

MANIPULADOR ROBÓTICO BORG ARM

Relatório Final do Projeto The Borg Arm

Montagem e testes em ambiente simulado e real

Autores:

Anderson Queiroz do Vale

Aziel Martins de Freitas Júnior

Diogo Alexandre Martins

Israel Cerqueira Motta Neto

Mateus Santos de Cerqueira

Pedro Paulo Ventura Tecchio

Facilitadores:

Marco Antonio dos Reis

Rebeca Tourinho Lima

Tiago Pereira de Souza

**Salvador
Bahia, Brasil**

Novembro de 2020

Título: Manipulador Robótico Borg Arm	
PROD. TEC. BIR - 302 / 2020	Versão
Classificação: () Confidencial (X) Restrito () Uso Interno () Público	01

Informações Confidenciais - Informações estratégicas para o BIR e Senai Cimatec.

Seu manuseio é restrito a usuários previamente autorizados pelo Gestor da área.

Informações Restritas - Informação cujo conhecimento, manuseio e controle de acesso devem estar limitados a um grupo restrito de pesquisadores que necessitam utilizá-la para exercer suas atividades profissionais.

Informações de Uso Interno - São informações destinadas à utilização interna por pesquisadores e parceiros.

Informações Públicas - Informações que podem ser distribuídas ao público externo, o que, usualmente, é feito através dos canais apropriados.

Dados Internacionais de Catalogação na Publicação (CIP)

000
Anderson Queiroz do Vale
Aziel Martins de Freitas Júnior
Diogo Alexandre Martins
Israel Cerqueira Motta Neto
Mateus Santos de Cerqueira
Pedro Paulo Ventura Tecchio

Marco Antonio dos Reis
Rebeca Tourinho Lima
Tiago Pereira de Souza

Manipulador Robótico Borg Arm
Salvador
Bahia, Brasil
Novembro de 2020

Keywords:

1. Manipulator. 2. Simulation. 3. Computer vision.

000

SUMÁRIO EXECUTIVO

O projeto de Manipuladores - Desafio 2, também conhecido como **The Borg Arm Project**, se configura sob o Programa de Formação de Novos Talentos do Serviço Nacional de Aprendizagem Industrial do Departamento Regional da Bahia - Senai/DR/BA, sendo este o principal fomentador do programa.

O projeto teve início técnico em 27 de Fevereiro de 2020.

Devido aos problemas ocasionado pela pandemia houve uma parada no desenvolvimento da parte mecânica do projeto. Por esse motivo, o prazo de execução planejado foi estendido para 1 mês após o retorno das atividades.

RESUMO

Este trabalho apresenta resultados do projeto **The Borg Arm** que propõe o desenvolvimento de um manipulador robótico, desde a sua conceptualização até a entrega de protótipo operacional. Este projeto faz parte de conjunto de desafios propostos para o desenvolvimento de uma plataforma robótica móvel capaz de desarmar uma bomba remotamente de forma autônoma. Para tanto, foi realizado um estudo do estado da arte de robôs com arquitetura e aplicações semelhantes. Com base em tal estudo e nos materiais disponíveis em laboratório, foi desenvolvido um projeto mecânico, elétrico e de *software* capaz de realizar a atividade inicial de localizar um marco fiducial e apertar um botão em uma caixa aleatoriamente existente no espaço de atuação do manipulador. São apresentados neste relatório os resultados obtidos através dos testes de repetibilidade do movimento do manipulador e da análise de variância dos objetivos realizados. Estes resultados mostram que o manipulador desenvolvido é capaz de realizar a tarefa proposta com alta confiabilidade e baixa variância.

ABSTRACT

This work presents partial results of the project **The Borg Arm** which aims to develop a robotic manipulator from concept creation until delivery of an operational prototype. This project is part of the set of challenges proposed for the development of a mobile robotic platform capable of autonomously disarming a bomb. To this end, a study was carried out on the state of the art of similar existing robots. Based on this study and materials available in the laboratory, a mechanical, electrical and software project was developed which is capable of performing, in an automated way, the simplified tasks of locating a fiducial tag and pressing a button, both contained in an electrical panel placed at random inside the manipulator workspace. In this partial report, results obtained through simulation done with ROS framework and Gazebo simulator are presented. These results show that the developed manipulator will be able to perform the proposed task.

