

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA E ELETRÔNICA
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA

João Pedro Volpato Goulart

**Desenvolvimento de uma unidade de atuação eletrônica para uma mão robótica
antropomórfica atuada por tendões**

Florianópolis

02/02/2026

João Pedro Volpato Goulart

**Desenvolvimento de uma unidade de atuação eletrônica para uma mão robótica
antropomórfica atuada por tendões**

Trabalho de Conclusão do Curso de Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal de Santa Catarina como requisito para a obtenção do título de Bacharel em Engenharia Elétrica.

Orientador: Prof. Saulo Güths, Dr.

Florianópolis

02/02/2026

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Volpato Goulart, João Pedro

Desenvolvimento de uma unidade de atuação eletrônica
para uma mão robótica antropomórfica atuada por tendões /
João Pedro Volpato Goulart ; orientador, Saulo Güths,
2026.

75 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Centro
Tecnológico, Graduação em Engenharia Elétrica,
Florianópolis, 2026.

Inclui referências.

1. Engenharia Elétrica. 2. Robótica antropomórfica. I.
Güths, Saulo. II. Universidade Federal de Santa Catarina.
Graduação em Engenharia Elétrica. III. Título.

João Pedro Volpato Goulart

**Desenvolvimento de uma unidade de atuação eletrônica para uma mão robótica
antropomórfica atuada por tendões**

Este Trabalho Conclusão de Curso foi julgado adequado para obtenção do Título de “Bacharel em Engenharia Elétrica” e aceito, em sua forma final, pelo Curso de Graduação em Engenharia Elétrica.

Florianópolis, 02 de fevereiro de 2026.



Documento assinado digitalmente
ROBERTO FRANCISCO COELHO
Data: 10/02/2026 08:42:28-0300
CPF: ***.034.249-**
Verifique as assinaturas em <https://v.ufsc.br>

Prof. Roberto Francisco Coelho, Dr.
Coordenador do Curso de Graduação em Engenharia Elétrica

Banca Examinadora:



Documento assinado digitalmente
Saulo Güths
Data: 04/02/2026 20:31:04-0300
CPF: ***.889.529-**
Verifique as assinaturas em <https://v.ufsc.br>

Prof. Saulo Güths, Dr.
Orientador
Universidade Federal de Santa Catarina



Documento assinado digitalmente
JULIO CESAR FRANTZ
Data: 09/02/2026 14:49:34-0300
CPF: ***.877.290-**
Verifique as assinaturas em <https://v.ufsc.br>

Prof. Julio Cesar Frantz, Dr.
Centro Universitário de Brusque



Documento assinado digitalmente
Cesar Ramos Rodrigues
Data: 06/02/2026 14:18:02-0300
CPF: ***.557.990-**
Verifique as assinaturas em <https://v.ufsc.br>

Prof. Cesar Ramos Rodrigues, Dr.
Universidade Federal de Santa Catarina

Este trabalho é dedicado aos meus pais, cujo apoio foi essencial para que eu chegasse até aqui.

AGRADECIMENTOS

Agradeço, primeiramente, aos meus pais, João e Vânia, que sempre me proporcionaram tudo o que estive ao seu alcance e não mediram esforços para que eu me tornasse quem sou hoje, oferecendo apoio, incentivo e exemplos que foram fundamentais ao longo de toda a minha formação.

À minha irmã Nina, pelo carinho, companheirismo e pela presença constante ao longo da minha vida. Ao meu irmão Vitor, que me inspira diariamente e tornou o percurso deste trabalho mais leve, doando seu tempo para me acompanhar.

Ao meu orientador, Saulo Güths, expresso minha profunda gratidão não apenas pela orientação neste trabalho, mas também pela mentoria e amizade construídas ao longo da graduação, contribuindo de forma decisiva para o meu crescimento acadêmico e pessoal.

À minha namorada Mariana, que estive ao meu lado tanto nas celebrações de vitória quanto nos momentos de dificuldade, mostrando, a cada dia, que a beleza da vida está em compartilhar caminhos, conquistas e aprendizados.

Aos meus amigos Eduardo, Paiva, Lukinhas e João, companheiros de mais de dez anos de jornada, que me ensinaram o verdadeiro valor de uma amizade sólida e duradoura.

Aos meus amigos da Apex, Felipe Jun e Bruninho, que, mesmo por vezes distantes, sempre se mantiveram presentes.

Aos meus tios, tias, primos, primas e à minha avó Branca, que acompanha com carinho a conquista de mais um neto formado, deixo meu sincero agradecimento pelo apoio e incentivo ao longo de toda a minha trajetória.

Aos meus amigos do Laboratório LAR/UFSC, grandes exemplos de dedicação e companheirismo, que me proporcionaram excelentes momentos, trocas enriquecedoras e muito aprendizado acadêmico e profissional.

Agradeço ao professor Julio Frantz, que se tornou meu mentor na área da robótica e me inspira a ser um profissional de excelência.

Ao meu orientador de projeto na Alemanha, Daniel Gossen, agradeço pela confiança ao acreditar no meu potencial e pela oportunidade de trabalhar no IGMR, proporcionando uma das experiências acadêmicas mais marcantes da minha formação.

Aos meus amigos alemães Johannes, Nils, Sebastian, Tom, Reza e Steffi, que me acolheram de forma excepcional e mostraram que as diferenças culturais são, na verdade, elementos que nos aproximam e nos fortalecem.

Ao meu amigo Bettanin, por mostrar o caminho das pedras e por todo o apoio e empenho para que minha experiência internacional se tornasse possível.

Aos meus amigos da Magic Island Robotics, que por causa deles eu coleciono dez anos de bons momentos na competição que marcou a minha vida.

Por fim, a todos os outros amigos e amigas que, mesmo não cabendo nesta página, ocupam um lugar especial no meu coração.

“A mente que se abre a uma nova ideia jamais volta ao seu tamanho original.”

(EINSTEIN, A.)

RESUMO

O desenvolvimento de mãos robóticas antropomórficas apresenta desafios significativos relacionados à compactação dos atuadores, à transmissão eficiente de força e à coordenação de múltiplos graus de liberdade. Nesse contexto, este trabalho propõe o projeto e a implementação de uma unidade de atuação dedicada para uma mão robótica antropomórfica acionada por tendões, contemplando desde a concepção mecânica até a validação funcional do sistema integrado.

Inicialmente, foi realizada a modelagem completa da unidade motora em software CAD, considerando aspectos de organização dos atuadores, guiamento dos tendões, integração estrutural e viabilidade de fabricação. A partir do modelo virtual, a unidade foi prototipada utilizando elementos impressos em filamento polimérico por manufatura aditiva, seguida da montagem mecânica e da realização das conexões elétricas entre os servomotores, a unidade de processamento e os módulos auxiliares de alimentação e comunicação.

Paralelamente, foi desenvolvida uma arquitetura de software baseada em ROS 2, responsável por receber comandos do usuário, interpretar instruções individuais ou coordenadas e enviar comandos de posição aos atuadores. Foram implementados nós específicos para o acionamento de motores individuais, controle simultâneo de múltiplos motores e execução de movimentos predefinidos por meio de *presets*, permitindo a realização de padrões coordenados de movimento da mão robótica.

Por fim, foram conduzidos testes experimentais com o objetivo de validar o funcionamento da unidade de atuação e a integração entre os subsistemas mecânico, elétrico e computacional. Os resultados demonstraram a viabilidade da solução proposta, evidenciando que a unidade desenvolvida é capaz de acionar de forma coordenada os tendões da mão robótica, atendendo aos requisitos funcionais do projeto. O trabalho contribui como uma base modular e replicável para pesquisas futuras em manipulação robótica antropomórfica e sistemas de atuação distribuída.

Palavras-chave: Mão robótica antropomórfica. Unidade de atuação. Transmissão por tendões. ROS 2.

ABSTRACT

The development of anthropomorphic robotic hands presents significant challenges related to actuator compactness, efficient force transmission, and the coordination of multiple degrees of freedom. In this context, this work proposes the design and implementation of a dedicated actuation unit for a tendon-driven anthropomorphic robotic hand, encompassing the process from mechanical conception to functional validation of the integrated system.

Initially, the complete modeling of the motor unit was carried out using CAD software, considering aspects such as actuator arrangement, tendon routing, structural integration, and manufacturing feasibility. Based on the virtual model, the unit was prototyped using polymer filament elements produced by additive manufacturing, followed by mechanical assembly and the execution of electrical connections among the servomotors, the processing unit, and auxiliary power and communication modules.

In parallel, a software architecture based on ROS 2 was developed to receive user commands, interpret individual or coordinated instructions, and send position commands to the actuators. Specific nodes were implemented for single motor actuation, simultaneous control of multiple motors, and execution of predefined movements through presets, enabling coordinated motion patterns of the robotic hand.

Finally, experimental tests were conducted to validate the operation of the actuation unit and the integration among the mechanical, electrical, and computational subsystems. The results demonstrated the feasibility of the proposed solution, showing that the developed unit is capable of coordinately actuating the tendons of the robotic hand while meeting the functional requirements of the project. This work contributes as a modular and replicable basis for future research in anthropomorphic robotic manipulation and distributed actuation systems.

Keywords: Anthropomorphic robotic hand. Actuation unit. Tendon-driven transmission. ROS 2.

LISTA DE FIGURAS

Figura 1: Nomenclatura das articulações da mão humana.	17
Figura 2: Planos e eixos de movimento.....	19
Figura 3: Osso e elemento rígido.	20
Figura 4: Juntas sinoviais e exemplos.	21
Figura 5: Ligamento anterolateral e fita elástica.	21
Figura 6: Protótipo de mão utilizado no projeto.....	24
Figura 7: Modelo tridimensional expandido do protótipo.....	27
Figura 8: Exemplo de comunicação adotada pelo ROS.	28
Figura 9: Gaveta de uma máquina de lavar roupa manipulada por mão humana.	30
Figura 10: Bancada de testes.	31
Figura 11: Esquemático e dimensões do servo XL-320.....	33
Figura 12: Painel do software Dynamixel Wizard.	35
Figura 13: Raspberry Pi 4B anexa ao robô.....	36
Figura 14: Diagrama de conexão interna do robô.	41
Figura 15: Ponto de partida do projeto.	42
Figura 16: Modelo tridimensional do acoplamento das placas eletrônicas.	44
Figura 17: Raspberry Pi fixada no robô.	45
Figura 18: U2D2 fixada no robô.	45
Figura 19: Modelo 3D da carcaça e peça manufaturada.	46
Figura 20: Conector e chave fixadas na carcaça do robô.	46
Figura 21: Teste preliminar para guia dos tendões.....	47
Figura 22: Vista de corte para visualização dos canais do modelo 3D.	48
Figura 23: Seções responsáveis por guiar tubos PTFE.	48
Figura 24: Tubos passando pelas seções da peça manufaturada.	49
Figura 25: Conjunto montado.....	49
Figura 26: Modelo CAD do robô completo.	50
Figura 27: Sistema de guia para os tendões com polias.	51
Figura 28: Robô completo montado.....	52
Figura 29: Modelo tridimensional do flange comercial utilizado.	53
Figura 30: Representação da estrutura de <i>software</i> do sistema robótico.	55
Figura 31: Interface de controle simultâneo de três motores escolhidos.	57
Figura 32: Robô com LEDs verdes acesos.....	59

Figura 33: Robô com LEDs roxos acesos.	60
Figura 34: Registro no console de movimento simultâneo de três servomotores.	61
Figura 35: Estrutura de definição de uma predefinição no código.....	62
Figura 36: Retorno no console da função de predefinições de movimento.	63
Figura 37: Exemplo de integração dos nós pela interface do usuário.	64
Figura 38: Registrador de <i>shutdown</i> do servomotor XL-320.....	64
Figura 39: Identificação da fadiga nas fitas elásticas de alguns dedos.	65
Figura 40: Adaptação com porcas na ponta dos dedos.	66
Figura 41: Posicionamento inclinado do robô com auxílio do braço robótico.	67

LISTA DE TABELAS

Tabela 1: Medidas das falanges de cada dedo.....	24
Tabela 2: Ângulos de movimento de cada dedo.....	25
Tabela 3: Principais especificações do servomotor XL-320.....	32
Tabela 4: Especificações técnicas de cada componente.....	37

LISTA DE ABREVIATURAS E SIGLAS

CAD	<i>Computer-Aided Design</i> (Projeto Assistido por Computador)
DC-DC	Conversor de Corrente Contínua para Corrente Contínua
DOF	<i>Degree of Freedom</i> (grau de liberdade)
LED	<i>Light Emitting Diode</i> (Diodo Emissor de Luz)
ROS 2	<i>Robot Operating System 2</i> (Sistema para Operação de Robô 2)
SDK	<i>Software Development Kit</i> (Kit de desenvolvimento de software)
SSH	Secure Shell
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus
UR	Universal Robots
UR5/UR20	Braços robóticos industriais da Universal Robots

LISTA DE SÍMBOLOS



Conexão elétrica de corrente contínua 7,50 V



Conexão elétrica de corrente contínua 5,00 V



Conexão de dados via USB



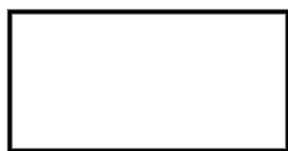
Conexão de comando dos servomotores



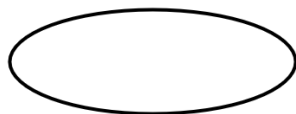
Comando de movimento direto



Comando de tradução



Representação de componentes no diagrama



Representação tópicos de mensagem no diagrama

A

Ampère

V

Volt

W

Watt

SUMÁRIO

1	Introdução	15
1.1	Objetivos.....	15
1.1.1	Objetivo geral.....	16
1.1.2	Objetivos específicos	16
2	Fundamentação teórica	17
2.1	Fisiologia da mão humana	17
2.1.1	Revisão da biomecânica da mão humana	17
2.1.2	A Relação entre elementos biomecânicos e elementos mecânicos	19
2.2	A mão robótica	22
2.2.1	O grau de antropomorfismo	22
2.2.2	Atuação de uma mão robótica	22
2.2.3	O protótipo de mão robótica antropomórfica.....	23
2.2.3.1	<i>Dimensões antropométricas e amplitude de movimento para cada junta.....</i>	<i>24</i>
2.2.3.2	<i>Transmissão do movimento e manufatura dos componentes</i>	<i>26</i>
2.3	Controle de mãos robóticas	27
3	Método	29
3.1	A origem do projeto.....	29
3.1.1	Oportunidade no IGMR/RWTH Aachen.....	29
3.1.2	Estudo do protótipo de mão robótica com auxílio da bancada de testes	30
3.2	Definições iniciais	31
3.2.1	Requisitos de operação	31
3.2.2	Definição do motor	32
3.2.2.1	<i>Configuração e atuação preliminar dos 16 servo motores</i>	<i>34</i>
3.2.3	Definição do microcontrolador e sistema operacional	35
3.2.4	Verificação dos requisitos de alimentação da eletrônica do robô	36

3.2.4.1	<i>Topologia de conexão elétrica.....</i>	36
3.2.4.2	<i>Levantamento de potência dos componentes.....</i>	37
3.2.4.3	<i>Diagrama eletrônico do sistema.....</i>	40
3.2.5	Definição do framework.....	41
3.3	Ajustes na unidade de acionamento.....	42
3.3.1	Suporte da eletrônica na estrutura central	43
3.3.2	Guia para os tendões da mão.....	46
3.3.3	Montagem do robô e passagem dos tendões pela estrutura.....	50
3.3.4	Fixação do robô de mão no braço robótico	53
3.4	Arquitetura de software	54
3.4.1	Configuração da interface de controle.....	54
3.4.2	Definição da estrutura do sistema.....	54
4	Resultados.....	57
4.1	Teste do firmware	57
4.1.1	Função de aquisição de teclas	57
4.1.2	Função de controle de um único servo.....	58
4.1.3	Função de controle de múltiplos servos.....	59
4.1.4	Função que define os padrões de movimento.....	61
4.2	Integração dos blocos de comando	63
4.3	Integração com o movimento do braço robótico	65
5	Conclusões	68
	Referências	70

1 INTRODUÇÃO

O desejo de substituir uma mão perdida é antigo. Ao longo da história, diversas soluções foram buscadas com o objetivo de restaurar a integridade corporal, seja por razões estéticas, seja para recuperar a autonomia funcional do indivíduo (Mendez et al., 2025).

A mão humana constitui uma das ferramentas mais complexas e versáteis do corpo, permitindo ao ser humano manipular objetos com elevada destreza e precisão. Essa capacidade foi determinante para o desenvolvimento de tecnologias, da cultura e, conseqüentemente, para a melhoria da qualidade de vida (Kivell et al., 2023). Em contrapartida, a perda desse membro acarreta impactos significativos na autonomia para a realização de tarefas cotidianas, ocasionando limitações físicas e, muitas vezes, efeitos psicológicos adversos. Tal condição decorre da elevada complexidade anatômica e funcional da mão, que combina força, precisão e agilidade em movimentos finos, estimando-se cerca de 24 graus de liberdade (Stillfried & Van Der Smagt, n.d.).

Diante dessa complexidade, o interesse de centros de pesquisa em desenvolver soluções tecnológicas inspiradas na mão humana tem crescido consideravelmente. Essas pesquisas não se restringem às aplicações médicas voltadas à substituição de membros perdidos, mas se estendem também a dispositivos tecnológicos aplicados na indústria, como os braços robóticos colaborativos, que buscam reproduzir a destreza e a adaptabilidade humanas em ambientes automatizados (Biagiotti et al., n.d.).

Nesse contexto, o presente trabalho tem como objetivo contribuir para a ampliação da acessibilidade de mãos robóticas antropomórficas como ferramentas de manipulação de objetos, passíveis de acoplamento a braços robóticos colaborativos. O projeto foi desenvolvido no Instituto IGMR da RWTH University e integra o projeto DiMonta, voltado ao desenvolvimento de uma ferramenta automatizada de desmontagem de equipamentos de alto valor agregado para fins de reciclagem.

