigned tasks. Therefore, the development approach places a premium on flexibility, allowing the control system to

dynamically modify its architecture based on the requirements of different applications. This adaptability ensures that the humanoid robot can seamlessly transition between tasks, emulating the versatility of the human control system in various scenarios.

The development approach strategically utilizes the servo-force compensation module of the SEHA actuator to improve physical safety. The servo-force compensation module enhances the actuator's physical safety of individual joints when the force exceeds the preset force limit, even though the response to an action requires moving all the limb joints together. The control system can generate the response motion pattern to control all the limb joints or overall robot joints, enhancing human-robot interaction.

The development approach focuses on enhancing the cycle time for achieving better robot compliance, which is at the highest levels in the decentralized control architecture due to accomplishing its control loop in the low-level control, as discussed in the state-of-the-art chapter.

Finally, to address the third and fourth challenges mentioned in the problematic section 4.2, the approach focuses on distributing the joint controller to divide the required processing capabilities into the low-level controllers and minimize the cable numbers to connect the associated sensors and actuators to their corresponding controller to improve the social appearance of the robot, those will be discussed in detail in the upcoming sections.

This approach draws the development roadmap but still needs to understand the context of the application and translate this approach to requirements to facilitate defining the characteristics of the developed control architecture. A more profound understanding of the studied application could be acquired by analyzing the HYDROiD state and requirements.

#### 4.3.1 HYDROiD Robot State

HYDROiD is a hydraulically-actuated full size humanoid robot, created in 2009 [46] with parallel mechanisms [47], [48] in its shoulders, torso, arms, and ankles as

shown in Figure 4.1. A patent for a spherical hinge with a coupled actuator was conducted [49]. The robot has 36 DoF with servo valve-based hydraulic actuation. The robot DoFs have motion ranges shown in Table 4.1 capable of executing the measured motion angles during the walking gait presented in the biomedical study chapter’s data collection section 1.4. Table 4.1 illustrates the motion ranges of the main involved joints during the walking gate alongside the rotational speed of these joints to be used as a speed limit of the robot actuators.

DoF	Motion	Speed
Hip Flexion	$\pm 90^\circ$	150°/sec
Hip Abduction	$\pm 12^\circ$	130°/sec
Hip Rotation	$\pm 10^\circ$	120°/sec
Knee Flexion	$-5^\circ, +90^\circ$	400°/sec
Ankle Flexion	$-45^\circ, +45^\circ$	200°/sec
Ankle Abduction	$-15^\circ, +15^\circ$	240°/sec
Ankle Rotation	$-15^\circ, +30^\circ$	250°/sec
Shoulder Flexion	$-90^\circ, +90^\circ$	90°/sec
Shoulder Abduction	$-10^\circ, +90^\circ$	90°/sec
Shoulder Rotation	$-10^\circ, +10^\circ$	90°/sec
Elbow Flexion	$-5^\circ, +120^\circ$	90°/sec

**Table 4.1:** HYDROiD: Degrees of Freedom, Motion Angle, and Speed for involved joints in the walking gait [126]

During the last decade, many research axes have grown through this project. Research on decreasing the weight of the robot [50], [51] is ongoing. The emotional head is studied and developed [52]. Mechanical design is under enhancement [53]. Artificial skin is under development [54]. In addition, some advanced actuation-based research like the Integrated Electro-Hydraulic Actuator (IEHA) and several versions of IEHA have been developed [55], [56]. And the patented Servo Electro-Hydraulic Actuator (SEHA) [57], but they are not yet implemented on the robot. This thesis’s studied version of HYDROiD is based on the main modules of the SEHA actuator, the SIC driven by the servo valve, and the MGH controlled by its motor speed. Each hydraulic actuator typically has one servo-valve, two oil-pressure sensors, one position sensor, and one force sensor. In order to develop a control architecture, data quantity was analyzed. The actuation and sensor signals



of the joint controllers were unified as a float-32 datatype. In addition, each head's electric motor has a 14-bit actuation signal and a 14-bit position signal. Whole body sensors are the IMU sensor with 6x16-bit measurements and the two sets of load cells in the feet; each set comprises up to 8 sensors of 16 bits. Those pieces of data sizes are organized in Table.4.2

Element	Quantity	$N^\circ$	Size	Total Number of bits
Servo valve	36	1	32 bits	1152
Position sensor	51	1	32 bits	1632
Force sensor	36	1	32 bits	1152
Pressure sensor	36	2	32 bits	2304
Electric motors	15	1	14 bits	210
IMU	1	6	16 bits	96
Foot load cells	2	8	16 bits	256
Total Variables	232			6802

**Table 4.2:** Sensors and actuation elements of HYDROiD robot

By calculating the sum of the last column, the total payload data to be transmitted between controllers and peripheral elements is about **6800** bits. This payload data will be exchanged at each cycle of the control loop.

### 4.3.2 Performance Requirements

The control architecture for versatile humanoid robotics platforms should satisfy many needs; the objective research on the platform is generally the autonomous dynamic locomotion on uneven terrains and human-robot interaction [127]. Ideally, the specification can be defined as follows:

R1- Flexibility to apply control laws: One of the main objectives is to provide a research platform to study human sensory-motor control. The robot control architecture should allow testing sensory-motor control loops as the local loop at the joint level or as a centralized one.

R2- High bandwidth in the hardware and software: The intended humanoid robot should have High dynamic agility. The hardware and software infrastructure

should not be a constraint and should not limit the dynamic bandwidth of the robot in different tasks.

R3- Deterministic performance: The high bandwidth is essential but not sufficient. In motion control for highly dynamic applications such as a humanoid robot, a delayed command becomes a wrong command; therefore, the deterministic time performance is crucial. This requirement fetches the possibility of developing highly dynamic controllers for safe human-robot interaction.

R4- Human-like appearance: The architecture should account for this feature by preserving the compactness and the required appearance of the robot preventing bulky elements during the design of the control architecture.

R5- Portability of control architecture: The goal application is an autonomous mobile robot; therefore, the control architecture's overall design should be embedded and not depend on fixed stations.

R6- Open-source-based design: In order to permit further sophisticated development, the design should be based on open-source elements.

R7- Ease of integration: HYDROiD's system is complex due to the presence of a high number of elements in the system. One of the essential objectives is to guarantee ease of integration at all times. Moreover, the robot is in the development phase. Additional elements can be added to the architecture later. Therefore, the architecture should accept the extension with easy integration possibilities.

R8- The ease of maintenance: Regarding the system's complexity, functional-based isolation is required to facilitate maintenance during operation and tests.

R9- Reliable and robust design against the external environment: In mobile robots, robustness against shocks and vibration is required. Besides, electromagnetic interference immunity is essential in electronic design.

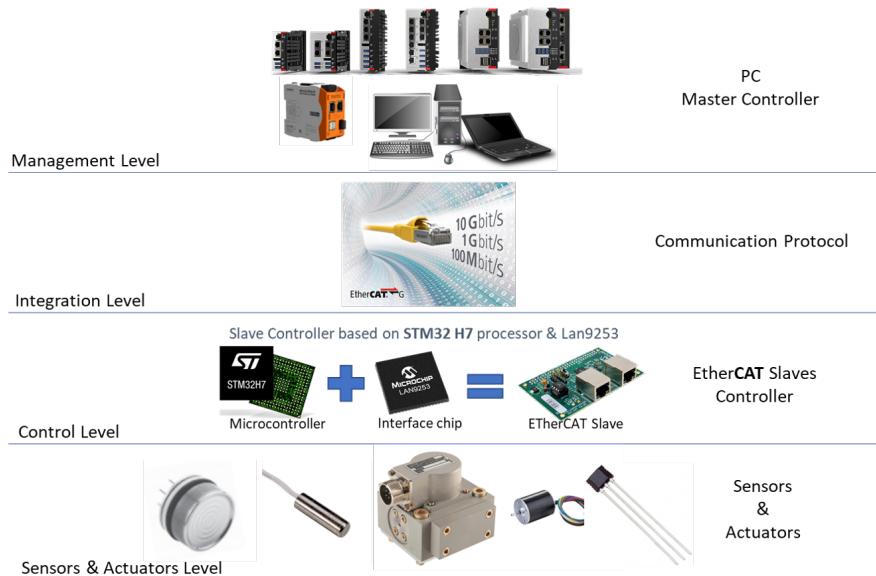
Analyzing the robot state and defining the application requirements represent a crucial step in advancing the development of a more effective control architecture. However, the hierarchical levels within the control architecture are general lines that outline the trajectory for creating a robust design.

### 4.3.3 Control Architecture' Levels

The control system's hierarchical levels can vary from one control system to another, depending on the application's specifications. Generally, four common levels provide a structured approach to designing control systems, enabling effective control and simplifying the implementation of complex processes. The control levels from bottom to top, as shown in Figure 4.2 are the following:

- Sensor and actuator level: it is the direct physical interaction level; it interacts directly with the surrounding environment of the application.
- Control level: represented by the joint controllers in robotic applications, it is responsible for implementing the desired behavior of the joint and achieving its setpoints or targets using sensor feedback.
- Integration level: focuses on coordinating the control level operations to optimize overall system performance by handling the interdependencies among microcontrollers.
- Management level: the highest level in the control system involves decision-making, planning, and coordination between the integrated subsystems.

In this development, the chosen elements in each control level work independently when some elements are replaced on one level, which will not affect the elements on the other control level. For instance, replacing a one-joint controller with a different multi-joint controller does not necessitate the replacement of associated sensors for individual joints. In this case, the modification is represented by disconnecting these sensors from the old controller and connecting them to the new one. Similarly, replacing one sensor must match the prepared interface with the corresponding controller. For this reason, the controller should provide a wide range of interfacing capabilities to adapt various types of sensors and actuators. Those considerations enhance the flexibility of the control architecture, enabling it to adapt to more intricate scenarios and cases.



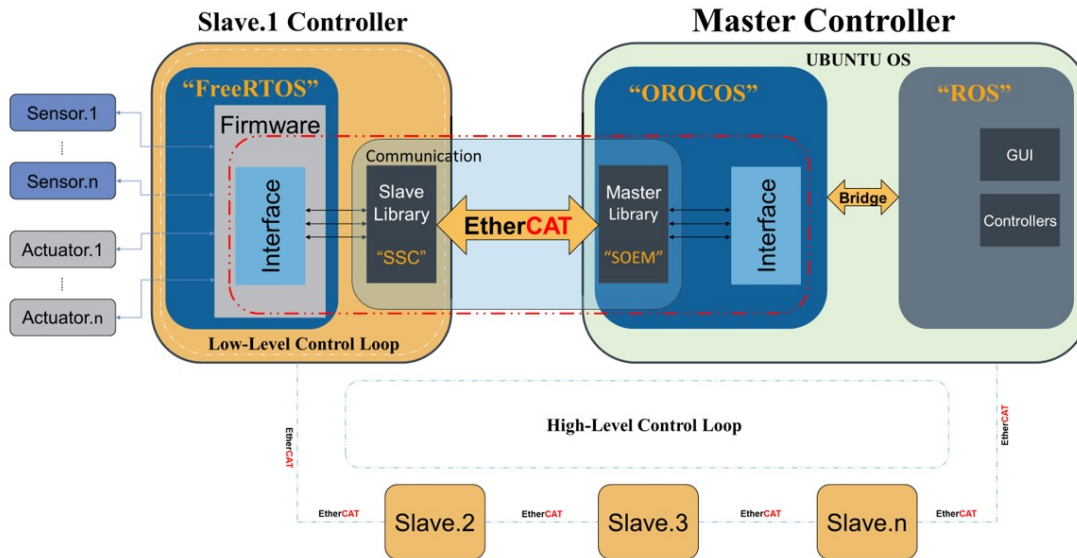
**Figure 4.2:** Control Architecture Levels

Following the previous study, we would like to design a control architecture that meets the mentioned requirement by choosing the suitable elements in every control level and ensuring compatibility by organizing the relationship among these elements.

#### 4.4 Control Architecture Hardware Development

The proposed control architecture is a distributed real-time control system where all the software and the hardware's elements are capable of performing in real-time, ensuring safe interaction with humans. as shown in Figure 4.3. It is based on Master-Slave daisy chain topology. The hardware's elements are represented by the master controller, communication network bus, and slave controllers. The software's elements are represented by the used operating system and the middleware in the master, the communication interface software in the master and the slaves, and the operating system and the firmware in the slaves.

The development of this solution was conducted with hardware and software



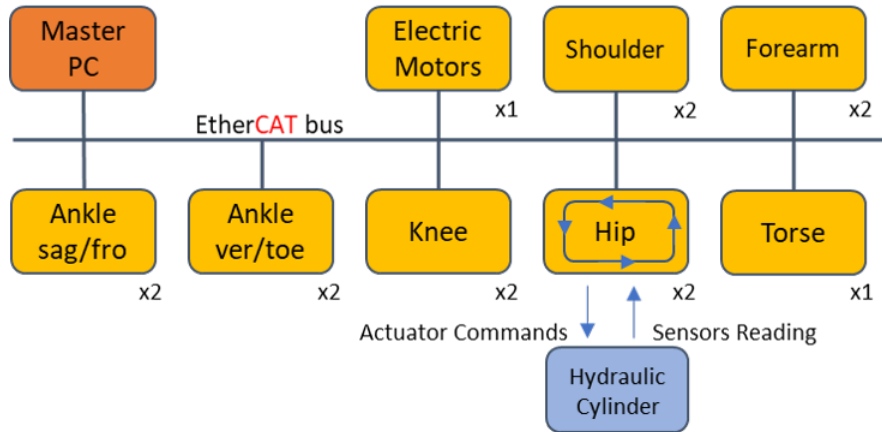
**Figure 4.3:** Software architecture defines the relation between different software.

of the high and low control levels.

The hardware architecture uses a centralized embedded computer and a communication field bus. The centralized computer communicates with a distributed network of computational units. The distributed nodes exchange data via the communication bus in order to provide states, commands, and measurements in real-time. The general structure of the design is shown in 4.4

#### 4.4.1 Master Controller

A humanoid robot needs a central computational unit in order to organize the operation of the joint controllers in a synchronized way. An industrial PC C6025 from Beckhoff has been chosen thanks to its compact size and powerful computing capabilities. The chosen PC has the following parameters: equipped with three EtherCAT ports that enable hard real-time communications, a quad-core processor Core i3, 8 GB of RAM memory, and an SSD disk. These are adequate parameters, as they allow the overall management of the control architecture.



**Figure 4.4:** Hardware architecture showing central PC, decentralized nodes and communication bus.

#### 4.4.2 Communication Protocol

According to the state-of-the-art study, there are some main constraints regarding the communication protocol, like the bandwidth, achievable update rate, and real-time capability. We have chosen the communication protocol EtherCAT, standardized in IEC 61158 and suitable for both hard and soft real-time computing requirements in automation technology [128]. Using the EtherCAT as the backbone of the control architecture is crucial for its high-performance, fast, and deterministic communication capabilities and the ability to handle large amounts of data and control multiple devices simultaneously. The proposed EtherCAT-based real-time control architecture is designed to take full advantage of these capabilities to provide advanced control and coordination of the robot's actuators. We are using approximately 6800 bits of payload data as control and monitoring signals through the 100Mbps network bandwidth. To calculate the cycle time, which refers to the required communication time by the master controller to collect and update the data memories of all sensors and actuators. According to [129]. If the payload  $x$  is less than 1486 bytes as the maximum Ethernet frame size, the corresponding equation to calculate the minimum achievable cycle time through the network is 4.1.

$$\Gamma = (2n - 1)\ell + 2n\delta + \frac{8}{C}(40 + \max(44, n(12 + x))) \quad (4.1)$$

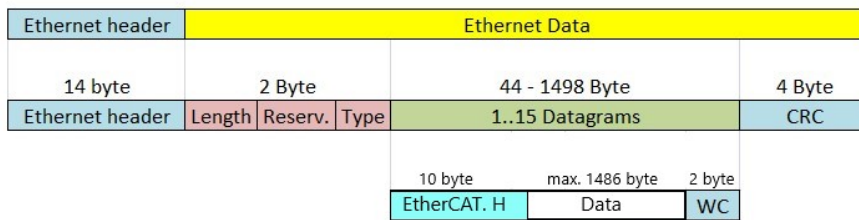
Where the transmission delay is:

$$\tau = \frac{8}{C}(40 + \max(44, n(12 + x)))$$

Terms	Notation	Units
Minimum cycle time	$\Gamma$	s
Transmission delay	$\tau$	s
Network device latency	$\ell$	s
Propagation delay	$\delta$	s
Link capacity	$C$	bits/s
Payload	$x$	bytes
Number of network devices (slaves)	$n$	- -

**Table 4.3:** Cycle Time Elements

We have 14 slaves sharing 232 bus variables with a payload of 6,800 bits, about 850 bytes. Taking into consideration, the typical propagation delay, as mentioned in [129], is 0.3  $\mu$ s/slave. if we assumed that the network device latency = 0.3  $\mu$ s/slave also; The minimum cycle time will be around 120  $\mu$ s. This configuration allows us to transmit and receive the payload data at approximately 8 kHz; this is the ideal control system without the master processor’s latency and unlimited software and middleware processing capabilities.



**Figure 4.5:** EtherCAT frame structure based on the Ethernet

The Network device latency and the propagation delay will be measured in the SEHA Control Architecture Realization and Validation chapter 5.

### 4.4.3 Distributed joint controllers and their components

The number of joint controllers on the bus is related to the number of Active Degrees of Freedom (ADoFs) in the robot. The trivial solution is to have a controller for each ADoF. Such a solution has two problems: The first problem is the bulkiness, with a compromise of the compactness of the computer architecture on the robot; therefore, having a controller at each ADoF will make the robot relatively bulky and will not respect the human form. The second problem is the existence of hybrid and parallel mechanisms that hold many DoFs see Figure 4.1. The hybrid and parallel mechanisms need synchronization between them in order to avoid mechanical problems and possible errors. Moreover, having one node that combines multi-joint controllers in one controller for each hybrid mechanism will give the possibility to integrate the Inverse Geometry Model (IGM) locally in the node. The solution of having a node at each ADoF can be imposed when choosing a commercial actuation driver, which is not our case. Therefore, we need to develop and unify the driver of each group of ADoF with the same mechanism as the robot. After examining the robot's mechanical design, it was determined that 14 control nodes were necessary to operate various mechanisms. These nodes were carefully placed on the robot's body with consideration for potential mechanical fixation points and minimizing the use of cables. The 14 nodes are separated into many categories, each category having a different number of channels where each channel is responsible for driving one joint, enabling the node to control the whole mechanism. Table 4.4 provides a list of the 14 nodes that correspond to the robot's mechanisms.

As a first prototype, a node board was developed as one channel node to control one hydraulic cylinder. The node is a highly compact unit, and it generally contains four functional elements:

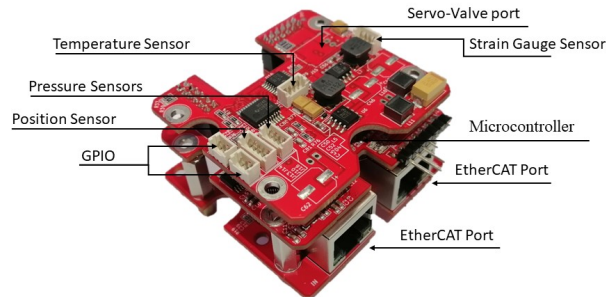
- Principal computational unit: this unit works as an electronic transceiver to read/write the frame of communicated data. Additional functions are required to be performed by this unit in order to realize distributed control.
- Communication interface: this functional part contains the needed electron-



Node	Quantity	Type	N° of ADoF
Ankle (sag/fro)	2 nodes	Type 2-ch	2x2
Ankle (Ver/Toe)	2 nodes	Type 2-ch	2x2
Knee	2 nodes	Type 1-ch	2x1
Hip	2 nodes	Type 3-ch	2x3
Torse	1 node	Type 4-ch	1x4
Shoulder	2 nodes	Type 4-ch	2x4
Forearm	2 nodes	Type 4-ch	2x4
Head	1 node	Type EM	1x15
Total	14 nodes	-	51

**Table 4.4:** Functional distribution of nodes

ics to exchange data with the master PC of HYDROiD via the communication protocol.



**Figure 4.6:** Node of the type 1-ch in three combined boards: computational unit, EtherCAT interface and peripheral interface.

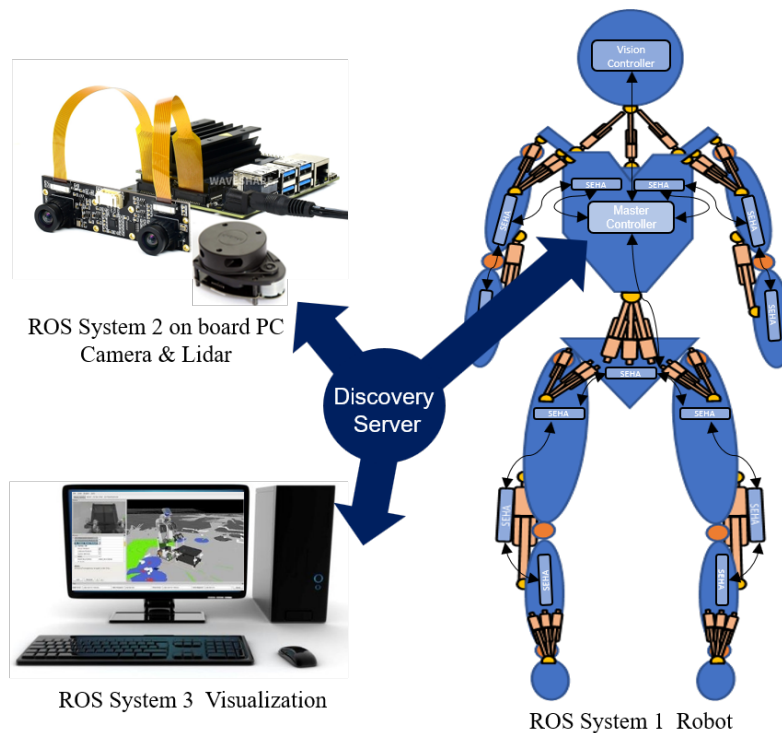
- Servovalve driver: a driver is required to manage the needed power to operate the servo valve. The driver will translate the control signal to a current signal to drive the servo valve.

- Sensor's interface: Each type of sensor in HYDROiD has certain precautions for the collection and the adaptation of signals to have meaningful data. A

conditioning circuit is needed for each sensor.

This node version consists of three boards combined in one block with dimensions 52x52x40 mm, shown in Figure 4.6. These boards divide the mentioned functionalities. The used microcontroller is the STM32H7, which possesses the ARM M7 architecture that suits the real-time application; the communication interface part consists of an integrated circuit LAN9252 with two standard RJ45 connectors, a servo-valve driver circuit, and the conditioning circuits of the associated sensors.

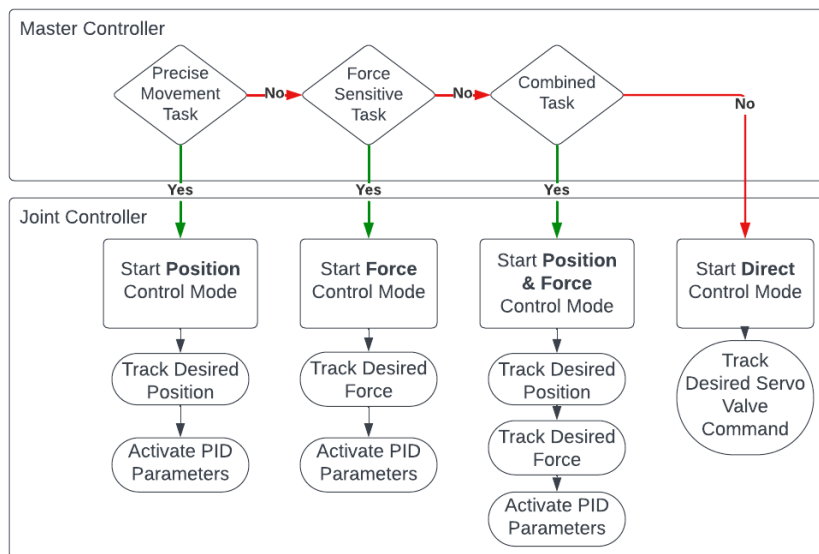
Figure 4.7 shows the final hardware control architecture, where 14 joint controllers are connected in daisy-chain topology to the master controller. Moreover, the master controller communicates with other subsystems via the discovery server provided by the ROS. This will facilitate expanding the application by integrating other systems via the ROS server.