LISTA DE FIGURAS

Figura 1:	Ambiente de operação	23
Figura 2:	Componentes do manipulador	30
Figura 3:	Diagrama dos motores	30
Figura 4:	Arquitetura geral do sistema	31
Figura 5:	Simulação da área de trabalho revelada pelo uso da cinemática direta do modelo desenvolvido	33
Figura 6:	Projeção no plano XY	33
Figura 7:	Projeção no plano XZ	33
Figura 8:	Estrutura analítica do protótipo	34
Figura 9:	Fluxograma Escaneamento	34
Figura 10:	Fluxograma Identificação	35
Figura 11:	Árvores de transformadas antes da detecção do ArUco	36
Figura 12:	Conexão entre árvores de transformadas	37
Figura 13:	Fluxograma Acionamento	37
Figura 14:	Pose de repouso	40
Figura 15:	Pose início de trabalho	40
Figura 16:	Alvo à direita	40
Figura 17:	Alvo à esquerda	40
Figura 18:	Posição 1	41
Figura 19:	Posição 2	41
Figura 20:	Quantidade dos testes em relação à variação da posição e velocidade	43
Figura 21:	Configuração utilizada no teste de repetibilidade	43
Figura 22:	Marcador acoplado no manipulador	44
Figura 23:	Quantidade dos testes em relação à variação da posição e velocidade	44
Figura 24:	Histograma do tempo até o pressionamento do botão	46
Figura 25:	Histograma do tempo da missão	47
Figura 26:	Histograma do tempo de <i>scan</i>	47
Figura 27:	Histograma do RMS da posição X	48
Figura 28:	Histograma do RMS da posição Y	48
Figura 29:	Histograma do RMS da posição Z	49
Figura 30:	Histograma do RMS da orientação X	49
Figura 31:	Histograma do RMS da orientação Y	50
Figura 32:	Histograma do RMS da orientação Z	50
Figura 33:	Figura representando o teste ANOVA	52
Figura 34:	Condição Estática	71
Figura 35:	Carga Estática	72

Figura 36:	Forças cortantes.	73
Figura 37:	Momentos fletores.	73
Figura 38:	Momentos fletores com a carga do <i>payload</i> .	74
Figura 39:	Desenho técnico do suporte da camera.	76
Figura 40:	Desenho técnico perfil de alumínio.	77
Figura 41:	Desenho técnico perfil de alumínio.	78
Figura 42:	Desenho técnico conector do motor 2 com o perfil de alumínio 1.	79
Figura 43:	Desenho técnico conector do motor 3 com o perfil de alumínio 1.	80
Figura 44:	Desenho técnico conector do motor 3 com o perfil de alumínio 2.	81
Figura 45:	Desenho técnico conector do motor 4 com o perfil de alumínio 2.	82
Figura 46:	Esquema elétrico de potência para motores 1 a 3	84
Figura 47:	Esquema elétrico de potência para motores 4 a 5	85
Figura 48:	Desenho mecânico do motor ROBOTIS DYNAMIXEL PH42-020-S300-R	128
Figura 49:	Desenho mecânico do motor ROBOTIS DYNAMIXEL PH54-100-S500-R.	133
Figura 50:	Desenho mecânico do motor ROBOTIS DYNAMIXEL PH54-200-S500-R.	138
Figura 51:	Desenho mecânico da estrutura ROBOTIS FRP54-A110.	140
Figura 52:	Desenho mecânico da estrutura ROBOTIS FRP42-A110.	141
Figura 53:	Desenho mecânico da estrutura ROBOTIS FRP54-I110K.	142
Figura 54:	Desenho mecânico da estrutura ROBOTIS FRP42-I110K.	143
Figura 55:	Desenho mecânico da estrutura ROBOTIS FRP54-H121K.	144
Figura 56:	Desenho mecânico da estrutura ROBOTIS FRP54-H221K.	145
Figura 57:	Desenho mecânico da estrutura ROBOTIS FRP42-H221K.	146
Figura 58:	Desenho mecânico do conversor ROBOTIS U2D2 (RS485 - USB).	157