1.1 OBJETIVOS

Nas seções abaixo estão descritos o objetivo geral e os objetivos específicos deste TCC.

1.1.1 Objetivo geral

O objetivo geral deste trabalho é controlar uma mão robótica antropomórfica atuada por tendões por meio de um sistema eletrônico de acionamento com servomotores, utilizando o framework ROS2 para integrar *hardware* e *software*.

1.1.2 Objetivos específicos

Como objetivos específicos desse trabalho, podem ser elencados:

- Modelar em *software* CAD a unidade motora completa do robô, responsável pela atuação dos tendões de uma mão antropomórfica;
- Prototipar a unidade motora com elementos impressos em 3D, executando montagem mecânica e conexão elétrica de todos os componentes;
- Implementar arquitetura de *software* responsável por coletar instruções do usuário, interpretar e enviar o comando para os atuadores do sistema;
- Executar testes para validação do dispositivo e da integração entre *software*, *hardware* e mecanismo.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 FISIOLOGIA DA MÃO HUMANA

2.1.1 Revisão da biomecânica da mão humana

As mãos humanas são estruturas altamente complexas compostas por ossos, articulações, ligamentos, tendões, músculos e pele, que atuam de forma integrada para possibilitar uma ampla gama de movimentos e manipulações precisas. Essa capacidade de controle fino e adaptabilidade de funções despertou o interesse de pesquisadores das áreas de biomecânica em buscar compreender e modelar sua estrutura de forma sistemática (Biagiotti et al., 2002).

A mão humana é formada por 27 ossos, distribuídos entre o carpo, metacarpo e falanges. Cada dedo, exceto o polegar, é composto por três falanges, proximal, média e distal, conectadas por articulações interfalangianas. O polegar, por sua vez, possui apenas duas falanges (proximal e distal) e articulações adaptadas que fornecem movimentos únicos de oposição. As articulações metacarpofalangianas (MCP) ligam as falanges proximais aos ossos metacarpais, enquanto as articulações carpometacarpais (CMC) conectam o metacarpo ao carpo, permitindo certa mobilidade da base da mão (Kerkhof; Van Leeuwen; Vereecke, 2018). As articulações da mão humana e suas respectivas nomenclaturas são apresentadas na Figura 1.

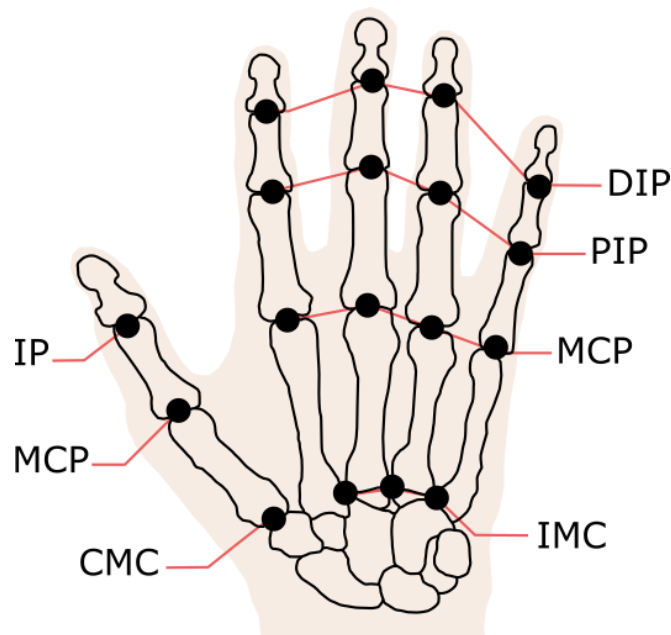


Figura 1: Nomenclatura das articulações da mão humana.

Fonte: Elaboração própria.

Cada articulação é classificada conforme seus graus de liberdade e tipo de movimento permitido. As articulações interfalangianas, por exemplo, são articulações do tipo dobradiça, permitindo movimentos de flexão e extensão. Já as articulações metacarpofalângicas possibilitam, além desses, movimentos de abdução e adução. Essa terminologia é fundamental para o entendimento da cinemática da mão e para o desenvolvimento de sistemas robóticos antropomórficos que buscam reproduzir tais movimentos (Stefano, 2023). Com a identificação das articulações e sua respectiva capacidade de movimento, foi possível estimar quantos graus de liberdade a mão humana é capaz de possuir. O artigo publicado por Stillfried e Van der Smagt, 2010, apresenta um estudo de movimento baseado em MRI para medição da cinemática da mão humana e demonstra que o modelo da mão apresenta 24 graus de liberdade, onde cinco estão no polegar, 16 nos dedos restantes e 3 na palma (Stillfried & Van Der Smagt, 2010).

Para identificar e classificar os movimentos realizados pelos dedos da mão utiliza-se a referência dos planos anatômicos de movimento, padronizados internacionalmente pela Terminologia Anatômica proposta pela Federative Committee on Anatomical Terminology (FCAT). A Figura 2 ilustra os três planos principais: sagital, frontal e transversal. Além dos planos, também são determinados os eixos que servem como linhas de referência que orientam tanto a descrição quanto a medição dos movimentos articulares.

O eixo distal–proximal corresponde ao alinhamento longitudinal do dedo e indica a direção que vai da extremidade distal em direção ao corpo, correspondendo ao plano frontal. O eixo dorsal–palmar descreve a orientação que atravessa o dedo no sentido posterior para anterior e está associado aos movimentos que ocorrem no plano sagital. Já o eixo ulnar–radial representa a direção lateral do dedo em relação aos ossos ulna e rádio, sendo fundamental para descrever movimentos de desvio lateral (Standring et al., 2016).

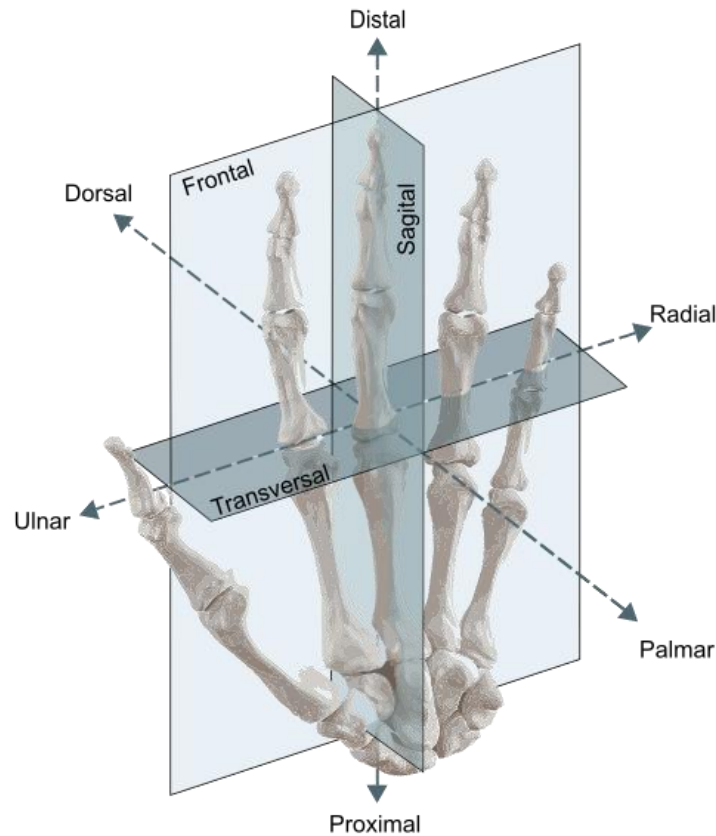


Figura 2: Planos e eixos de movimento.

Fonte: Elaboração própria.

A partir dos planos e eixos de movimento que compõe o espaço tridimensional, foi possível assimilar o movimento do corpo, sobretudo das falanges da mão humana, com conceitos mecânicos apresentados no capítulo seguinte.

2.1.2 A relação entre elementos biomecânicos e elementos mecânicos

O movimento do corpo humano pode ser descrito em termos de graus de liberdade, que, segundo Martins e Murai, representa o número de variáveis independentes necessárias para definir de forma unívoca a configuração de um sistema. De maneira análoga a um mecanismo articulado, cada articulação da mão humana fornece um ou mais graus de liberdade, permitindo a execução de movimentos que se tornam mais complexos na medida que os graus de liberdade do mecanismo aumentam. Já o grau de atuação refere-se ao número de elementos ativos, como músculos ou atuadores, capazes de gerar movimento (Martins; Murai, 2019).

Estabelecendo uma analogia entre a biomecânica e a engenharia mecânica, os ossos podem ser comparados aos elos rígidos de um mecanismo, conforme apresentado na Figura 3. Eles fornecem a estrutura necessária para a transmissão do movimento entre as articulações, da mesma forma que os elos fazem em sistemas robóticos articulados (Salgado Da Silva, 2024).



Figura 3: Osso e elemento rígido.

Fonte: Salgado Da Silva, 2024.

Entre esses ossos, a cartilagem articular atua como um material de conexão que permite deslizamento suave e reduz o atrito, além de conferir certa complacência ao sistema. Essa flexibilidade, em conjunto com o formato geométrico das extremidades ósseas, define o tipo de junta correspondente. Articulações esféricas, como a do ombro, por exemplo, referem-se ao par cinemático esférico, que permite que um elo faça as três rotações em relação a outro elo (Martins; Murai, 2019). Assim, as articulações humanas podem ser comparadas diretamente aos tipos de pares cinemáticos empregados em robôs e mecanismos industriais, conforme apresentado na Figura 4.

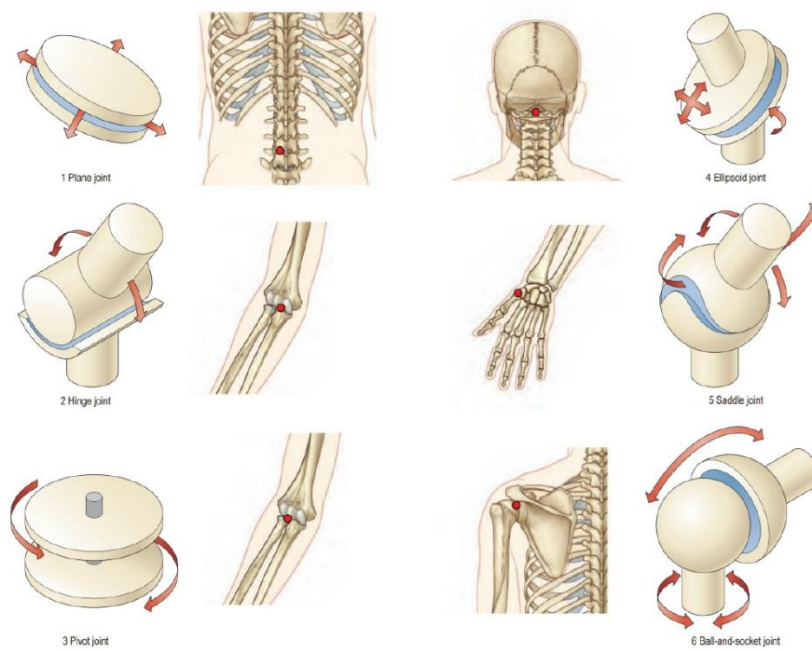


Figura 4: Juntas sinoviais e exemplos.

Fonte: Standring et al., 2016.

Os ligamentos, por sua vez, podem ser comparados a cabos ou fitas elásticas que conectam dois ossos, limitando sua amplitude de movimento e garantindo a estabilidade da articulação. De forma semelhante, os tendões também atuam como elementos de tração, mas conectam músculos aos ossos. Quando o músculo se contrai, o tendão transmite essa força ao osso, provocando o movimento articular. Em uma analogia mecânica, o comportamento elástico dos tendões pode ser comparado a um coeficiente de elasticidade de cabos ou molas, sendo essencial para o controle fino de força e posição durante o movimento da mão de contração e de relaxamento (Salgado Da Silva, 2024). A Figura 5 ilustra a representação de um tendão com seu equivalente em componente mecânico.

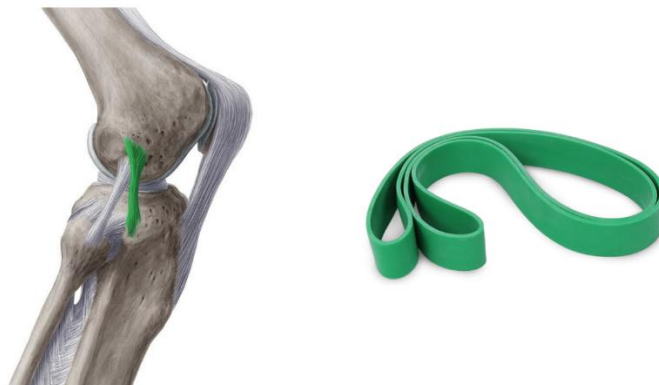


Figura 5: Ligamento anterolateral e fita elástica.

Fonte: Salgado Da Silva, 2024.

2.2 A MÃO ROBÓTICA

2.2.1 O grau de antropomorfismo

O conceito de antropomorfismo em sistemas robóticos refere-se à capacidade do atuador terminal de combinar dois atributos fundamentais: forma e funcionalidade. Segundo Biagiotti et al., 2002, um manipulador antropomórfico deve possuir simultaneamente uma forma que remeta à mão humana e um nível de destreza que garanta funcionalidade. Dessa forma, o grau de antropomorfismo é definido como a medida em que o dispositivo consegue equilibrar semelhança estrutural e habilidade funcional, integrando características estéticas e operacionais de forma coerente (Biagiotti et al., 2002).

Ainda de acordo com esse trabalho, o antropomorfismo pode ser compreendido a partir de três princípios fundamentais. O primeiro princípio é a semelhança morfológica, que consiste em projetar a mão robótica com formato e proporções comparáveis às da mão humana, incluindo a disposição dos dedos, do polegar e da palma. O segundo princípio é a equivalência cinemática, que busca garantir um número e uma distribuição de graus de liberdade capazes de reproduzir os movimentos naturais da mão humana. Por fim, o terceiro princípio é a compatibilidade funcional, que assegura que a mão seja capaz de executar tarefas de preensão e manipulação com precisão e naturalidade, reproduzindo o comportamento biomecânico humano.

Já o conceito de destreza, de acordo o trabalho em questão, é definido por dois aspectos principais. O primeiro é a capacidade do manipulador de adaptar-se a diferentes formas e tamanhos de objetos, o que permite a realização de preensões variadas e seguras. O segundo é a capacidade de executar manipulações complexas, que envolvem movimentos finos e controle preciso da força aplicada sobre os objetos. Assim, a destreza constitui o componente funcional que complementa o antropomorfismo estrutural: uma mão robótica verdadeiramente antropomórfica deve não apenas se assemelhar à mão humana, mas também agir de forma semelhante a ela, expressando fluidez, adaptabilidade e precisão durante o movimento.

2.2.2 Atuação de uma mão robótica

As mãos robóticas podem ser classificadas de acordo com o modo como seus atuadores geram movimento nas articulações. Um dos métodos mais diretos é aquele em que cada junta é acionada por um micro atuador localizado diretamente na junta. Esse tipo de configuração, frequentemente denominada mão robótica de cadeia aberta com acionamento direto por motor, caracteriza-se por possuir um atuador independente em cada grau de liberdade. Assim, cada

motor é responsável por movimentar uma junta específica do dedo, o que permite um controle preciso e desacoplado entre os diferentes eixos de movimento (Zhang et al., 2018).

De acordo com o estudo publicado por Zhang et al., 2018, esse tipo de topologia apresenta a vantagem de desacoplar os movimentos das juntas, o que simplifica o método de controle e facilita a implementação de um sistema de manipulação com movimentos independentes para cada articulação. No entanto, esse método geralmente impõe limitações de desempenho, especialmente relacionadas à carga útil. Isso ocorre porque os atuadores localizados nas juntas iniciais de cada dedo precisam suportar não apenas o peso da estrutura, mas também o peso adicional das juntas subsequentes, que contêm seus próprios atuadores. Como consequência, o aumento de massa ao longo da cadeia cinemática reduz a eficiência do sistema e compromete sua relação peso-potência.

Em contrapartida, o modelo de mão robótica atuada por tendões oferece uma alternativa mais leve e eficiente. Nesse tipo de configuração, os motores são instalados em regiões fixas, geralmente no antebraço, e o movimento é transmitido por meio de cabos flexíveis que percorrem a estrutura do dedo. Essa abordagem permite uma distribuição mais equilibrada de massa, mantendo o peso concentrado na base do sistema e liberando os dedos de cargas excessivas. Além disso, essa topologia proporciona uma estrutura interna mais limpa, sem o acúmulo de motores e fiação ao longo dos dedos, o que favorece a integração mecânica.

Entretanto, o acionamento por cabos também apresenta desafios próprios. Para reproduzir o número de graus de liberdade da mão humana, é necessário utilizar muitos cabos. Além disso, como o movimento gerado por tração do cabo atua em um único sentido, torna-se necessário empregar mecanismos adicionais de retorno para que o dedo volte à posição de repouso. Esse retorno pode ser obtido por meio de molas, elásticos ou cabos complementares, compensando a ausência de atuação bidirecional direta presente nos sistemas de junta motorizada.

2.2.3 O protótipo de mão robótica antropomórfica

O protótipo desenvolvido integra elementos mecânicos e eletrônicos e é constituído por duas seções principais: a mão robótica e a unidade de atuação dos tendões. A mão robótica foi previamente projetada como parte da dissertação de mestrado de Sebastian Polzin e apresenta 24 graus de liberdade e 17 tendões, conforme ilustrado na Figura 6. Já a unidade de atuação foi desenvolvida especificamente para este trabalho e será detalhada na seção de método (Polzin, 2024).



Figura 6: Protótipo de mão utilizado no projeto.

Fonte: Polzin, 2024.