**Figure 4.7:** Hardware Control Architecture, Joint Controllers' Distribution, ROS Discovery Server.

## 4.5 Control Architecture Software Development

The software architecture development aims to meet the required specifications in 4.3.2, especially implementing control in a flexible way, real-time capability, and ease of integration. The flexibility in our development is represented by the capability of the high-level controller to control the joint directly or switch between a set of pre-developed sensorimotor programs within the joint controllers. The master controller will choose which control mode will be activated, as shown in Figure 4.8, based on the robot's task; these control strategies enhance the robot's ability to perform tasks effectively in various real-world scenarios. Appendix B presents the detailed control mode selection development beside the data structure, type, and size on the EtherCAT bus.



**Figure 4.8:** Sensorimotor programs selection.

The distributed software over all the system levels was developed in the high and low levels of control as follows:

### 4.5.1 Master Controller Software

A Linux-based solution was chosen to facilitate the integration of various softwares. Among available Linux-based solutions, Ubuntu is a commonly used operating system in robotics and software development. The standard Ubuntu Kernel does not provide real-time guarantees, resulting in non-deterministic response and stability problems. Different approaches are available to patch its kernel and make it able to perform real-time tasks. The choice of real-time patching is a trade-off between deterministic performance and ease of use during development. A quantitative comparison [130] helps to decide how fast is fast enough in a robotics application for the actual development phase. As a result, the standard kernel was patched with the Preempt RT Kernel, which provides the necessary real-time capabilities for our application. Alongside the operating system, middleware is still needed to manage real-time tasks. Open ROBot Control Software (OROCOS) was chosen among many available middlewares. It has real-time capability and supports four C++ libraries: the Real-Time Toolkit (RTT), the Kinematics and Dynamics Library, the Bayesian Filtering Library, and the OROCOS Component Library. The RTT provides the infrastructure and the functionalities to build robotics applications, with emphasis on real-time, online interactive, and component-based applications. It facilitates the implementation of tasks and ensures that network communication through software ports is performed promptly and accurately. OROCOS has a bridge that facilitates the integration with ROS for non-real-time actions. ROS was used to enhance the integration level across a diverse range of hardware and software platforms. ROS is implementing applications like Simultaneous Localization and Mapping (SLAM). This accomplishes auto navigation, path planning, and collision avoidance for HYDROiD. Moreover, Graphic User Interface (GUI) was developed as a part of ROS; in other words, ROS was used for non-real-time tasks. The bridge between the ROS and the OROCOS is the *rtt - ros - comm* [131] package; it was integrated to provide an OROCOS RTT Service for publishing and subscribing (transmitting and receiving data) to ROS topics (data bus).

The interface between the EtherCAT bus and the application layer in the mas-

ter was developed as an OROCOS workspace to operate the communication in real time. It contains the Simple Open Master EtherCAT (SOME) library that we integrated into the workspace as another OROCOS package, which dominates the communication bus and manages the data transmitting and receiving, the *rtt – soem* [132] package used as a dependency for the conducted interface. In practice, each EtherCAT hardware needs a particular software driver inside the master library. SOEM provides some drivers for commonly used Beckhoff devices. For our application, a custom driver has been created for our custom EtherCAT slave interface.

#### 4.5.2 Joint Controller Software

Among the many Real-Time Operating Systems (RTOS) available for embedded systems, the three open-source options are FreeRTOS, Zephyr, and RTEMS. For our STM32H7 microcontrollers, we chose FreeRTOS due to its wide adoption, official STMicroelectronics support, small footprint, real-time capabilities, and open-source licensing that meets our requirements. The communication interface in the slave is the EtherCAT Slave Stack Code (SSC); it is an example source code in ANSI C supporting both the  $\mu$ C and the SPI interface; the code serves as a development basis for the implementation of EtherCAT in devices with its processor.

The flexibility of the proposed control architecture extends beyond application operation; it can also be utilized during the application's development process. This adaptability enables the seamless replacement of elements under development with validated ones, contributing to the validation of other components within the control architecture. The application uses the hydraulic cylinders as the main actuators in its actuation system. As mentioned previously, this actuator uses many sensors that should be validated separately by testing them with certificated sensors, validating the actuator controller, and then validating the mechanical functioning of the utilized actuator. Upon validating the controller with its associated sensors, they could be used in the identification process of the hydraulic cylinder, as discussed in the following section.

## 4.6 Conclusion

In conclusion, This chapter presented a comprehensive methodology and approach to developing the SEHA-based HYDROiD control system. Addressing crucial aspects of hardware and software development and the definition of key control architecture levels, this chapter laid the foundation for a versatile and adaptive control framework influenced by the application-based diverse control architectures insights gained from the State-of-the-Art chapter 2. The systematic methodology allows for the seamless integration of various elements at different control levels, ensuring independence and flexibility in the system. The approach facilitates the application's operation and proves valuable during the actuation system's developmental stages. The capability to replace developmental elements with validated counterparts enhances the efficiency of the validation process for other components within the control architecture. As a transition to the next chapter, "SEHA-Based HYDROiD Control Architecture Realization and Validation," the focus will shift toward the practical implementation and testing of the proposed solution elements. This next phase encompasses the design, realization, and rigorous testing and validation processes to bring the SEHA-based HYDROiD control architecture from concept to reality.

# Chapter 5

## SEHA-Based HYDROïD Control Architecture Realization and Validation

### Contents

---

<b>5.1</b>	<b>Introduction</b>	<b>104</b>
<b>5.2</b>	<b>Main Hydraulic Group (MGH) Tests and Validation</b>	<b>104</b>
5.2.1	Board Development	105
5.2.2	Testing SEHA Controller	107
5.2.3	ISO Test Control System	115
<b>5.3</b>	<b>Servo-Instrumented Cylinder Tests and Validation</b>	<b>119</b>
5.3.1	Board Development	119
5.3.2	SIC Controller Servo Valve Tests	120
5.3.3	SIC Controller Pressure Sensors Tests	124
5.3.4	SIC Controller Position Sensor Tests	125
<b>5.4</b>	<b>System Identification For Servo Instrumented Cylinder</b>	<b>127</b>
5.4.1	System Essential Dynamics	130
5.4.2	Experiment Setup and Data Collection	133

5.4.3	Model Structure Selection . . . . .	139
5.4.4	Model Identification . . . . .	139
5.4.5	Model Validation . . . . .	141
<b>5.5</b>	<b>SEHA Control Architecture tests and results . . . . .</b>	<b>143</b>
5.5.1	Master controller performance test . . . . .	143
5.5.2	Network performance test . . . . .	145
5.5.3	Walking gait Experiment . . . . .	146
<b>5.6</b>	<b>Conclusion . . . . .</b>	<b>147</b>

---

## 5.1 Introduction

The previous chapter discussed the proposed control architecture development methodology and the employed approaches in constructing humanoid robots utilizing SEHA as an actuation system. This chapter dives deeper into the details of the development, mainly for the substantial actuator modules, the main group hydraulic MGH, and the servo-instrumented cylinder SIC. These modules' realization and validation tests are separated into testing the mechanical functionality, the associated sensors and actuators, and the corresponding controllers. After validating the main functionalities, this chapter discusses a system identification process for the SIC module. Then, it investigates the test and validation process of the control architecture elements. Moreover, it demonstrates a robot walking gait performance test conducted using the proposed control architecture.

## 5.2 Main Hydraulic Group (MGH) Tests and Validation

The mechanical development of the MGH module was explored in 3.3.1. This section explores the development of the module controller and the required test that validates the overall module as a stand-alone product and as a main controller in case of adding some of the actuator modules.



### 5.2.1 Board Development

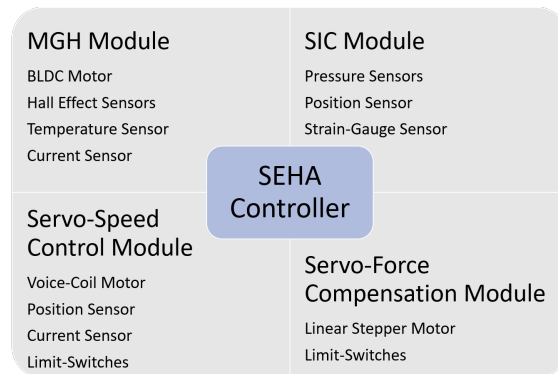
This board is the main controller of the SEHA actuator. Therefore, it should contain all modules' associated sensors and actuators. This controller combines the defined module's monitoring and controlling points in the Servo Electro-Hydraulic Actuator (SEHA) chapter 3. This combination facilitates adding or removing the involved modules in the robotic system based on its requirements. Figure 5.1 illustrates the involved inputs and outputs ports in the SEHA controller classified by their corresponding module.

Designing the board takes into account the necessity of General Purpose Input Output (GPIO) to make the board adaptable to a variety of common sensors that are utilized in robotic applications to handle the case of replacing one of the associated sensors or actuators. The controller provides the following GPIO and communication interface ports:

- Two ADC input channels with 16 bits resolution, 10kHz sampling rate, 0-5 VDC range, and  $\pm 0.25\%$  accuracy.
- One DAC output channel with 16 bits resolution, 10kHz sampling rate, 0-5 VDC range, and  $\pm 0.25\%$  accuracy.
- 3.3V and 5V power supplies to power multiple sensors or actuators in case of replacement or extension of the system.
- Serial Peripheral Interface (SPI) communication protocol port
- Inter-Integrated Circuit (I<sup>2</sup>C) communication protocol port

In order to limit the heat released during system operation and avoid the noise produced by the power electronic components, the hardware architecture is divided into four separate boards. These cards are linked to the assembly by available direct connectors or dedicated cabling. These boards are the following:

- **Board 1:** The microprocessor STM32H7 is the main MCU installed on the first board of the controller, and it has a parallel connection to the EtherCAT



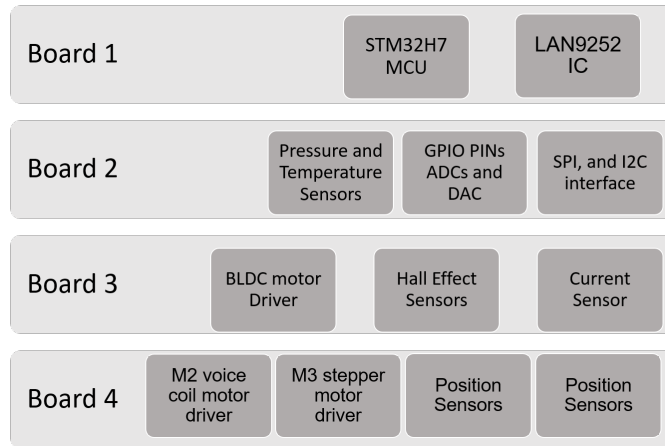
**Figure 5.1:** SEHA general controller combining the functionalities of the actuator modules.

LAN9252 integrated circuit on the same board. The RJ45 connection was installed on another board to minimize the size of the electronic system. The RJ45 connector type is IP68, suitable for outdoor environments, and ensures that dust and water will not affect the connection operation.

- **Board 2:** is the peripherals manager; it has conditioning circuits for the connected sensors with their connectors (two pressure sensors, two temperature sensors, and the GPIO PINs for two ADC, one DAC, SPI, and I2C interface).
- **Board 3:** This board is dedicated to BLDC motor control; it has all the required connections to integrate the BLDC motor, such as the hall effect sensors and the motor phases
- **Board 4:** This board controls the two M2 and M3 motors. the M2 is a voice coil motor AVM12-13-C48 with current feedback, and the M3 stepper motor 20DAMXXD2B. It contains the conditioning circuits of the two position sensors, AMS AS5171 and AMS AS5311, and their limit switches. In addition, two voltage regulators will provide the power supplies for the connected sensors, 3.3V and 5V with GND.

Operating the controller peripherals and components requires powering up the

controller with two power supplies 12 VDC power supply for the main MCU, EtherCAT, M2, and M3 motors, and 325VDC, 10A BLDC motor power supply.



**Figure 5.2:** SEHA-Controller boards and corresponding functional blocks

The SEHA controller prototype included a cooling system (heatsink and fan) to maintain its temperature below 70°C with full load. This division serves the modularity approach of the actuator; for instance, in the case of using only the MGH module without the servo-speed control module and the servo-force compensation module, it is sufficient to use the first three boards and avoid using the fourth one that is controlling these two modules and monitoring their feedback. Appendix A provides additional documents SEHA development, starting from defining the actuator’s sequence of operation and control mode selection beside the data structure, type, and size on the EtherCAT bus. In addition, it provides the performance and requirements of a graphic user interface that facilitates controlling the actuator.

### 5.2.2 Testing SEHA Controller

In this section, the focus is on validating the functionalities of the developed controller blocks shown in Figure 5.2

### Power supply 3.3 V AND 5V Test

Powering up multiple modules requires 3.3V and 5V power supplies. These power supplies are transformed from the 12V power supplied by the external system. The test procedure is to measure the 3.3V and 5V voltage outputs when the 12V supply changes between 10 and 15V. This validation test aims to check the variation of the 3.3V and 5V outputs in operation. The acceptable variation is  $\pm 3\%$  from the nominal voltage. Table 5.1 presents the results captured during the variation of the input power supply.

Power Supply (V)	Measured 5V pin	Error(%)	measured 3.3V pin	Error(%)
10Vdc	5.05V	1(%)	3.33V	1(%)
12Vdc	5.05V	1(%)	3.36V	2(%)
15Vdc	5.1V	2(%)	3.38V	2.4(%)

**Table 5.1:** Power supply 3.3 V and 5V Test

### Pressure sensor test

The selected Keller 4LC pressure sensor is a ratiometric sensor that provides a proportional output signal to a ratio of the measured input power supply. This type of sensor needs an ADC input for its analog output and another ADC to measure its input power supply. The controller provided two ADC channels with 16-bit resolution, 10kHz sampling rate, 10%-90% of the input VDC range, 2-300 bar reading range, and  $\pm 0.25\%$  accuracy. In addition, an additional ADC channel is added to measure the variation in the voltage supplied to the sensor (ADC3).

The validation process of the designed pressure transducer is to power up the sensor input ADC channel with a variety of voltages instead of connecting the sensor itself and operating the system to provide a range of pressure to validate the input ports. That will minimize the error source and give the advantage of more systematic testing procedures. Table 5.2 illustrates the results during the variation of the applied voltage on the mentioned ADC channels.

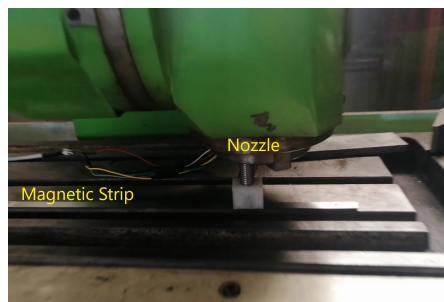
Applied (V)	ADC-1	Error(%)	ADC-2	Error(%)	ADC-3	Error(%)
0	0.012	0.24	0.009	0.18	0.005	0.1
0.5	0.505	0.1	0.509	0.18	0.5001	0.002
2	2.014	0.28	2.011	0.22	2.001	0.02
4.5	4.509	0.18	4.505	0.1	4.501	0.02
5	4.992	-0.16	4.988	-0.24	4.996	-0.08

**Table 5.2:** Pressure sensor input test results, the figure illustrates the captured signals with 16-bit resolution converted to an integer number and the corresponding error percentage.

### Position sensor test

The utilized position sensor is AMS AS5311, 5V/30mA, with incremental output configuration with a resolution of 1.95 $\mu$ m per step. An index pulse is generated once for every pole pair, corresponding to a 2.0mm interval. The sensor supports a traveling speed of up to 650mm/second in incremental mode.

The sensor readings should be verified by comparing them with an absolute sensor reading installed in the same condition or by installing the sensor on a mechanical system that can provide motion steps meeting the sensor resolution. The repeatability and stability of the sensor reading should also be validated by performing multiple tests to assess the repeatability of sensor readings and checking for any drift or instability in the readings over time.



**Figure 5.3:** Position Sensor Test Bench: The figure shows the position sensor installed on the CNC machine's nozzle and the magnetic strip installed on the movable plate.

This sensor was validated by installing the sensor on a CNC machine nozzle

and a 2cm strip magnet with a 2.0mm interval pole pair length on the machine's movable plate, as shown in Figure 5.3. The machine can provide linear displacement with  $10(\mu\text{m})$  resolution. Table 5.3 presents the implemented test; the first column is the setpoint displacement in  $(\mu\text{m})$ , the second is the sensor displacement reading, the third is the step displacement sensor reading, and the final is the calculated error in  $(\mu\text{m})$ . The first and the final displacements have the larger error because a side of the sensor chip went out of the strip edge. According to the shown results, it was noted that the sensor reading is reliable with an accuracy range  $\pm 50(\mu\text{m})$ , which is sufficient for the application.

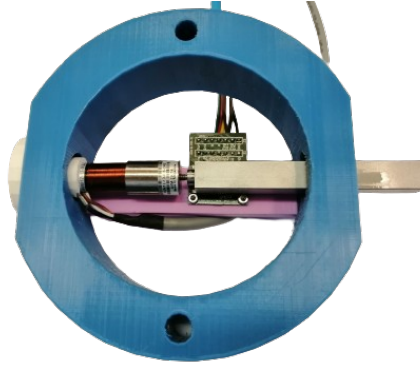
Displacement ( $\mu\text{m}$ )	Sensor reading ( $\mu\text{m}$ )	Sensor dis- placement ( $\mu\text{m}$ )	Error ( $\mu\text{m}$ )
0	0	0	0
1000	1041	1041	-41
2000	2041	1000	0
3000	3041	1000	0
4000	4043	1002	-2
5000	5046	1003	-3
6000	6046	1000	0
7000	7041	995	5
8000	8081	1040	-40
9000	9056	975	25
10000	9710	654	346

**Table 5.3:** Position sensor test results, the figure illustrates the captured displacement of a CNC machine, sensor measurement reading, and the corresponding error in  $(\mu\text{m})$ .

### M2 Voice coil motor test

Driving the voice coil motor needs to supply the current to make the coil move inside the motor. Directional control is required to move the coil forward or backward. The voltage supplied to the controller is 12 VDC. A PWM control solution was chosen to decrease the generated heat. However, stabilizing the current supplied to the motor is not a control priority, so the variation of this value during the

control phase is not part of the validation criteria (there will be no control loop for this parameter).



**Figure 5.4:** M2 test bench: The figure shows the voice coil motor AVM12-13-C48 installed on the motion axis with the position sensor circuit AMS AS5311.

The current supplied will be limited to prevent it from exceeding the limit accepted by the motor (3A max). The control command will be in the form (PWM, DIR) with the direction of movement DIR and the corresponding voltage level PWM. The current feedback is taken for monitoring. Position control is programmed with 1 mm accuracy (without dynamic or overshoot constraints because it is related to the mechanical test bench shown in Figure 5.4). The test was conducted by setting many desired position steps to control the motor movement and measuring this movement by the validated position sensor in the previous section. Table 5.4 illustrates the desired position setpoint in both directions and some large steps, position sensor measurement reading, and the corresponding error in (mm).

### M3 Stepper motor test

The A4988 integrated circuit driver controls the M3 linear motor. Similar to the M2 motor, two directions are controlled. To make the motor turn, we drive the PULSE variable, which defines the number of pulses to make the motor move. No position sensor is used to control the motor position; the motor may miss some requested moves (loss step), particularly when there is a load on the motor. The fastest change between two pulses is 100 us.

Direction	Setpoint(mm)	Measured Position(mm)	Error(mm)
Positive	1	0.517	0.483
	2	1.59	0.41
	3	2.6	0.4
	4	3.77	0.23
	5	4.79	0.21
Negative	4	4.3	-0.3
	3	3.27	-0.27
	2	2.25	-0.25
	1	1.25	-0.25
	0	0.24	-0.24
Step	0 -->5	5.14	-0.14
	5 -->1	1.19	-0.19
	1 -->4	4.17	-0.17
	4 -->0	0.15	-0.15

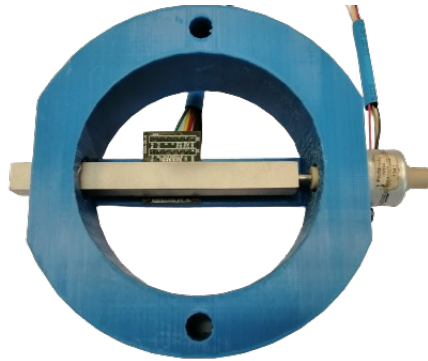
**Table 5.4:** M2 Voice coil motor test results, the figure illustrates the position setpoint, sensor measurement reading, and the corresponding error in (mm).

Validation tests were carried out under no-load conditions. Drive the motor with two parameters (Pulse and Dir) via the EtherCAT interface. See if the motor can change direction as required. A wire will be installed on the motor cover to check the motor’s displacement in an approximate way, as there is no position sensor installed to check the motor’s correct operation. The position control is programmed with an accuracy of 1 mm without dynamic or overshoot constraints because it is related to the mechanical test bench shown in Figure 5.5. Similar to the M2 voice coil motor test, the conducted test set many desired positions with an open control loop (no feedback) and measured the movement using the same linear position sensor installed on the stepper motor test bench. Table 5.5 illustrates the desired position setpoints, position sensor measurement reading, and the corresponding error in (mm).

### M1 BLDC motor test

Driving the BLDC motor is critical and sensitive due to its high-rated voltage (325VDC). The motor is controlled in both direction with a wide speed range





**Figure 5.5:** M3 test bench: The figure shows the stepper motor 20DAMXXD2B installed on the motion axis with the position sensor circuit AMS AS5311.

Setpoint(mm)	Board Position(mm)	Error(mm)
1	0.8	0.2
2	1.89	0.11
3	2.84	0.16
4	3.91	0.09
5	4.9	0.1
6	5.95	0.05
7	6.94	0.06
8	7.97	0.03
9	8.95	0.05

**Table 5.5:** M3 Stepper motor test results, the figure illustrates the motor position setpoint, position sensor measurement reading, and the corresponding error in (mm).

( $\pm 5000$ ). The driver board contains an MCU specialized in driving BLDC motors. The communication between the main MCU and the driver is established through a UART connection that supports BLDC motor control data exchange at a 5 ms update rate. The discussion between these two boards has the BLDC motor control setpoints, feedback, and emergency parameters.

This test aims to validate the functionality of driving the BLDC motor accurately and efficiently by the developed controller. The driving speed accuracy was validated by operating the motor at the desired speeds and measuring it with a certified tachometer. The controller and motor temperature was measured during the motor operation to ensure it was not reaching the allowed limit for robotic application.

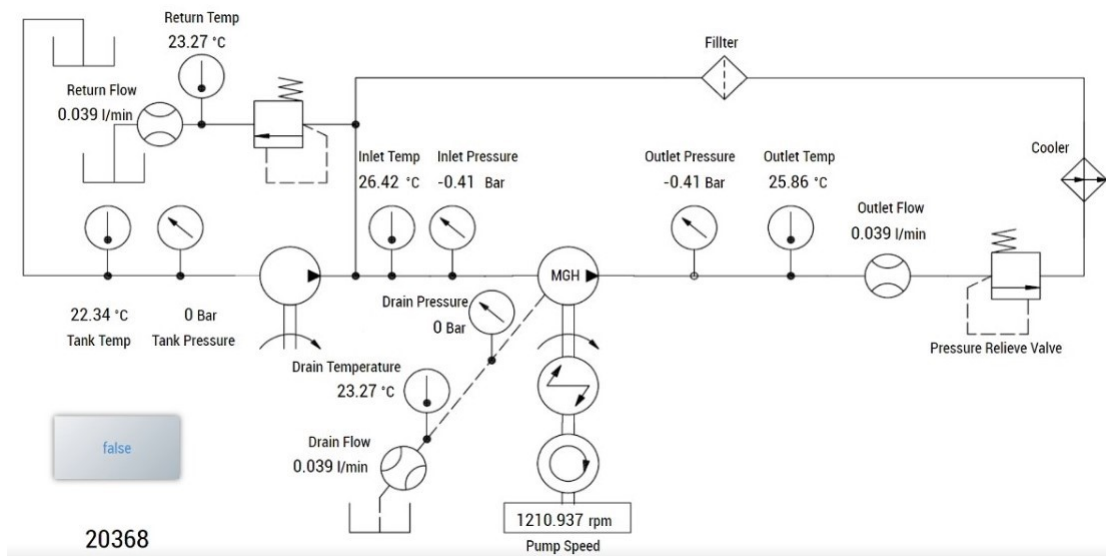
Desired Speed [RPM]	Tachometer reading [RPM]	Motor Temperature [°C]	Driver Temperature [°C]	Speed Error (RPM)	Error (%)
-1000	1010	33.6	29	10	0.2
-1500	1513	34.5	29.4	13	0.26
-2000	2018	34.7	29.6	18	0.36
-3000	3025	35	30	25	0.5
-4000	4033	35	30.6	33	0.66
-5000	5045	36	34.7	45	0.9
1000	1010	36.9	35	10	0.2
2000	2018	37.2	35.2	18	0.36
3000	3026	37.6	35.3	26	0.52
4000	4035	38	36	35	0.7
5000	5044	38.2	36.4	44	0.88

**Table 5.6:** M1 BLDC motor test results, the figure illustrates the motor speed setpoint, Tachometer measurement reading, controller and motor temperature, and the corresponding error in (RPM) and (%).