LISTA DE TABELAS

Tabela 1:	Especificações dos motores	32
Tabela 2:	Especificações do manipulador Borg Arm	32
Tabela 3:	Lista de componentes utilizados.	32
Tabela 4:	Parâmetros de configuração dos motores ROBOTIS DYNAMIXEL	39
Tabela 5:	Valores de configuração dos motores referentes à Velocidade 1.	42
Tabela 6:	Valores de configuração dos motores referentes à Velocidade 2.	42
Tabela 7:	Valores de configuração dos motores referentes à posição.	42
Tabela 8:	Resultados referentes à execução do objetivo.	45
Tabela 9:	Teste ANOVA estudo de contribuição para o manipulador.	52
Tabela 10:	T-teste.	53
Tabela 11:	F-teste.	53
Tabela 12:	Coeficiente de Prioridade de Risco	54
Tabela 13:	Estudo dos modos e efeitos de falha: Motor	54
Tabela 14:	Estudo dos modos e efeitos de falha: Câmera	55
Tabela 15:	Estudo dos modos e efeitos de falha: NUC	55
Tabela 16:	Estudo dos modos e efeitos de falha: Conector Mecânico	55
Tabela 17:	Estudo dos modos e efeitos de falha: Cabo	55
Tabela 18:	Lições aprendidas	58
Tabela 19:	Matriz Denavit-Hartenberg para o manipulador proposto.	87
Tabela 20:	Dados obtidos com a realização dos experimentos parte 1.	98
Tabela 21:	Dados obtidos com a realização dos experimentos parte 2.	99
Tabela 22:	Especificações do computador Intel NUC 6i5SYB.	109
Tabela 23:	Especificações da câmera Basler ace.	112
Tabela 24:	Especificações da lente Kowa LM8HC para sensor de 1 polegada de diâmetro.	121
Tabela 25:	Especificações do motor ROBOTIS DYNAMIXEL PH42-20-S300-R.	124
Tabela 26:	Especificações do motor ROBOTIS DYNAMIXEL PH54-100-S500-R.	129
Tabela 27:	Especificações do motor ROBOTIS DYNAMIXEL PH54-200-S500-R.	134
Tabela 28:	Especificações do conversor ROBOTIS U2D2 (RS485-USB).	147

LISTA DE SÍMBOLOS E ABREVIATURAS

ArUco *Augmented reality library from the University of Córdoba*

COTS *Commercial Off-The-Shelf*

DH Denavit-Hartenberg

DoF *Degrees of Freedom*

EE *End Effector*

EOD *Explosive Ordnance Disposal*

FMECA *Failure Mode Effects and Criticality Analysis*

IEDs *Improvised Explosive Devices*

NUC *Next Unit of Computing*

OMPL *Open Motion Planning Library*

OpenCV *Open Source Computer Vision*

ROS *Robot Operating System*

URDF *Unified Robot Description Format*

USBDC *United States Bomb Data Centre*

XML *Extensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO	15
1.1 Objetivos	15
1.2 Justificativa	16
1.3 Organização do relatório	17
2 CONCEITO DO SISTEMA	19
2.1 Parâmetros básicos	19
2.1.1 Requisitos do cliente	19
2.1.2 Requisitos técnicos	19
2.1.3 Premissas	20
2.1.4 Estudo do estado da arte	20
2.1.5 Ambiente de operação	23
2.1.6 Normas utilizadas	24
3 DESENVOLVIMENTO DO SISTEMA	29
3.1 Descrição do sistema	29
3.1.1 Arquitetura geral	31
3.1.2 Especificação técnica	31
3.1.3 Estrutura analítica do protótipo	33
3.2 Especificação funcional	33
3.2.1 Escaneamento	34
3.2.1.1 Descrição	34
3.2.1.2 Premissas necessárias	35
3.2.1.3 Dependências	35
3.2.1.4 Saídas	35
3.2.2 Identificação	35
3.2.2.1 Descrição	35
3.2.2.2 Premissas necessárias	36
3.2.2.3 Dependências	36
3.2.2.4 Saídas	36