2.2.3.1 Dimensões antropométricas e amplitude de movimento para cada junta

Após a definição da configuração básica da mão robótica, torna-se necessário estabelecer um conjunto de dimensões antropométricas para possibilitar sua construção completa. Essas dimensões incluem a largura e a altura da palma, a largura, altura e espessura de cada dedo, bem como os ângulos máximo e mínimo de cada articulação. Esses parâmetros são fundamentais para garantir que a mão robótica apresente proporções realistas e capacidade de movimento compatível com a mão humana.

O protótipo de mão utilizado foi previamente desenvolvido no IGMR na dissertação de mestrado de Sebastian Polzin. Para definir as dimensões adequadas, Sebastian adotou como base o estudo de Vergara et al., 2018, que apresenta 99 dimensões medidas em 134 indivíduos, abrangendo as principais variáveis geométricas necessárias para a modelagem da mão. As medidas do estudo incluem o comprimento, a largura e a espessura de cada falange, além da espessura da palma em nove diferentes posições. Os valores médios são apresentados separadamente para homens e mulheres, fornecendo um conjunto de dados completo e confiável para o dimensionamento do protótipo. As medidas são apresentadas na Tabela 1.

Tabela 1: Medidas das falanges de cada dedo.

Fonte: Polzin, 2024.

Dedo	Falange Proximal [mm]	Falange Medial [mm]	Falange Distal [mm]	Largura [mm]	Espessura [mm]
Indicador	47,775	27,885	18,915	20,5	16
Médio	51,87	33,15	21,06	19	16
Anelar	47,58	32,175	20,865	18	16
Mínimo	39,78	22,815	18,135	16	16

Já para definição do alcance de movimento angular, os valores adotados no projeto foram organizados em uma tabela que relaciona cada articulação ao respectivo alcance angular utilizado. As medidas referentes aos dedos foram obtidas de Cobos et al., 2008, complementadas por informações de El-shennawy et al., 2001, nos casos em que faltavam dados sobre as articulações CMC. Os valores de alcance de movimento do polegar foram extraídos de Peña-Pitarch et al., 2014, garantindo uma correspondência fiel entre o modelo robótico e a amplitude real de movimento da mão humana. Os ângulos são apresentados na Tabela 2.

Tabela 2: Ângulos de movimento de cada dedo.

Fonte: Polzin, 2024.

Dedo	Articulação	Flexão/Extensão [°]	Abdução/Adução [°]
Polegar	CMC	60/60	15/60
Polegar	MCP	55/0	0/30
Polegar	IP	80/0	-
Indicador	MCP	80/0	30/30
Indicador	PIP	100/0	-
Indicador	DIP	90/0	-
Médio	MCP	80/0	22.5/22.5
Médio	PIP	100/0	-
Médio	DIP	90/0	-
Anelar	CMC	10/0	22.5/22.5
Anelar	MCP	80/0	-
Anelar	PIP	100/0	-

Dedo	Articulação	Flexão/Extensão [°]	Abdução/Adução [°]
Anelar	DIP	90/0	-
Mínimo	CMC	20/0	25/25
Mínimo	MCP	80/0	-
Mínimo	PIP	100/0	-
Mínimo	DIP	90/0	-

2.2.3.2 Transmissão do movimento e manufatura dos componentes

A transmissão do movimento é o elemento intermediário entre o sistema de atuação e o movimento efetivo das articulações. Conforme documentado por Sebastian Polzin, optou-se pelo uso de um sistema de transmissão por tendão e polias, uma escolha fundamentada no estudo de Clark et al., 2021, no qual são comparadas nove arquiteturas distintas de orientação de cabos em dedos robóticos. Segundo a análise desses autores, a variante selecionada oferece o melhor equilíbrio entre desempenho de preensão e eficiência na transmissão de força, justificando sua adoção como padrão para o roteamento de tendões dentro dos dedos do protótipo.

Além da escolha do sistema principal de transmissão, Polzin adotou a estratégia denominada *Agonistic Tendon with Elastic Band*, que utiliza fitas elásticas para promover o retorno passivo das articulações. A decisão de empregar fitas elásticas em vez de molas metálicas está relacionada principalmente ao custo reduzido desse material. No entanto, o uso de elastômeros apresenta desvantagens relevantes, como desgaste progressivo da capacidade elástica e variações na pré-carga ocasionadas pela necessidade de corte manual e colagem. Essas incertezas podem impactar o desempenho da preensão e a repetibilidade da mão, especialmente em cenários que exigem precisão ou cargas variáveis.

Outro aspecto considerado por Polzin está relacionado ao atrito presente ao longo do roteamento dos tendões. Para minimizar perdas de eficiência, foram utilizados tubos de PTFE nos pontos em que o cabo atravessa o interior da estrutura, reduzindo o atrito superficial e prolongando a vida útil do sistema. Da mesma forma, as polias presentes nas falanges foram montadas com o mesmo tipo de rolamento empregado nas juntas, de forma a garantir rotação suave, reduzir resistência mecânica e possibilitar transmissão de força mais estável.

A dissertação também detalha o roteamento dos cabos na palma e no polegar, com atenção especial às juntas MCP e à articulação CMC do polegar. A complexidade desses

elementos exigiu uma descrição aprofundada do processo de construção, uma vez que envolvem múltiplas direções de força e variações geométricas que influenciam diretamente a precisão dos movimentos. Já o roteamento no interior das falanges não é abordado em profundidade, pois segue os princípios previamente descritos.

No que diz respeito à manufatura dos componentes, Polzin utilizou majoritariamente impressão 3D do tipo FDM. Esta técnica foi escolhida por permitir a produção de peças de maneira econômica, com precisão adequada ao propósito experimental e com liberdade geométrica suficiente para a criação de estruturas complexas. Todos os componentes estruturais foram impressos em PLA com densidade interna de 30 por cento. Além das peças impressas, alguns componentes comerciais foram incorporados ao protótipo, tais como parafusos, porcas e a linha utilizada como tendão. A Figura 7 ilustra os componentes na montagem do protótipo.

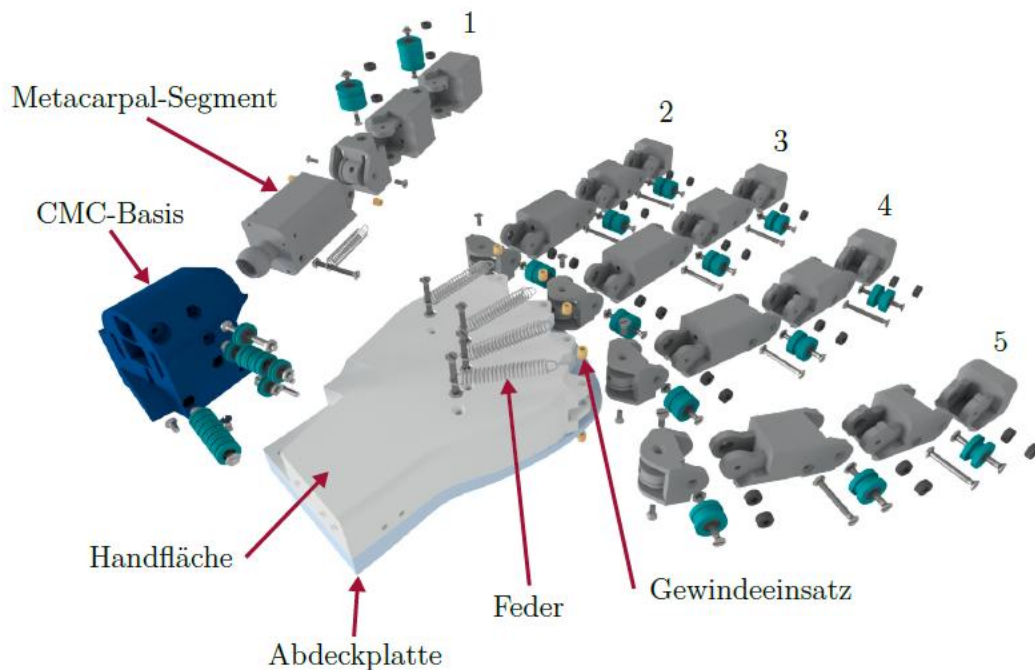


Figura 7: Modelo tridimensional expandido do protótipo.

Fonte: Polzin, 2024.

2.3 CONTROLE DE MÃOS ROBÓTICAS

O controle de mãos robóticas envolve a integração entre o *hardware*, composto por motores e controladores, e o *software* responsável pela execução das tarefas de movimento e coordenação. Dessa forma, o Robot Operating System (ROS) se destaca como um dos *frameworks* mais adequados e amplamente utilizados em aplicações de robótica. O ROS

fornece uma estrutura modular que facilita o desenvolvimento, a comunicação e a reutilização de código em sistemas robóticos complexos.

A arquitetura do ROS é baseada em uma estrutura distribuída de nós de comunicação, como representada na Figura 8, organizada segundo a lógica de publicadores (*publishers*) e assinantes (*subscribers*). Cada nó executa uma função específica do sistema, como a leitura de sensores, o controle de motores ou o processamento de teclas de comando, e se comunica com outros nós por meio de tópicos. Um nó publicador envia mensagens em um determinado tópico, enquanto um nó assinante recebe e processa essas mensagens.

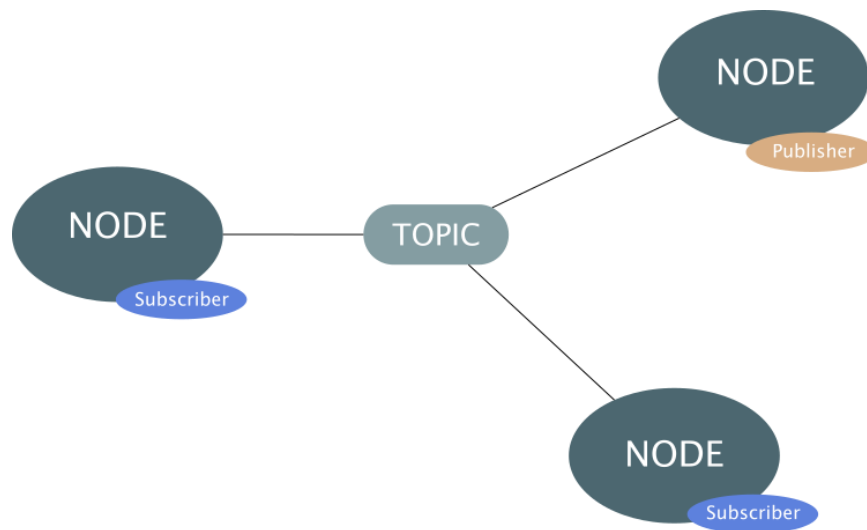


Figura 8: Exemplo de comunicação adotada pelo ROS.

Fonte: Elaboração própria.

Além da modularidade, o ROS oferece um conjunto de bibliotecas e ferramentas para simulação, visualização e controle de *hardware* real. Essa infraestrutura facilita o desenvolvimento de algoritmos de controle e a integração de diferentes componentes, promovendo a padronização e a cooperação entre sistemas robóticos.

No desenvolvimento deste trabalho, o Dynamixel Wizard foi utilizado como ferramenta inicial para a configuração dos servomotores Dynamixel XL-320 empregados na mão robótica. O Dynamixel Wizard é um *software* desenvolvido pela empresa Robotis que permite realizar o diagnóstico e o ajuste de motores compatíveis com o protocolo Dynamixel. Por meio dessa ferramenta foi possível configurar o protocolo de comunicação adequado, definir a taxa de transmissão de dados, atribuir identificadores únicos para cada servomotor e realizar testes preliminares de movimentação. Esses testes foram essenciais para validar o funcionamento individual de cada atuador antes da montagem final do sistema.

3 MÉTODO

3.1 A ORIGEM DO PROJETO

3.1.1 Oportunidade no IGMR/RWTH Aachen

A oportunidade de desenvolver este trabalho surgiu a partir de um projeto maior, previamente iniciado no Instituto de Teoria de Mecanismos, Dinâmica de Máquinas e Robótica (IGMR) da RWTH Aachen, pertencente à Universidade RWTH Aachen. Financiado pelo programa Start-up Transfer.NRW por meio do Ministério da Economia, Indústria, Proteção Climática e Energia da Renânia-do-Norte-Vestfália e com cofinanciamento de fundos da União Europeia (EFRE), o projeto DiMonta tem como propósito o desenvolvimento de um sistema de desmontagem automatizada de equipamentos para reciclagem. A solução integra *software* e *hardware* em braços robóticos colaborativos comerciais para detectar, processar, planejar e desmontar equipamentos com alto valor agregado para reciclagem, como baterias de veículos elétricos e eletrodomésticos que utilizam componentes ricos em terras raras (IGMR, 2025)

A arquitetura do projeto é estruturada em torno de um sistema operacional de desmontagem denominado di.core, responsável por orquestrar diversas ferramentas modulares que abrangem desde o reconhecimento do componente até o planejamento, o agarramento e a desconexão de elementos, incluindo a remoção de parafusos, adesivos e conectores. Todo o processo é controlado por técnicas de inteligência artificial adaptativa que permitem a execução eficiente das tarefas.

A proposta também considera desafios práticos relevantes para a desmontagem automatizada, como a grande diversidade de produtos a serem desmontados e suas diferentes condições físicas, que podem incluir desgaste, peças ausentes ou danos estruturais. Além disso, lacunas nos dados do produto, como a ausência de modelos tridimensionais e de especificações completas, representam obstáculos que são tratados por meio de estratégias adaptativas baseadas em inteligência artificial.

O núcleo da plataforma é composto por ferramentas organizadas em cinco categorias principais:

- Detectar: identificar os componentes presentes no produto;
- Entender: determinar quais etapas são necessárias para a desmontagem;
- Planejar: estabelecer a sequência de ações mais eficiente;
- Agarrar: posicionar, estabilizar e segurar partes do produto;

- Desconectar: executar a desmontagem mecânica, incluindo a remoção de fixadores e elementos de junção.

Considerando esse escopo de automatização da desmontagem de produtos, a proposta de desenvolvimento de uma garra antropomórfica torna-se particularmente relevante, pois muitos equipamentos de uso cotidiano são projetados para serem manipulados por mãos humanas. Um exemplo é a gaveta para *dispenser* de sabão em máquinas de lavar roupa, ilustrada na Figura 9, concebida originalmente para operação manual. Assim, uma mão robótica antropomórfica acoplada a um braço robótico comercial combina as vantagens da automação industrial com a forma e a destreza características de uma mão humana, possibilitando a execução de tarefas complexas de desmontagem que exigem precisão e adaptabilidade. O exemplo em questão é estudado pelo projeto DiMonta como desafio prático a ser resolvido no ano de 2025.

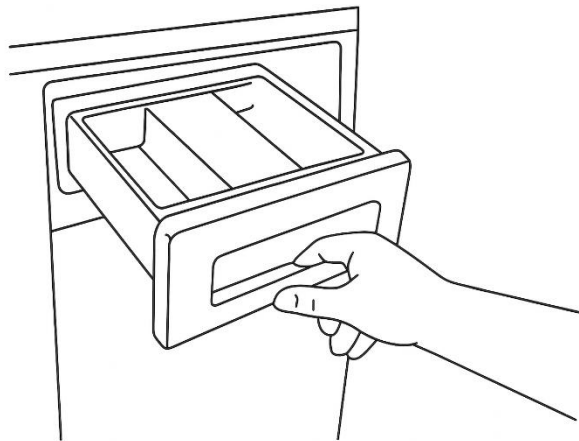


Figura 9: Gaveta de uma máquina de lavar roupa manipulada por mão humana.

Fonte: Elaboração própria, gerada por IA.

3.1.2 Estudo do protótipo de mão robótica com auxílio da bancada de testes

A fim de entender a capacidade e as restrições do protótipo de mão robótica a ser atuado pela unidade motora, utilizou-se de uma bancada de testes desenvolvida pelo Sebastian Polzin, ilustrada na Figura 10, que conta com manivelas para fazer a atuação manual de cada tendão individualmente. Dessa forma, pode-se identificar não só a capacidade do protótipo em efetuar movimentos coordenados que posteriormente viriam a ser reproduzidos de forma automatizada, mas também pontos de melhoria que poderiam acarretar restrições no protótipo final do robô interligado ao módulo de controle.

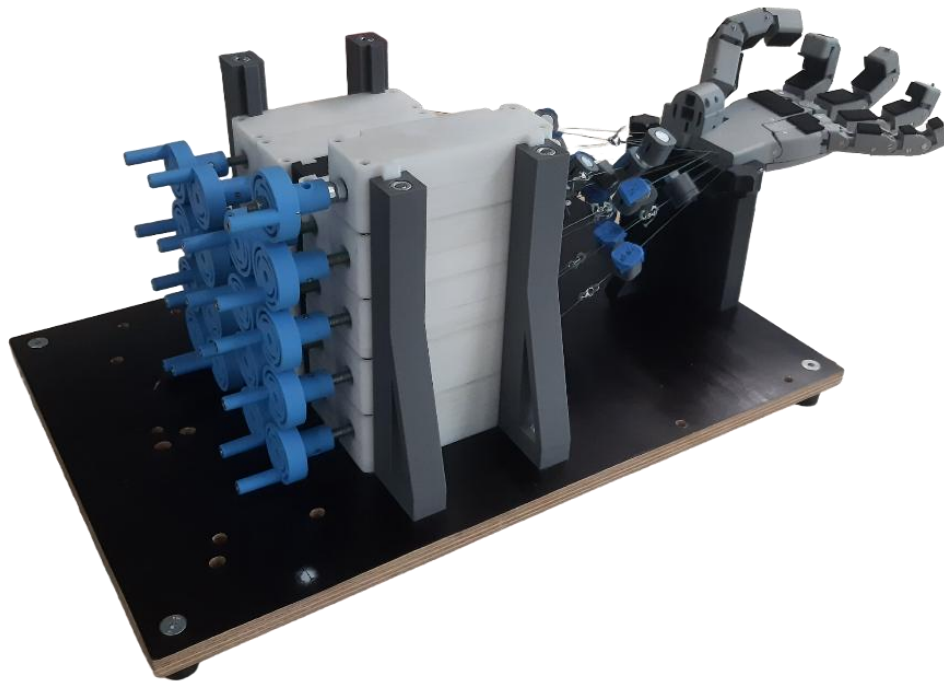


Figura 10: Bancada de testes.