Table 5.6 illustrates the motor speed setpoint, Tachometer measurement reading, controller and motor temperature, and the corresponding error in (rpm)

### 5.2.3 ISO Test Control System

The proposed control architecture was utilized to test and evaluate the actuator components' performance during development. The MGH was evaluated by a performance test conducted according to the ISO 4409:2019 [3], (Hydraulic fluid power - Positive-displacement pumps, motors and integral transmissions - Methods of testing and presenting basic steady state performance). The standard provides hydraulic circuit diagrams and methodologies for testing hydraulic pumps as shown in 5.6.



**Figure 5.6:** ISO hydraulic circuit diagram

### ISO Control system requirements

The implemented control system for evaluating the pump under steady-state conditions to determine its performance characteristics. The system can be used with open or closed circuits mentioned in the standard document. The standard categorizes results into three measurement accuracy classes, A, B, and C, based on the allowable systematic errors of flow, torque, pressure, and temperature measuring instruments as shown in Table 5.7.

Parameter/Class	A	B	C
Rotational frequency, %	±0,5	±1,0	±2,0
Torque, %	±0,5	±1,0	±2,0
Volume flow rate, %	±0,5	±1,5	±2,5
Pressure, Pa ( $Pe < 2 \times 10^5$ Pa) <i>a</i>	$\pm 1 \times 10^3$	$\pm 3 \times 10^3$	$\pm 5 \times 10^3$
Pressure, % ( $Pe \geq 2 \times 10^5$ Pa)	±0,5	±1,5	±2,5

**Table 5.7:** Permissible variation for classes of measurement accuracy: The permissible variations listed in this table concern deviation of the indicated instrument reading and do not refer to limits of error of the instrument reading. These variations are used as an indicator of steady state and are also used where graphical results are presented for a parameter of fixed values. The actual indicated value should be used in any subsequent calculation of power, efficiency or power losses. *a* 1 Pa = 1 N/m<sup>2</sup>. [3]

According to the standard, each set of readings taken for a controlled value of a selected parameter shall be recorded only where the indicated value of the controlled parameter is within the limits shown in Table 5.7. If multiple readings of a variable are recorded, the mean values shall be documented while the controlled parameter is within the operating limits. The maximum suggested time period to acquire each reading is 10 seconds at a minimum data acquisition rate of 1000 Hz. Such readings should include zero displacement and idle operating conditions.

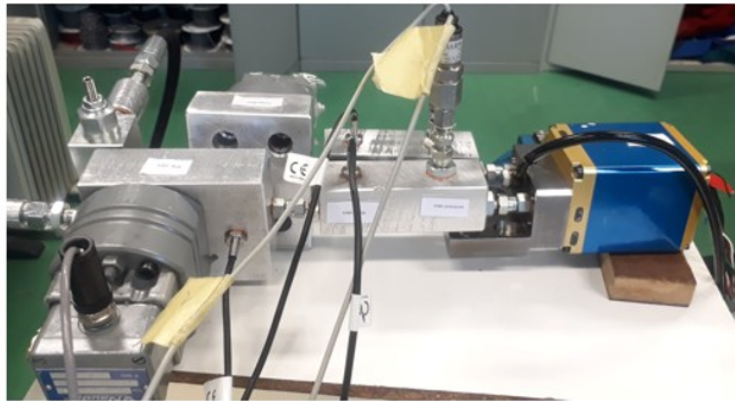
### Test bench components selection

Figure 5.6 illustrates the required component in the test bench to determine the performance curve characteristics by capturing the pressure, temperature, and flow values at various rotational speeds of the pump. Table 5.8 defines the quantities and characteristics of the utilized sensors and components that meet the ISO standard requirements. These components are installed on i) four manifolds combine the temperature and the pressure sensors, ii) a manifold that combines a flow meter and temperature sensors, iii) two manifolds for the flow sensor alone. Figure 5.7

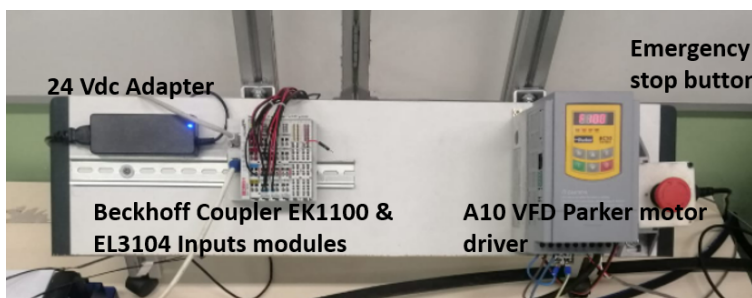
## 5.2. Main Hydraulic Group (MGH) Tests and Validation

Component	Qty	Reference	Range	Output Signal
Temperature sensor	5	NTC 10K	-40 to 150 °C	0 to 10 K $\Omega$
Pressure sensor	4	Keller 4LC	2.5 - 300 Bar	0.5 to 4.5 Vdc
Flow meter	3	VES VS-2	0.2 - 18 L/min	5000 Pulses/Litter
Pressure relief valve	2	Inaltis-VMDR 40	40 - 350 Bar	-
Driver	1	PARKER A-10	0 - 5000 rpm	0 - 5 Vdc
Cooler	1	SCM A45-1 EAP	50-80 °C	Dry contact

**Table 5.8:** Test bench components: the table presents the used sensors and components in the ISO test bench with their quantities and characteristics.



**Figure 5.7:** ISO-4409 Implemented Hydraulic circuit

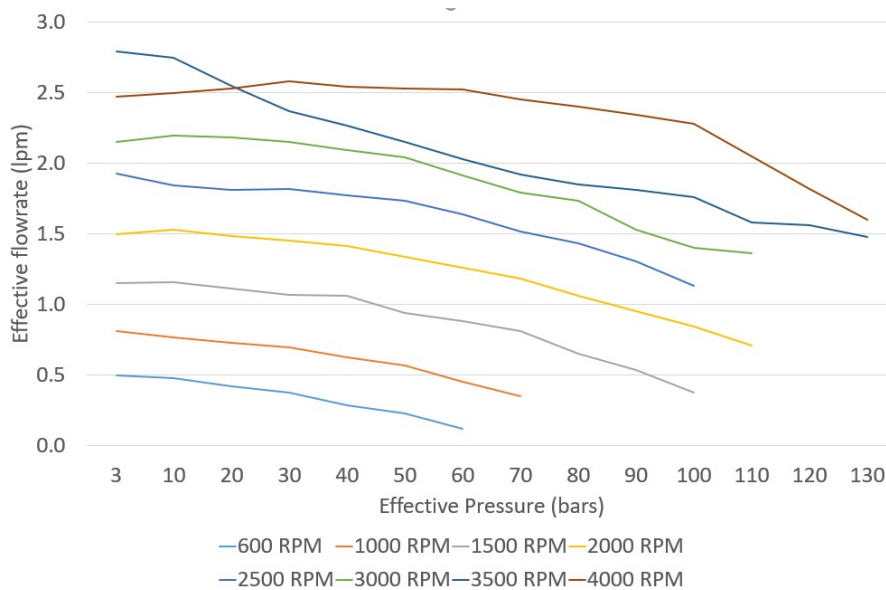


**Figure 5.8:** ISO-test bench Beckhoff controller and Parker A10 motor driver

The test bench sensors are connected to a Beckhoff coupler EK1100 with analog input modules EL3104 shown in Figure 5.8. Beckhoff coupler EK1100 is an EtherCAT slave; it was connected to a PC with Windows 10 OS, and TwinCAT Beckhoff software was installed to perform the data acquisition in real time. TwinCAT software acts as a communication master and a master controller. It dominates the data bus and allows the user to program a master loop control for monitoring all the input variables and controlling all the output variables on the bus regardless of which slave sends or receives these data.

### ISO-4409 test results

At the steady state, the variation of mean indicated values for measured parameters aligns with the Class A specifications of permissible variation outlined in ISO 4409:2019. This class requires a stringent allowable variation of  $\pm 0.5\%$  for volume flow rate and pressure, making it the category with the most stringent acceptable deviation from the mean values of measured parameters.



**Figure 5.9:** The effective flow rates at different rotational speeds and pressures.

Figure 5.9 shows the flow rate against the measured pressure at a constant speed. Notably, as the applied pressure increases, the flow rate experiences a

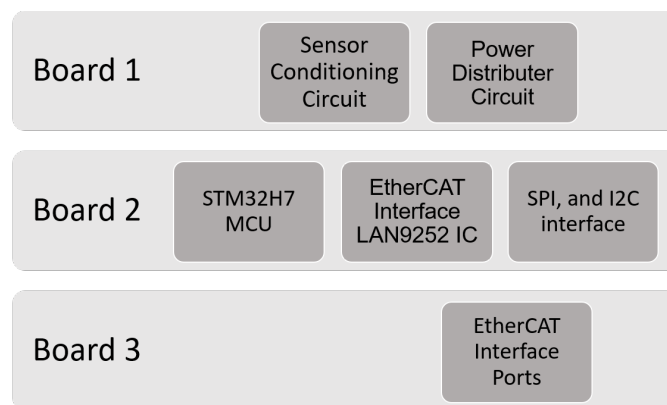
decline. Beyond 3000 RPM, the pump's curve deviates from the observed pattern at lower speeds, indicating the presence of an emerging defect in the pump.

### 5.3 Servo-Instrumented Cylinder Tests and Validation

The mechanical development of the SIC module was presented in 3.3.5, and the functional blocks were explored in the 4.4.3. This section discusses the module controller development and the test procedures.

#### 5.3.1 Board Development

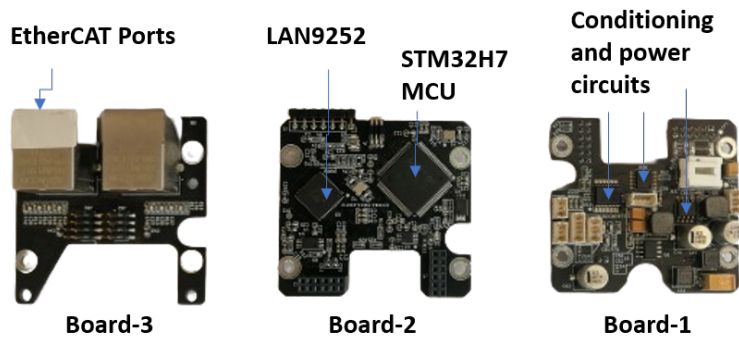
The SIC module integrates a servo valve controlling the oil flow through the piston chambers, a linear position sensor measuring the rod motion, two pressure sensors measuring the piston chambers' pressures, a bidirectional force sensor measuring the applied force on the rod, and a temperature sensor measuring the oil temperature. The controller contains conditioning circuits for the associated sensors, a servo valve driver, a power circuit, an EtherCAT communication interface IC and ports, and a microcontroller STM32H7. Similarly to the SEHA controller development, the SIC controller was divided into three separate boards. The functional blocks were distributed in these boards as follows:



**Figure 5.10:** SIC-Controller functional blocks distributed on three boards

- **Board 1:** is the peripherals manager; it has conditioning circuits for the connected sensors with their connectors (two pressure Keller 4LC, a temperature PT1000, position AMS AS5311, full bridge strain gauges sensors).
- **Board 2:** The microprocessor STM32H7 is the main MCU installed on the first board of the controller, and it has a parallel connection to the EtherCAT LAN9252 integrated circuit on the same board. In addition, the GPIO PINs for two ADC, one DAC, SPI, and I2C interface are provided by this board.
- **Board 3:** Has the two RJ45 connectors type IP68, suitable for outdoor environments, and ensures that dust and water will not affect the connection operation.

Figure 5.11 illustrates the manufactured controller board.



**Figure 5.11:** SIC Manufactured controller boards.

The test described in this section is divided into **seven** tests examining the performance of the different parts of the SIC controller board, including the servo valve driving circuit, the power circuit, the EtherCAT interface circuit, and the conditioning circuit of the pressure sensors, the temperature sensor, the position sensor, and the force.

### 5.3.2 SIC Controller Servo Valve Tests

This section is dedicated to examining the driving circuit of the servo valve. Current measurement is accomplished by assessing the voltage drop across the mea-



surement resistance, and this current measurement is visually displayed on the oscilloscope. Simultaneously, the hydraulic flow rate is gauged using a flow rate meter, with the corresponding flow signal (Vflow) showcased on the oscilloscope. An ammeter could be employed instead of Rsh to measure the current directly instead of assessing the measurement resistance. A test point (TP7) is also available for measuring a voltage corresponding to the current.

### Servo Valve Static Error Test

The test has been initiated by specifying a constant desired value for the current, with the rated current ranging between  $\pm 10$ mA. Subsequently, the voltage  $V_o$  on Rsh (or the DC current on an ammeter) is measured and monitored using an accurate meter. For values of  $-10$ mA and  $-5$ mA, measurement is conducted via TwinCAT, as the ammeter is incapable. The objective is to ensure that the static error (real current/desired current) remains below 0.5%FS (0.1mA). To achieve this, the test is repeated on various samples of the ordered value, encompassing  $\pm 10\%$ ,  $\pm 25\%$ ,  $\pm 50\%$ ,  $\pm 75\%$ , and  $\pm 100\%$  of the rated current. Table 5.9 presents the static error of captured current measurements with variation in the desired current.

Desired current	Measured current min(mA)	Measured current max(mA)	average current (mA)	Static error
-10	-10.02	-10.06	-10.04	-0.04
-5	-5.01	-5.04	-5.025	-0.025
-1	-0.99	-1.01	-1	0
0	0.08 $\mu$ A	20 $\mu$ A	0.0104	0.0104
+1	1.01	1.07	1.04	0.04
+5	5.03	5.13	5.08	0.08
+10	10.00	10.09	10.045	0.045

**Table 5.9:** SIC Controller: Servo Valve Output Current Measurements and Static Error Results

### Servo Valve Settling time Test

Applying an entry square signal with an amplitude of  $\pm 100\%FS$  as the desired value to the servo valve driver has been executed. The value is set using TwinCAT entry fields, ensuring a sufficiently high frequency (e.g., 10Hz) to capture all transient periods. Subsequently, the voltage  $V_o$  on Rsh is measured and monitored on the oscilloscope, with note that test point TP7 on the card is utilized for voltage measurement corresponding to the current. The observation of step response characteristics is mandated, and particular attention is given to measuring the settling time at 95%, which should be less than 0.5 ms in the worst-case scenario. Table 5.10 presents the settling time at various current desired values.

Desired value (mA)	Measured voltage (mV)	measured current (mA)	Settling time 95% (ms)	Overshoot (%)
-10 to +10	736	x	2.4	0 %
-5 to +5	246	x	1.46	0 %
-1 to +1	24	x	0.365	0 %
-5 to 0	204	x	1.45	0 %
-10 to 0	348	x	1.30	0 %
0 to +5	276	x	1.18	0 %
0 to +10	336	x	1.68	0 %
-10 to +10	736	x	2.22	0 %

**Table 5.10:** SIC Controller: Servo Valve Output Settling Time and Current Overshoot Measurements Results.

### Servo Valve Current Overshoot Test

Examining the current overshoot of the servo valve driver circuit involves applying an entry square signal with an amplitude of  $\pm 100\%FS$  as the desired value. Subsequently, observation of step response characteristics is imperative. If a maximum overshoot is present, it must be limited to less than  $10\%FS$ . The test readings and results are presented in Table 5.10 at different current entry square signals as current desired values.

**Electric Frequency Analysis (cut off & phase margin)**

Applying an entry sinewave signal with an amplitude of  $\pm 100\%$ FS as the desired value to the servo valve driver has been executed, with the value being set using TwinCAT entry fields. The voltage  $V_o$ , measured at test point TP7, is monitored on the oscilloscope, and simultaneous monitoring of the desired signal is attempted (not successfully). The frequency is systematically varied from 1 Hz up to 1 kHz. Subsequent measurements include assessing the amplitude of the output signal and determining the phase shift (unavailable). Throughout all frequencies, the amplitude is expected to exceed 50% of the static signal's amplitude (at 1 Hz), and the phase shift should be less than  $90^\circ$  of the static signal (at 1 Hz). Table 5.11 shows the measured voltage amplitudes at the mentioned frequency range and the output signal gain.

Frequency (Hz)	Voltage Amplitude (mV)	Gain (%)
1	813	100%
10	813	100%
50	766	94%
100	766	94%
150	746	92%
200	716	88%
300	593	73%
400	530	65%
500	436	54%
600	366	45%
1000	233	29%

**Table 5.11:** SIC Controller: Servo Valve Cut-Off Frequency Test Results.

**Mechanic Frequency Analysis (cut off & phase margin)**

This test was conducted by applying an entry sinewave signal with an amplitude of  $\pm 100\%$ FS as the desired value to the servo valve driver has been implemented, with the value set via TwinCAT entry fields. The flow rate meter (Vflow) signal

is measured and observed on the oscilloscope, and the desired signal is monitored. The frequency of the entry sine wave in TwinCAT is systematically adjusted from 1 Hz up to 200 Hz. Measurements include determining the amplitude of the Vflow signal and evaluating the phase shift of the Vflow signal. Throughout all frequencies, the amplitude is expected to surpass 50% of the amplitude of the static signal (at 1 Hz), and the phase shift should not exceed 90° of the static signal (at 1 Hz). Table 5.12 shows the measured flow rates through the servo valve at the mentioned frequency range and the output signal gain.

Sin-wave Frequency	Servo Valve Flow Rate (lpm)	Gain (%)
1	7.017	100
2	6.89	98.3
10	6.66	95
20	7.06	100.6
30	6.79	96.8
50	6.54	93.3
80	6.31	90
100	5.71	81.4
200		50

**Table 5.12:** SIC Controller: Servo Valve Cut-Off Flow Test Results.

### 5.3.3 SIC Controller Pressure Sensors Tests

The developed controller and a referential data acquisition system, Dspace DS1104, measured the pressure sensor in this test. The integrated pressure limiting valve controlled the hydraulic pump pressure setpoint, with pressure variation between 10 and 210 bar according to the specified range. Following this, a waiting period ensues until the pressure stabilizes. Subsequently, the output is measured using the SIC through the TwinCAT software, while the pressure referential measurement system is employed for a parallel measurement. The voltage signal ( $V_p$ ) and the supply signal ( $V_{cc}$ ) are measured using dSpace for the referential measurement. The real pressure is defined in the datasheet of 4LC by the equation Pressure =

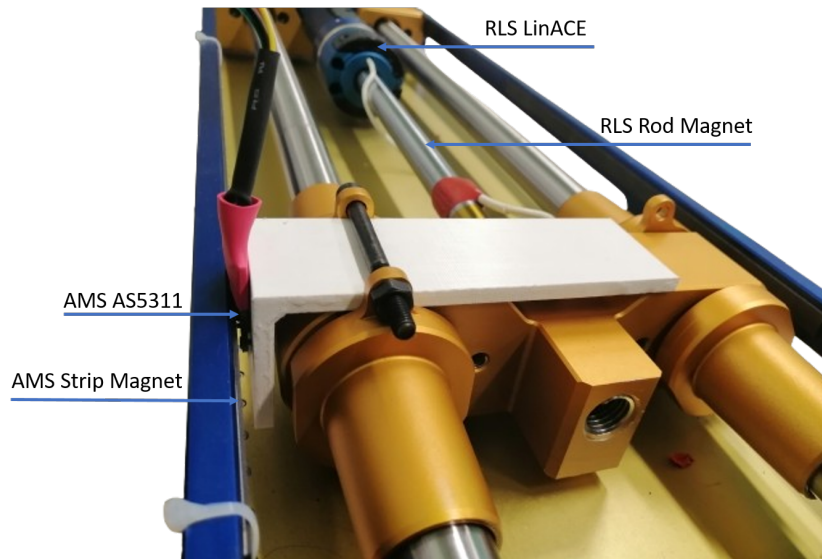
$200 * ((V_p - 0.1V_{cc}) / (0.8 V_{cc}))$  since the sensor is ratiometric, and it is related to the ratio  $V_p/V_{cc}$ . The two readings are compared, and the test is repeated while varying the environmental conditions of the card. The acceptable difference between the readings should be at most 0.25% FS. The repeatability criterion for the reading is set at less than 0.1%FS. Table 5.2 illustrates the pressure sensor's measured power supply and output signals by two different controllers and calculates the reading error according to the sensor specification document.

Pump Pressure (bar)	SIC $V_p$ (Volt)	SIC $V_c$ (Volt)	$V_p$ dSpace (Volt)	$V_c$ dSpace (Volt)	SIC Calculated Pressure (bar)	dSpace Calculated Pressure (bar)	Error $100*(P_{dsp} - P_{sic})/300$ (%)
start	513	4975	501	4961	3.16	2.36	0.26
10	638	4975	635	4970	12.519	12.34	0.058
20	780	4975	770	4970	23.15	22.46	0.23
50	1182	4975	1180	4970	53.25	53.19	0.02
100	1850	4975	1848	4970	103.26	103.25	0.003
150	2500	4975	2492	4970	151.93	151.52	0.137
180	2905	4975	2981	4970	182.26	188.17	-1.97

**Table 5.13:** SIC Controller: Pressure Sensors Test Results.

### 5.3.4 SIC Controller Position Sensor Tests

The SEHA controller position sensor was installed on a CNC machine nozzle to set the displacement and measure the position. The position sensor test in the SIC controller was conducted by comparing the AMS AS5311 reading with an absolute linear cylindrical encoder RLS LinACE customized for the SIC with  $0.5 \mu\text{m}$  resolution and  $\pm 5 \mu\text{m}$  accuracy [133]. The RLS sensor integrated into the cylinder cap and its magnet in the cylinder rod serves as the reference for the measurement as shown in Figure 5.12. The figure shows the cylinder installed on a development kit, the AMS AS5311 linear position sensor with its strip magnet, and the RLS LinACE sensor with its rod magnet.



**Figure 5.12:** The SIC installed on development kit with both position sensors

The position is set to different points for testing the reading accuracy and resolution. The output is measured using the SIC controller through the Twin-CAT software. Simultaneously, the position is measured using the RLS. The two readings are then compared. The test is iterated while varying the environmental conditions of the card, with a stipulation that the difference should not exceed  $20 \mu\text{m}$ . Additionally, the repeatability criterion for the reading is set at less than  $20 \mu\text{m}$ . The measurement resolution is determined with a sensitivity set to  $20 \mu\text{m}$ . Table 5.14 shows the captured readings of both position sensors at different steps.

The previous tests offer a comprehensive insight into the performance of each module controller and its functional blocks. For effective testing of the developed sensorimotor programs, a profound understanding of the system dynamics is crucial to ascertain the actuator's capability to attain the desired values. The subsequent section will explore system identification for the primary actuator in the HYDROiD robot.

RLS Reading ( $\mu\text{m}$ )	SIC Reading ( $\mu\text{m}$ )	RLS real Distance ( $\mu\text{m}$ )	SIC real Distance $\mu\text{m}$	Error (mm)
874230	874230	0	0	0
1204420	1194850	16509.5	16031	0.4785
1668250	1691450	39701	40861	-1.16
2404500	2422250	76513.5	77401	-0.8875
2731050	2807850	92841	96681	-3.84
3873330	3944850	149955	153531	-3.576
4863970	4967250	199487	204651	-5.164
873690	874050	-27	-9	-0.018
2566250	2611850	84601	86881	-2.28
2507830	2524850	81680	82531	-0.851

**Table 5.14:** SIC Controller: Position Sensor Test Results.

## 5.4 System Identification For Servo Instrumented Cylinder

System identification concerns developing a dynamic system model using data rather than physics principles to identify the essential dynamics either responsible for or resulting in the observed system behavior.

These models are generally simplified representations of the real system. In fact, developing a model that mimics the essential features of the target system and then excluding all other aspects will facilitate the system model-based design processes.

These processes include: i) Controller design, where many tools can assist in finding the optimal solution if there is a mathematical representation of the system, especially a linear representation. ii) Improving estimation state by combining the predictions from simple models with sensor measurements. iii) Ensuring system performance or safety by using models to perform formal analysis. iv) Replacing physical systems and testing environments by simulating them. In addition, we can reduce these models to even more simplified versions to speed up these simulations.