3.2.3	Acionamento	37
3.2.3.1	Descrição	37
3.2.3.2	Premissas necessárias	37
3.2.3.3	Dependências	38
3.2.3.4	Saídas	38
3.3	Simulação do sistema	38
3.3.1	Estrutura de Pacotes	38
3.4	Integração	39
3.5	Testes realizados	40
4	CONFIABILIDADE DO SISTEMA	41
4.1	Experimentos	41
4.1.1	Teste de repetibilidade	43
4.2	Resultados dos testes de desempenho	45
4.2.1	Teste Shapiro-Wilk e Kolmogorov–Smirnov	45
4.2.2	Histogramas	46
4.2.3	ANOVA	51
4.2.4	T-teste	53
4.2.5	F-teste	53
4.3	Estudo dos modos e efeitos de falhas	53
5	GESTÃO DO CONHECIMENTO	57
5.1	Lições aprendidas	57
5.2	Guia de uso	58
5.2.1	Exigências	59
5.2.1.1	Pacotes	59
5.2.1.2	Pacote Externo	60
5.2.1.3	Programas	60
5.2.2	Como executar	60
6	CONCLUSÃO	63
REFERÊNCIAS		65

APÊNDICE A Projeto mecânico	71
A.1 Análise estática	71
A.2 Desenho técnico mecânico	75
A.2.1 Suporte da câmera	76
A.2.2 Perfil de alumínio 1	77
A.2.3 Perfil de alumínio 2	78
A.2.4 Conector do motor 2 com perfil de alumínio 1	79
A.2.5 Conector do motor 3 com perfil de alumínio 1	80
A.2.6 Conector dos motor 3 com perfil de alumínio 2	81
A.2.7 Conector do motor 4 com perfil de alumínio 2	82
APÊNDICE B Projeto elétrico	83
B.1 Esquema elétrico de potência	83
B.2 Potência demandada	86
APÊNDICE C Aplicação da teoria de manipuladores robóticos em experimentos simulados	87
C.1 Inicialização da classe	87
C.2 Obtenção das matrizes de transformação homogênea	88
C.3 Cálculo da cinemática direta	89
C.4 Estimação do espaço de trabalho do manipulador	90
C.5 Obtenção da matriz Jacobiana do manipulador	92
C.6 Um algoritmo para a Cinemática Inversa	93
C.7 Exemplo de uso da classe <i>Manipulator</i>	94
APÊNDICE D Dados coletados durante experimentos	97
APÊNDICE E Aplicação de testes estatísticos nos dados obtidos	101
ANEXO A Especificação de componentes adquiridos	109
A.1 Computador Intel NUC	109
A.2 Câmera Basler ace acA4600	112
A.3 Lente Kowa LM8HC	121

A.4 Motores ROBOTIS	123
A.4.1 DYNAMIXEL PH42-020-S300-R	124
A.4.2 DYNAMIXEL PH54-100-S500-R	129
A.4.3 DYNAMIXEL PH54-200-S500-R	134
A.5 Estruturas ROBOTIS	139
A.5.1 DYNAMIXEL Pro 54: <i>Adaptor Set</i> (FRP54-A110) . .	140
A.5.2 DYNAMIXEL Pro 42: <i>Adaptor Set</i> (FRP42-A110K) . .	141
A.5.3 DYNAMIXEL Pro 54: <i>Idler Bearing Set</i> (FRP54-I110K)	142
A.5.4 DYNAMIXEL Pro 42-20: <i>Idler Bearing Set</i> (FRP42-I110K)	143
A.5.5 DYNAMIXEL Pro 54-100: <i>Single Hinge Set</i> (FRP54-H121K)	144
A.5.6 DYNAMIXEL Pro 54-200: <i>Single Hinge Set</i> (FRP54-H221K)	145
A.5.7 DYNAMIXEL Pro 42-20: <i>Single Hinge Set</i> (FRP42-H221K)	146
A.6 Conversor ROBOTIS U2D2 (RS485 - USB)	147

1 INTRODUÇÃO

Este relatório apresenta os resultados relacionados à montagem mecânica de um manipulador robótico real, obtidos no projeto **The Borg Arm**, que constitui uma das etapas de desenvolvimento de um robô móvel autônomo capaz de desarmar uma bomba. Este projeto foi desenvolvido durante o Programa de Formação de Novos Talentos do Laboratório de Robótica e Sistemas Autônomos do SENAI Cimatec.

De modo geral, a construção de um manipulador robótico exige a união de conhecimentos extraídos de diversas áreas da ciência e da engenharia. Dos conceitos de engenharia mecânica faz-se uso do estudo das máquinas em situações estáticas e dinâmicas. A matemática fornece ferramentas para a descrição de movimentos espaciais e outros atributos dos manipuladores. A teoria de controle provê ferramentas para projetar e avaliar algoritmos que permitem a realização de movimentos ou a aplicação de força desejadas. As técnicas de engenharia elétrica são aplicadas no projeto de sensores e interfaces para robôs e a ciência da computação contribuiu com a base para a programação desses dispositivos a fim de que desempenhem a tarefa desejada ([CRAIG, 2005](#)).