Fonte: Elaboração própria.

3.2 DEFINIÇÕES INICIAIS

3.2.1 Requisitos de operação

A operação da mão robótica proposta neste trabalho exige o atendimento a requisitos funcionais e estruturais capazes de garantir a integração com o ambiente de testes, em especial com o braço robótico onde será instalada. Estabeleceu-se que a mão robótica deve ser totalmente compatível com o braço robótico UR20, da fabricante Universal Robots, permitindo acoplamento direto à flange do manipulador sem a necessidade de adaptações complexas. Essa integração estrutural deve assegurar rigidez mecânica, alinhamento adequado e facilidade de montagem e manutenção.

Do ponto de vista de alimentação, definiu-se que o sistema deve operar com apenas um cabo, responsável por energizar todos os atuadores e circuitos embarcados, ajustando os níveis de tensão conforme a necessidade do circuito interno. No que se refere à conectividade, estabeleceu-se que toda a comunicação com a mão robótica deve ocorrer exclusivamente por meio de interface wireless, sem o uso de cabos de dados. Essas decisões visam reduzir o número de conexões físicas, minimizar interferências mecânicas durante movimentos amplos do braço e aumentar a confiabilidade da operação, especialmente em aplicações que exigem grande liberdade de movimento.

3.2.2 Definição do motor

Para efetuar a atuação dos tendões da mão robótica, optou-se pelo uso do servo Dynamixel XL-320, da Robotis, por uma combinação de fatores técnicos, mecânicos e de integração com o sistema de controle. Segundo o datasheet da fabricante, o XL-320 é um atuador compacto que integra motor, caixa de redução, controle e sensores em um módulo único. Algumas de suas especificações principais, apresentadas na Tabela 3, fazem este motor ser uma escolha atrativa para a aplicação em questão (Robotis, 2025).

Tabela 3: Principais especificações do servomotor XL-320.

Fonte: Robotis, 2025.

Especificação	Valor	Unidade
Baud rate	7,343 - 1000	kbps
Resolução	0,29	°
Percurso angular	0 - 300	°
Peso	16,7	g
Dimensões (L × A × P)	24 × 36 × 27	mm
Redução	238:1	-
Torque	0,39	N·m
Velocidade a vazio	14	rev/min
ID	0 - 252	-

A combinação de torque, velocidade, feedback e tamanho torna o XL-320 especialmente adequado para aplicações em que o atuador é acoplado em espaços limitados, fornecendo força suficiente para manipulação, mas sem adicionar peso excessivo ou dimensões que comprometam o volume adequado da unidade de atuação. O desenho tridimensional do motor com suas medidas é apresentado a seguir, na Figura 11.

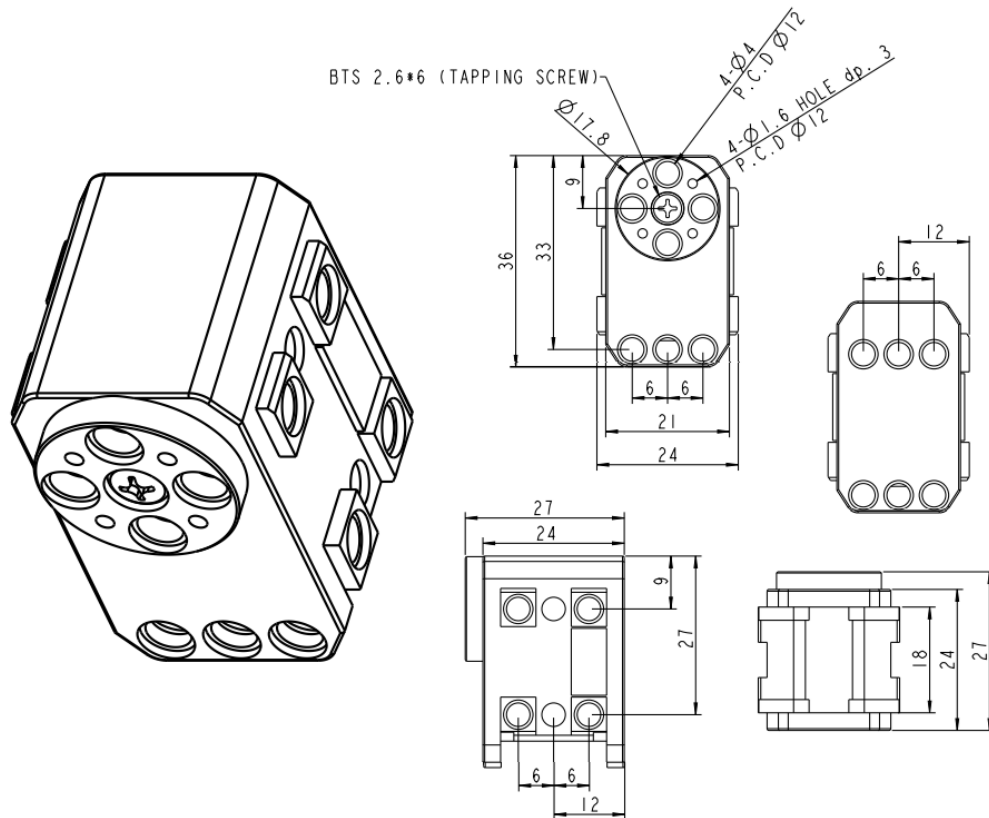


Figura 11: Esquemático e dimensões do servo XL-320.

Fonte: Robotis, 2025.

Outro aspecto decisivo para a escolha foi a integração com o ROS. A série Dynamixel, incluindo o XL-320, é amplamente suportada por bibliotecas e pacotes ROS, como o `dynamixel_sdk` utilizado neste projeto, permitindo controle por tópicos, monitoramento dos estados de operação e configuração de parâmetros de funcionamento diretamente via ROS. Essa compatibilidade facilita a implementação dos controladores, a sincronia entre vários servos e proporciona um ambiente de testes mais seguro, tanto para proteção do *hardware* quanto para a estrutura do robô.

Além disso, o protocolo de comunicação UART TTL Half-duplex e a possibilidade de conectar múltiplos servos em um único barramento em *daisy-chain*, isto é, cada motor é conectado ao próximo por meio dos conectores embutidos nas laterais do servo, simplificam o cabeamento para uma única cadeia série e reduzem a complexidade física do robô. Dessa forma, todos os atuadores compartilham a mesma linha de alimentação e o mesmo fio de comunicação, reduzindo drasticamente o número de cabos necessários.

Esse modelo emprega conectores de 3 pinos do tipo JST-PH, padrão da linha XL, sendo dois conectores idênticos posicionados lateralmente no corpo do servo, com pinos de VCC (alimentação positiva), GND (terra), e dados.

O XL-320 também possui mecanismos internos de segurança que protegem as engrenagens da caixa de redução do servo ao identificar um torque acima do valor máximo permitido. O mecanismo de embreagem, segundo a fabricante, protege fisicamente as engrenagens e retoma o torque para o eixo do motor apenas após receber um comando de habilitação do torque depois do travamento.

A presença de LEDs programáveis de status no corpo do servo também contribui para o diagnóstico e monitoramento do sistema. Esses indicadores visuais foram empregados tanto para sinalizar a ativação de cada servo quanto para indicar estados específicos associados às funções de atuação, possibilitando a identificação rápida de atividade, erros e condições anômalas ao longo dos testes.

3.2.2.1 Configuração e atuação preliminar dos 16 servo motores

Esta etapa de preparação dos servomotores envolve o uso do *software* Dynamixel Wizard, disponibilizado pela própria fabricante, com o objetivo de ajustar os principais parâmetros de comunicação necessários para a operação adequada da unidade de atuação. O *software* permite configurar o protocolo utilizado pelo servo, definir o *baud rate*, atribuir um identificador único a cada unidade e realizar testes básicos de movimentação, assegurando o correto funcionamento de todos os atuadores antes de sua integração ao sistema. A Figura 12 traz a interface do *software* e as opções de configuração dos parâmetros citados.

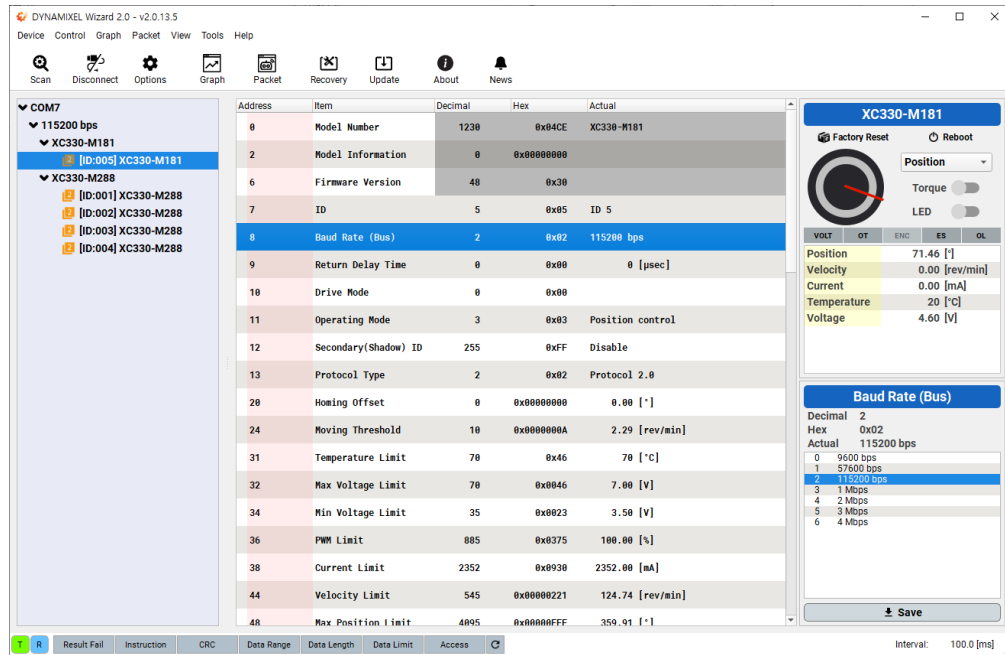


Figura 12: Painel do software Dynamixel Wizard.

Fonte: Robotis, 2025.

O modelo de servomotor escolhido suporta até 253 identificadores distintos, numerados de 0 a 252. Dessa forma, foi possível atribuir aos 16 servomotores utilizados IDs consecutivos, variando de 0 a 15, em correspondência direta com o marcador numérico fixado na carcaça de cada motor. Essa organização contribuiu para a estruturação lógica do sistema e para o mapeamento adequado dos tendões atuados.

3.2.3 Definição do microcontrolador e sistema operacional

A escolha da Raspberry Pi 4 Model B deve-se à sua ampla compatibilidade com o framework ROS 2, já que a operação é bem estabelecida nessa plataforma. A placa oferece desempenho suficiente para executar os nós de controle, comunicação e monitoramento em tempo real, além de integrar nativamente interfaces necessárias ao sistema, como UART, USB e redes sem fio Wi-Fi e Bluetooth, além de ter as dimensões adequadas para ser embarcada no robô, conforme ilustra a Figura 13.



Figura 13: Raspberry Pi 4B anexa ao robô.

Fonte: Elaboração própria.

O Ubuntu Server 22.04 foi adotado como sistema operacional por ser a distribuição recomendada pelo próprio time de desenvolvimento do ROS 2 em sua documentação oficial, especificamente para a versão Humble do *framework* selecionada para este projeto. A versão Server, sem interface gráfica, foi escolhida estrategicamente para otimizar o desempenho, uma vez que todo o sistema foi projetado para execução local no microcontrolador embarcado no robô. A ausência de interface gráfica evita consumo desnecessário de memória e processamento, permitindo que os recursos da placa sejam integralmente dedicados aos processos essenciais de controle, comunicação e monitoramento do sistema robótico (Robotis, 2025).

3.2.4 Verificação dos requisitos de alimentação da eletrônica do robô

3.2.4.1 Topologia de conexão elétrica

A alimentação elétrica do sistema foi centralizada em um único cabo de entrada, conectado diretamente ao módulo U2D2 Power Hub, que recebe 7,50 V da fonte externa e distribui essa tensão para todos os servomotores. A partir desse mesmo ponto de entrada, a alimentação é ramificada para um conversor DC-DC chaveado, responsável por reduzir a tensão para 5,00 V e fornecer à Raspberry Pi 4 Model. Essa solução permite unificar todo o sistema em um único cabo de energia, simplificando o cabeamento e garantindo que tanto o módulo de

controle dos servos quanto o microcontrolador operem dentro de suas faixas de tensão especificadas.

3.2.4.2 Levantamento de potência dos componentes

A determinação da potência necessária para o sistema foi realizada com o objetivo de dimensionar adequadamente a fonte de alimentação responsável por suprir todos os componentes eletrônicos do robô. Embora as tensões de entrada fossem previamente definidas pelos próprios dispositivos, sendo 7,50 V para o barramento dos servomotores e 5,00 V para a Raspberry Pi, a corrente total requerida variava de acordo com a combinação de componentes em operação. Dessa forma, tornou-se necessário avaliar o consumo em diferentes cenários de funcionamento, incluindo situações mínimas, típicas e críticas, de modo a garantir que a fonte selecionada fosse capaz de atender de forma segura a todas as possíveis condições de carga do sistema.

Para os servomotores XL-320, foram considerados os valores de corrente fornecidos no manual do fabricante, que especifica uma corrente sem carga de 0,18 A e corrente de estol de 1,10 A por motor, ambos operando a 7,50 V (Robotis, 2025). Para a Raspberry Pi 4 Model B, adotou-se a recomendação oficial da Raspberry Pi Foundation, que indica tensão de alimentação de 5,00 V e corrente típica de até 2,50 A durante uso regular, podendo atingir 3,00 A em condições de carga elevada (Raspberry Pi Ltd., 2025). As especificações do módulo U2D2 Power Hub indicam capacidade máxima de corrente próxima de 10,00 A, o que representa um limite prático para a alimentação simultânea dos servos. Como o rendimento do conversor DC-DC utilizado para alimentar a Raspberry Pi não foi especificado, adotou-se uma eficiência conservadora de 90%, condizente com conversores chaveados de uso geral. A Tabela 4 apresenta os parâmetros de tensão, corrente e rendimento dos componentes do sistema eletrônico, que serão utilizados a seguir para os cálculos de consumo em diferentes cenários.

Tabela 4: Especificações técnicas de cada componente.

Fonte: Elaboração própria.

Componente	Parâmetro	Símbolo	Valor	Unidade
Servomotor Dynamixel XL-320	Tensão de operação	V_{servos}	7,50	V
Servomotor Dynamixel XL-320	Corrente sem carga	I_{vazio}	0,18	A
Servomotor Dynamixel XL-320	Corrente de estol	I_{estol}	1,10	A
Servomotor Dynamixel XL-320	Corrente média	$I_{\text{média}}$	0,65	A

Raspberry Pi 4 Model B	Tensão de alimentação	V_{pi}	5,00	V
Raspberry Pi 4 Model B	Corrente típica	I_{pi}	2,50	A
Raspberry Pi 4 Model B	Corrente máxima	$I_{pi,máx}$	3,00	A
Conversor DC-DC	Tensão de entrada	V_{fonte}	7,50	V
Conversor DC-DC	Tensão de saída	V_{pi}	5,00	V
Conversor DC-DC	Rendimento	η	0,90	-
U2D2 Power Hub	Tensão de operação	V_{fonte}	7,50	V
U2D2 Power Hub	Corrente auxiliar	I_{U2D2}	0,10	A
Fonte de alimentação	Tensão nominal	V_{fonte}	7,50	V
Fonte de alimentação	Corrente total	I_{total}	Variável	A

Os cálculos de potência seguem as relações elétricas básicas de corrente e potência. A potência consumida por cada componente é dada pela Equação [1]:

$$P = V * I \quad [1]$$

A corrente exigida do lado de 7,50 V para a alimentação da Raspberry Pi foi calculada considerando a eficiência do conversor DC-DC. Utilizou-se a relação da Equação [2]:

$$I_{entrada} = \frac{V_{pi} * I_{pi}}{V_{fonte} * \eta} \quad [2]$$

V_{pi} é a tensão de saída do conversor, I_{pi} é a corrente consumida pela Raspberry Pi, V_{fonte} é a tensão da fonte de 7,50 V e η é a eficiência do conversor. A corrente total requerida do sistema resulta da soma da corrente consumida pelos servomotores, pela Raspberry Pi e pelo consumo auxiliar aproximado do módulo U2D2 e circuitos de sinal, adotado como 0,10 A por não possuir valor oficial detalhado.

Como o consumo total do sistema depende diretamente da quantidade de servos acionados simultaneamente e de suas condições de operação, estabeleceu-se três cenários representativos. O cenário extremo considera a situação improvável em que todos os 16 servos operam em corrente de estol simultaneamente, o que permite identificar o limite absoluto do sistema. O cenário mínimo considera apenas um servo operando em plena carga enquanto os demais permanecem sem carga, representando situações de ajustes finos ou calibração de movimentos individuais. O cenário comum, mais próximo da operação típica da mão robótica antropomórfica, considera seis servos atuando simultaneamente, correspondendo

aproximadamente ao número de tendões envolvidos em movimentos usuais de preensão e articulação coordenada.

A seguir são apresentados os cálculos detalhados para cada um dos três casos.