However, The effective model should sufficiently capture the system's essential features. These features are determined by the purpose of the model development and a certain amount of understanding of the system. For example, in

the studied system (Electro-Hydraulic Actuators), to understand how much the servo-valve should be opened for a particular cylinder velocity, the system's model should consider the dynamics that play a significant part in determining it. These features include i) the cylinder geometric parameters, ii) the servo-valve characteristics, iii) the hydraulic system parameters such as the oil pressure and flow rate. Nevertheless, friction, manufacturing defects, oil characteristics, environment parameters, and vibration through the internal structure are not essential features. For example, even though vibration plays a vital role in material fatigue or sensor disturbances, it does not affect the system dynamics directly.

In general, system models comprise two main parts: structure and parameters. The identification process is mainly about determining the most suitable structure and optimal parameters that accurately capture the system's dynamics. There is more than one way to structure a model. A simple system with known physical parameters could be modeled with an ordinary differential equation. But with unknown physical parameters for the same system, a more general second-order differential equation could be used like the equation 5.1 with the parameters A, B, C, and D. Even though these two models can represent the same dynamics, they are represented slightly differently.

$$f = Ax'' + Bx' + Cx + D \quad (5.1)$$

Selecting a model structure is not limited to just differential equations. Other structures could be selected, like process models with time delay, non-parametric models like frequency response models, or even more abstract, like neural networks. Irrespective of the structure, the model would still have to capture the essential underlying dynamics of the real system to be useful.

There are many ways to approach system identification for linear, nonlinear, and recursive online system identification. Creating a model that can capture these dynamics depends on the knowledge level of the real system. If all of the essential features and how they contribute to its dynamic behavior are known, then a white-box approach could be adopted to create a model. The model is written



directly in this approach, using first principles and physics.

The black box method is on the opposite side of the spectrum; the essential dynamics are unknown, and the equations cannot be written directly. Therefore, the system behavior could be predicted by exciting the system with input signals, which affect the output. Assuming there is an ability to excite all the system's essential dynamics, These data could be used to learn a mathematical model that mimics the observed I/O relationship. Still, there is a need to choose a model structure representing the observed behavior and then fit that model to the data by selecting the best parameters.

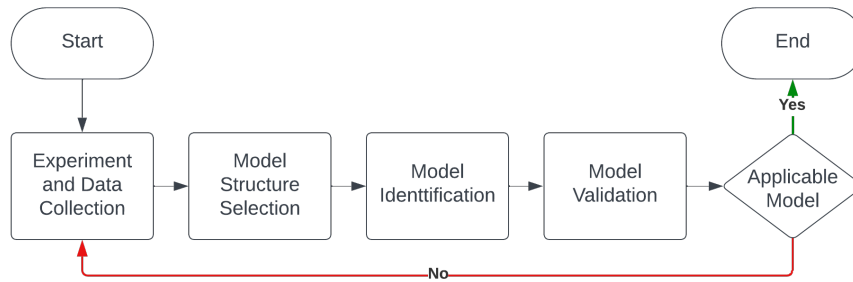
Noting that a substantial difference exists between modeling the system dynamics and curve fitting.

Curve fitting to predict the system output in the future by fitting a curve to the output data and extrapolating that curve out into the future. The curve fitting is based on an equation with structure and parameters. It does not describe the underlying mechanisms that created the data; it just describes the particular data sequence. If the system were initialized with a different state or given different input signals, the curve fitting would not give any future prediction.

Whereas the system identification gives an advantage of the correlation between the points in the data to fit a dynamic model. In dynamic systems, the output is a function of the inputs at a specific moment added to previous system values. The final model predicts the data from different starting conditions and with different inputs. This is a lot more powerful than just curve fitting because in most cases, there are multiple inputs and outputs and many tunable parameters to fit the model with a numerical method that adjusts the parameters automatically to minimize a cost function.

Validating the identified model should ensure that the output behaves the same, not only for a particular input but also under a different set of inputs. As a validation process, the step input will be replaced with a series of different step inputs to validate the model. If the real system and the model were run with these inputs, the results could be checked to see if they would match the identified model.

If the results were not met, more data should be gathered, the model structure should be changed, and then the identification process should be repeated. This trial-and-error process continues until the results are met and end up with a viable identified model.



**Figure 5.13:** System Identification Workflow.

For concluding the system identification workflow, and since system identification is a data-driven method, Figure 5.13 simplifies the process with the main four steps as follows

- First, set up an experiment or test to collect that data from the real system.
- Then, choose a model structure to fit into the collected data.
- Then, identify that model by actually fitting it to the data.
- Finally, validate the model and observe its performance. To establish system identification on a hydraulic cylinder operated by a servo valve, there is a need to understand the system’s essential dynamics as motioned before. The cylinder geometric parameters are constant parameters, whereas the servo-valve characteristics and the hydraulic parameters, like the oil pressure and flow rate are not.

#### 5.4.1 System Essential Dynamics

The used servo-valve is nozzle-flapper Moog.30 equivalent. It works at a rated input current from -10 to 10 mA that controls a 12 lpm flow rate through the valve control at an operational pressure of 210 bar. Figure 5.14 shows the relationship

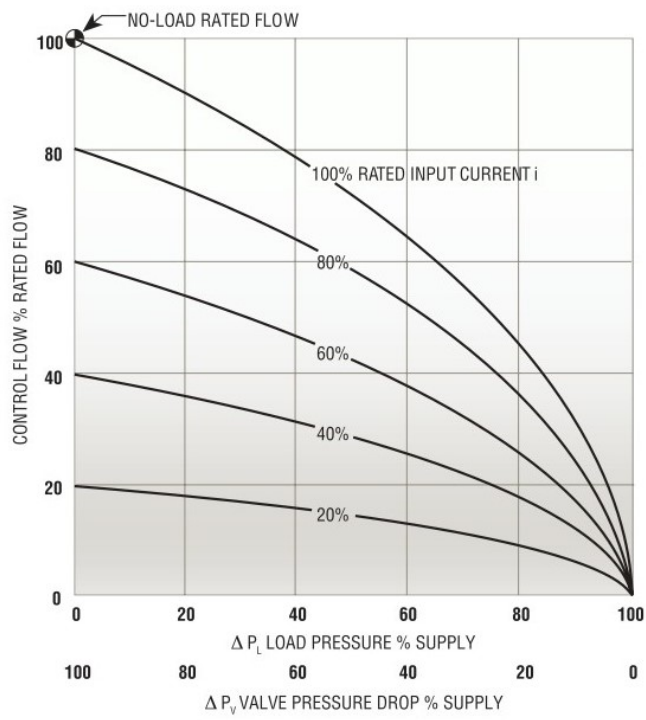


Figure 5.14: Moog30 Flow-Load Characteristics

between the load pressure and the flow rate through the servo valve at different rated input currents.

The servo valve dynamics behavior is represented by a model's structure of second-order differential equation, as shown in Figure. 5.15, even when using another commercialized servo-valve that could represent the model structure.

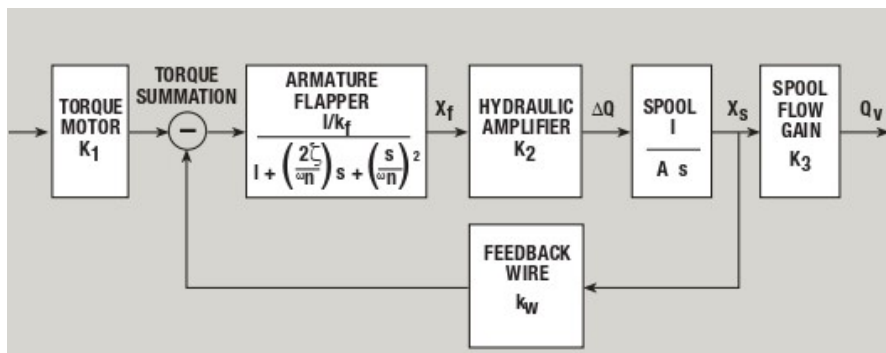


Figure 5.15: Moog.30 Servo Valve Internal Dynamics

This model captures the servo valve's essential dynamics but cannot capture the overall system dynamics or non-linear dynamics, and it does not capture other factors like disturbances. However, it has the essential dynamics to predict the general motion of the cylinder with some level of accuracy. This model can predict the cylinder velocity given an input current, where "I" is the input current, and "Q" is the output flow. The cylinder velocity could be calculated easily using the following linear equation 5.2.

$$PistonVelocity(V) = FlowRate(Q)/PistonArea(A) \tag{5.2}$$

However, we are not approaching the system with a total lack of information; we have some amount of knowledge that could help to choose the proper structure and develop the model. The chosen second-order model would be sufficient to capture the essential dynamics and then use data and system identification to estimate the parameters of this model. This is called a gray box method, which combines the transfer function of the servo-valve to set up the initial problem and

then uses data to learn the remaining portion of the model structure or parameter set.

Therefore, and according to the previous section 5.4, the first step is to set up an experiment to collect data for identifying the system.

### 5.4.2 Experiment Setup and Data Collection

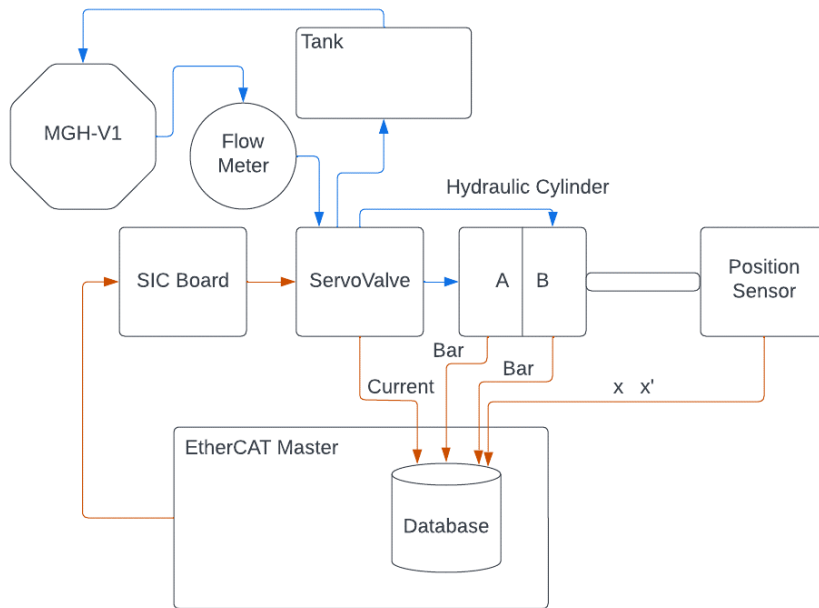
In this system, with some unknown dynamics, a step input is used to excite the system, and then the response is recorded to find a model that can capture the remaining dynamics. A mini group hydraulic (MGH) is connected to a flow meter, delivering the hydraulic power to the hydraulic cylinder operated by the developed SIC board. The used peripherals are the following:

- Position Sensor AMS AS-5311 liner installed on the rod's end
- Two Pressure sensors Keller 4LC measuring the pressure of the piston chambers.
- Current sensor integrated into the SIC board to measure the servo valve current
- Flow meter on the SIC hydraulic input
- Servo Valve

Figure 5.16, show the test circuits. The used AMS position sensor is an absolute sensor, but in the incremental encoder mode limit switches needed to set the initial position by resetting the pulses counter. These experiments was conducted with no limit switches, that let to adjust the initial point before starting the reading to avoid resetting the counter to the maximum value because of the workbench vibration or even the increased displacement at high pressure. The initial position was adjusted, with 50 micrometers.

#### **Setting up the flow rate:**

To ensure capturing all the essential dynamics, the test should be performed at the peak values of the effective parameters like the flow rate. This test was conducted



**Figure 5.16:** Hydraulic and Electrical Test Circuit.

at a constant flow rate; flow measurements were executed at different pump speeds with and without load (with no load: the MGH was connected to the flow meter, then the flow meter’s outlet to the tank. With load: The SIC was connected to the flow meter’s outlet, and the servo valve was operated at maximum range +10 mA), which gives accurate readings and ensures the MGH flow rate. Table. 5.15 show these measurements.

RPM	Flow rate (lpm) No load	Flow rate (lpm) at load
1250	0.85	0.73
1600	1	0.85
3600	1.8	1.66

**Table 5.15:** MGH Speeds with Corresponding Flow Rates

The test was performed at a 3600 rpm speed with 1.8 lpm, the corresponding rod max velocity calculated according to 5.2, For 1.66 lpm flow rate with the used SIC 18-30 mm, the theoretical speed at the extraction movement is 0.0589 m/s and 0.0377 m/s at the retraction movement. The piston’s maximum velocity

occurs when the piston is in the retraction movement with the full opening range of the servo valve, where the area in the retraction phase is smaller than the area in the extension phase because of the area of the rod that existed in the high-pressure chamber during the retraction movement. This expected velocity requires a certain sampling time, considering the position sensor resolution of 1.95 micrometers.

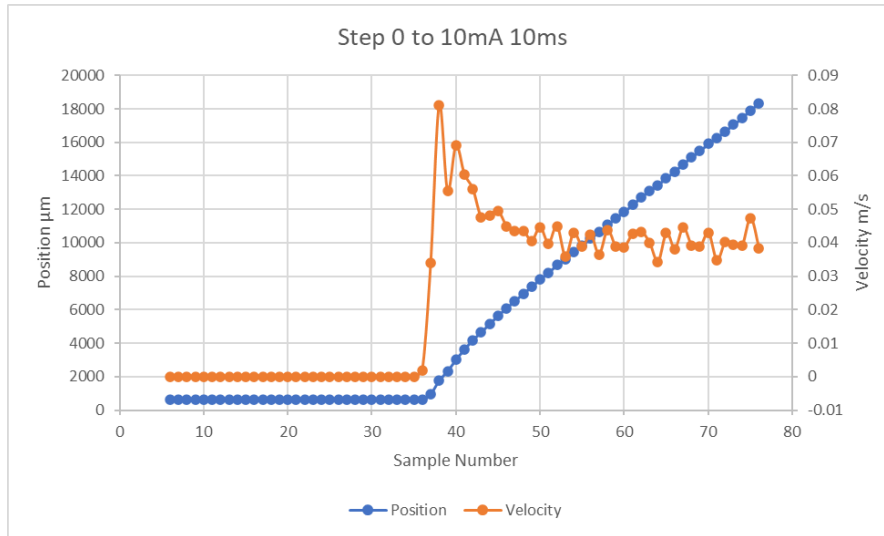
### **Setting up the sampling time:**

The sampling time is important to set in this test to suit the position sensor resolution and calculate the rod velocity, which is the position derivative. The working frequency is set to 10 kHz as a reading and writing I/O task. The derivative of the position in the same frequency could not be able to calculate the velocity with the tiny difference between every two consecutive readings. A velocity task was conducted every 10ms, resulting in a limited number of samples during the transition phase until the signal stabilized, potentially missing certain aspects of the signal dynamics. Acquiring an increased number of samples throughout the transition phase enhances the accuracy of system identification.

Figure 5.17 illustrates the captured signals for the position and the calculated velocity when a step signal was applied to the servo valve's coil. Limited samples were detected during the transition phase, including the overshoot peak.

The measured position curves show the non-linearity of the system even at the constant flow rate. The piston's internal leakage and the manufacturing defects vary the friction along the piston, causing nonlinear movement. And since these factors were not measured, they are treated as disturbances.

Another experiment involved altering the rod's movement direction by reversing the provided current for the servo valve's coil. This change was implemented by applying a negative signal immediately following a positive signal. The purpose was to capture the signal and determine the quantity of samples acquired during this directional alternation, as shown in Figure 5.18. A limited sample quantity was detected for this wide-range movement, which could result in omitting essential system dynamics.



**Figure 5.17:** Measured Velocity and Position with a Step Input 10 mA at 10ms Sampling Time

The current step was applied while the rod was extracting; the captured data could not be used in the system identification as step data, and it is not a sinusoidal signal with repetitive alternation in the signal.

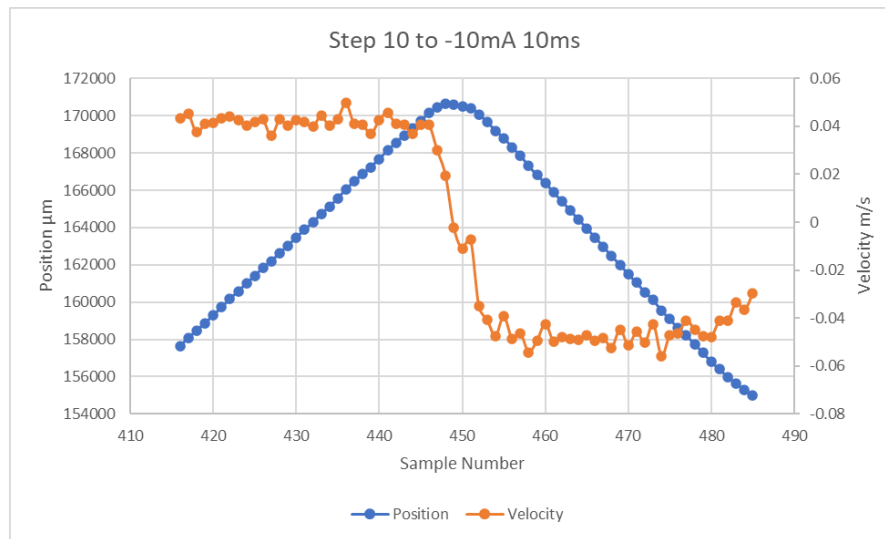
An additional experiment was carried out to observe the system’s behavior as the rod retracts, showcasing the variation in velocity due to the cylinder’s smaller area. This is intended to identify the sample quantities at different speeds. As well as the sample quantities in the rod during the stopping phase, as shown in Figure. 5.19.

The figure illustrates the position returned to its initial state, and only one sample was detected during the transition phase. Which is undoubtedly a lack of dynamic data.

Therefore, decreasing the sampling time is essential to enhance the number of detected samples, leading to an improved system identification process. This adjustment provides a more comprehensive understanding of the system dynamics by acquiring additional information.

The test was conducted again by changing the sampling time to **3ms** to catch more data in the transition phase. The results shown in the following figures





**Figure 5.18:** Measured Velocity and Position with a Step Input 10 to -10 mA at 10ms Sampling Time

5.20,5.21 illustrate more dynamic data was captured, especially at the beginning of the transition phase. That will facilitate the identification process through MATLAB by implementing the **ident** function.

In Figure. 5.20, The rod starting position was measured with the added 50-micrometer offset, disturbances deriving from system vibrations, and the pressure variety that might be compressing the piston while it resides in its initial position. These factors led to changes in the position sensor readings, acquiring data handling to remove the offset and these disturbances.

This data was collected from the real system and modified to suit the system identification process; the next step is to select a model structure. This set of data could be used alongside the servo-valve model to estimate the overall system model.

Another experiment was carried out with a 5 mA step input at the constant flow to prepare different data sets to be used in validating the model. That ensures the model will be valid even with different input steps.

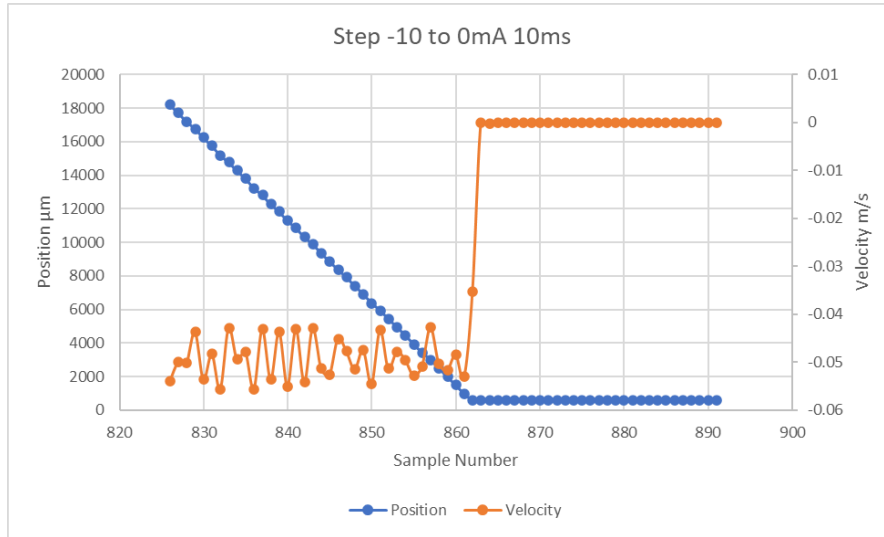


Figure 5.19: Measured Velocity and Position with a Step Input -10 mA at 10ms Sampling Time

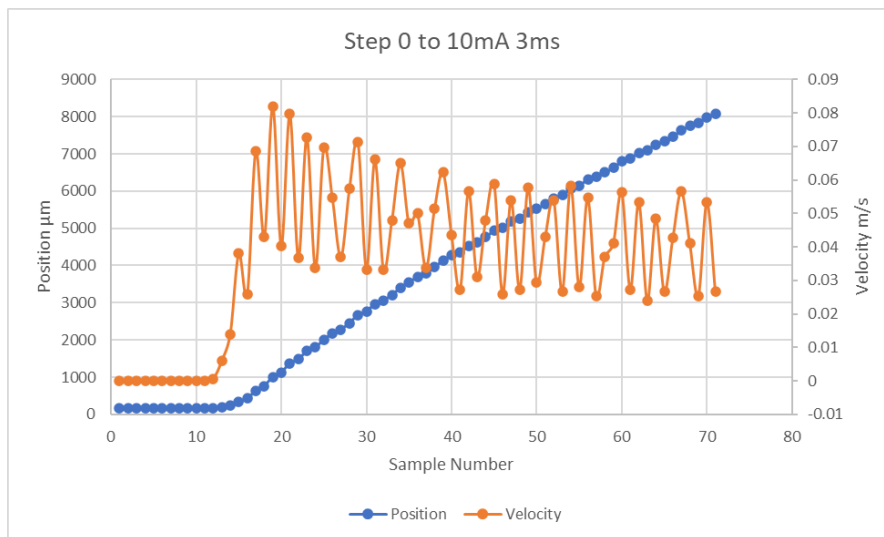
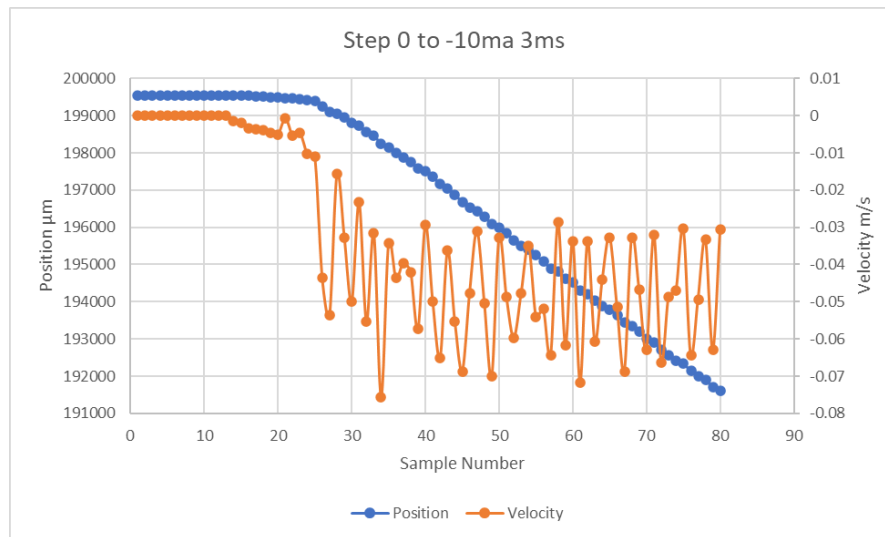


Figure 5.20: Measured Velocity and Position with a Step Input 10 mA at 3ms Sampling Time



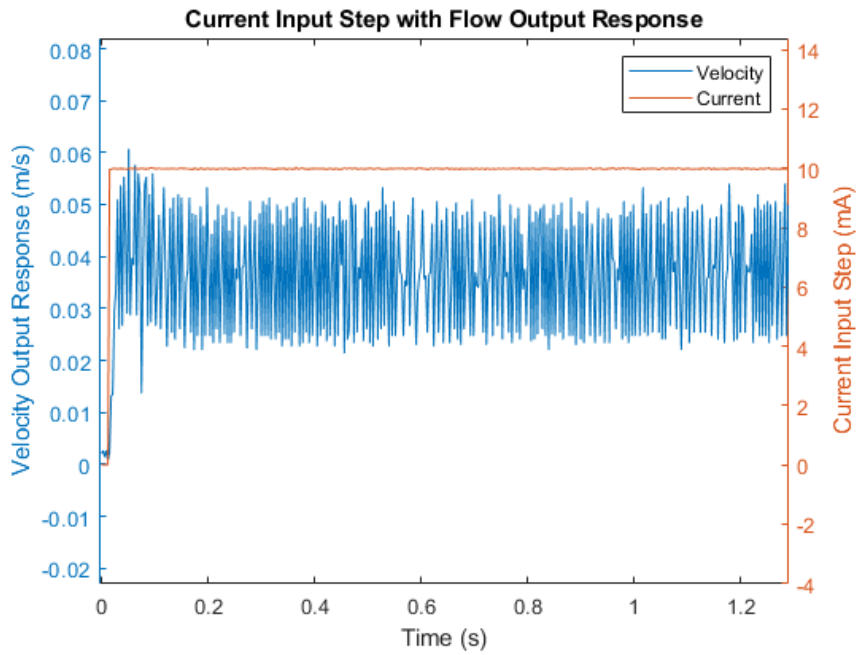
**Figure 5.21:** Measured Velocity and Position with a Step Input -10 mA at 3ms Sampling Time

### 5.4.3 Model Structure Selection

This step establishes a model structure that aligns with the collected data set. The system input and output relationship needs to be determined and represented by a transfer function as a model structure. Figure 5.22 shows the current input step and the velocity output response. A slight overshoot in the response was observed before the flow reached its steady state; this indicates that the studied system is at least a second-order system. Therefore, the given servo valve's transfer function could be used to facilitate the identification workflow and avoid trial and error in finding the right structure. This model structure cannot capture the non-linear behavior of the system, but it could be a starting point to identify the system with a reliable model structure and then tune the proper parameters.