Assim, faz-se necessário o uso de diversas técnicas, em especial alguns algoritmos de localização, mapeamento e visão computacional. Logo, a escolha de uma plataforma que permita a integração e o gerenciamento destes algoritmos é de fundamental importância. Dentre as plataformas existentes, o *framework ROS - Robotic Operating System* provê vantagens tais como extensa biblioteca de pacotes e *softwares* livres e de código aberto, grande comunidade de usuários e desenvolvedores e aceitação em ambientes acadêmicos e industriais. Além destas facilidades, a existência de pacotes e plataformas orientadas ao desenvolvimento de robôs móveis autônomos e de manipuladores robóticos influenciou fortemente a escolha desta plataforma de integração para este projeto.

Tendo-se em vista a complexidade de desenvolvimento de um manipulador robótico, em especial, um manipulador que funcionará acoplado à um robô móvel em etapas futuras, à caracterização dos objetivos deste projeto em si e sua justificativa, estão dispostas à seguir.

1.1 Objetivos

O projeto **The Borg Arm** tem como objetivo principal a construção de um manipulador robótico capaz de realizar tarefas simples de forma automática, desde que à ele esteja acoplada a ferramenta necessária para a execução da atividade, usualmente chamada em inglês de *EE - End Effector*. No escopo deste projeto, a tarefa escolhida foi caracterizada pela ação de pressionar uma botoeira existente em um painel elétrico aleatoriamente posicionado na área de trabalho do manipulador robótico. Ao acionar a botoeira, um indicador luminoso fornecerá um *feedback* visual. Um marco fiducial, terceiro e último

elemento deste painel elétrico, está disposto na superfície contendo a batoeira e o indicador luminoso.

O objetivo secundário deste projeto é caracterizado pelo intercâmbio de conhecimento entre equipes e seus tutores, para o desenvolvimento de habilidades de pesquisa e compartilhamento de idéias no âmbito da robótica vivenciando a criação, construção e implementação de um projeto partindo do seu princípio.

1.2 Justificativa

Através do uso de novas tecnologias, dispositivos e programação aplicada, robôs podem realizar uma gama cada vez maior de atividades com extrema precisão, exatidão e rapidez. Com a constante necessidade de diversos setores em melhorar seus processos e serviços, cada vez mais existem robôs realizando atividades que antes eram exercidas por humanos. Dentre as atividades realizadas, estão aquelas que são perigosas, demoradas ou repetitivas.

O descarte de objetos explosivos é uma tarefa altamente arriscada para todas as pessoas envolvidas e é realizada tanto na esfera civil quanto na militar. No contexto militar, por exemplo, existe o Descarte de Munição Explosiva (do inglês *EOD* - *Explosive Ordnance Disposal*), e pode ser classificada em três subcategorias: remoção de minas, manuseio de munição não explodida e manuseio de dispositivos explosivos improvisados (do inglês *IEDs* - *Improvised Explosive Devices*) (LUNDBERG; REINHOLD; CHRISTENSEN, 2007). Os *IEDs* são a ameaça mais comum encontrada, podendo ser acionado de várias maneiras, incluindo através de transmissores e interruptores eletrônicos, sensores de inclinação, sensores térmicos e detectores de movimento, e também podem conter uma ampla variedade de materiais explosivos ou incendiários, incluindo pólvora, gasolina, TNT ou C-4 (DEROOS; PRICE; REIDY, 2001).

O problema dos *IEDs* é recorrente em todo o mundo, e segundo o Relatório Anual de Incidentes de Explosivos, publicado anualmente pelo *USBDC* - *United States Bomb Data Centre*, que é o resultado de várias investigações forenses sobre ataques terroristas envolvendo explosivos em solo americano, no ano de 2018 foram registradas 666 recuperações e 131 casos de incidentes de explosão relacionados a *IEDs* (USBDC, 2016). Os dispositivos geralmente utilizados nesta atividade são os robôs teleoperados, que são dispositivos operados remotamente à distância por um operador humano (CUI et al., 2003).