No cenário extremo, todos os 16 servos operam simultaneamente em corrente de estol. A corrente total consumida pelos servomotores é dada pela Equação [3], que utilizou do valor da corrente de estol da Tabela 4:

$$\begin{aligned} I_{\text{servos}} &= 16 * I_{\text{estol}} \\ I_{\text{servos}} &= 16 * 1,10 = 17,60 \text{ A} \end{aligned} \quad [3]$$

A corrente exigida da fonte de 7,50 V para alimentar a Raspberry Pi, considerando uma eficiência de 90 por cento do conversor DC-DC, pode ser calculada pela aplicação da Equação [2], conforme mostrado na Equação [4]:

$$I_{\text{pi,entrada}} = \frac{5,00 * 2,50}{7,50 * 0,9} \approx 1,85 \text{ A} \quad [4]$$

A corrente total estimada é apresentada na Equação [5]:

$$\begin{aligned} I_{\text{total}} &= I_{\text{servos}} + I_{\text{pi,entrada}} + I_{\text{U2D2}} \\ I_{\text{total}} &= 17,60 + 1,85 + 0,10 = 19,55 \text{ A} \end{aligned} \quad [5]$$

A potência total consumida, mostrada pela Equação [6], é:

$$\begin{aligned} P_{\text{total}} &= V_{\text{fonte}} * I_{\text{total}} \\ P_{\text{total}} &= 7,50 * 19,55 \approx 146,60 \text{ W} \end{aligned} \quad [6]$$

Esse cenário demonstra que a soma das correntes ultrapassa significativamente o limite máximo contínuo suportado pelo U2D2 Power Hub, que é da ordem de 10,00 A. Consequentemente, este caso serve exclusivamente como análise teórica e não como modo de operação permitido. É importante salientar que cada servomotor apresenta um sistema de proteção único contra sobrecorrente, logo o caso extremo apresentado é protegido pelo mecanismo dos servos, assim como pelo limite de corrente imposto futuramente pela fonte de alimentação.

O cenário mínimo considera apenas um servo em plena operação enquanto os demais operam em corrente sem carga. Assim, a corrente total dos servos pode ser vista na Equação [7]:

$$\begin{aligned} I_{\text{servos}} &= 1 * I_{\text{estol}} + 15 * I_{\text{vazio}} \\ I_{\text{servos}} &= 1 * 1,10 + 15 * 0,18 = 3,80 \text{ A} \end{aligned} \quad [7]$$

A corrente total é calculada pela aplicação dos valores obtidos nas Equações [4] e [7] na Equação [5], mantendo o valor de I_{U2D2} de acordo com a Tabela 4, conforme mostrado na Equação [8]:

$$I_{\text{total}} = 3,80 + 1,85 + 0,10 = 5,75 \text{ A} \quad [8]$$

Já a potência correspondente é calculada pela Equação [6], como pode ser observado na Equação [9]:

$$P_{\text{total}} = 7,50 * 5,75 \approx 43,10 \text{ W} \quad [9]$$

Este cenário demonstra um consumo bastante reduzido em relação aos demais e representa os momentos de calibração ou teste isolado de cada atuador.

O cenário comum busca representar o comportamento típico da mão robótica durante uma preensão, na qual vários tendões são acionados simultaneamente. Considerou-se que seis servos operam em condição de carga moderada enquanto os demais permanecem sem carga. Para efeito de cálculo, adotou-se uma corrente média de operação de 0,65 A por servo em carga moderada, valor condizente com o comportamento típico desses motores em esforços intermediários (Robotis, 2025).

Assim, a corrente dos servos é dada pela aplicação da Equação [3], como é notado na Equação [10]:

$$\begin{aligned} I_{\text{servos}} &= 6 * I_{\text{média}} + 10 * I_{\text{vazio}} \\ I_{\text{servos}} &= 6 * 0,65 + 10 * 0,18 = 5,70 \text{ A} \end{aligned} \quad [10]$$

A corrente total, calculada a partir da Equação [5], resulta no valor apresentado na Equação [11]:

$$I_{\text{total}} = 5,70 + 1,85 + 0,10 = 7,65 \text{ A} \quad [11]$$

Utilizando da Equação [6], a potência total consumida neste cenário aproximado é calculada pela Equação [12]:

$$P_{\text{total}} = 7,50 * 7,65 \approx 57,40 \text{ W} \quad [12]$$

Este cenário representa de forma mais realista o consumo típico esperado durante a operação normal da unidade de atuação dos tendões.

3.2.4.3 Diagrama eletrônico do sistema

Com base na topologia de conexão adotada e nos níveis de tensão apresentados na seção anterior, a Figura 14 apresenta o diagrama elétrico completo do robô, destacando as interligações entre os componentes principais e o fluxo de alimentação ao longo do sistema.

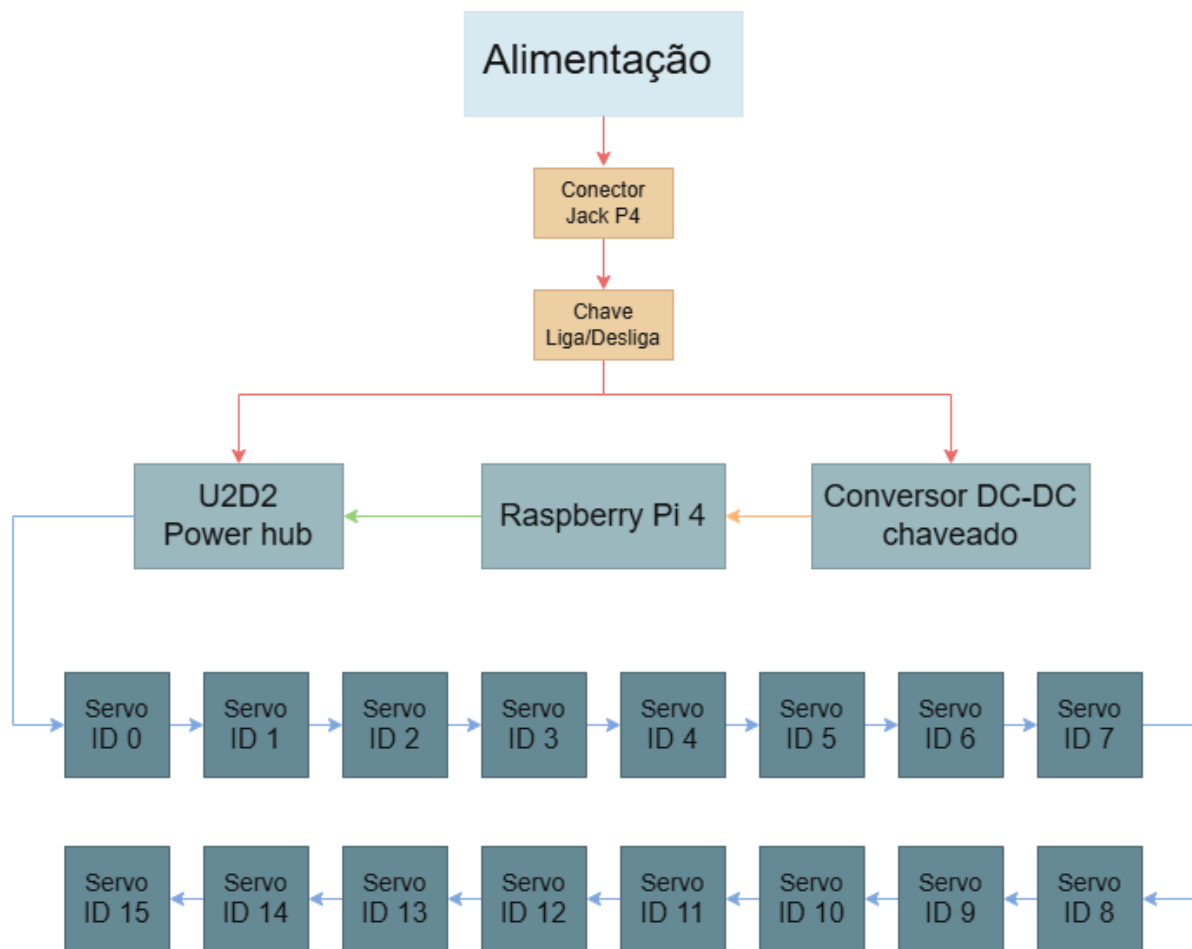


Figura 14: Diagrama de conexão interna do robô.

Fonte: Elaboração própria.

3.2.5 Definição do framework

A escolha do ROS 2 Humble como framework de desenvolvimento é fundamentada essencialmente na ampla compatibilidade com microcontroladores baseados na arquitetura ARM, como a Raspberry Pi 4B utilizada neste projeto. Entre as versões disponíveis do ROS 2, o Humble apresenta melhor equilíbrio entre maturidade, suporte comunitário e longevidade, sendo a versão mais recomendada. Versões mais recentes, como Iron ou Jazzy, embora disponham de recursos adicionais, ainda apresentam menor estabilidade em *hardware* embarcado, enquanto versões anteriores, como Foxy, alcançaram fim de suporte ou possuem limitações de compatibilidade com bibliotecas modernas.

Foi adotado o Dynamixel SDK para a comunicação com os servomotores, pacote oficial fornecido pela ROBOTIS e compatível com o ROS 2. Essa escolha se justifica pela

integração nativa com os servos XL-320, pela disponibilidade de exemplos funcionais e pelo amplo suporte da comunidade no fórum do ROS.

3.3 AJUSTES NA UNIDADE DE ACIONAMENTO

Quando este trabalho foi iniciado, já havia sido desenvolvido um protótipo preliminar da unidade de acionamento da mão robótica. Este protótipo era composto de uma estrutura central manufaturada em impressão 3D com furos previstos para fixação dos servos definidos no escopo do projeto. Além disso, também havia sido usinada a placa responsável pela conexão entre a unidade de acionamento e o braço robótico UR5 da fabricante Universal Robots. A Figura 15 ilustra o ponto de partida do projeto, onde pode-se identificar os elementos descritos conectados a um protótipo de mão robótica, também previamente desenvolvido no IGMR.



Figura 15: Ponto de partida do projeto.

Fonte: Elaboração própria.

Durante as etapas de integração da unidade de acionamento a mão robótica, verificou-se a necessidade de realizar modificações estruturais adicionais para garantir o funcionamento completo do conjunto. Embora o projeto original da mão robótica contemplasse o arranjo mecânico e a transmissão por tendões, a implementação prática evidenciou demandas específicas relacionadas à organização dos componentes eletrônicos, à condução adequada dos

tendões e ao acoplamento mecânico com o braço robótico a ser utilizado posteriormente na etapa de testes. Dessa forma, foram desenvolvidos e incorporados novos elementos à estrutura, com o objetivo de assegurar estabilidade, otimizar o roteamento dos tendões e permitir a interface física adequada com o manipulador UR20.

3.3.1 Suporte da eletrônica na estrutura central

A estrutura central da unidade de acionamento havia sido originalmente projetada apenas para acomodar os servomotores responsáveis pelo tensionamento dos tendões, sem prever o espaço necessário para a instalação dos componentes eletrônicos adicionais. Para isso, foram modeladas duas peças responsáveis fixar as placas eletrônicas nas faces laterais livres da estrutura central, parafusadas diretamente nos furos já presentes, evitando modificações na geometria original e mantendo a integridade da unidade previamente construída. A solução permitiu a fixação da Raspberry Pi em uma das faces e do módulo U2D2 na face oposta, garantindo distribuição adequada dos componentes e boa acessibilidade às conexões. A Figura 16 ilustra as duas peças em modelo 3D.

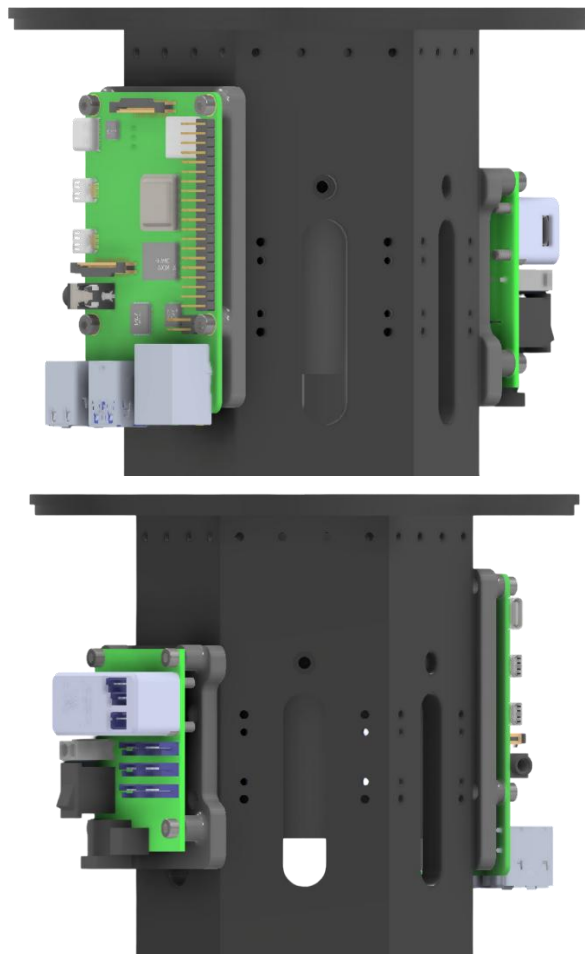


Figura 16: Modelo tridimensional do acoplamento das placas eletrônicas.

Fonte: Elaboração própria.

Já a Figura 17 e a Figura 18 apresentam fotos das placas fixadas na estrutura central do robô, depois de manufaturadas em impressão 3D, para fins de comparação.

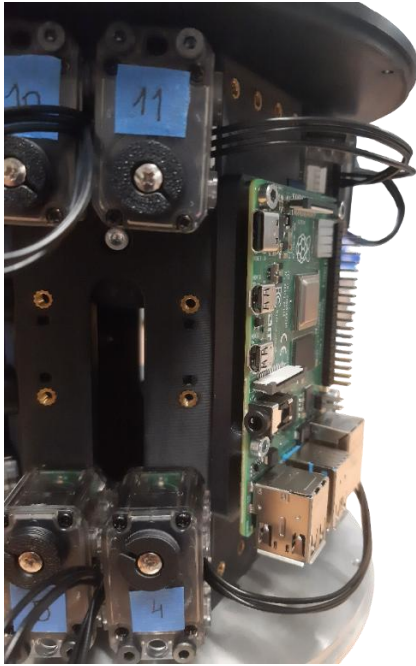


Figura 17: Raspberry Pi fixada no robô.

Fonte: Elaboração própria.

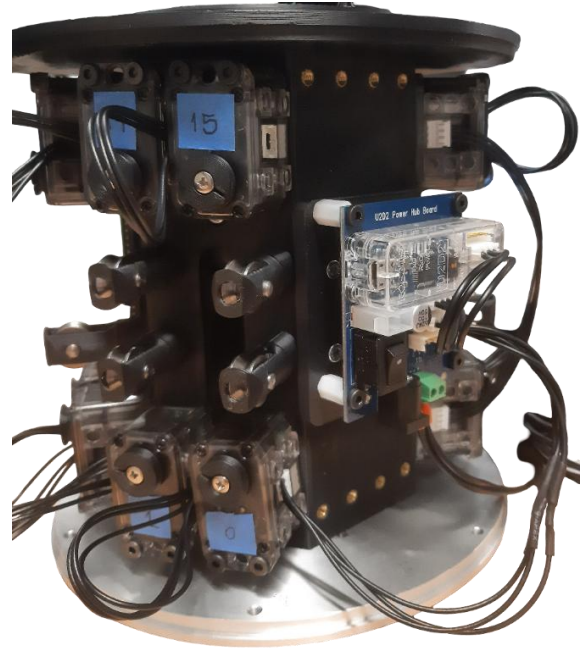


Figura 18: U2D2 fixada no robô.

Fonte: Elaboração própria.

Além dessas modificações estruturais já descritas, foram incorporados à unidade de acionamento um conector P4 e uma chave liga/desliga destinados à alimentação geral do sistema. Ambos os componentes foram fixados a uma carcaça lateral de proteção projetada para evitar o contato de agentes externos com os elementos internos do robô, conforme ilustrado na figura apresentada a seguir. A Figura 19 apresenta a comparação entre o modelo tridimensional da carcaça de proteção e a peça manufaturada.

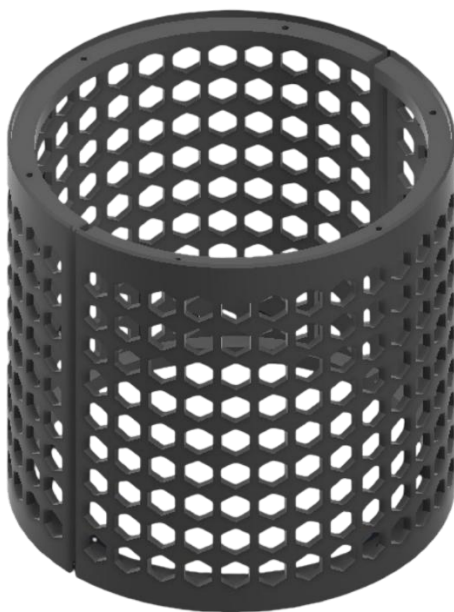


Figura 19: Modelo 3D da carcaça e peça manufaturada.

Fonte: Elaboração própria.

Já a Figura 20 ilustra a fixação do conector P4 e da chave de acionamento na estrutura lateral do robô.



Figura 20: Conector e chave fixadas na carcaça do robô.

Fonte: Elaboração própria.

3.3.2 Guia para os tendões da mão

A condução adequada dos tendões entre a mão robótica e os servomotores instalados nas laterais da unidade de acionamento constitui um elemento essencial para o funcionamento confiável do sistema. Considerando que a unidade de atuação foi projetada para operar até dezesseis tendões simultaneamente, tornou-se necessário desenvolver uma solução que permitisse encaminhar cada tendão ao seu respectivo motor de forma organizada, livre de interferências e com atrito minimizado. A partir dessa demanda, foram definidos dois requisitos principais: cada tendão deveria percorrer um trajeto exclusivo até o seu servo, evitando cruzamentos e enrolamentos, e esse trajeto deveria apresentar atrito reduzido de modo a preservar a precisão do movimento ao longo de várias iterações.