### 5.4.4 Model Identification

The identification was carried out with two data sets collected at a constant flow rate while changing the servo valve current step. The model structure was chosen as a second-order system according to the servo-valve model, and then the identi-



**Figure 5.22:** Current Input Step with Flow Output Response

fication specified the parameters. The identified model represented in the transfer function 5.3

$$\frac{As + B}{s^2 + Cs + D} \tag{5.3}$$

The servo-valve input was excited with a step of 10 mA and 5 mA at a flow rate of 1.66 lpm. The identification tuned the parameters for every step and gave the parameters shown in Table. 5.16

Step input	5 mA	10 mA
A	1.378	0.4442
B	19.54	11.62
C	102.1	113.3
D	2474	3082

**Table 5.16:** System Parameters at step input 5 and 10 mA

### 5.4.5 Model Validation

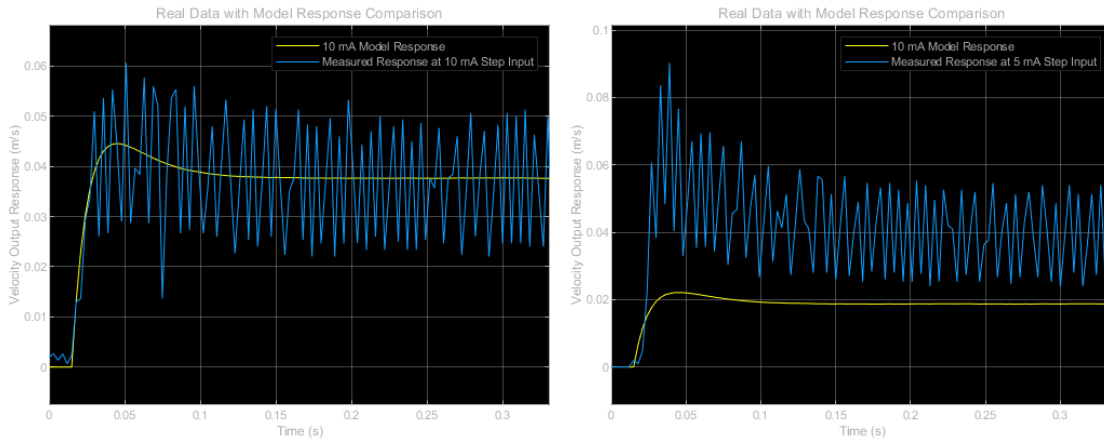
The collected data from many tests were involved in the validation process to validate the identified model. The process is to apply one data set with certain conditions to the identified model based on another data set. This intersection in using different data sets to validate models identified with other data sets allows to compare these models and choose the most identical one. The validation guarantees a favorable system response, even when confronted with varying initial conditions within the system.

The first validation step compares the measured data with the model response at a particular input step, where the used model in this step is identified based on the same input step. The validation is conducted by providing the data set to the captured transfer function and comparing the results with the measured data set in Figure 5.23a; it shows how close the model response is to the measured response relative to the mean of the measured response.

The second validation step is providing a set of the measured data with different input steps to the captured model and comparing this data with the model response. This step will determine if the model is reliable and applicable to different initial conditions of the system. Figure 5.23b shows the measured data amplitude mismatching the model response amplitude but having the same curve.

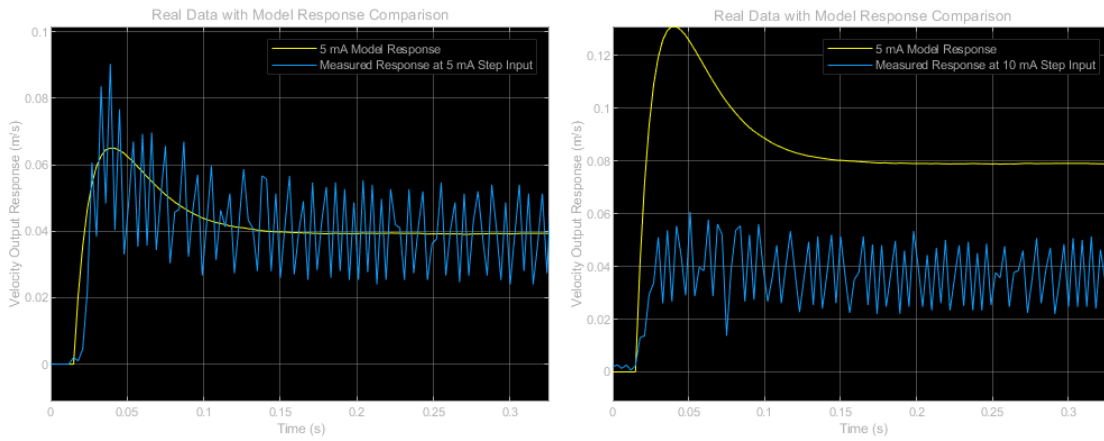
Since using the 10mA-based model does not give a close response to the measured data, the 5 mA model response was tested, and the results were compared while using the same data set and other data sets as input of the transfer function. The validation with the captured model is conducted while exciting the transfer function input with the 5 mA data set and the other case while switching to the 10 mA data set. The transfer function output was also compared with the measured responses separately.

Figure 5.24a, shows how close the model response is to the measured response relative to the mean of the measured response. And Figure 5.24b shows the measured data amplitude mismatching the model response amplitude, especially in the overshoot phase; it has a higher overshoot than the 10 mA model response.



(a) Real Data with Model Response Comparison (10 mA model with 10 mA data) (b) Real Data with Model Response Comparison 10 mA model with 5 mA data

**Figure 5.23:** Real Data with Model Response Comparison



(a) Real Data with Model Response Comparison (5 mA model with 5 mA data) (b) Real Data with Model Response Comparison 5 mA model with 10 mA data

**Figure 5.24:** Real Data with Model Response Comparison

In conclusion, the identified model gives a reliable response while using the same input step that specified its parameters. While using a model structure with another initial system condition, the model response does not match the measured response. There is an inverse relationship between the step input and the overshoot peak.

In this section, the system with all the servo-valve's essential dynamics was tested at a constant flow of 1.66 lpm, whereas the servo-valve's rated flow is around 8 lpm at the rated supply pressure 70 bar, and the maximum flow is around 12 lpm at the maximum supply pressure 210 bar.

Some essential overall system dynamics were missing in this test due to the lack of supplying the system with the required flow rate. The identified model could be used but with other sets of data that should be collected at the rated parameters of the system.

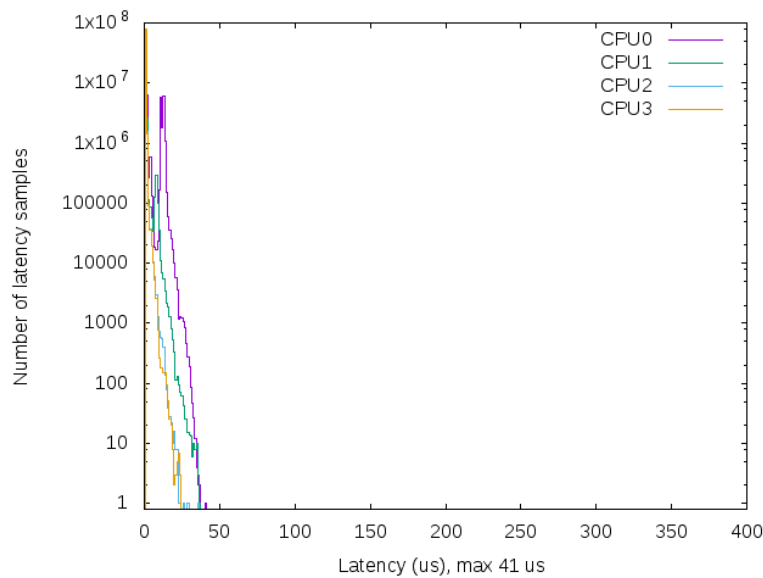
## 5.5 SEHA Control Architecture tests and results

Our test strategy focuses on the system's real-time capability and the robot's overall performance. The real-time tasks are affected by the master processor's latency, transmission delay, and slave latency.

### 5.5.1 Master controller performance test

To test the master latency, we installed Ubuntu 16.04 on the C6025 PC and patched it with the real-time kernel RT-Preempt. During the installation and patching, some essential configurations are required. The kernel preemption model should be configured to a fully preemptable kernel. This modification allows the high-priority task to interrupt the lower-priority task. Moreover, it allows the operating system to respect the defined deadlines. Another substantial modification is to disable a task's scheduling on several microprocessor cores. Some other kernel configurations are essential concerning the power management of the processor. We disable the frequency scaling and the idle state of the processor (The idle state is to power off

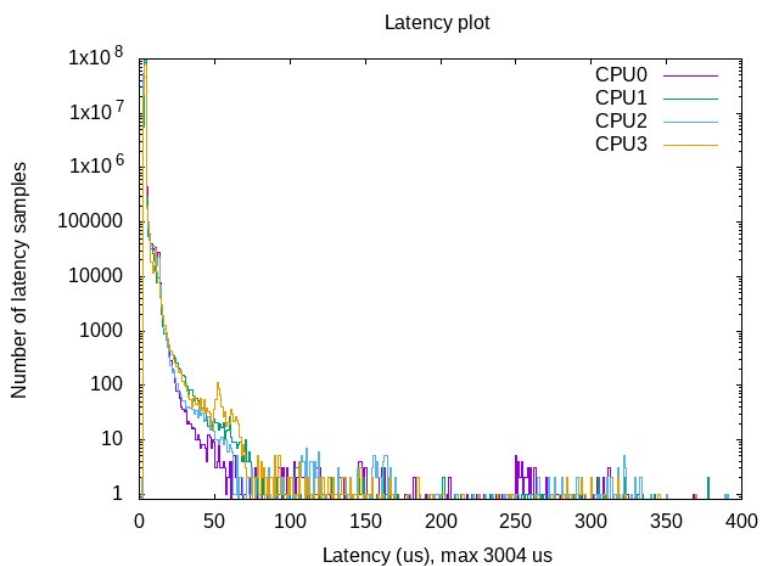
the CPU core after a defined period of inactivity) in order to avoid the delay of switching on the processor after an idle state. We test the performance after the installation to measure the latency of the patched kernel. We used the Cyclictest application [134] while simultaneously running many threads using CPU stress utility, network flood, and graphical stress. The test was launched for 100 million cycles; it takes 5 hours (the goal is to mimic the autonomy of the robot). The results are shown in Figure 5.25, for comparison, where we apply the same test on the Ubuntu system without real-time patching. We showed the results in Figure 5.26.



**Figure 5.25:** Real-Time Kernel Latency Experiment

Figure 5.25 shows that the test application has been launched on 4 CPUs; during the operation period, the maximum latency did not generally exceed 41 us. Some exceptions rarely happened (less than 100 cases over the 100 Million cases) Regarding the same test on the Ubuntu system without the real-time patching showed the results in Figure 5.26, we found that it produced more latency, often exceeding several milliseconds. This result means that the patch on our hardware guarantees the soft real-time performance.

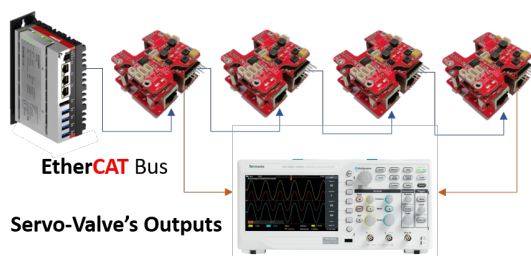




**Figure 5.26:** Non Real-Time Kernel Latency Experiment

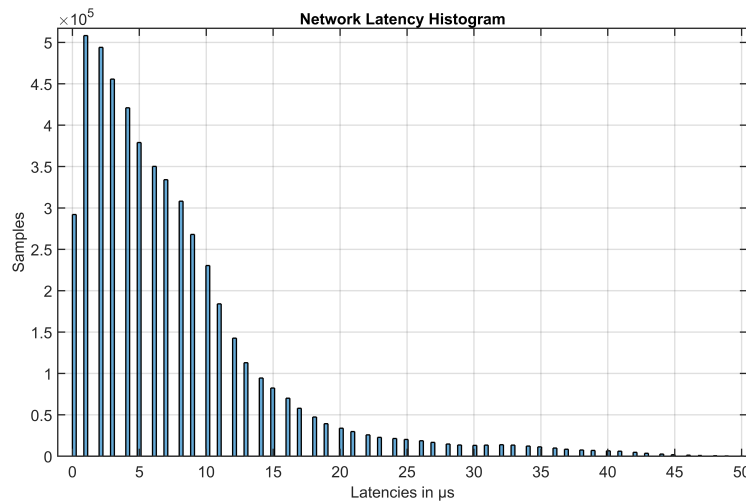
### 5.5.2 Network performance test

To test the network latency, we connected four of our joint controllers to the master PC. The Distributed Clocks (DC) were used to ensure synchronous communication. Then, we developed an OROCOS component to toggle the servo valve's outputs with 2kHz, which will generate a rectangular wave with 1kHz. We measured the delay between the first and last connected slaves' outputs. Figure 5.28 is the histogram for over one million measurements.



**Figure 5.27:** Latency Network Test

The results show that the central tendency is around the 0 to 6  $\mu\text{s}$  latency, and minority samples crossed this range but did not step over 282  $\mu\text{s}$ . In this test,



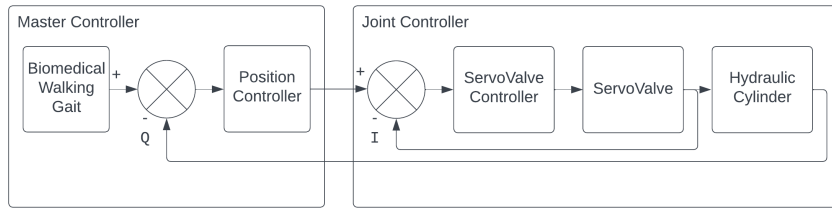
**Figure 5.28:** Network latency measurements

we connected only four customized joint controllers as shown in Figure 5.27 with less than the total payload. After adding all the joint controllers to the network, the estimated update rate is 3kHz.

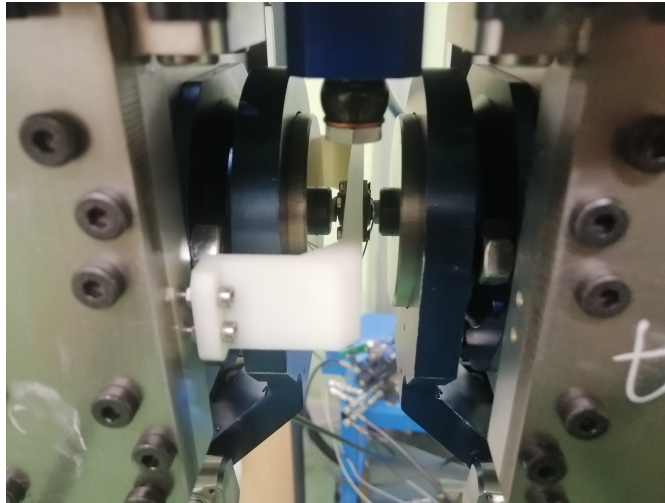
### 5.5.3 Walking gait Experiment

The developed architecture allows us to test the robot’s overall performance by easily implementing a gait control. The collected biomedical data was modified to fit the operation condition of the robot. The master applied a walking gait on four DoFs, controlling two hip pitches and two knees. The position controller was conducted in the master, taking the desired signal from the biomedical data set, activating the direct control mode in the joint controllers to implement the distributed sensorimotor program as shown in Figure 5.29

In this control mode, all the required data to be transmitted through the communication bus are the servo valve command and position feedback signals for each joint controller. This sensorimotor program is performed by the master, unlike other control modes where the joint controller will measure the feedback signal and command the servo valve based on it.



**Figure 5.29:** Distributed Sensorimotor Program with gait walking



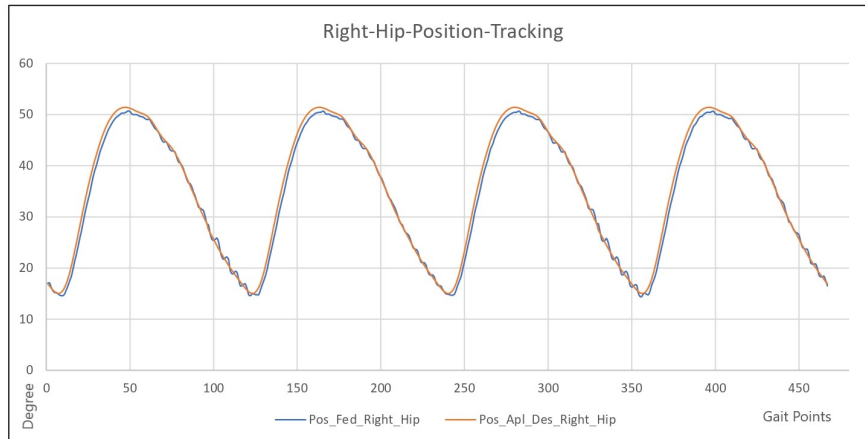
**Figure 5.30:** Position Sensors AMS AS5045B Installed on the Robot Hip Axis

The joint's rotational positions were measured with the AMS AS-5045B sensor installed on the joint rotation axis as shown in Figure 5.30. The position tracking test for the right hip is shown in Figure 5.31. It shows a good system response with small unpleasant vibrations.

Figure 5.32 demonstrates the movement during the gait walking experiment using the direct control mode.

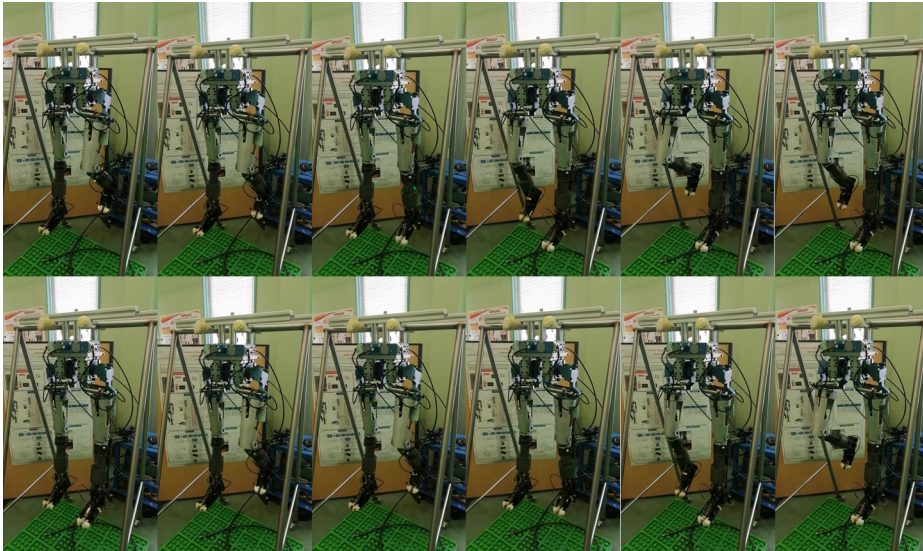
## 5.6 Conclusion

This chapter presented the practical development of SEHA modules' controllers and discussed their validation test processes. The ISO-4099 test bench control system was also developed based on the proposed control architecture and presented



**Figure 5.31:** Hip-Pitch position tracing measurements during the gait walking

in this chapter. The validation tests of the proposed control architecture elements for the HYDROiD robot were presented and discussed. In addition, after validating the module controller and the control architecture elements and as a first step toward integrating this actuator into HYDROiD, system identification was implemented for the SIC module to understand the system dynamics better and improve controlling the actuator. Finally, the chapter presented the results of the gait walking experiment while the master controller was executing the sensorimotor loop. The proposed architecture gives the ability to examine the sensorimotor loop not only in the joint controllers through the low-level control loop but also in the master controller through the high-level control loop. It provides real-time capability i) in the slave controller where the FreeRTOS is used, ii) in the master controller where the Preempt-RT Kernel acts as RTOS beside the real-time middleware OROCOS, and iii) in the EtherCAT communication protocol. Furthermore, the master-slave daisy chain topology allows connecting the sensors to their slave controllers to minimize the installed cables along the movable joints, making it more mobility and socially acceptable appearance.



**Figure 5.32:** HYDROiD Implementing Gait Walking



# Chapter 6

## Conclusions and Perspectives

### 6.1 Conclusions

This thesis approached a control architecture for the electro-hydraulic humanoid robot HYDROiD that emulates the human nervous system in controlling the muscles during various motions. The conclusion of this research has produced profound insights into the complicated domain of humanoid robotics, combining fields of biomechanics, state-of-the-art technologies, innovative actuation systems, and advanced control architectures.

The biomedical study chapter 1 provided profound insights into one method of human muscle control mechanisms during various motions. It presented these motions and the postural coordination as sensorimotor programs controlled by adaptive control strategies. The chapter analyzed a walking gait data set of healthy humans with the same morphology as the studied robot HYDROiD. These insights provided the groundwork for the subsequent chapters, guiding the development of HYDROiD towards a robot capable of emulating human motions and interacting seamlessly in real-world scenarios.

The state-of-the-art chapter 2 presented the historical evolution of humanoid

robots and investigated their control architecture systems and corresponding elements like communication technologies, software, middleware, joint controllers, and actuation systems. By analyzing actuation technologies, a hybrid actuation system that combines the advantages of both technologies is needed to push forward to an exciting future for humanoid robots. The chapter presented a unique factor: the robot application influences the control architectures in humanoid robots, emphasizing the need for flexibility in the control system for versatile humanoid robots.

The HYDROïD actuation system is presented in chapter 3; it relies on a Servo Electro-Hydraulic Actuator, a hybrid and modular actuator. It describes the actuator features, structure, development, and operation. It comprises i) a power generation module named Main Group Hydraulic (MGH), ii) a hydraulic actuator module named Servo Instrumented Cylinder (SIC), iii) the speed control module, iv) the force compensation module, v) a hydraulic circuit module. This chapter defined each module's monitoring and controlling points to perform sufficient control strategies. SEHA generates punctual power at active joints instead of the same pressure for all active joints delivered by a central hydraulic group in a classical hydraulic actuation system. It ensures a high level of physical safe interaction through its servo-force compensation module. It effectively manages the power consumption by securing the cylinder's position to remain in place during static postures without additional energy consumption.

The SEHA-based HYDROïD control architecture chapter 4 presents the proposed architecture approach and its development. This chapter laid the foundation for a versatile and adaptive control framework. The topology of the hardware architecture is centralized around a master PC with distributed joint controllers, while the software architecture is an adaptable based on the assigned task. It is distributed while performing precise motions that do not demand force control and decentralized while performing tasks demand compliance and force control. The proposed architecture gives the ability to examine the sensorimotor loop not only in the joint controllers through the low-level control loop but also in the master



controller through the high-level control loop. The systematic approach allows for the seamless integration of various elements at different control levels, ensuring independence and flexibility in the system. The approach facilitates the application's operation and proves valuable during the actuation system's developmental stages. The capability to replace developmental elements with validated counterparts enhances the efficiency of the validation process for other components within the control architecture.