A proposta do conjunto de trabalhos em desenvolvimento no Programa de Formação de Novos Talentos do SENAI - Cimatec é o desenvolvimento de um robô autônomo que realize a atividade de localização e desarme de um artefato explosivo. Para tanto, faz-se necessário o desenvolvimento de um manipulador robótico compatível com uma plataforma robótica móvel capaz de atuar sobre e executar operações capazes de desarmar tais artefatos. A construção de um manipulador robótico com estas características constitui o principal

objetivo deste projeto, tal como descrito anteriormente.

1.3 Organização do relatório

Este documento está organizado como descrito a seguir: O capítulo 2 trata dos conceitos gerais do sistema, tais como parâmetros básicos, requisitos do cliente e técnicos, estudo do estado da arte, ambiente de operação e normas utilizadas. No capítulo 3 estão dispostos temas relacionados ao desenvolvimento do sistema como sua descrição, especificação funcional, arquitetura de *software*, simulação e integração. Assuntos e resultados relacionados à confiabilidade do sistema projetado estão dispostos no capítulo 4. O capítulo 4 também apresenta os resultados externos provenientes deste trabalho como as lições aprendidas pela equipe de desenvolvimento e o guia de uso do manipulador desenvolvido. Por fim, a conclusão deste trabalho é discutida no capítulo 6.

2 CONCEITO DO SISTEMA

Um manipulador robótico consiste em um mecanismo composto de uma série de segmentos, articulados ou deslizantes entre si, que tem como objetivo manipular objetos, podendo ser controlado por um operador ou por algum sistema lógico ([DIS, 2012](#)).

O presente projeto visa a concepção e desenvolvimento de um manipulador robótico composto de 5 graus de liberdade, cujo objetivo é o pressionamento de uma botoeira, presente em um painel elétrico, de forma automatizada.

2.1 Parâmetros básicos

O sistema foi desenvolvido para atender aos requisitos fornecidos pelo cliente, baseando-se nos requisitos técnicos e premissas adotadas e de acordo com o estudo do estado da arte sobre manipuladores robóticos autônomos.

2.1.1 Requisitos do cliente

Na concepção do projeto, os requisitos do cliente são um conjunto de necessidades e expectativas provenientes do cliente, relacionadas ao funcionamento do manipulador robótico, que devem ser cumpridas. Os seguintes requisitos para o presente projeto foram determinados pelo cliente:

- Projetar um manipulador robótico;
- Pressionar uma botoeira instalada em uma caixa;
- A caixa colocado a cerca de 1 m da base do manipulador;
- Existir um (e apenas um) marco fiducial na caixa;
- O manipulador deve ser capaz de detectar o alvo e localizá-lo no espaço;

2.1.2 Requisitos técnicos

Os requisitos técnicos são as condições necessárias para satisfazer o objetivo. No projeto, são as características do sistema (funções, objetivos, propriedades e restrições) para satisfazer as características e especificações solicitadas pelo cliente, as seguintes características foram definidas para o sistema:

- O manipulador deve possuir 5 graus de liberdade;
- O manipulador deve ter um alcance de 1 m em relação à sua base;
- O manipulador deve ter pelo menos uma carga útil de 0,5 kg;
- Ser capaz de identificar uma marco fiducial por uma única câmera RGB;
- A câmera deve ter no mínimo resolução full HD;

- Utilizar o ROS Melodic;
- Realizar simulação no Gazebo;
- Utilizar o MoveIt;
- Integrar ao sistema um mini computador para realizar o processamento;
- Estar fixo em uma base, posicionada em $x = -0.6$ m, $y = 0.0$, tomando-se como referencial o centro da bancada com eixo x paralelo ao maior lado e em direção à lateral contendo as tomadas elétricas;
- A base deve ser projetada para futura integração a um robô móvel;
- A junta situada aproximadamente ao meio do alcance máximo do manipulador deve ser projetada de modo que o atuador esteja em ângulo de 45° , o que implica em um arranjo de componentes de força que reduz o esforço sobre ele.