A primeira solução avaliada, ilustrada na Figura 21, consistiu no uso de tubos flexíveis de Politetrafluoretileno (PTFE, conhecido comercialmente como Teflon®) fixados a um disco superior e encaminhados pelo interior da unidade de acionamento até os pontos de saída dos tendões na estrutura central. Essa alternativa atendia aos requisitos de baixa fricção e de separação individual dos cabos, no entanto, devido à flexibilidade elevada do PTFE, verificou-

se que a tensão aplicada ao tendão durante o acionamento deformava o tubo, resultando em variações no movimento final do dedo. Assim, constatou-se que seria necessário incorporar algum elemento estrutural rígido capaz de manter os tubos alinhados e imobilizados.



Figura 21: Teste preliminar para guia dos tendões.

Fonte: Elaboração própria.

Com base nesse diagnóstico, projetou-se a peça ilustrada na Figura 22, que consiste em um cilindro de 70,50 milímetros de diâmetro, majoritariamente maciço, contendo em seu interior um conjunto de canais curvos para acomodação dos tubos de PTFE. Esses canais garantem o trajeto correto e independente para cada tendão, evitando interferências e preservando a geometria do guia ao longo da operação do robô.

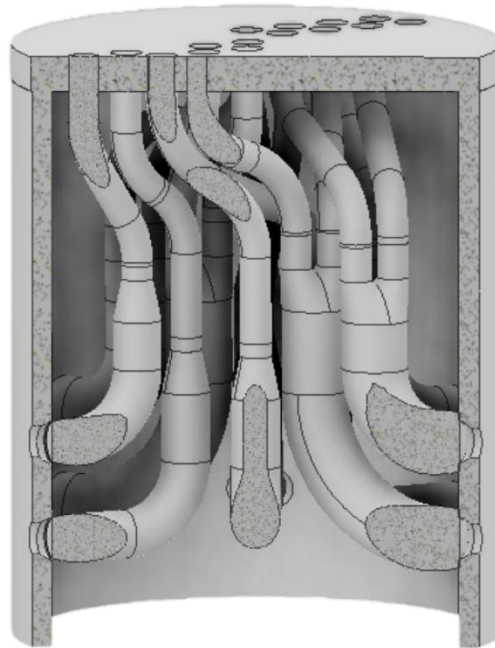


Figura 22: Vista de corte para visualização dos canais do modelo 3D.

Fonte: Elaboração própria.

Para viabilizar a montagem, a peça foi dividida em quatro seções empilháveis, unidas por parafusos M3 e roscas de inserção, conforme apresentado na Figura 23. As duas primeiras seções são responsáveis por desviar progressivamente os canais para as laterais do cilindro, acompanhando o posicionamento dos servomotores na estrutura central. As duas seções subsequentes direcionam os canais da orientação horizontal para a vertical e organizam os dezesseis túneis na mesma topologia em que os tendões emergem da mão robótica.

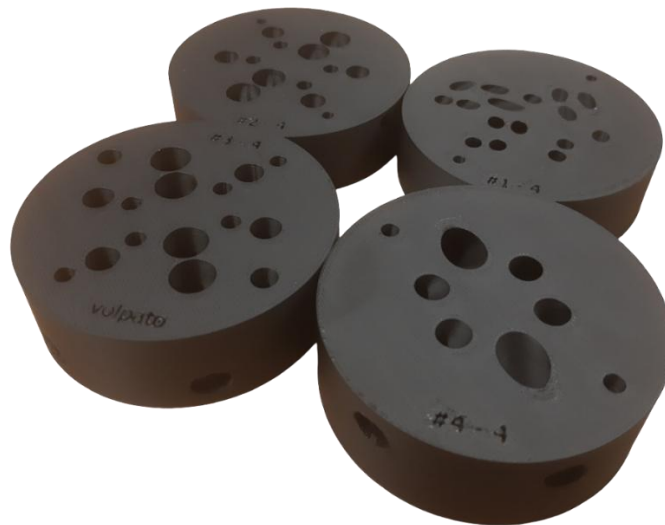


Figura 23: Seções responsáveis por guiar tubos PTFE.

Fonte: Elaboração própria.

A Figura 24 mostra o processo de inserção dos tubos de PTFE ao longo das seções internas da peça, ilustrando a transição de direção horizontal para vertical do tubo por meio dos canais modelados.



Figura 24: Tubos passando pelas seções da peça manufaturada.

Fonte: Elaboração própria.

A Figura 25 apresenta o conjunto final montado, onde é possível observar as roscas de inserção dispostas radialmente na superfície superior da peça, utilizadas para fixação firme da guia ao corpo estrutural central do robô.

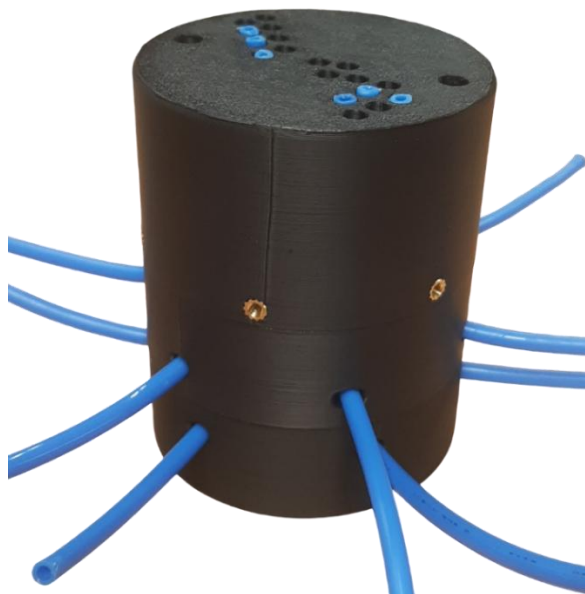


Figura 25: Conjunto montado.

Fonte: Elaboração própria.

3.3.3 Montagem do robô e passagem dos tendões pela estrutura

A etapa seguinte do desenvolvimento consistiu na montagem do robô a partir das peças previamente projetadas em ambiente CAD, bem como na integração do protótipo de mão antropomórfica ao conjunto de atuação motora. Essa fase envolveu tanto a montagem mecânica dos componentes estruturais quanto a passagem e fixação dos tendões ao longo dos canais de guia individuais previstos no projeto. A Figura 26 apresenta o modelo tridimensional completo do robô, utilizado como referência durante todo o processo de montagem.

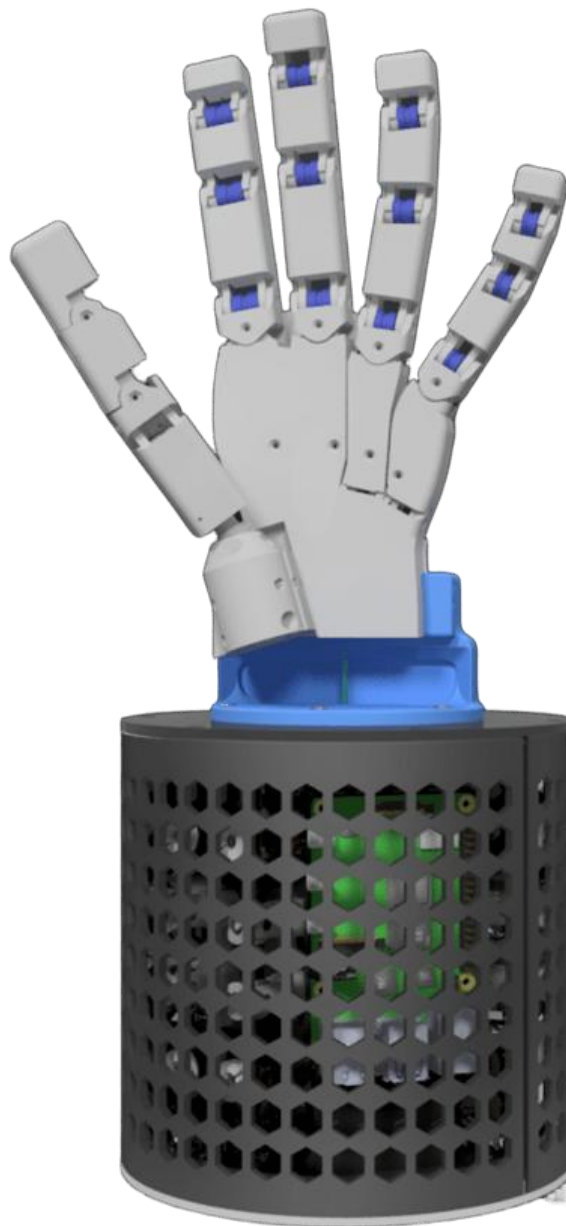


Figura 26: Modelo CAD do robô completo.

Fonte: Elaboração própria.

Para a passagem dos tendões pelos canais internos da estrutura, adotou-se uma técnica baseada no uso de uma linha guia. Inicialmente, essa linha foi inserida pelo furo de saída inferior do canal e conduzida até o topo com o auxílio de uma pistola de ar comprimido, que facilitou o deslocamento da guia ao longo do túnel interno. Em seguida, a extremidade da linha guia foi amarrada ao tendão, permitindo que este fosse puxado de volta pelo mesmo percurso. Após a condução de todos os tendões pelos respectivos canais, cada cabo foi direcionado ao seu sistema de polia, responsável por redirecionar o esforço mecânico até o eixo do servomotor correspondente. Em seguida, os tendões foram fixados a pequenas peças impressas em 3D, projetadas especificamente para serem acopladas ao eixo dos motores. Esse conjunto, composto pelo canal de guia, polia e elemento de fixação no eixo do servo, pode ser identificado no modelo tridimensional ilustrado na Figura 27.

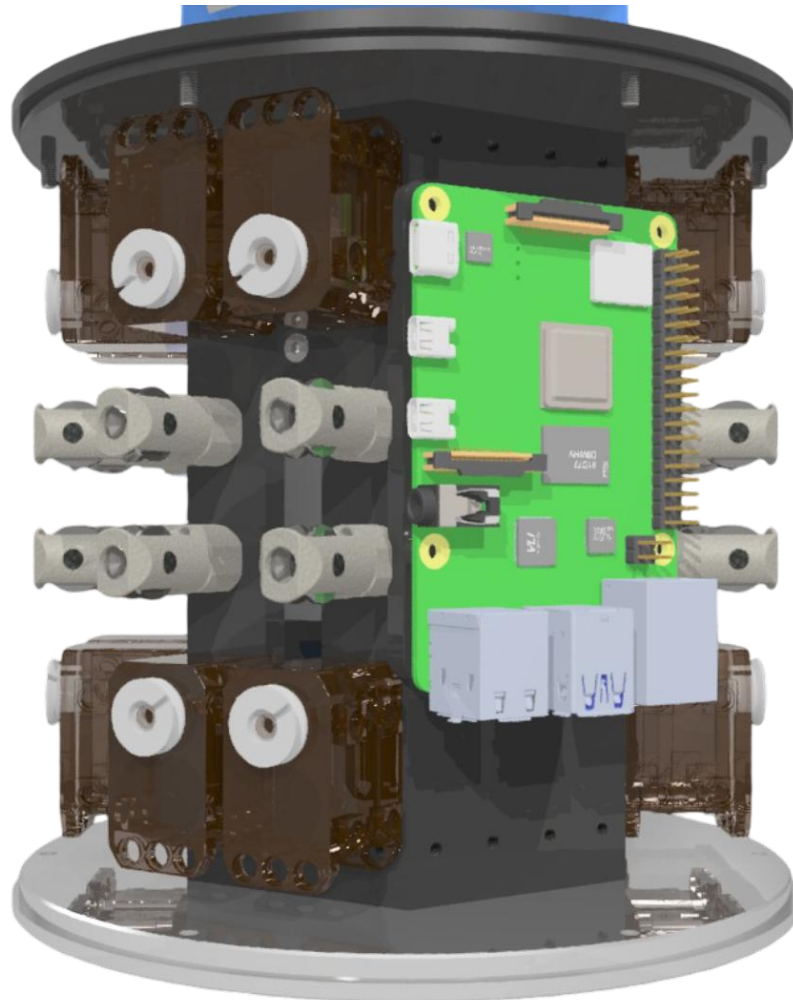


Figura 27: Sistema de guia para os tendões com polias.

Fonte: Elaboração própria.

Por fim, o resultado da montagem completa do robô, incluindo a integração estrutural e funcional com o protótipo de mão antropomórfica, é apresentado na Figura 28. Essa etapa conclui a fase de integração mecânica do sistema, preparando o conjunto para a fixação no braço robótico utilizado nos testes e descrito na seção seguinte.



Figura 28: Robô completo montado.

Fonte: Elaboração própria.

3.3.4 Fixação do robô de mão no braço robótico

Para viabilizar a integração mecânica entre a unidade de acionamento da mão robótica e o manipulador colaborativo utilizado nos experimentos, foi necessário desenvolver uma solução de acoplamento compatível com o padrão de montagem do braço UR20. A placa de alumínio, que faz parte do material preliminar desenvolvido previamente, apresentava o padrão de furação destinado ao modelo UR5, manipulador com capacidade de carga de 5 kg (Universal Robots, 2025). Considerando que diversos acessórios e garras projetados para o UR5 são comumente utilizados também no UR20, selecionado como braço colaborativo para este trabalho, optou-se pela aquisição de um flange comercial capaz de realizar a adaptação entre os dois modelos. O flange, ilustrado na Figura 29, permite que dispositivos originalmente concebidos para o UR5 sejam acoplados ao UR20 sem necessidade de alterações adicionais na interface mecânica.

Após a fixação da placa de alumínio ao flange adaptador e, conseqüentemente, ao braço robótico UR20, identificou-se a necessidade de estabelecer um método mais robusto para a fixação da unidade de acionamento. A solução inicialmente prevista, que utilizava a carcaça lateral impressa em 3D como elemento estrutural, não oferecia a rigidez mecânica necessária para garantir estabilidade durante a operação. Diante disso, foram usinados quatro espaçadores de alumínio, projetados para posicionar corretamente a estrutura central do robô em relação à placa de base e assegurar uma conexão firme, alinhada e estruturalmente confiável entre os componentes.



Figura 29: Modelo tridimensional do flange comercial utilizado.

Fonte: Elaboração própria.

3.4 ARQUITETURA DE SOFTWARE

3.4.1 Configuração da interface de controle

A interação entre o operador e a Raspberry Pi utilizada no sistema foi realizada de forma totalmente remota, por meio de uma conexão SSH estabelecida a partir de um computador com sistema operacional Windows. Para isso, adotou-se o Visual Studio Code como ambiente de desenvolvimento, aproveitando sua extensão nativa para acesso remoto via SSH, que oferece uma integração bem estruturada para o desenvolvimento em sistemas embarcados.

A configuração inicial consiste em instalar a extensão *Remote – SSH* no VS Code e registrar o endereço IP da Raspberry Pi dentro do arquivo de configuração da extensão. Uma vez estabelecida a conexão, o VS Code passa a operar diretamente no ambiente Linux da placa, permitindo ao usuário executar comandos no terminal da Raspberry Pi sem sair da interface gráfica do *software*. Dessa forma, tornou-se possível compilar pacotes ROS 2, iniciar nós, monitorar tópicos, editar arquivos de configuração e manipular o sistema de arquivos da placa de maneira fluida.

Além do terminal integrado, o VS Code disponibiliza a visualização completa da árvore de diretórios da Raspberry Pi, simplificando a navegação entre pacotes, nós e arquivos de código. Essa funcionalidade é especialmente útil durante o desenvolvimento do firmware no ROS, pois viabiliza alterações rápidas no código, reexecução imediata dos nós e acompanhamento dos resultados diretamente no console do *software*.

3.4.2 Definição da estrutura do sistema

A estrutura de *software* do sistema foi organizada em três blocos operacionais, implementados como nós ROS 2 distintos que se comunicam entre si por meio de tópicos. Cada nó desempenha uma função específica dentro da arquitetura geral e interage com os demais apenas quando necessário. A Figura 30 apresenta o fluxograma geral da arquitetura definida.

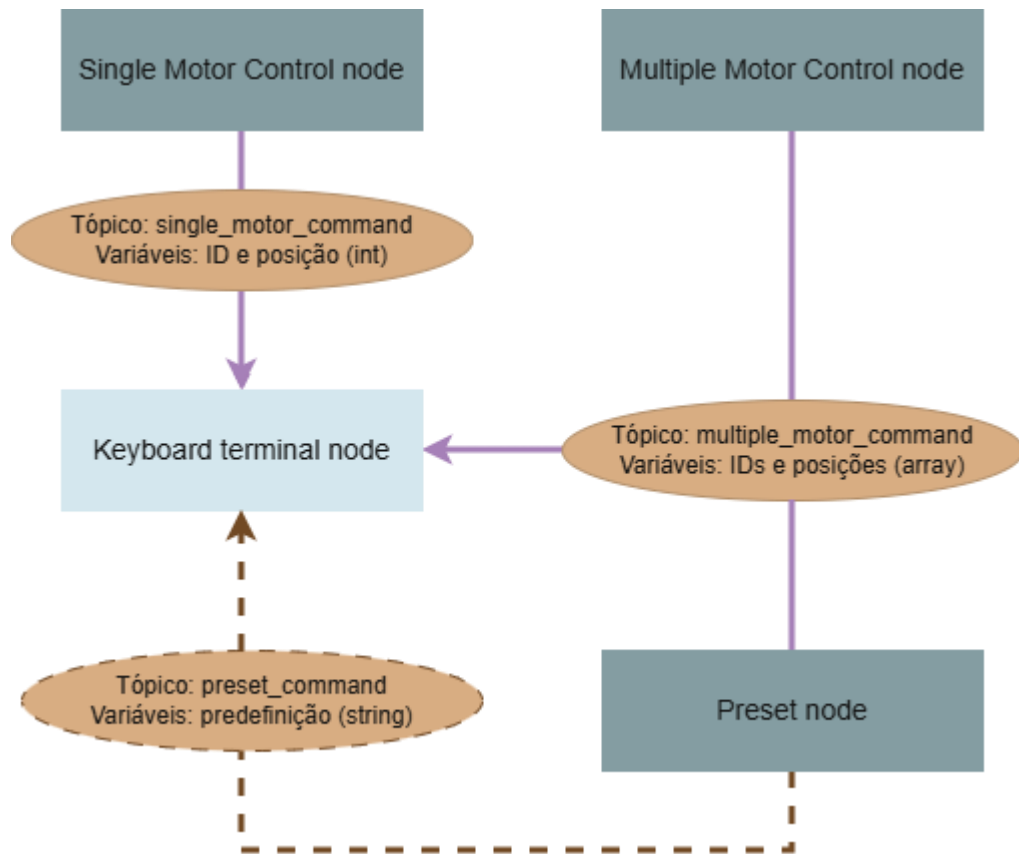


Figura 30: Representação da estrutura de *software* do sistema robótico.