The practical implementation and testing of the proposed solution elements is presented in the SEHA-Based HYDROiD Control Architecture Realization and Validation chapter 5. It encompasses the design, realization, and accurate testing and validation processes of SEHA modules' controllers as control architecture elements. The development of the ISO-4099 test bench control system was presented. A system identification was implemented for the SIC module to improve the actuator control. Finally, the chapter presented the results of the gait walking experiment while the master controller was executing the sensorimotor loop. The proposed architecture gives the ability to examine the sensorimotor loop not only in the joint controllers through the low-level control loop but also in the master controller through the high-level control loop. It provides real-time capability i) in the slave controller where the FreeRTOS is used, ii) in the master controller where the Preempt-RT Kernel acts as RTOS beside the real-time middleware OROCOS, and iii) in the EtherCAT communication protocol. Furthermore, the master-slave daisy chain topology allows connecting the sensors to their slave controllers to minimize the installed cables along the movable joints, making it more mobility and socially acceptable appearance. The results showed 50% advancements in the update rate compared to [76], where they reached a 2kHz update rate and 30% advancements in the control task latencies compared to [135]. This thesis paints a comprehensive picture of SEHA-based HYDROiD control architecture and brings it from concept to reality to potentially impact on the field.

## 6.2 Perspectives

The central perspective of this work is to validate the control architecture's flexibility in adapting various scenarios or tasks by changing its topology. The demonstrated walking gait test implemented the sensorimotor in the master controller with a distributed behavior of the joint controllers. The next test should implement the sensorimotor program in the joint controllers with a decentralized topology, where they can make decisions and control their corresponding joints. This validation will assist in testing various methods of the human control system to improve our understanding of human locomotion and manipulation.

A further perspective is to perform a system identification for the SIC module with the rated flow to validate the prepared sensorimotor program in the module controller and enhance the actuator's reliability when operating under varying loads.

A long-term perspective is about performing the walking gait with the 16DoFs of HYDROiD's lower body using six customized joint controllers as EtherCAT slaves to test and validate controlling the robot mechanisms locally. This will facilitate the development processes of humanoid robots and allow studying the integration of different elements in mechatronic and control systems.

# Publications

## International Journal

- S.Jleilaty, A. Ammounah, G. Abdulmalek, L. Nouveliere, S. AlFayad, Distributed Real-Time Control Architecture for Electro-Hydraulic Humanoid Robots, *Robotic Intelligence and Automation Journal*, February 2024

## National conference

- JNRH 2023: S.Jleilaty, M. Ghandour, G. Abdulmalek, M. Sleiman, A. Ammounah, S. AlFayad, EtherCAT-Based Real-Time Control Architecture for HYDROiD: Electro-Hydraulic Humanoid



# Bibliography

- [1] en. URL: <https://www.campusfrance.org/en/pause-program-urgent-aid-scientists-exile>.
- [2] Benoît G. Bardy et al. “Dynamics of human postural transitions.” en. In: *Journal of Experimental Psychology: Human Perception and Performance* 28.3 (2002), pp. 499–514. ISSN: 1939-1277, 0096-1523. DOI: [10.1037/0096-1523.28.3.499](https://doi.org/10.1037/0096-1523.28.3.499).
- [3] ISO standard. *ISO 4409:2019*. en. URL: <https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/06/84/68410.html>.
- [4] L. A. Peyré-Tartaruga and M. Coertjens. “Locomotion as a Powerful Model to Study Integrative Physiology: Efficiency, Economy, and Power Relationship”. In: *Frontiers in Physiology* 9 (2018). DOI: [10.3389/fphys.2018.01789](https://doi.org/10.3389/fphys.2018.01789).
- [5] *Nervous System Encyclopedia.com*. URL: <https://www.encyclopedia.com/medicine/anatomy-and-physiology/anatomy-and-physiology/nervous-system>.
- [6] *Overview of neuron structure and function*. en. URL: <https://www.khanacademy.org/science/biology/human-biology/neuron-nervous-system/a/overview-of-neuron-structure-and-function>.
- [7] R. J. Peterka. “Sensorimotor Integration in Human Postural Control”. en. In: *Journal of Neurophysiology* 88.3 (Sept. 2002), pp. 1097–1118. ISSN: 0022-3077, 1522-1598. DOI: [10.1152/jn.2002.88.3.1097](https://doi.org/10.1152/jn.2002.88.3.1097).

- [8] J. F. Yang, D. A. Winter, and R. P. Wells. “Postural dynamics in the standing human”. eng. In: *Biological Cybernetics* 62.4 (1990), pp. 309–320. ISSN: 0340-1200. DOI: [10.1007/BF00201445](https://doi.org/10.1007/BF00201445).
- [9] Turgay Akay and Andrew J. Murray. “Relative Contribution of Proprioceptive and Vestibular Sensory Systems to Locomotion: Opportunities for Discovery in the Age of Molecular Science”. In: *International Journal of Molecular Sciences* 22.3 (Feb. 2021), p. 1467. ISSN: 1422-0067. DOI: [10.3390/ijms22031467](https://doi.org/10.3390/ijms22031467).
- [10] Maziar Ahmad Sharbafi and Andre Seyfarth. “How locomotion sub-functions can control walking at different speeds?” en. In: *Journal of Biomechanics* 53 (Feb. 2017), pp. 163–170. ISSN: 00219290. DOI: [10.1016/j.jbiomech.2017.01.018](https://doi.org/10.1016/j.jbiomech.2017.01.018).
- [11] R. Baker et al. “The conventional gait model - success and limitations”. In: (2018). DOI: [10.1007/978-3-319-14418-4\\_25](https://doi.org/10.1007/978-3-319-14418-4_25).
- [12] F. Leboeuf et al. “The conventional gait model, an open-source implementation that reproduces the past but prepares for the future.” In: *Gait & posture* 69 (2019), pp. 126–129. DOI: [10.1016/j.gaitpost.2019.01.034](https://doi.org/10.1016/j.gaitpost.2019.01.034).
- [13] James Robinson and Gary Smidt. “Quantitative Gait Evaluation in the Clinic”. In: *Physical therapy* 61 (Apr. 1981), pp. 351–3. DOI: [10.1093/ptj/61.3.351](https://doi.org/10.1093/ptj/61.3.351).
- [14] Samer Alfayad. “Robot humanoïde HYDROiD: actionnement, structure cinématique et stratégie de contrôle”. These de doctorat. Versailles-St Quentin en Yvelines, Jan. 2009. URL: <https://www.theses.fr/2009VERS0016>.
- [15] Ye Xie et al. “A Review: Robust Locomotion for Biped Humanoid Robots”. en. In: *Journal of Physics: Conference Series* 1487.1 (Mar. 2020), p. 012048. ISSN: 1742-6596. DOI: [10.1088/1742-6596/1487/1/012048](https://doi.org/10.1088/1742-6596/1487/1/012048).

- [16] J.E. Pratt et al. “The RoboKnee: an exoskeleton for enhancing strength and endurance during walking”. In: *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004.* Vol. 3. Apr. 2004, 2430–2435 Vol.3. DOI: [10.1109/ROBOT.2004.1307425](https://doi.org/10.1109/ROBOT.2004.1307425). URL: <https://ieeexplore.ieee.org/document/1307425>.
- [17] URL: [https://www.humanoid.waseda.ac.jp/booklet/kato\\_2.html](https://www.humanoid.waseda.ac.jp/booklet/kato_2.html).
- [18] S. Sugano and I. Kato. “WABOT-2: Autonomous robot with dexterous finger-arm–Finger-arm coordination control in keyboard performance”. In: *1987 IEEE International Conference on Robotics and Automation Proceedings.* Vol. 4. Mar. 1987, pp. 90–97. DOI: [10.1109/ROBOT.1987.1088025](https://doi.org/10.1109/ROBOT.1987.1088025). URL: <https://ieeexplore.ieee.org/document/1088025>.
- [19] K. Hirai et al. “The development of Honda humanoid robot”. In: *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146).* Vol. 2. May 1998, 1321–1326 vol.2. DOI: [10.1109/ROBOT.1998.677288](https://doi.org/10.1109/ROBOT.1998.677288). URL: <https://ieeexplore.ieee.org/document/677288>.
- [20] Satoshi Shigemi. “ASIMO and Humanoid Robot Research at Honda”. en. In: *Humanoid Robotics: A Reference.* Ed. by Ambarish Goswami and Prahlad Vadakkepat. Dordrecht: Springer Netherlands, 2019, pp. 55–90. ISBN: 978-94-007-6045-5. DOI: [10.1007/978-94-007-6046-2\\_9](https://doi.org/10.1007/978-94-007-6046-2_9). URL: [http://link.springer.com/10.1007/978-94-007-6046-2\\_9](http://link.springer.com/10.1007/978-94-007-6046-2_9).
- [21] Y. Sakagami et al. “The intelligent ASIMO: system overview and integration”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems.* Vol. 3. Sept. 2002, 2478–2483 vol.3. DOI: [10.1109/IRDS.2002.1041641](https://doi.org/10.1109/IRDS.2002.1041641). URL: <https://ieeexplore.ieee.org/abstract/document/1041641>.
- [22] URL: <https://pc.watch.impress.co.jp/docs/2002/0411/hrp1.htm>.
- [23] K. Kaneko et al. “Design of prototype humanoid robotics platform for HRP”. In: *IEEE/RSJ International Conference on Intelligent Robots and*

- Systems*. Vol. 3. Sept. 2002, 2431–2436 vol.3. DOI: [10.1109/IRDS.2002.1041632](https://doi.org/10.1109/IRDS.2002.1041632). URL: <https://ieeexplore.ieee.org/document/1041632>.
- [24] Kazuhito Yokoi et al. “Experimental Study of Humanoid Robot HRP-1S”. In: *The International Journal of Robotics Research* 23.4–5 (Apr. 2004), pp. 351–362. ISSN: 0278-3649. DOI: [10.1177/0278364904042194](https://doi.org/10.1177/0278364904042194).
- [25] Kenji Kaneko et al. “Humanoid Robot HRP-2”. en. In: ().
- [26] Kensuke Harada and Takakatsu Isozumi. “Humanoid robot HRP-3”. en. In: (Jan. 2008). URL: [https://www.academia.edu/3211567/Humanoid\\_robot\\_HRP\\_3](https://www.academia.edu/3211567/Humanoid_robot_HRP_3).
- [27] Kenji Kaneko et al. “Humanoid robot HRP-4 - Humanoid robotics platform with lightweight and slim body”. In: *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Sept. 2011, pp. 4400–4407. DOI: [10.1109/IROS.2011.6094465](https://doi.org/10.1109/IROS.2011.6094465). URL: [https://ieeexplore.ieee.org/abstract/document/6094465?casa\\_token=\\_UqjHN\\_mJuMAAAAA:LNQWUVPtb2xD22VnT60iSsiv-ssnbcUx3e5NhN3kfNgvmrnfbvDP9TfpVwk\\_02BY503XAh\\_oSjc](https://ieeexplore.ieee.org/abstract/document/6094465?casa_token=_UqjHN_mJuMAAAAA:LNQWUVPtb2xD22VnT60iSsiv-ssnbcUx3e5NhN3kfNgvmrnfbvDP9TfpVwk_02BY503XAh_oSjc).
- [28] Shuuji Kajita et al. *Cybernetic Human HRP-4C: A Humanoid Robot with Human-Like Proportions*. Jan. 2009, p. 314.
- [29] Kenji Kaneko et al. “Humanoid Robot HRP-5P: An Electrically Actuated Humanoid Robot With High-Power and Wide-Range Joints”. In: *IEEE Robotics and Automation Letters* 4.2 (Apr. 2019), pp. 1431–1438. ISSN: 2377-3766. DOI: [10.1109/LRA.2019.2896465](https://doi.org/10.1109/LRA.2019.2896465).
- [30] Jung-Woo Heo, In-Ho Lee, and Jun-Ho Oh. “Development of Humanoid Robots in HUBO Laboratory, KAIST”. en. In: 30.4 (2012).
- [31] Baek-Kyu Cho, Sang-Sin Park, and Jun-ho Oh. “Controllers for running in the humanoid robot, HUBO”. In: *2009 9th IEEE-RAS International Conference on Humanoid Robots*. Dec. 2009, pp. 385–390. DOI: [10.1109/ICHR.2009.5379574](https://doi.org/10.1109/ICHR.2009.5379574). URL: <https://ieeexplore.ieee.org/abstract/document/5379574>.



- [32] Y.F. Zheng et al. “Humanoid robots walking on grass, sands and rocks”. In: *2013 IEEE Conference on Technologies for Practical Robot Applications (TePRA)*. Apr. 2013, pp. 1–6. DOI: [10.1109/TePRA.2013.6556367](https://doi.org/10.1109/TePRA.2013.6556367). URL: <https://ieeexplore.ieee.org/document/6556367>.
- [33] Taejin Jung et al. “Development of the Humanoid Disaster Response Platform DRC-HUBO+”. In: *IEEE Transactions on Robotics* 34.1 (Feb. 2018), pp. 1–17. ISSN: 1552-3098, 1941-0468. DOI: [10.1109/TR0.2017.2776287](https://doi.org/10.1109/TR0.2017.2776287).
- [34] Alberto Parmiggiani et al. “THE DESIGN OF THE iCub HUMANOID ROBOT”. en. In: *International Journal of Humanoid Robotics* 09.04 (Dec. 2012), p. 1250027. ISSN: 0219-8436, 1793-6942. DOI: [10.1142/S0219843612500272](https://doi.org/10.1142/S0219843612500272).
- [35] Giorgio Metta et al. “The iCub humanoid robot: An open-systems platform for research in cognitive development”. In: *Neural Networks. Social Cognition: From Babies to Robots* 23.8 (Oct. 2010), pp. 1125–1134. ISSN: 0893-6080. DOI: [10.1016/j.neunet.2010.08.010](https://doi.org/10.1016/j.neunet.2010.08.010).
- [36] M.A. Diftler et al. “Robonaut 2 - The first humanoid robot in space”. In: *2011 IEEE International Conference on Robotics and Automation*. May 2011, pp. 2178–2183. DOI: [10.1109/ICRA.2011.5979830](https://doi.org/10.1109/ICRA.2011.5979830). URL: <https://ieeexplore.ieee.org/document/5979830>.
- [37] Nicolaus Radford et al. “Valkyrie: NASA’s First bipedal humanoid robot”. In: *Journal of Field Robotics* 32 (May 2015), pp. 397–419. DOI: [10.1002/rob.21560](https://doi.org/10.1002/rob.21560).
- [38] Nicholas Paine et al. “Actuator Control for the NASA-JSC Valkyrie Humanoid Robot: A Decoupled Dynamics Approach for Torque Control of Series Elastic Robots”. en. In: *Journal of Field Robotics* 32.3 (May 2015), pp. 378–396. ISSN: 1556-4959, 1556-4967. DOI: [10.1002/rob.21556](https://doi.org/10.1002/rob.21556).
- [39] Nikos G. Tsagarakis et al. “COMpliant huMANoid COMAN: Optimal joint stiffness tuning for modal frequency control”. In: *2013 IEEE International Conference on Robotics and Automation*. May 2013, pp. 673–678. DOI:

- 10.1109/ICRA.2013.6630645. URL: <https://ieeexplore.ieee.org/document/6630645>.
- [40] Arash Ajoudani et al. “A manipulation framework for compliant humanoid COMAN: Application to a valve turning task”. In: *2014 IEEE-RAS International Conference on Humanoid Robots*. Nov. 2014, pp. 664–670. DOI: 10.1109/HUMANOIDS.2014.7041434. URL: [https://ieeexplore.ieee.org/abstract/document/7041434?casa\\_token=90gcJ86KpkUAAAAA:\\_vb68UnLrvpDIp8mZoCFBQsBSOCWut405mTTdfKfx-BPOP966hoB0i4E6m7\\_1X\\_WZ4h04ezolSOJ1g](https://ieeexplore.ieee.org/abstract/document/7041434?casa_token=90gcJ86KpkUAAAAA:_vb68UnLrvpDIp8mZoCFBQsBSOCWut405mTTdfKfx-BPOP966hoB0i4E6m7_1X_WZ4h04ezolSOJ1g).
- [41] Mirko Ferrati et al. “The Walk-Man Robot Software Architecture”. In: *Frontiers in Robotics and AI* 3 (May 2016), p. 25. ISSN: 2296-9144. DOI: 10.3389/frobt.2016.00025.
- [42] N. G. Tsagarakis et al. “WALK-MAN: A High-Performance Humanoid Platform for Realistic Environments”. en. In: *Journal of Field Robotics* 34.7 (2017), pp. 1225–1259. ISSN: 1556-4967. DOI: 10.1002/rob.21702.
- [43] O. Stasse et al. “TALOS: A new humanoid research platform targeted for industrial applications”. In: *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*. Nov. 2017, pp. 689–695. DOI: 10.1109/HUMANOIDS.2017.8246947. URL: [https://ieeexplore.ieee.org/abstract/document/8246947?casa\\_token=zBy90QZTOoYAAAAA:AUjJOY708cvLQUAOpVLMdCxi\\_Q41ES0kyRV4Iy\\_MjdYOH3TeXaUPFQ9XIynwsXW-y\\_LRREyEgHLQug](https://ieeexplore.ieee.org/abstract/document/8246947?casa_token=zBy90QZTOoYAAAAA:AUjJOY708cvLQUAOpVLMdCxi_Q41ES0kyRV4Iy_MjdYOH3TeXaUPFQ9XIynwsXW-y_LRREyEgHLQug).
- [44] Eric C Whitman. “Robust Optimal Walking on the Sarcos Humanoid Robot”. en. In: ().
- [45] Gordon Cheng et al. “CB: a humanoid research platform for exploring neuroscience”. In: *Advanced Robotics* 21.10 (Jan. 2007), pp. 1097–1114. ISSN: 0169-1864. DOI: 10.1163/156855307781389356.
- [46] S. Alfayad, F.B. Ouezdou, and F. Namoun. “New three DOF ankle mechanism for humanoid robotic application: Modeling, design and realization”.

- In: *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Oct. 2009, pp. 4969–4976. DOI: [10.1109/IR0S.2009.5354806](https://doi.org/10.1109/IR0S.2009.5354806).
- [47] S. Alfayad et al. “Three DOF hybrid mechanism for humanoid robotic application: Modeling, design and realization”. In: *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Oct. 2009, pp. 4955–4961. DOI: [10.1109/IR0S.2009.5354811](https://doi.org/10.1109/IR0S.2009.5354811).
- [48] Samer Alfayad, Fethi B. Ouezdou, and Faycal Namoun. “New 3-DOFs Hybrid Mechanism for Ankle and Wrist of Humanoid Robot: Modeling, Simulation, and Experiments”. In: *Journal of Mechanical Design* 133.021005 (Jan. 2011). ISSN: 1050-0472. DOI: [10.1115/1.4003250](https://doi.org/10.1115/1.4003250). URL: <https://doi.org/10.1115/1.4003250>.
- [49] Samer Alfayad et al. “(54) HUMANOID ROBOT IMPLEMENTING A SPHERICAL HINGE WITH COUPLED ACTUATORS”. en. In: ().
- [50] M. Ellasswad et al. “Development of lightweight hydraulic cylinder for humanoid robots applications”. In: *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 232.18 (2018), pp. 3351–3364. ISSN: 20412983. DOI: [10.1177/0954406217731794](https://doi.org/10.1177/0954406217731794).
- [51] S. Alfayad et al. “Lightweight high performance integrated actuator for humanoid robotic applications: Modeling, design and realization”. In: *2009 IEEE International Conference on Robotics and Automation*. May 2009, pp. 562–567. DOI: [10.1109/ROBOT.2009.5152286](https://doi.org/10.1109/ROBOT.2009.5152286).
- [52] Samer Alfayad et al. “HYDROiD humanoid robot head with perception and emotion capabilities: Modeling, design, and experimental results”. In: *Frontiers Robotics AI* 3.APR (2016), pp. 1–16. ISSN: 22969144. DOI: [10.3389/frobt.2016.00015](https://doi.org/10.3389/frobt.2016.00015).
- [53] A. Abdellatif et al. “Development of a New Hydraulic Ankle for HYDROiD Humanoid Robot”. In: *Journal of Intelligent and Robotic Systems* 92.2 (Oct. 2018), pp. 293–308. ISSN: 1573-0409. DOI: [10.1007/s10846-017-0750-z](https://doi.org/10.1007/s10846-017-0750-z).

- [54] M. Tahoun et al. “Force localized interaction sensing system for HYDRO#x00EF;D humanoid robot”. In: *IEEE Sensors Journal* PP.c (2019), p. 1. ISSN: 15581748. DOI: [10.1109/JSEN.2019.2962281](https://doi.org/10.1109/JSEN.2019.2962281).
- [55] Samer Alfayad et al. “High performance integrated electro-hydraulic actuator for robotics - Part I: Principle, prototype design and first experiments”. In: *Sensors and Actuators, A: Physical* 169.1 (2011), pp. 115–123. ISSN: 09244247. DOI: [10.1016/j.sna.2010.10.026](https://doi.org/10.1016/j.sna.2010.10.026). URL: <http://dx.doi.org/10.1016/j.sna.2010.10.026>.
- [56] Samer Alfayad. “Modeling and Simulation of a New Integrated Electrohydraulic Actuator for Humanoid Robots”. In: *International Journal of Advanced Robotics and Automation* 1.3 (2016), pp. 1–12. DOI: [10.15226/2473-3032/1/3/00114](https://doi.org/10.15226/2473-3032/1/3/00114).
- [57] Samer Alfayad, Mohamad Kardofaki, and Maya Sleiman. *Actionneur hydraulique à compensation de surpression*. fr. July 2022. URL: <https://patents.google.com/patent/FR3093138B1/en?inventor=Samer+Alfayad>.
- [58] Gabe Nelson et al. “PETMAN: A Humanoid Robot for Testing Chemical Protective Clothing”. In: *Journal of the Robotics Society of Japan* 30.4 (2012), pp. 372–377. DOI: [10.7210/jrsj.30.372](https://doi.org/10.7210/jrsj.30.372).
- [59] en. URL: <https://robotsguide.com/robots/atlas2016>.
- [60] URL: <https://www.darpa.mil/about-us/timeline/debut-atlas-robot>.
- [61] Christopher G Atkeson et al. “Achieving reliable humanoid robot operations in the DARPA robotics challenge: team WPI-CMU’s approach”. In: *The DARPA Robotics Challenge Finals: Humanoid Robots To The Rescue* (2018), pp. 271–307.
- [62] Siyuan Feng et al. “Optimization based controller design and implementation for the Atlas robot in the DARPA Robotics Challenge Finals”. In: *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. Nov. 2015, pp. 1028–1035. DOI: [10.1109/HUMANOIDS.2015](https://doi.org/10.1109/HUMANOIDS.2015).

7363480. URL: [https://ieeexplore.ieee.org/abstract/document/7363480?casa\\_token=GvLt5QOUybKAAAAA:m2iZdtqCp9CWRklc\\_P14g0-a2VdT-huIoo8uLDIoo2JpgH28uvq2QM7qwxexEub07XR9z6fWU8byIw](https://ieeexplore.ieee.org/abstract/document/7363480?casa_token=GvLt5QOUybKAAAAA:m2iZdtqCp9CWRklc_P14g0-a2VdT-huIoo8uLDIoo2JpgH28uvq2QM7qwxexEub07XR9z6fWU8byIw).
- [63] Nandan Banerjee et al. “Human-supervised control of the ATLAS humanoid robot for traversing doors”. In: *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. Nov. 2015, pp. 722–729. DOI: [10.1109/HUMANOIDS.2015.7363442](https://doi.org/10.1109/HUMANOIDS.2015.7363442). URL: [https://ieeexplore.ieee.org/abstract/document/7363442?casa\\_token=qdKGLLZ0cK8AAAAA:iTl\\_RjVIPLlVz-999usEDxiUvi9cdpv\\_QH-lW3-M-U3DbhfCx2h3z4PoFW0x386EiWhsItUgQzXnbg](https://ieeexplore.ieee.org/abstract/document/7363442?casa_token=qdKGLLZ0cK8AAAAA:iTl_RjVIPLlVz-999usEDxiUvi9cdpv_QH-lW3-M-U3DbhfCx2h3z4PoFW0x386EiWhsItUgQzXnbg).
- [64] en-US. URL: <https://bostondynamics.com/atlas/>.
- [65] Ko Yamamoto et al. “Experimental Validation of Resolved Viscoelasticity Control on Hydrostatically Driven Humanoid Hydra”. en. In: *Proceedings of the 2018 International Symposium on Experimental Robotics*. Ed. by Jing Xiao, Torsten Kröger, and Oussama Khatib. Springer Proceedings in Advanced Robotics. Cham: Springer International Publishing, 2020, pp. 619–628. ISBN: 978-3-030-33950-0. DOI: [10.1007/978-3-030-33950-0\\_53](https://doi.org/10.1007/978-3-030-33950-0_53).
- [66] Tianyi Ko et al. “Whole-Body Compliant Motion by Sensor Integration of an EHA-Driven Humanoid Hydra”. In: *International Journal of Humanoid Robotics* 18.01 (Feb. 2021), p. 2150002. ISSN: 0219-8436. DOI: [10.1142/S021984362150002X](https://doi.org/10.1142/S021984362150002X).
- [67] K. Akachi et al. “Development of humanoid robot HRP-3P”. In: *5th IEEE-RAS International Conference on Humanoid Robots, 2005*. Dec. 2005, pp. 50–55. DOI: [10.1109/ICHR.2005.1573544](https://doi.org/10.1109/ICHR.2005.1573544). URL: <https://ieeexplore.ieee.org/abstract/document/1573544>.
- [68] Fumio Kanehiro et al. “Distributed control system of humanoid robots based on real-time ethernet”. In: *IEEE International Conference on Intelligent Robots and Systems (2006)*, pp. 2471–2477. DOI: [10.1109/IROS.2006.281691](https://doi.org/10.1109/IROS.2006.281691).

- [69] T. Matsui et al. “Distributed real-time processing for humanoid robots”. In: *11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA '05)*. Aug. 2005, pp. 205–210. DOI: [10.1109/RTCSA.2005.39](https://doi.org/10.1109/RTCSA.2005.39). URL: <https://ieeexplore.ieee.org/document/1541082>.
- [70] Zhangguo Yu et al. “Distributed Control System for a Humanoid Robot”. In: *2007 International Conference on Mechatronics and Automation*. Aug. 2007, pp. 1166–1171. DOI: [10.1109/ICMA.2007.4303713](https://doi.org/10.1109/ICMA.2007.4303713). URL: <https://ieeexplore.ieee.org/document/4303713>.
- [71] Brennan Pierce and Gordon Cheng. “Herbert: Design and realization of a full-sized anthropometrically correct humanoid robot”. In: *Frontiers Robotics AI 2*. JUN (2015), pp. 1–22. ISSN: 22969144. DOI: [10.3389/frobt.2015.00014](https://doi.org/10.3389/frobt.2015.00014).
- [72] M. Omar Faruque Sarker et al. “An IEEE-1394 based real-time robot control system for efficient controlling of humanoids”. In: *IEEE International Conference on Intelligent Robots and Systems (2006)*, pp. 1416–1421. DOI: [10.1109/IR0S.2006.281933](https://doi.org/10.1109/IR0S.2006.281933).
- [73] K. Kaneko et al. “Humanoid robot HRP-3”. In: *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2008, pp. 2471–2478.
- [74] *Website KAIST, Korea*. <https://robots.ieee.org/robots/drchubo/>. Accessed: 2020-01-28.
- [75] Gabe Nelson, Aaron Saunders, and Robert Playter. “The PETMAN and Atlas Robots at Boston Dynamics”. In: *Humanoid Robotics: A Reference (2017)*, pp. 1–18. DOI: [10.1007/978-94-007-7194-9\\_15-1](https://doi.org/10.1007/978-94-007-7194-9_15-1).
- [76] Felix Sygulla et al. “An EtherCAT-Based Real-Time Control System Architecture for Humanoid Robots”. In: *IEEE International Conference on Automation Science and Engineering 2018-Augus (2018)*, pp. 483–490. ISSN: 21618089. DOI: [10.1109/COASE.2018.8560532](https://doi.org/10.1109/COASE.2018.8560532).

- [77] Olivier Stasse et al. “TALOS : A new humanoid research platform targeted for industrial applications To cite this version : HAL Id : hal-01485519 TALOS : A new humanoid research platform targeted for industrial applications”. In: (2018).
- [78] Giuseppe F. Rigano et al. “Towards a robot hardware abstraction layer (R-HAL) leveraging the XBot software framework”. In: *Proceedings - 2nd IEEE International Conference on Robotic Computing, IRC 2018* 2018-Janua (2018), pp. 175–176. DOI: [10.1109/IRC.2018.00036](https://doi.org/10.1109/IRC.2018.00036).
- [79] Martin Rostan, Joseph E. Stubbs, and Dmitry Dzilno. “EtherCAT enabled advanced control architecture”. en. In: *2010 IEEE/SEMI Advanced Semiconductor Manufacturing Conference (ASMC)*. San Francisco, CA, USA: IEEE, July 2010, pp. 39–44. ISBN: 978-1-4244-6517-0. DOI: [10.1109/ASMC.2010.5551414](https://doi.org/10.1109/ASMC.2010.5551414). URL: <http://ieeexplore.ieee.org/document/5551414/>.
- [80] Kenji Kaneko, Fumio Kanehiro, and Mitsuharu Morisawa. “Humanoid Robot HRP-4 - Humanoid Robotics Platform with Lightweight and Slim Body -”. en. In: ().
- [81] O. Stasse and T. Flayols. “An Overview of Humanoid Robots Technologies”. en. In: *Biomechanics of Anthropomorphic Systems*. Ed. by Gentiane Venture, Jean-Paul Laumond, and Bruno Watier. Vol. 124. Springer Tracts in Advanced Robotics. Cham: Springer International Publishing, 2019, pp. 281–310. ISBN: 978-3-319-93869-1. DOI: [10.1007/978-3-319-93870-7\\_13](https://doi.org/10.1007/978-3-319-93870-7_13). URL: [http://link.springer.com/10.1007/978-3-319-93870-7\\_13](http://link.springer.com/10.1007/978-3-319-93870-7_13).
- [82] Johannes Engelsberger et al. “Torque-Based Dynamic Walking - A Long Way from Simulation to Experiment”. en. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. Brisbane, QLD: IEEE, May 2018, pp. 440–447. ISBN: 978-1-5386-3081-5. DOI: [10.1109/ICRA.2018.8462862](https://doi.org/10.1109/ICRA.2018.8462862). URL: <https://ieeexplore.ieee.org/document/8462862/>.



- [83] Valerio Favot et al. “The sensor-controller network of the humanoid robot LOLA”. In: *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)* (2012), pp. 805–810.
- [84] Sisir Karumanchi et al. “Team RoboSimian: Semi-autonomous Mobile Manipulation at the 2015 DARPA Robotics Challenge Finals”. en. In: ().
- [85] Federico Reghenzani, Giuseppe Massari, and William Fornaciari. “The real-time linux kernel: A survey on Preempt\_RT”. In: *ACM Computing Surveys* 52.1 (2019). ISSN: 15577341. DOI: [10.1145/3297714](https://doi.org/10.1145/3297714).
- [86] Krishna Kavi, Robert Akl, and Ali Hurson. “Real-Time Systems: An Introduction and the State-of-the-Art”. In: *Wiley Encyclopedia of Computer Science and Engineering* (2009). DOI: [10.1002/9780470050118.ecse344](https://doi.org/10.1002/9780470050118.ecse344).
- [87] “M. Quigley, K. Conley, B. Gerkey, J. Faust, T. B. Foote, J. Leibs, R. Wheeler, A.Y. Ng, "ROS: An open-source robot operating system", Proc. ICRA Open-Source Softw. Workshop, 2009.” In: ().
- [88] Giorgio Metta, Paul Fitzpatrick, and Lorenzo Natale. “YARP: Yet another robot platform”. In: *International Journal of Advanced Robotic Systems* 3.1 (2006), pp. 043–048. ISSN: 17298806. DOI: [10.5772/5761](https://doi.org/10.5772/5761).
- [89] H. Bruyninckx. “Open robot control software: The OROCOS project”. In: *Proceedings - IEEE International Conference on Robotics and Automation* 3 (2001), pp. 2523–2528. ISSN: 10504729. DOI: [10.1109/robot.2001.933002](https://doi.org/10.1109/robot.2001.933002).
- [90] Herman Bruyninckx, Peter Soetens, and Bob Koninckx. “The real-time motion control core of the OROCOS project”. In: *Proceedings - IEEE International Conference on Robotics and Automation* 2 (2003), pp. 2766–2771. ISSN: 10504729. DOI: [10.1109/robot.2003.1242011](https://doi.org/10.1109/robot.2003.1242011).
- [91] David Doose, Christophe Grand, and Charles Lesire. “MAUVE runtime: A component-based middleware to reconfigure software architectures in real-time”. In: *Proceedings - 2017 1st IEEE International Conference on Robotic Computing, IRC 2017* 1 (2017), pp. 208–211. DOI: [10.1109/IRC.2017.47](https://doi.org/10.1109/IRC.2017.47).



- [92] Noriaki Ando et al. “Software deployment infrastructure for component based RT-systems”. In: *Journal of Robotics and Mechatronics* 23.3 (2011), pp. 350–359. ISSN: 18838049. DOI: [10.20965/jrm.2011.p0350](https://doi.org/10.20965/jrm.2011.p0350).
- [93] Luca Muratore et al. “XBotCore: A real-time cross-robot software platform”. In: *Proceedings - 2017 1st IEEE International Conference on Robotic Computing, IRC 2017* April (2017), pp. 77–80. DOI: [10.1109/IRC.2017.45](https://doi.org/10.1109/IRC.2017.45).
- [94] Harrisson Fischer et al. “An industrial standard based control architecture for multi-robot real time coordination”. In: *IEEE International Conference on Industrial Informatics (INDIN)* 0 (2016), pp. 207–212. ISSN: 19354576. DOI: [10.1109/INDIN.2016.7819160](https://doi.org/10.1109/INDIN.2016.7819160).
- [95] H. Fischer et al. “RTRobMultiAxisControl: A Framework for Real-Time Multiaxis and Multirobot Control”. In: *IEEE Transactions on Automation Science and Engineering* 16.3 (2019), pp. 1205–1217. ISSN: 15583783. DOI: [10.1109/TASE.2018.2889813](https://doi.org/10.1109/TASE.2018.2889813).
- [96] Kenji Kaneko et al. “Humanoid robot HRP-2”. In: *Proceedings - IEEE International Conference on Robotics and Automation* 2004.2 (2004), pp. 1083–1090. ISSN: 10504729. DOI: [10.1299/jsmemag.109.1051\\_438](https://doi.org/10.1299/jsmemag.109.1051_438).
- [97] Stefano Dafarra et al. “A Control Architecture with Online Predictive Planning for Position and Torque Controlled Walking of Humanoid Robots”. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Oct. 2018, pp. 1–9. DOI: [10.1109/IROS.2018.8594277](https://doi.org/10.1109/IROS.2018.8594277). URL: <https://ieeexplore.ieee.org/document/8594277>.
- [98] KangKyu Lee et al. “Implementing Full-body Torque Control in Humanoid Robot with High Gear Ratio Using Pulse Width Modulation Voltage”. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Oct. 2018, pp. 726–732. DOI: [10.1109/IROS.2018.8593908](https://doi.org/10.1109/IROS.2018.8593908). URL: <https://ieeexplore.ieee.org/document/8593908>.
- [99] George Mesesan et al. “Dynamic Walking on Compliant and Uneven Terrain using DCM and Passivity-based Whole-body Control”. In: *2019 IEEE-RAS*

- 19th International Conference on Humanoid Robots (Humanoids)* (2019), pp. 25–32. DOI: [10.1109/Humanoids43949.2019.9035053](https://doi.org/10.1109/Humanoids43949.2019.9035053).
- [100] Noëlie Ramuzat et al. “Actuator Model, Identification and Differential Dynamic Programming for a TALOS Humanoid Robot”. In: *2020 European Control Conference (ECC)* (2020), pp. 724–730. DOI: [10.23919/ecc51009.2020.9143817](https://doi.org/10.23919/ecc51009.2020.9143817).
- [101] N. Paine et al. “Actuator Control for the NASA-JSC Valkyrie Humanoid Robot: A Decoupled Dynamics Approach for Torque Control of Series Elastic Robots”. In: *Journal of Field Robotics* 32 (2015). DOI: [10.1002/rob.21556](https://doi.org/10.1002/rob.21556).
- [102] Yangwei You et al. “Straight leg walking strategy for torque-controlled humanoid robots”. In: *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)* (2016), pp. 2014–2019. DOI: [10.1109/ROBIO.2016.7866625](https://doi.org/10.1109/ROBIO.2016.7866625).
- [103] Mingxing Liu, Ryan Lober, and Vincent Padois. “Whole-body hierarchical motion and force control for humanoid robots”. en. In: *Autonomous Robots* 40.3 (Mar. 2016), pp. 493–504. ISSN: 1573-7527. DOI: [10.1007/s10514-015-9513-5](https://doi.org/10.1007/s10514-015-9513-5).
- [104] Usine Digitale. “Xiaomi présente CyberOne, un robot humanoïde”. fr. In: (Aug. 2022). URL: <https://www.usine-digitale.fr/article/xiaomi-presente-cyberone-un-robot-humanoide.N2033982>.
- [105] en. URL: <https://www.generationrobots.com/en/404154-h1-humanoid-robot.html>.
- [106] Jamie K. Paik et al. “Development of an Anthropomorphic Robotic Arm and Hand for Interactive Humanoids”. en. In: *Journal of Bionic Engineering* 9.2 (June 2012), pp. 133–142. ISSN: 2543-2141. DOI: [10.1016/S1672-6529\(11\)60107-8](https://doi.org/10.1016/S1672-6529(11)60107-8).

- [107] Ming-Hsun Chiang and Fan-Ren Chang. “Anthropomorphic Design of the Human-Like Walking Robot”. en. In: *Journal of Bionic Engineering* 10.2 (June 2013), pp. 186–193. ISSN: 2543-2141. DOI: [10.1016/S1672-6529\(13\)60214-0](https://doi.org/10.1016/S1672-6529(13)60214-0).
- [108] Michiel Plooij et al. “Lock Your Robot: A Review of Locking Devices in Robotics”. In: *IEEE Robotics and Automation Magazine* 22.1 (Mar. 2015), pp. 106–117. ISSN: 1558-223X. DOI: [10.1109/MRA.2014.2381368](https://doi.org/10.1109/MRA.2014.2381368).
- [109] Jerry E. Pratt and Benjamin T. Krupp. “Series Elastic Actuators for legged robots”. In: *Unmanned Ground Vehicle Technology VI*. Vol. 5422. SPIE, Sept. 2004, pp. 135–144. DOI: [10.1117/12.548000](https://doi.org/10.1117/12.548000). URL: <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/5422/0000/Series-Elastic-Actuators-for-legged-robots/10.1117/12.548000.full>.
- [110] B. Vanderborght et al. “Variable impedance actuators: A review”. In: *Robotics and Autonomous Systems* 61.12 (Dec. 2013), pp. 1601–1614. ISSN: 0921-8890. DOI: [10.1016/j.robot.2013.06.009](https://doi.org/10.1016/j.robot.2013.06.009).
- [111] Jacob Reher, Wen-Loong Ma, and Aaron D. Ames. “Dynamic Walking with Compliance on a Cassie Bipedal Robot”. In: *2019 18th European Control Conference (ECC)*. June 2019, pp. 2589–2595. DOI: [10.23919/ECC.2019.8796090](https://doi.org/10.23919/ECC.2019.8796090). URL: <https://ieeexplore.ieee.org/abstract/document/8796090>.
- [112] URL: <https://agilityrobotics.com/news/2023/this-is-digit-made-for-work>.
- [113] Tianyi Ko et al. “Compliant Biped Locomotion of Hydra, an Electro-Hydrostatically Driven Humanoid”. In: *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*. IEEE, Nov. 2018, pp. 280–283. ISBN: 978-1-5386-7283-9. DOI: [10.1109/HUMANOIDS.2018.8624973](https://doi.org/10.1109/HUMANOIDS.2018.8624973). URL: <https://ieeexplore.ieee.org/document/8624973/>.

- [114] Maya Sleiman. “New actuation technologies for humanoid robotics and assistive devices”. These de doctorat. université Paris-Saclay, Sept. 2022. URL: <https://theses.fr/2022UPAST110>.
- [115] Chun-Che Huang and Andrew Kusiak. “Modularity in design of products and systems”. In: *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 28.1 (1998), pp. 66–77.
- [116] Gunnar Erixon. *Modular Function Deployment: A Method for Product Modularisation*. Google-Books-ID: keWTtgAACAAJ. Royal Inst. of Technology, Department of Manufacturing Systems, Assembly Systems Division, 1998. ISBN: 978-9925-7360-5-8.
- [117] Gerhard Pahl et al. *Engineering Design: A Systematic Approach*. Google-Books-ID: 57aWTCE3gE0C. Springer Science and Business Media, Aug. 2007. ISBN: 978-1-84628-319-2.
- [118] Anthony Esposito. *Fluid power with applications*. Prentice Hall Upper Saddle River, New Jersey, 2000.
- [119] George E. Totten. *Handbook of hydraulic fluid technology*. CRC press, 2011.
- [120] Mituso Komagata, Tianyi Ko, and Yoshihiko Nakamura. “Small Size Hydraulic Pumps with Low Heat Generation for Electro Hydrostatic Actuation of Humanoid Robots”. In: *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*. IEEE, Nov. 2018, pp. 1–6. ISBN: 978-1-5386-7283-9. DOI: [10.1109/HUMANOIDS.2018.8625055](https://doi.org/10.1109/HUMANOIDS.2018.8625055). URL: <https://ieeexplore.ieee.org/document/8625055/>.
- [121] Kawasaki. *No. 181 Precision Machinery | Kawasaki Heavy Industries, Ltd.* URL: <https://global.kawasaki.com/en/corp/rd/magazine/181/index.html>.
- [122] arnaldo Leal. *Series Elastic Actuator: Design, Analysis and Comparison - Google Scholar*. URL: [https://scholar.google.fr/scholar?hl=fr&as\\_sdt=0%5C%C5&q=Series+Elastic+Actuator%5C%3A+Design%5C%2C+Analysis+and+Comparison&btnG=](https://scholar.google.fr/scholar?hl=fr&as_sdt=0%5C%C5&q=Series+Elastic+Actuator%5C%3A+Design%5C%2C+Analysis+and+Comparison&btnG=).

- [123] Liu Baojun et al. *Integrated electro-hydraulic actuator*. June 2011. URL: <https://patents.google.com/patent/W02011072502A1/en>.
- [124] Matthew Olson and James Prazak. *Electro-hydraulic actuator*. Dec. 2011. URL: <https://patents.google.com/patent/US20110289912A1/en?q=US20110289912A1>.
- [125] Ghiath Abdulmalek. “Electronics and Control Development of Servo Electro-Hydraulic Actuators (SEHA)”. These de doctorat. université Paris-Saclay, Dec. 2023.
- [126] Samer ALFAYAD. “Robot humanoïde HYDROïD : actionnement, structure cinématique et stratégie de contrôle”. PhD thesis. PhD thesis, Université de Versailles St Quentin en Yvelines, 2009.
- [127] Diego Torricelli et al. “Human-like compliant locomotion: State of the art of robotic implementations”. In: *Bioinspiration and Biomimetics* 11.5 (2016). ISSN: 17483190. DOI: [10.1088/1748-3190/11/5/051002](https://doi.org/10.1088/1748-3190/11/5/051002).
- [128] Martin Rostan. “High Speed Industrial Ethernet for Semiconductor Equipment”. en. In: (2004).
- [129] Jérémy Robert et al. “Minimum Cycle Time Analysis of Ethernet-Based Real-Time Protocols”. en. In: *International Journal of Computers Communications and Control* 7.4 (Sept. 2014), p. 744. ISSN: 1841-9844, 1841-9836. DOI: [10.15837/ijccc.2012.4.1372](https://doi.org/10.15837/ijccc.2012.4.1372).
- [130] Jeremy H Brown and Brad Martin. “How fast is fast enough? Choosing between Xenomai and Linux for real-time applications”. In: *Proceedings of the 12th Real-Time Linux Workshop (RTLWS'12)* (2012), p. 17. URL: <https://www.osadl.org/fileadmin/dam/rtlws/12/Brown.pdf>.
- [131] C++. Sept. 2023. URL: [https://github.com/orocos/rtt\\_ros\\_integration](https://github.com/orocos/rtt_ros_integration).
- [132] C++. July 2022. URL: [https://github.com/orocos/rtt\\_soem](https://github.com/orocos/rtt_soem).
- [133] URL: <https://www.rls.si/eng/linace-absolute-linear-shaft-encoder>.

- [134] *Website cyclicttest - High resolution test program*. <http://manpages.ubuntu.com/manpages/cosmic/man8/cyclicttest.8.html>. Accessed: 2020-01-28.
- [135] Lennart Puck et al. “Performance Evaluation of Real-Time ROS2 Robotic Control in a Time-Synchronized Distributed Network”. In: Aug. 2021, pp. 1670–1676. DOI: [10.1109/CASE49439.2021.9551447](https://doi.org/10.1109/CASE49439.2021.9551447).

# Appendix A

## SEHA Control System Design Documents

### A.1 Sequence of SEHA Operations

## SEQUENCE OF OPERATIONS

### 1 Introduction

This document describes the control system operations while running the Dev-Kit. This description defines the responsibility for each part and relationship between the different elements of the system to have a full operation.

### 2 Objective

To have a clear vision about how each part of our system will act during functional operations of the system, that will lead to predefine the requirements like the firmware capability (modes of control), the EtherCAT frame structure (bus variables), the master capability (to operate the control mode), and the GUI capability as well

### 3 Prerequisite



The sequence of operation starting after ensuring everything is powered up and the communication established, the main steps to follow are the following:

#### 3.1 Power up the Dev-Kit

Connect the power supply unit to the power source 220 VAC, connect the pump power cable as well.

#### 3.2 Link up the Dev-Kit

Ensure the EtherCAT connection is established between the PC and the SIC board, as well between the SIC board and the MGH board in the following order:

PC  SIC Board  MGH Board



Make sure of the port sequencing from one device's output port to the next device's input port

### 3.3 Run the software

Run KALYSTA Studio software, Scan the network and select the Dev-Kit product to be

## 4 Sequence of Operations

### 4.1 Read states

Mainly to examine the direction of the applied force on the Dev-Kit, whatever the user installed the springs or hang the load.

### 4.2 Operate the servo-valve

By pressing the start button of the MGH from the GUI and based on the force direction, the servovalve will be directed to release the load and reset the force. If the force was caused by the springs, directing the servovalve will lead the rod to go to zero position. If the force was caused by the load, directing the servovalve will lead the rod to go to the end of stroke position and the load will fell down.

### 4.3 Starting the MGH

When the force released, the pump will start accelerating until it reaches the initial speed, the initial speed value is **600 RPM** configured in the MGH firmware. The user has no access to modify it but can increase the speed from the GUI to match the required flow and pressure. The user will have a table for the speeds and the corresponding pressures and flows.

The user should set the desired speed before operating the SIC with one of its modes of control, the system will be disabling the selected mode of control until the MGH reach the steady state of the adjusted desired speed.

### 4.4 Operating the SIC

The user can select one of the modes of control from the GUI, when the user changes the mode of control the system will reset the mode of control before enabling the selected mode of control and it will reset all the command variables that used in multiple mode of control.

## 4.5 Stop the SIC

When the user presses the stop button of the SIC from the GUI, the system will do the following:

- Disabling the control mode (**close the servo valve**).
- Decelerating the MGH speed to its initial value **600 RPM**

## 4.6 Stop the MGH

When the user presses the stop button of the MGH from the GUI, the system will do the following:

- Decelerating the MGH to **zero RPM**
- Examine the applied force and enable the servo valve to release the force

## 4.7 Emergency state

In case of the physical **emergency button** is pressed, the **GUI emergency button**, or the **lid** is opened:

- the MGH will be decelerated with its maximum deceleration value (the acceleration and deceleration values will be determined in the MGH firmware) the user has no access to modify these values).
- The servovalve will be closed to hold the applied force

## 4.8 Reset emergency state

Reset the emergency state need to reset the physical emergency button if it is pressed, and press the GUI reset button, the system will do the following

Examine the force direction and operate the servovlave to reset the applied force

Enabling starting the MGH as described in [4.3](#)

## 5 Conclusion

This document specifies the required action in every system state and organizes how these actions will take place. Starting from this document, a **schedule of control point** could be created to specify the soft and hard points and their corresponding variables.

## A.2 SEHA Control Mode Selection

### A.2.1 Control Mode Description

#### Velocity Control

When powering up the board, the M2 and M3 should reset and then go to the initial values that are set by their corresponding variables.

When the index variable is set to 1, the velocity control mode will be activated and the pump will start accelerating until it reaches the initial speed, the initial speed value is 600 RPM configured in the MGH firmware. The user has no access to modify it but can increase the speed setting of the Data1 variable. Controlling the M2 and M3 occurred by setting the position

The acceleration and the deceleration values should be configured in the firmware. The Acceleration value is 5 sec from zero to reach the operational speed (5000 RPM). The Deceleration value is 5 sec from the operational speed to zero speed.

When the emergency flag is activated by the frame variable or by one of the GPIO inputs, the pump should stop immediately, the M2 value should reset, and the M3 position return to zero as well; the reset should done by resetting the GPIO input, and the emergency variable By resetting: the M2 and M3 will return to their values that were received by the frame variables.