Fonte: Elaboração própria.

O primeiro nó, em azul claro, é responsável pela interface de controle direto da mão robótica. Ele interpreta os comandos inseridos pelo usuário por meio do teclado conectado à Raspberry Pi via SSH, identifica a tecla pressionada, converte a entrada em um comando digital compatível com o protocolo dos servomotores e determina o identificador do motor selecionado. Esse nó funciona como a principal interface de interação entre o operador e o sistema.

O segundo nó, denominado *preset node*, é dedicado ao gerenciamento de predefinições de movimento coordenado e funciona como um tradutor de teclas de predefinição para comandos do motor. Nele são definidas funções de movimento compostas, nas quais cada predefinição especifica os servomotores envolvidos e suas posições finais desejadas. Para adicionar uma nova predefinição, basta registrar uma função de movimento associada a uma tecla de ativação. Quando o usuário aciona uma dessas teclas, o nó consulta a predefinição correspondente e encaminha os comandos necessários ao nó de atuação.

Os dois últimos nós executam a atuação dos servomotores e foi desenvolvido a partir do nó de referência *read_write_node* do pacote Dynamixel SDK. Ele recebe, como assinante, as mensagens publicadas pelos dois nós anteriores, contendo o ID do servomotor a ser movimentado e a posição de destino. A partir dessas informações, publica o comando final no tópico *set_position*, realizando o acionamento. Trata-se, portanto, do nó responsável pela interface direta com o *hardware*.

4 RESULTADOS

4.1 TESTE DO FIRMWARE

4.1.1 Função de aquisição de teclas

O nó de aquisição de teclas monitora continuamente três teclas principais: A, D e Q. A tecla A reduz o valor da posição enviada ao servo, fazendo com que o motor rotacione no sentido correspondente à diminuição de ângulo, enquanto a tecla D aumenta esse valor, gerando o movimento no sentido oposto. Já a tecla Q encerra a execução do nó e finaliza o controle por teclado. Todas as demais teclas são ignoradas, mantendo o sistema restrito aos comandos definidamente implementados.

Quando o nó é iniciado, o usuário deve escolher entre dois modos: controle de um único motor ou controle simultâneo de múltiplos motores, conforme ilustra a Figura 31. No modo de motor único, o sistema solicita que o usuário informe o ID do servo a ser controlado e armazena esse valor na variável `self.id`. A cada pressionamento das teclas A ou D, o console exibe uma mensagem no formato “*M: ID → P: posição*”, indicando de forma explícita qual motor está sendo atualizado e qual o novo valor de posição enviado ao nó de atuação, valor esse que foi definido para incrementar a cada 20 passos, correspondendo a aproximadamente 5,86 graus uma vez que o curso total do motor é de 300 graus divididos em 1024 unidades. Já no modo múltiplo, o sistema solicita quantos motores serão controlados e seus respectivos IDs. Nesse caso, a cada comando enviado, o console apresenta “*Motors: [lista de IDs] → Positions: [lista de posições]*”, refletindo o deslocamento simultâneo de todos os servos selecionados.

```
Choose mode: [s]ingle / [m]ultiple: m
How many motors to control? 3
Type motor ID #1: 0
Type motor ID #2: 1
Type motor ID #3: 2
Keyboard control started for motors [0, 1, 2]. Use A/D to move all together, Q to exit.
```

Figura 31: Interface de controle simultâneo de três motores escolhidos.

Fonte: Elaboração própria.

No modo de controle individual, o nó publica mensagens do tipo *SingleMotorCommand* no tópico chamado *single_motor_command*. Cada mensagem contém dois parâmetros principais: o ID do motor selecionado na variável `msg.id` e a posição desejada em `msg.position`, que é atualizada conforme o usuário pressiona as teclas A ou D. Da mesma forma, no modo múltiplo, o nó publica mensagens do tipo *MultipleMotorCommand* no tópico

multiple_motor_command, contendo uma lista de IDs e as respectivas posições de cada motor controlado simultaneamente.

Essas mensagens são transmitidas continuamente sempre que uma tecla válida é acionada, de modo que nós inscritos nesses tópicos (*subscribers*), como o nó de atuação dos servos descrito na seção seguinte, recebem instantaneamente as atualizações e executam o movimento correspondente. Dessa forma, o nó de aquisição de teclado não atua diretamente nos motores, mas sim envia comandos para os demais blocos do sistema por meio dos mecanismos nativos de comunicação assíncrona do ROS 2.

4.1.2 Função de controle de um único servo

O nó responsável pelo controle individual de cada servomotor tem como função receber, interpretar e executar os comandos publicados pelo nó de interface de teclado. Assim que iniciado, ele realiza a configuração de comunicação com o barramento DYNAMIXEL, abrindo a porta serial, definindo a taxa de transmissão para 1MHz e inicializando o protocolo utilizado pelos servos. Após o estabelecimento dessa comunicação, o nó realiza uma etapa de inicialização em que todos os servos previamente instalados no sistema são configurados para operar no modo de posição, têm seu torque habilitado e recebem um sinal visual por meio da iluminação de seus LEDs em cor ciano. Durante essa etapa, o console apresenta mensagens informando os motores detectados e eventuais erros de comunicação, o que facilita o diagnóstico do sistema.

Após a inicialização, o nó passa a atuar como um assinante do tópico destinado ao controle individual, chamado *single_motor_command*. Assim que a mensagem é recebida, o nó executa o seguinte procedimento: ativa temporariamente o LED do motor selecionado para a cor verde, conforme a Figura 32, indicando visualmente que aquele atuador está sendo acionado, e então envia o comando de posição ao motor correspondente. Em seguida, são produzidas mensagens no console informando o ID do servo e a posição para a qual ele foi movimentado. Caso ocorra algum erro de comunicação ou resposta inválida do motor, o nó também registra essas informações no console de forma clara, permitindo ao operador identificar rapidamente falhas no barramento ou no servo específico.



Figura 32: Robô com LEDs verdes acesos.

Fonte: Elaboração própria.

Ao final da operação, quando o sistema é encerrado pelo usuário, o nó executa um procedimento de desligamento seguro, apagando o LED de todos os servos previamente configurados e encerrando a comunicação com o barramento.

4.1.3 Função de controle de múltiplos servos

O nó responsável pelo controle simultâneo de múltiplos servomotores segue a mesma estrutura lógica do nó dedicado ao acionamento individual, porém incorpora modificações importantes para permitir o envio de comandos conjuntos para vários atuadores. O nó foi construído a partir do código precursor de controle unitário, ampliando sua capacidade por meio de uma nova estrutura de mensagem, denominada *MultipleMotorCommand*, que contém dois vetores: um vetor de identificadores dos motores e um vetor correspondente de posições desejadas.

Assim como no código anterior, o nó realiza inicialmente a configuração completa dos dezesseis motores presentes na unidade de atuação, definindo o modo de operação, habilitando o torque e acendendo os LEDs na cor roxa para indicar que o processo de inicialização foi concluído. A principal diferença em relação ao nó dedicado ao controle de um único motor está na estrutura das mensagens e no tópico em que o nó se inscreve. Enquanto o nó anterior recebe uma única combinação de ID e posição por meio do tópico *single_motor_command*, este nó se inscreve no tópico *multiple_motor_command*, que transporta duas listas paralelas: uma contendo todos os IDs dos servos que devem ser acionados e outra contendo as posições alvo correspondentes. Dessa forma, cada elemento da lista de posições é aplicado ao servo localizado na mesma posição relativa da lista de IDs, permitindo o acionamento simultâneo e coordenado de vários motores. A Figura 33 representa o robô com os LEDs acesos em roxo, cor que indica que a função de controle de múltiplos servos está ativa.



Figura 33: Robô com LEDs roxos acesos.

Fonte: Elaboração própria.

Quando uma mensagem contendo a lista de identificadores e a lista de posições chega ao nó, ele percorre cada par ID–posição e envia o valor correspondente ao motor adequado.

Esse envio é realizado por meio da função de escrita do Dynamixel SDK, que transmite ao servo o valor numérico da posição desejada exatamente como definido no protocolo dos motores XL-320. A chamada do método `write2ByteTxRx` envia dois bytes ao registrador interno do servo destinado ao controle de posição, o que permite que cada motor receba seu comando de maneira independente e precisa. A Figura 34 abaixo mostra a publicação de uma mensagem de sucesso no console ao realizar o movimento de motores simultaneamente.

```
[INFO] [1749058639.182282361] [read_write_node]: Succeeded to open the port.
[INFO] [1749058639.186969639] [read_write_node]: Succeeded to set the baudrate.
[INFO] [1749058639.207495972] [read_write_node]: Succeeded to enable torque.
[INFO] [1749058770.178249968] [read_write_node]: Set [ID: 0] [Goal Position: 0]
[INFO] [1749058781.489648647] [read_write_node]: Set [ID: 1] [Goal Position: 0]
[INFO] [1749058795.024351565] [read_write_node]: Set [ID: 2] [Goal Position: 0]
[INFO] [1749058803.713324263] [read_write_node]: Set [ID: 0] [Goal Position: 20]
[INFO] [1749058895.051888283] [read_write_node]: Set [ID: 1] [Goal Position: 20]
[INFO] [1749058900.859663464] [read_write_node]: Set [ID: 2] [Goal Position: 20]
```

Figura 34: Registro no console de movimento simultâneo de três servomotores.

Fonte: Elaboração própria.

Se ocorrer qualquer falha durante essa transmissão, como erro de comunicação ou rejeição do pacote pelo motor, o nó registra a ocorrência no console para fins de diagnóstico. Caso contrário, ele confirma a movimentação imprimindo o ID do motor e a nova posição aplicada. Esse comportamento garante que o operador possa acompanhar a atuação simultânea dos motores e verificar, em tempo real, se todos receberam corretamente seus respectivos comandos.

4.1.4 Função que define os padrões de movimento

A função de *presets* foi desenvolvida com o objetivo de permitir que a mão robótica realizasse movimentos coordenados de forma automática, sem que o operador precisasse atuar motor por motor ou enviar comandos incrementais pelo teclado. Essa abordagem permite a criação de padrões de movimento repetidos, como a abertura total da mão ou combinações específicas de articulações a serem utilizadas em testes. Para isso, foi implementado um nó dedicado denominado *preset_node*, responsável por receber o nome de um *preset*, identificar os motores associados ao movimento e publicar os comandos apropriados para que o nó de controle múltiplo execute as posições programadas.

O nó inicia criando um *subscriber* para o tópico que recebe nomes de predefinições, denominado *preset_command*, que utiliza mensagens do tipo *PresetCommand*. Sempre que um nome de preset é publicado pelo nó de leitura do teclado, ele é recebido e processado pela

função *preset_callback*. Paralelamente, o nó cria um *publisher* para o tópico *multiple_motor_command*, que utiliza mensagens do tipo *MultipleMotorCommand* e é consumido pelo nó de controle simultâneo de servos. Dessa forma, o nó atua como um componente intermediário na arquitetura do sistema, convertendo uma solicitação em comandos de posição compreensíveis pelos motores.

Dentro do próprio nó é mantida uma estrutura de dados que armazena os *presets* disponíveis, cada um contendo uma lista de IDs de motores e uma lista correspondente de posições desejadas. Um exemplo dessa estrutura pode ser observado nos dois *presets* utilizados durante os testes iniciais. O *preset* denominado *open_hand* contém cinco motores e suas respectivas posições finais configuradas para a abertura completa dos dedos, enquanto o *preset test* consiste nos quatro motores que controlam a flexão e extensão de todos os dedos com exceção do polegar selecionados para fechar a mão completamente. Essa organização pode ser ilustrada na Figura 35.

```
self.presets = {
    "open_hand": {
        "ids": [1,2,3,4,5],
        "positions": [0, 0, 0, 0, 0]
    },
    "test": {
        "ids": [0,6,10,12],
        "positions": [600, 800, 480, 400]
    },
}
```

Figura 35: Estrutura de definição de uma predefinição no código.

Fonte: Elaboração própria.

Quando um comando é recebido, o *preset* é identificado e validado. O nó verifica se o nome enviado corresponde a alguma das predefinições disponíveis e garante que o número de IDs e o número de posições associados ao *preset* são iguais, evitando falhas de execução. Após essa verificação, o nó constrói uma mensagem do tipo *MultipleMotorCommand*, preenche os campos referentes aos IDs dos motores e suas posições e publica a mensagem no tópico destinado ao controle simultâneo. O nó responsável por acionar fisicamente os servos recebe essa mensagem e a executa motor a motor, permitindo que todos os movimentos correspondentes ao *preset* ocorram de forma sincronizada.

O console apresenta mensagens informativas durante todo o processo, como a confirmação de inicialização do nó, avisos quando *presets* vazios ou inexistentes são recebidos e relatórios detalhando quais motores e posições foram enviados para execução. Essas informações são essenciais para depuração e monitoramento durante os testes. Quando um *preset* válido é acionado, o console exibe a mensagem “*Published preset 'nome_do_preset' -> ids: [...], positions: [...]*”, permitindo ao usuário acompanhar exatamente o comando que foi encaminhado ao sistema, conforme ilustrado na Figura 36.

```
[INFO] Preset node started. Waiting for preset commands.
[INFO] Published preset 'open_hand' -> ids: [1, 2, 3, 4, 5], positions: [0, 0, 0, 0, 0]
[INFO] Published preset 'test' -> ids: [0, 6, 10, 12], positions: [600, 800, 480, 400]
[WARN] Empty preset name received.
[ERROR] Preset 'close_hand' not found.
```

Figura 36: Retorno no console da função de predefinições de movimento.

Fonte: Elaboração própria.

4.2 INTEGRAÇÃO DOS BLOCOS DE COMANDO

A integração completa entre os nós no ROS permitiu validar o funcionamento conjunto da interface de controle por teclado, dos módulos de atuação individual e múltipla dos servomotores e da lógica de predefinições de movimento. A comunicação entre esses nós ocorre exclusivamente por meio de tópicos, garantindo modularidade e permitindo que cada componente do sistema execute sua função específica sem dependências diretas. Durante os testes de integração, foi verificado tanto o funcionamento adequado das funções implementadas quanto limitações importantes que serviram de base para o aperfeiçoamento do sistema, especialmente no que diz respeito à detecção e tratamento de erros dos servomotores.

O processo de integração inicia pela sequência de execução dos três consoles principais, conforme ilustrado na Figura 37. O primeiro console corresponde ao nó de controle de teclado, que monitora continuamente as teclas pressionadas pelo usuário via SSH e publica comandos no tópico correspondente. O segundo console executa o nó responsável pelo controle direto dos servos e se inscreve no tópico no qual os dados são publicados. O terceiro console apresenta o nó de predefinições de movimento, que interpreta comandos recebidos e redireciona para ações coordenadas dos servomotores.

```
[INFO] [1749058639.182282361] [read_write_node]: Succeeded to open the port.
[INFO] [1749058639.186969639] [read_write_node]: Succeeded to set the baudrate.
[INFO] [1749058639.207495972] [read_write_node]: Succeeded to enable torque.
[INFO] [1749058803.713324263] [read_write_node]: Set [ID: 0] [Goal Position: 600]
[INFO] [1749058895.051888283] [read_write_node]: Set [ID: 6] [Goal Position: 800]
[INFO] [1749058900.859663464] [read_write_node]: Set [ID: 10] [Goal Position: 480]
[INFO] [1749058900.859663464] [read_write_node]: Set [ID: 12] [Goal Position: 400]

Choose mode: [s]ingle / [m]ultiple: m
How many motors to control? 3
Type motor ID #1: 0
Type motor ID #2: 1
Type motor ID #3: 2
Keyboard control started for motors [0, 1, 2]. Use A/D to move all together, Q to exit.
Preset command 'test' triggered by keyboard input (key: 'p').

[INFO] Preset node started. Waiting for preset commands.
[INFO] Published preset 'test' -> ids: [0, 6, 10, 12], positions:
[600, 800, 480, 400]
```

Figura 37: Exemplo de integração dos nós pela interface do usuário.

Fonte: Elaboração própria.

Apesar desse funcionamento consistente, a fase de integração evidenciou um conjunto de limitações que dificultaram a realização de testes contínuos com a mão robótica. O principal problema identificado foi a ausência de um nó dedicado à leitura dos registradores de erro e de *shutdown* dos servomotores, informações essenciais para detecção de falhas como sobrecarga, sobreaquecimento, tensão fora da faixa operacional ou travamento mecânico do eixo. De acordo com a Figura 38, esses erros são sinalizados por meio dos bits do registrador de *shutdown*, cada um indicando uma condição específica.

Bit	Item	Description
Bit 3~7	-	-
Bit 2	Input Voltage Error	Voltage is out of operational voatage range
Bit 1	OverHeating Error	Temperature is out of operational temperature range
Bit 0	Overload Error	Motor cannot output max load due to load being applied continously

Figura 38: Registrador de *shutdown* do servomotor XL-320.