#### Position Control

When powering up the board, the M2 and M3 should reset and then go to the initial values that are set by their corresponding variables.

When the index variable is set to 2, the Position control mode will be activated and the motor will consider the current position as an initial position link it to zero value, and break the motor on this position until a new position set by the Data1 variable

The acceleration and the deceleration values should be configured in the firmware. The Acceleration value is: 5 sec from zero to reach the desired position. The Deceleration value is: 5 sec from the operational speed to reach the desired position.

When the emergency flag is activated by the frame variable or by one of the GPIO inputs, the pump should stop immediately, the M2 value should reset, and the M3 position return to zero as well; the reset should be done by resetting the GPIO input, and the emergency variable. By resetting: the M2 and M3 will return to their values that were received by the frame variables.

Product:	MGH	Document Name:	Firmware Performance	By :	Subhi Jleilaty
Rev.:	01	Document N.:	1.1.3	Date:	10/05/2023

N°	Mode of Control	Control mode selection information				EtherCAT frame information				
		Parameters	Variable	Range	Unit	Variable data type	Data Object type	M/O/C	Comments	
1	Velocity Control	M1 Activation	Index	0 is OFF, 1 is ON	..	Int	PDO	M	Fixed in the frame	
		RPM Value	Data1	-5000 to 5000	RPM	Float	PDO	M		
		N/A	Data2	0 is OFF, 1 is ON	..	Float	PDO	M		
		Emergency	Emergency	0 is OFF, 1 is ON	..	Bool	PDO	M		
		N/A	M1kp		..	Float	PDO	O		
		N/A	M1ki		..	Float	PDO	O		
		N/A	M1kd		..	Float	PDO	O		
		M3 Desired Position	M3Position	0 is OFF, 1 is Position, 2 is Current	mm	Float	PDO	M		Fixed in the frame
		M2 Activation	M2Index		..	Int	PDO	M		
		M2 Desired Value (Position or Current)	M2Value	Position 0 to 11 Current -10 to 10	mm mAmps	Float	PDO	M		Position for outer control loop Current for inner control loop
M2 PID Parameter	M2kp		..	Float	PDO	O	To be written once upon the starting we can stop using these variables after we define the parameters			
M2 PID Parameter	M2ki		..	Float	PDO	O				
M2 PID Parameter	M2kd		..	Float	PDO	O				
2	Position Control	M1 Activation	Index	0 is OFF, 2 is ON	..	Int	PDO	M	Not used in this control mode	
		Desired Position Value	Data1	0 to 360000	mDegree	Float	PDO	M		
		N/A	Data2	0 is OFF, 1 is ON	..	Float	PDO	M		
		Emergency	Emergency	0 is OFF, 1 is ON	..	Bool	PDO	M		
		PID Parameter	M1kp		..	Float	PDO	O		
		PID Parameter	M1ki		..	Float	PDO	O		
		PID Parameter	M1kd		..	Float	PDO	O		
		M3 Desired Position	M3Position	0 is OFF, 1 is Position, 2 is Current	mm	Float	PDO	M		To be written once upon the starting we can stop using these variables after we define the parameters
		M2 Activation	M2Index		..	Int	PDO	M		
		M2 Desired Value (Position or Current)	M2Value	Position 0 to 11 Current -10 to 10	mm mAmps	Float	PDO	M		
M2 PID Parameter	M2kp		..	Float	PDO	O	To be written once upon the starting we			
M2 PID Parameter	M2ki		..	Float	PDO	O				

	M2 PID Parameter	M2Kd			..	Float	PDO	O	can stop using these variables after we define the parameters
3	M1 Activation	Index			..				
	Desired Torque Value	Data1	0 is OFF 3 is ON 0 to 20000		mN/m	Int	PDO	M	
	N/A	Data2			..	Float	PDO	M	Not used in this control mode
	Emergency		0 is OFF 1 is ON		..	Bool	PDO	M	
	PID Parameter	M1Kp			..	Float	PDO	O	To be written once upon the starting we can stop using these variables after we define the parameters
	PID Parameter	M1Ki			..	Float	PDO	O	
	PID Parameter	M1Kd			..	Float	PDO	O	
	M3 Desired Position	M3Position			mm	Float	PDO	M	
	M2 Activation	M2Index	0 is OFF, 1 is Position, 2 is Current		..	Int	PDO	M	Fixed in the frame
	M2 Desired Value (Position or Current)	M2Value	Position 0 to 11 Current -10 to 10		mm mAmps	Float	PDO	M	
	M2 PID Parameter	M2Kp			..	Float	PDO	O	To be written once upon the starting we can stop using these variables after we define the parameters
	M2 PID Parameter	M2Ki			..	Float	PDO	O	
M2 PID Parameter	M2Kd			..	Float	PDO	O		
4	M1 Activation	Index			..	Int	PDO	M	
	Desired Position Value	Data1	0 is OFF 4 is ON 0 to 360000		mDegree	Float	PDO	M	Fixed in the frame
	Torque Limit Value	Data2	0 to 20000		mN/m	Float	PDO	M	
	Emergency		0 is OFF 1 is ON		..	Bool	PDO	M	
	PID Parameter	M1Kp			..	Float	PDO	O	To be written once upon the starting we can stop using these variables after we define the parameters
	PID Parameter	M1Ki			..	Float	PDO	O	
	PID Parameter	M1Kd			..	Float	PDO	O	
	M3 Desired Position	M3Position			mm	Float	PDO	M	
	M2 Activation	M2Index	0 is OFF, 1 is Position, 2 is Current		..	Int	PDO	M	Fixed in the frame
	M2 Desired Value (Position or Current)	M2Value	Position 0 to 11 Current -10 to 10		mm mAmps	Float	PDO	M	
	M2 PID Parameter	M2Kp			..	Float	PDO	O	To be written once upon the starting we can stop using these variables after we define the parameters
	M2 PID Parameter	M2Ki			..	Float	PDO	O	
M2 PID Parameter	M2Kd			..	Float	PDO	O		
5	M1 Activation	Index			..	Int	PDO	M	
	Desired Position Value	Data1	0 is OFF ... is ON 0 to 360000		mDegree	Float	PDO	M	Fixed in the frame
	Velocity Limit Value	Data2	-5000 to 5000		RPM	Float	PDO	M	
	Emergency		0 is OFF 1 is ON		..	Bool	PDO	M	
	PID Parameter	M1Kp			..	Float	PDO	O	To be written once upon the starting we can stop using these variables
	PID Parameter	M1Ki			..	Float	PDO	O	
	PID Parameter	M1Kd			..	Float	PDO	O	
	M3 Desired Position	M3Position			mm	Float	PDO	M	
	M2 Activation	M2Index	0 is OFF, 1 is Position, 2 is Current		..	Int	PDO	M	Fixed in the frame
	M2 Desired Value (Position or Current)	M2Value	Position 0 to 11 Current -10 to 10		mm mAmps	Float	PDO	M	
	M2 PID Parameter	M2Kp			..	Float	PDO	O	To be written once upon the starting we can stop using these variables after we define the parameters
	M2 PID Parameter	M2Ki			..	Float	PDO	O	
M2 PID Parameter	M2Kd			..	Float	PDO	O		
	M1 Activation	Index			..	Int	PDO	M	

6	Torque and velocity Control	Desired Torque Value	Data1	0 to 20000	mN/m	Float	PDO	M	Fixed in the frame	
		Velocity Limit Value	Data2	-5000 to 5000	RPM	Float	PDO	M		
		Emergency	Emergency	0 is OFF 1 is ON		Bool	PDO	M		
		PID Parameter	M1Kp		..	Float	PDO	O		To be written once upon the starting we can stop using these variables
		PID Parameter	M1Ki		..	Float	PDO	O		
		PID Parameter	M1Kd		..	Float	PDO	O		
		M3 Desired Position	M3Position		mm	Float	PDO	M		
		M2 Activation	M2Index	0 is OFF, 1 is Position , 2 is Current	..	Int	PDO	M		Fixed in the frame
		M2 Desired Value (Position or Current)	M2Value	Position 0 to 11 Current -10 to 10	mm mAmps	Float	PDO	M		
		M2 PID Parameter	M2Kp		..	Float	PDO	O		To be written once upon the starting we can stop using these variables after we define the parameters
		M2 PID Parameter	M2Ki		..	Float	PDO	O		
		M2 PID Parameter	M2Kd		..	Float	PDO	O		

### **A.3 GUI Performance and Requirements**



## GUI PERFORMANCE

### 1 Introduction

This document describes the GUI performance while operating the Dev-Kit. This description defines the functionalities of the used displays and their areas, button, and to operate this application.

### 2 Objective

To have a clear vision about how each tool of the GUI will act during functional operations of the system, that will lead to predefine the requirements like the front-end development, the back-end development (ROS nodes and OROCOS Components), the EtherCAT frame structure (bus variables).

### 3 Prerequisite

The GUI operation starting after ensuring everything is powered up and the communication established, the main steps to follow are the following:

#### 3.1 Power up the Dev-Kit

Connect the power supply unit to the power source 220 VAC, connect the pump power cable as well.

#### 3.2 Link up the Dev-Kit

Ensure the EtherCAT connection is established between the PC and the SIC board, as well between the SIC board and the MGH board in the following order:

PC  $\Rightarrow$  SIC Board  $\Rightarrow$  MGH Board

Make sure of the port sequencing from one device's output port to the next device's input port

#### 3.3 Run the software

Run KALYSTA Studio software (GUI), Scan the network and select the **Dev-Kit** product to be

## 4 GUI Performance

### 4.1 Home display

#### Home display windows description

**Scan window:** this window will print out the list of the existed devices on the network

**Application window:** this window will print out only the connected devices selected by the user (corresponding the user application)

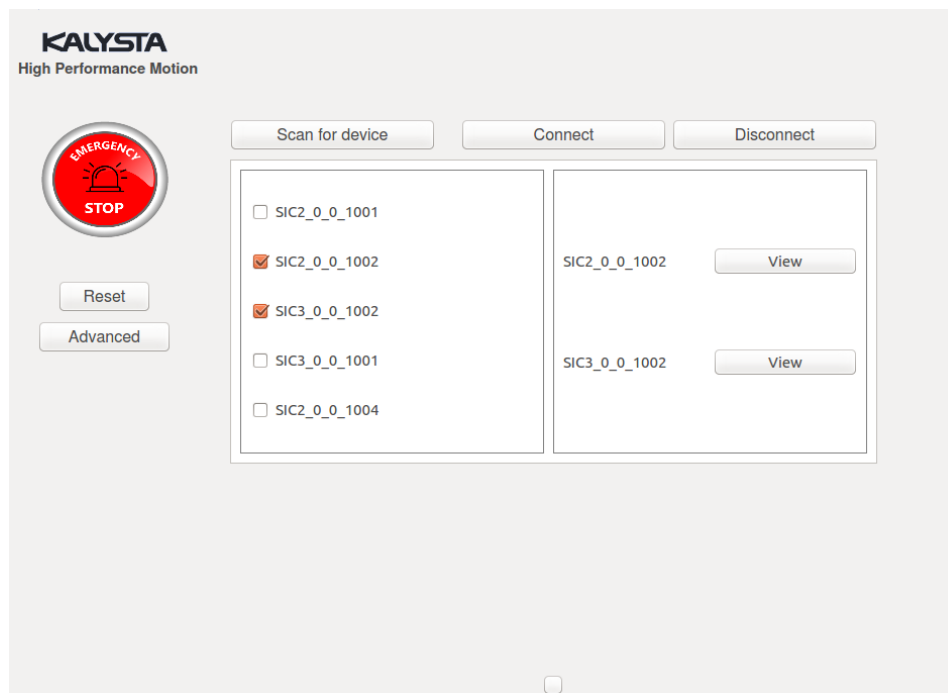


Figure 1 Home display

#### Home display button description

**Scan button:** to show the connected devices to the network and display them in list accompanying each device with check box

When it pressed: it will invoke the slaveinfo program in the background through the terminal and saves the connected slaves in a vector (their name and the correspondant address) then it displays in a list the available devices connected to the pc.

If no slaves are connected to the pc : "No device scanned" will appear on the screen.

**Connect button:** to enable the selected slaves to be controlled by the GUI

When it pressed: it will prepare a list of the selected slaves and print them out in the application window with a View button to each slave. It will examine the type of the slave (SIC, MGH, or SEHA) and link the view button with the corresponding display.

**View button:** when clicked , it will open a new window for the chosen device and it will publish the address of the devices on a new topic so we can access it in the productwindow and work with the right topics

**Disconnect button:** to disable the selected slaves to be controlled by the GUI

When it pressed: it will reset all the control signals of the GUI to the initial state and remove the disconnected devices of the of the application window.

**Emergency stop button:** to activate the emergency procedure and stop the application.

When it pressed: it will reset all the control signals coming from the GUI.

**Reset button:** to return the application control signals to the last values before the emergency stop was pressed.

**Advanced button:** this button will show the user the run time data as I/O variables with their values.

**View button:** next to each connected device, this button will pop up the device corresponding display

## 4.2 SIC display

### SIC display windows description

**Control mode window:** this window contains all the necessary commands to select mode of control and modify its parameters

**SIC window:** this window has the SIC picture in its background and the sensor readings located approximately as the actual location on the SIC for better understanding of the application.

**Logging window:** for printing the state of the product, the warning messages, and the alarms if existed.



Figure 2 SIC Display

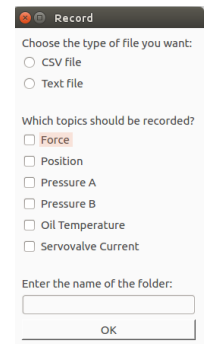
### SIC display button description

**Display button:** this button will pop up the GUI of the `rqt_plot` package (ROS package for monitoring the run time I/O signals).

**Reset button:** this button allows the user to reset the error, warning, and the emergency procedures, and reset the control signals as well to the initial state.

**Record Button:** this button will launch the ROS bag to be used as a data logger.

When pressed: a small window will appear, demanding the user the type of file they want, and what information they would like to save in .csv files. Also, it asks the user to give a file name where he can find the files + the rosbag file. This folder is then created inside the `/bagfiles` folder (directory).



**Visualize Button:** this button opens a terminal and launches "PlotJuggler", an application where he can visualize the data he recorded (using the rosbag file or the .csv files). => better when the user uses the rosbag files because we have the real time on the x axis of the file.

Next to the emergency stop button: the name of the slave + the address is displayed.

### 4.3 MGH Display

#### MGH display windows description

**Control mode window:** this window contains all the necessary commands to select mode of control and modify its parameters

**MGH window:** this window has the MGH picture in its background and the sensor readings located approximately as the actual location on the MGH for better understanding of the application.

**Logging window:** for printing the state of the product, the warning messages, and the alarms if existed.

The **SystemState** variable (EtherCAT frame variable) should be examined to detect the warning and the alarm states of the system as the following:

SystemState.00 = 1 then Emergency State: Print out (Emergency State)

SystemState.01 = 1 then Temperature range: Print out (Temperature sensor out of range)

SystemState.02 = 1 then Pressure range: Print out (Pressure sensor out of range)

SystemState.03 = 1 then Position range: Print out (Position sensor out of range)

SystemState.04 = 1 then Oil Temperature range: Print (Oil Temperature sensor out of range)

SystemState.05 = 1 then Current: Print out (Over current)

SystemState.06 = 1 then Voltage: Print out (Over voltage)

SystemState.07 = 1 then Reserved: Print out (Reserved)

SystemState.08 = 1 then Reserved: Print out (Reserved)

SystemState.09 = 1 then Reserved: Print out (Reserved)

SystemState.10 = 1 then Reserved: Print out (Reserved)

SystemState.11= 1 then Reserved: Print out (Reserved)

SystemState.12 = 1 then Reserved: Print out (Reserved)

SystemState.13 = 1 then Reserved: Print out (Reserved)

SystemState.14 = 1 then Reserved: Print out (Reserved)

SystemState.15 = 1 then Reserved: Print out (Reserved)

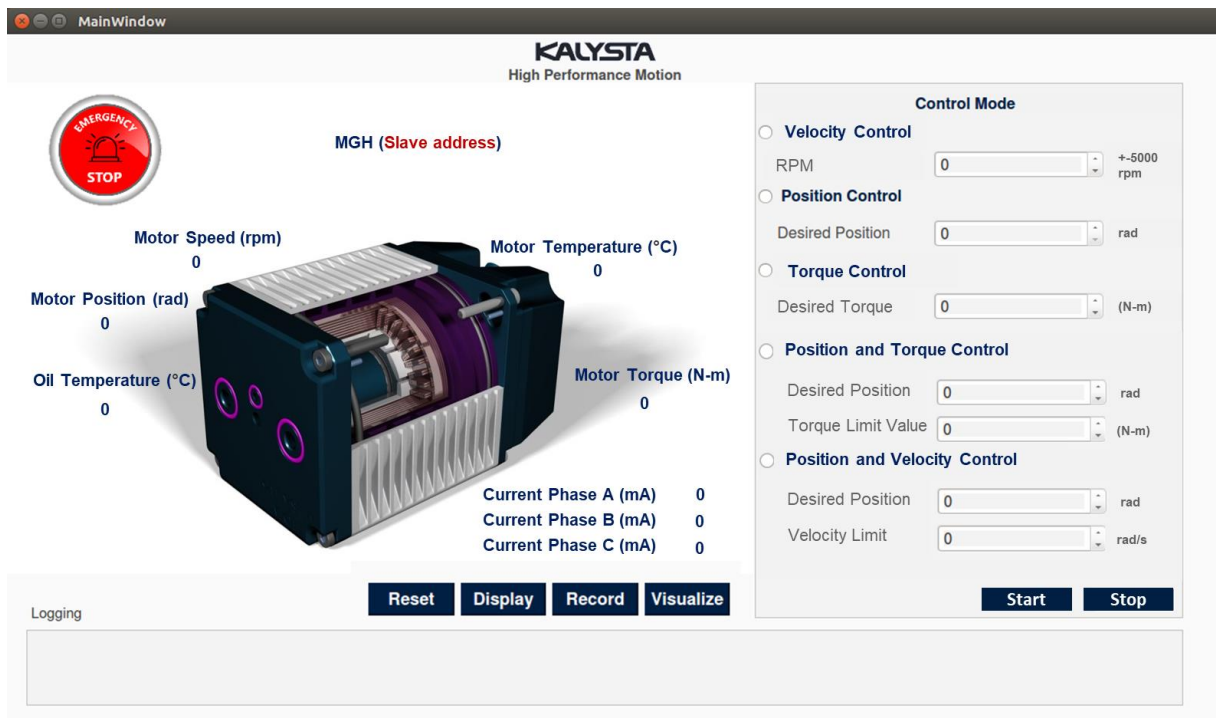


Figure 3 MGH Display

### MGH display button description

**Start button:** this button will set the **index** variable (EtherCAT frame variable) to 1

**Stop button:** this button will set the **index** variable (EtherCAT frame variable) to 0

**Emergency stop button:** to activate the emergency procedure and stop the application.

When it pressed: set the **Emergency** variable (EtherCAT frame variable) to 1 and reset all the control signals coming from the GUI.

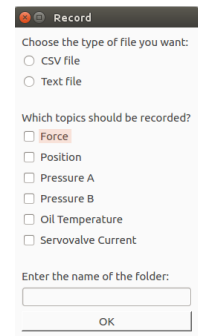
**Reset button:** this button allows the user to reset the error, warning, and the emergency procedures, and reset the control signals as well to the initial state.

When it pressed: set the **Emergency** variable (EtherCAT frame variable) to 0

**Display button:** this button will pop up the GUI of the **rqt\_plot** package (ROS package for monitoring the run time I/O signals).

**Record Button:** this button will launch the ROS bag to be used as a data logger.

When pressed: a small window will appear, demanding the user the type of file they want, and what information they would like to save in .csv files. Also , it asks the user to give a file name where he can find the files + the rosbag file. This folder is then created inside the /bagfiles folder (directory).



**Visualize Button:** this button open a terminal and launches “PlotJuggler” , an application where he can visualize the data he recorded (using the rosbag file or the .csv files). => better when the user uses the rosbag files because we have the real time on the x axis of the file.

In **the top-middle** of the display: the name of the slave + the address is displayed.

## 5 Conclusion

This document defines the GUI performance functionalities of the used displays and their areas, button, and to operate this application. It provides an understandable requirement to build the GUI and ensures the integration with the system.





# Appendix B

## SIC Control System Design Documents

### B.1 SIC Control Mode Selection

Product:	MGH	Document Name:	Firmware Performance	By:	Subhi Jleilaty
		Rev.:	01	Date:	10/05/2023
			Document N:	1.1.3	

Parameter information							EtherCAT frame information		
N°	Mode of Control	Parameters	Variable	Range	Unit	Variable data type	Data Object type	M/O/C	Comments
1	Direct Control	Activation	Index	0 is OFF 1 is ON	..	Int	PDO	M	Fixed in the frame
		Current Value	Data1	-10 to 10	mA	Float	PDO	M	
		N/A	Data2		..	Float	PDO	M	
		AccuracyForce	AccuracyForce	0 to 1000	..	Float	PDO	O	Not used in this control mode
		N/A	Kp		..	Float	PDO	O	
		N/A	Ki		..	Float	PDO	O	
		N/A	Kd		..	Float	PDO	O	
2	Position Control	Activation	Index	0 is OFF 2 is ON	..	Int	PDO	M	Not used in this control mode
		Desired Position Value	Data1	0 to 300000	µm	Float	PDO	M	
		N/A	Data2		..	Float	PDO	M	
		AccuracyForce	AccuracyForce	0 to 1000	..	Float	PDO	O	To be written once upon the starting we can stop using these variables after we define the parameters
		PID Parameter	Kp		..	Float	PDO	O	
		PID Parameter	Ki		..	Float	PDO	O	
		PID Parameter	Kd		..	Float	PDO	O	
3	Modified Position Control	Activation	Index	0 is OFF 3 is ON	..	Int	PDO	M	Not used in this control mode
		Desired Position Value	Data1	0 to 300000	µm	Float	PDO	M	
		N/A	Data2		..	Float	PDO	M	
		AccuracyForce	AccuracyForce	0 to 1000	..	Float	PDO	O	To be written once upon the starting we can stop using these variables after we define the parameters
		PID Parameter	Kp		..	Float	PDO	O	
		PID Parameter	Ki		..	Float	PDO	O	
		PID Parameter	Kd		..	Float	PDO	O	
4	Force Control	Activation	Index	0 is OFF 4 is ON	..	Int	PDO	M	Not used in this control mode
		Desired Force Value	Data1	-10 to 10	kN	Float	PDO	M	
		N/A	Data2		..	Float	PDO	M	
		AccuracyForce	AccuracyForce	0 to 1000	..	Float	PDO	O	

		PID Parameter	Kp		..	Float	PDO	O	To be written once upon the starting we can stop using these variables after we define the parameters
		PID Parameter	Ki		..	Float	PDO	O	
		PID Parameter	Kd		..	Float	PDO	O	
		Activation	Index	0 is OFF 5 is ON	..	Int	PDO	M	
5	Position and Force Control	Desired Position Value	Data1	0 to 30000	µm	Float	PDO	M	
		Force Limit Value	Data2	-10 to 10	kN	Float	PDO	M	
		AccuracyForce	AccuracyForce	0 to 1000	..	Float	PDO	O	To be written once upon the starting we can stop using these variables after we define the parameters
		PID Parameter	Kp		..	Float	PDO	O	
		PID Parameter	Ki		..	Float	PDO	O	
		PID Parameter	Kd		..	Float	PDO	O	
		Activation	Index	0 is OFF ... is ON	..	Int	PDO	M	
6	Direct Sinwave Control	Amplitude	Data1	-10 to 10	mA	Float	PDO	M	
		Frequency	Data2	0 to 1000	Hz	Float	PDO	M	To be written once upon the starting
		AccuracyForce	AccuracyForce	0 to 1000	..	Float	PDO	O	
		N/A	Kp		..	Float	PDO	O	Not used in this control mode
		N/A	Ki		..	Float	PDO	O	
		N/A	Kd		..	Float	PDO	O	
		Activation	Index	0 is OFF ... is ON	..	Int	PDO	M	
7	Direct Rectangle wave Control	Duty	Data1	0 to 100	µm	Float	PDO	M	
		Frequency	Data2	0 to 1000	Hz	Float	PDO	M	To be written once upon the starting
		AccuracyForce	AccuracyForce	0 to 1000	..	Float	PDO	O	we can stop using these variables after we define the parameters
		High Amplitude	Kp	-10 to 10	mA	Float	PDO	O	
		Low Amplitude	Ki	-10 to 10	mA	Float	PDO	O	
		N/A	Kd		..	Float	PDO	O	Not used in this control mode