Fonte: Robotis, 2025.

O manual do fabricante explica que, quando qualquer uma dessas condições ocorre, o motor interrompe a saída de torque e passa a piscar o LED em amarelo ou vermelho, sinalizando o estado de alarme. O LED permanece nesse padrão até que o registrador responsável seja

alterado manualmente, o que caracteriza a necessidade de realizar um *hard reset*. Um nó dedicado a essa função seria capaz de ler continuamente os registradores apropriados, identificar qual bit foi ativado e exibir, no console, uma descrição compreensível do erro ocorrido, assim como o motor afetado. Além disso, tal nó permitiria implementar automaticamente o procedimento de reset do motor, escrevendo o valor adequado no registrador responsável para liberar novamente o torque (Robotis, 2025).

4.3 INTEGRAÇÃO COM O MOVIMENTO DO BRAÇO ROBÓTICO

A etapa de integração entre a unidade de atuação dos tendões e o braço robótico UR20 permitiu avaliar o comportamento da mão robótica em situações reais de movimento dentro do espaço de trabalho do manipulador. A possibilidade de deslocar o conjunto através de diferentes orientações no espaço possibilitou observar a resposta dos dedos sob influência simultânea da gravidade, do tensionamento dos tendões e da ação das fitas elásticas responsáveis pelo retorno ao estado de repouso.

Durante os testes iniciais, verificou-se que os dedos da mão apresentavam dificuldade em retornar completamente à posição de repouso após a execução de movimentos repetidos. Essa limitação estava associada ao desgaste das fitas elásticas utilizadas como elementos de retorno passivo, mecanismo previamente discutido na seção de fundamentação teórica. Como ilustrado na Figura 39, as fitas já apresentavam sinais de fadiga e perda significativa de sua capacidade de recuperação, reduzindo a eficiência do retorno elástico.

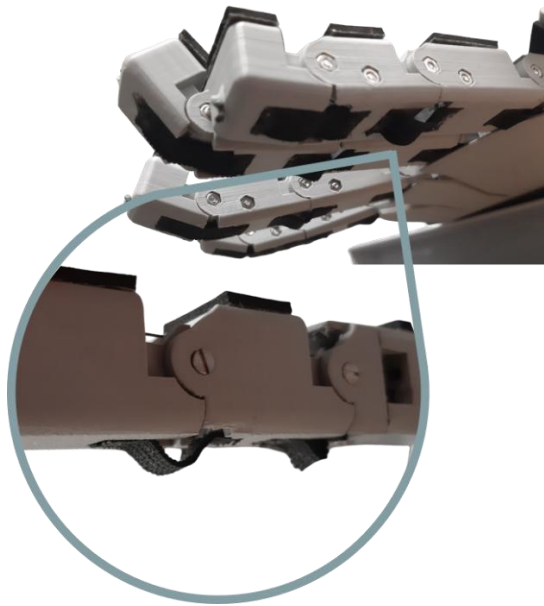


Figura 39: Identificação da fadiga nas fitas elásticas de alguns dedos.

Fonte: Elaboração própria.

Considerando que alterações estruturais mais profundas seriam realizadas apenas na próxima versão do protótipo de mão, que substituiria as fitas elásticas por molas helicoidais, uma solução provisória foi implementada com o objetivo de viabilizar os testes. Porcas M8 foram instaladas nas pontas dos dedos, adicionando massa às falanges distais, com a finalidade de acrescentar massa para utilizar a gravidade como auxílio ao retorno dos dedos, compensando parcialmente a degradação das fitas elásticas. A Figura 40 apresenta essa adaptação temporária, que permitiu testar movimentos repetidos sem comprometer a integridade dos tendões ou dos servomotores.

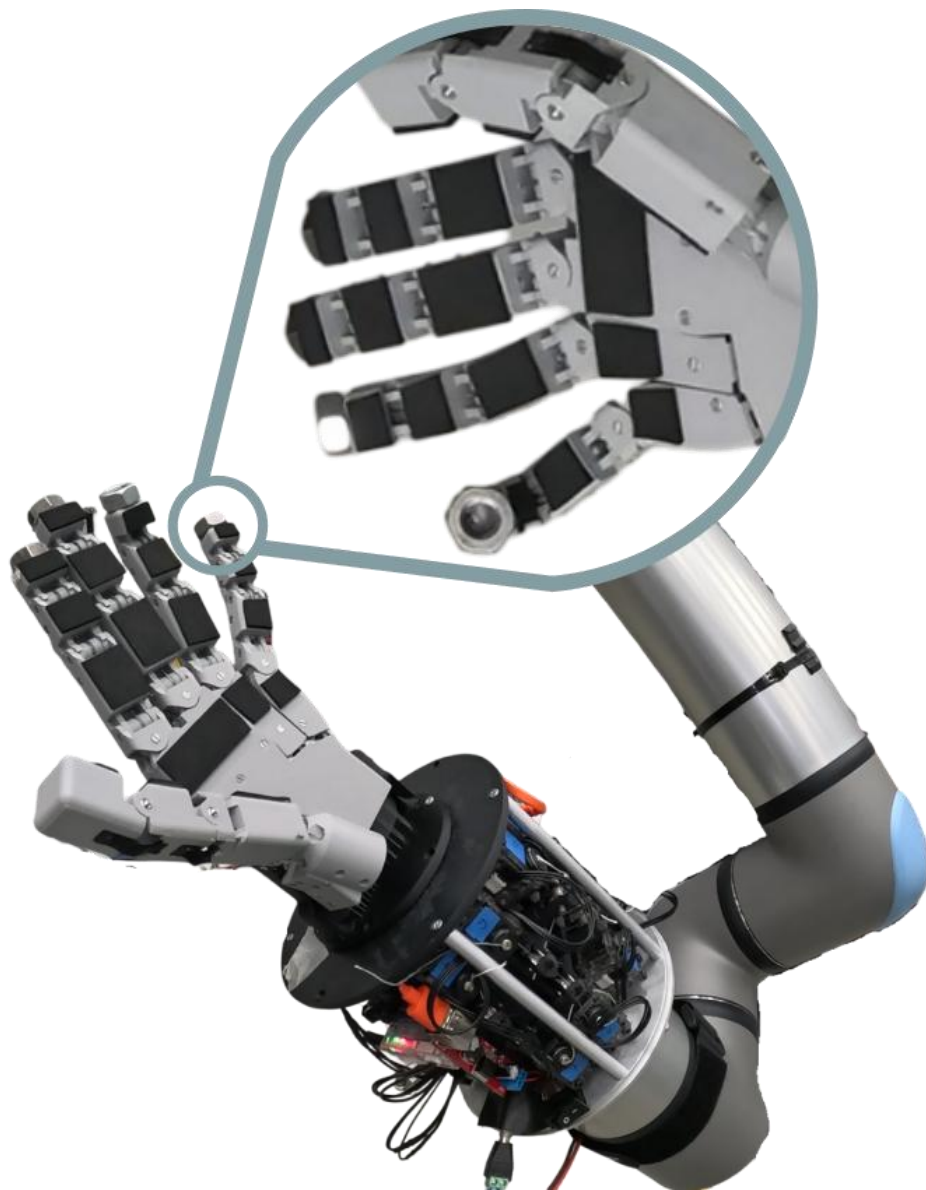


Figura 40: Adaptação com porcas na ponta dos dedos.

Fonte: Elaboração própria.

Além dessa modificação, foi possível explorar o uso do próprio movimento do braço robótico para otimizar o retorno dos dedos. Uma vez que o conjunto estava firmemente acoplado ao UR20, foi possível posicionar a mão em diferentes orientações no espaço. Uma das abordagens consistiu em inclinar o manipulador de modo que a mão permanecesse quase invertida, conforme ilustrado na Figura 41, permitindo que a ação gravitacional favorecesse a retração natural dos dedos. Embora essa medida não tenha sido suficiente para restaurar completamente o retorno de repouso, contribuiu para reduzir a variabilidade dos testes e possibilitou a execução de ciclos de movimento de forma repetitiva.

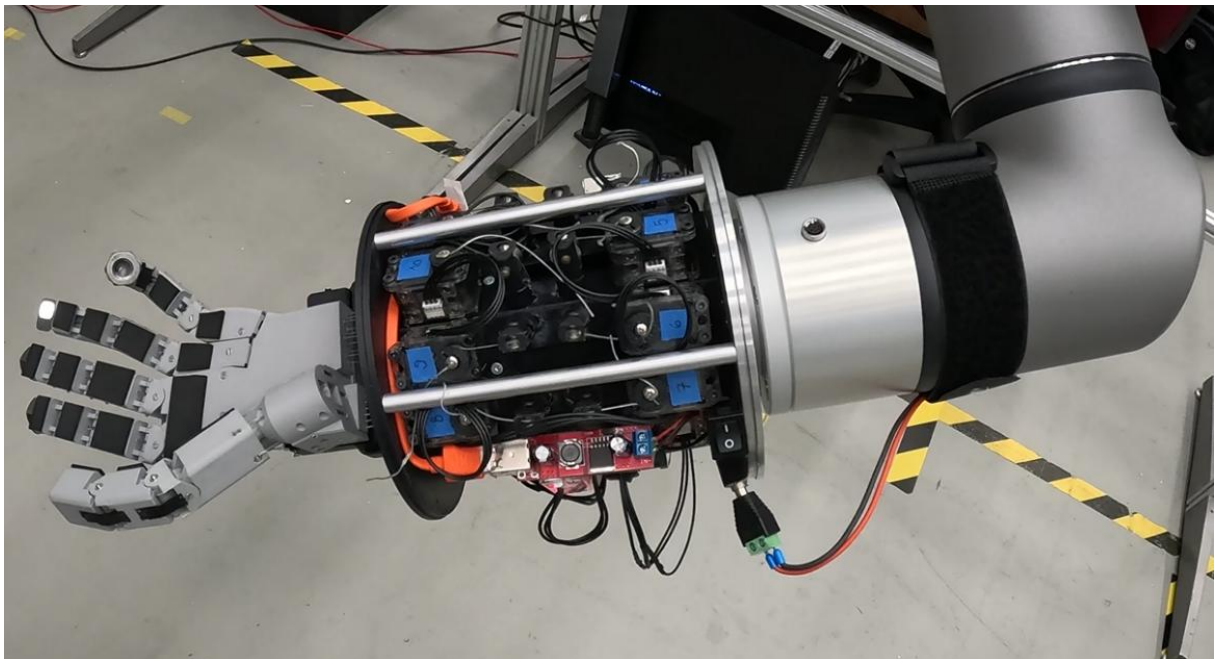


Figura 41: Posicionamento inclinado do robô com auxílio do braço robótico.

Fonte: Elaboração própria.

5 CONCLUSÕES

A realização deste trabalho demonstrou a viabilidade técnica do desenvolvimento de uma unidade motora dedicada à atuação de tendões de uma mão robótica antropomórfica, integrando aspectos de modelagem mecânica, prototipagem, arquitetura de *software* e validação experimental. Os resultados obtidos evidenciam que a abordagem adotada é adequada para aplicações em robótica de manipulação, especialmente em sistemas que demandam coordenação entre múltiplos atuadores e integração com plataformas robóticas de maior porte.

A etapa de modelagem em *software* CAD permitiu a concepção completa da unidade motora, considerando não apenas a acomodação dos servomotores e dos elementos de transmissão por tendões, mas também restrições geométricas, pontos de fixação e compatibilidade com a mão antropomórfica e com o braço robótico utilizado nos testes. As modelagens tridimensionais desenvolvidas ao longo do trabalho mostraram-se fundamentais para a visualização do conjunto, antecipação de interferências mecânicas e suporte às decisões de projeto, constituindo uma base sólida para futuras iterações e aprimoramentos do protótipo.

A prototipagem da unidade motora por meio de componentes impressos em filamento 3D viabilizou a montagem física do sistema e a integração mecânica e elétrica de todos os elementos. Esse processo evidenciou tanto as vantagens da manufatura aditiva, como rapidez de iteração e baixo custo, quanto suas limitações, especialmente no que diz respeito à resistência mecânica e ao comportamento de componentes auxiliares, como os elementos elásticos utilizados para o retorno passivo dos dedos. Ainda assim, o protótipo permitiu a validação prática do conceito proposto e forneceu subsídios importantes para melhorias a serem incorporadas em versões futuras.

No âmbito do *software*, a implementação de uma arquitetura baseada em ROS 2 mostrou-se eficaz para a coleta de instruções do usuário, interpretação de comandos e envio das informações aos atuadores do sistema. A separação funcional em nós distintos, responsáveis pela aquisição de entradas, definição de *presets* e atuação dos servomotores, contribuiu para a modularidade, clareza e escalabilidade do sistema. Essa organização facilita tanto a depuração quanto a expansão do projeto, possibilitando a incorporação de novas funcionalidades, como estratégias de controle mais avançadas ou mecanismos automáticos de detecção e tratamento de falhas.

Os testes realizados para validação do dispositivo permitiram verificar a integração entre *software*, *hardware* e mecanismo, além de evidenciar limitações práticas do sistema, como dificuldades de retorno à posição de repouso dos dedos e a ausência de um nó dedicado

ao diagnóstico e tratamento de erros dos servomotores. Essas limitações, longe de invalidarem o projeto, reforçam a importância da fase experimental como ferramenta de aprendizado e refinamento, indicando caminhos claros para evoluções futuras, como a adoção de molas no lugar de fitas elásticas e o desenvolvimento de rotinas de monitoramento e recuperação de falhas.

Acredita-se que o desenvolvimento deste trabalho contribua para o avanço de estudos relacionados à robótica antropomórfica e à integração de sistemas mecatrônicos complexos, especialmente no contexto acadêmico. Além de validar uma solução funcional, o projeto gera um material técnico detalhado que pode servir como referência para estudantes e pesquisadores interessados em arquitetura de controle, prototipagem rápida e integração de atuadores inteligentes em sistemas robóticos. Dessa forma, espera-se que este trabalho não apenas consolide os objetivos propostos, mas também fomente novas investigações e aprimoramentos na área de robótica de manipulação.

REFERÊNCIAS

BIAGIOTTI, L. *et al.* How Far Is the Human Hand? A Review on Anthropomorphic Robotic End-effectors. *[S.l.: S.n.]*.

CLARK, Angus B.; LIOW, Lois; ROJAS, Nicolas. Force Evaluation of Tendon Routing for Underactuated Grasping. *Journal of Mechanical Design*, v. 143, n. 10, 1 out. 2021.

COBOS, Salvador *et al.* 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems. *[S.l.]*: I E E E, 2008.

EL-SHENNAWY, Maged *et al.* Three-dimensional kinematic analysis of the second through fifth carpometacarpal joints. *The Journal of Hand Surgery*, v. 26, n. 6, p. 1030–1035, nov. 2001.

IGMR. Intelligent Robotic Skills for the Circular Economy. Disponível em: <<https://www.igmr.rwth-aachen.de/cms/igmr/forschung/projekte/konzepte-und-eigenentwicklungen/~bjgxgw/di-monta/?lidx=1>>. Acesso em: 19 nov. 2025.

KERKHOF, Faes D.; VAN LEEUWEN, Timo; VEREECKE, Evie E. The digital human forearm and hand. *Journal of Anatomy*, v. 233, n. 5, p. 557–566, 1 nov. 2018.

KIVELL, Tracy L. *et al.* Form, function and evolution of the human hand. *American Journal of Biological Anthropology*, v. 181, n. S76, p. 6–57, 1 ago. 2023.

MARTINS, Daniel; MURAI, Estevan Hideki. *Mecanismos: Síntese e análise com aplicações em robótica*. 1. ed. Florianópolis: EdUFSC, 2019.

MENDEZ, Vincent *et al.* Current Solutions and Future Trends for Robotic Prosthetic Hands Neuroprosthesis: a device that connects to the nervous system and either replaces missing parts of it or improves it. Downloaded from www.annualreviews.org. Guest (guest, v. 4, p. 1, 2025).

PEÑA-PITARCH, Esteban; FALGUERA, Neus Ticó; YANG, Jingzhou (James). Virtual human hand: model and kinematics. *Computer Methods in Biomechanics and Biomedical Engineering*, v. 17, n. 5, p. 568–579, 2014.

POLZIN, Sebastian. Einfluss der Gelenkkonfiguration auf die Greifperformance einer anthropomorphen Roboterhand. Aachen: [S.n.].

RASPBERRY PI LTD. Raspberry Pi 4 Model B Specifications.

ROBOTIS. Manual eletrônico do servo motor Dynamixel XL-320. Disponível em: <<https://emanual.robotis.com/docs/en/dxl/x/xl320/>>. Acesso em: 19 nov. 2025.

SALGADO DA SILVA, Esdras. Optimized Dimensional Synthesis for Limb Prostheses and Orthoses Mechanisms. [S.l.: S.n.].

STANDRING, Susan *et al.* Gray's Anatomy. The Anatomical Basis of Clinical Practice. [S.l.]: Elsevier Limited, 2016.

STEFANO, Aliberti. Journal of Morphology and Anatomy Perspective The Fascinating Functional Anatomy of the Human Hand. 2023.

STILLFRIED, Georg; VAN DER SMAGT, Patrick. Movement model of a human hand based on magnetic resonance imaging (MRI). [S.l.: S.n.].

UNIVERSAL ROBOTS. UR5 Technical Specifications. [S.l.: S.n.].

VERGARA, M.; AGOST, M. J.; GRACIA-IBÁÑEZ, V. Dorsal and palmar aspect dimensions of hand anthropometry for designing hand tools and protections. *Human Factors and Ergonomics In Manufacturing*, v. 28, n. 1, p. 17–28, 1 jan. 2018.

ZHANG, Zhong *et al.* CATCH-919 Hand: Design of a 9-actuator 19-DOF Anthropomorphic Robotic Hand. 12 set. 2018.