2.1.3 Premissas

As premissas do projeto são suposições, de eventos ou circunstâncias externas, tomadas como verdadeiras para fins de planejamento e execução do projeto. As premissas deste projeto são:

- Deve-se priorizar os materiais existentes no laboratório;
- Integrar ao sistema uma Intel NUC i5 6a geração com 16GB de RAM DDR-4, sob o Ubuntu 18.04 LTS e conexão sem fio;
- Usar os motores ROBOTIS DYNAMIXEL disponíveis no laboratório;
- Operar com fonte de alimentação capaz de fornecer 24 V e pelo menos 500 W.

2.1.4 Estudo do estado da arte

Foi realizado um estudo bibliográfico com o intuito de verificar projetos desenvolvidos utilizando manipuladores, e realizar a aplicação dos métodos abordados na proposta deste projeto. Existem alguns trabalhos que descrevem o uso de manipuladores que fazem uso de controle servo-visual, em tarefas de manuseio de objetos e utilizando marcos fiduciais. Foram identificados trabalhos que apresentam a produção de dispositivos robóticos móveis, equipados com um manipulador, que tem como objetivo final a coleta ou descarte de material perigoso. O estudo sobre projetos de manipuladores com 5 graus de liberdade em conjunto com o estudo da cinemática direta e inversa também foi englobado no estudo bibliográfico realizado.

Alguns autores abordam o processo de estudo da cinemática de manipuladores compostos de 5 graus de liberdade, desenvolvendo o estabelecimento do modelo cinemático ([LU; XU; WANG, 2014](#)) ([HUANG; TSENG, 2016](#)) ([MURZAKAEV; POLIAKOV; ARTEMEV, 2018](#)), analisando o espaço de trabalho ([CHEN et al., 2014](#)) ([LI et al., 2019](#)), obtendo a

solução da cinemática inversa (ZHANG; LI; XIAO, 2015) (ILIUHKHIN et al., 2017) (CHEN et al., 2017) e planejando a trajetória (KIM; CHEN; LEE, 2014).

O desenvolvimento, estudo e modelagem para manipuladores compostos de 5 graus de liberdade é tratado por (HAO; RU, 2013) e (ISSA et al., 2019). O trabalho descrito por (HVILSHØJ; BØGH, 2011) demonstra o desenvolvimento de um manipulador industrial dotado de componentes *Commercial Off-The-Shelf* (COTS) de hardware e software, acoplado à uma plataforma móvel autônoma, que utiliza visão computacional para recolher dados do ambiente. (PAUL et al., 2011) propôs a utilização de um manipulador Denso (QUANSER, 2017), de 6 graus de liberdade, para a exploração e mapeamento de um ambiente de manutenção de pontes. O trabalho descrito por (SHARP et al., 2017) aborda a integração de dois manipuladores de 6 graus de liberdade Universal Robotics UR5 (ROBOTICS, 2008), e a integração desse sistema semiautônomo ao ROS, Moveit! e RViz.

Manipuladores robóticos podem ser utilizados em conjunto com um sistema servo-visual, no qual é possível estimar a pose e a orientação da ferramenta devido à identificação de um referencial através da captura de imagens. O controle servo visual proposto por (DONG; ZHU, 2018) permite que um algoritmo de visão computacional detecte um alvo não-cooperativo (um alvo desconhecido comparado a um banco de dados de alvos em potencial para determinar a correspondência mais próxima) e identifique a pose do manipulador. Um manipulador de 6 graus de liberdade foi projetado e construído por (LAROCHE; ZHU, 2014) para utilizar controle servo-visual para manipulação autônoma de um alvo não-cooperativo.

O trabalho proposto por (TIYU et al., 2018) apresenta a utilização de um robô antropomórfico Hawk Robot (ROBOT, 2009), que possui dois manipuladores de 6 graus de liberdade, utilizando um sistema de visão computacional para pressionar um botão de elevador autonomamente. Outro controlador visual, baseado em um manipulador CYTON Alpha (ROBAI, 2008) utilizando 3 graus de liberdade em uma base móvel é utilizado por (ANDALUZ et al., 2015). (CHENG; CHEN; LIU, 2013) utiliza um manipulador móvel industrial, composto de um manipulador ABB IRB 120 (ABB, 2014), com 6 graus de liberdade montado em uma plataforma móvel, utilizando técnicas de controle servo-visual para o manuseio de objetos.

(DONG; ZHU, 2016) propõe a utilização de um controle autônomo servo-visual em um manipulador de 3 graus de liberdade para a manipulação de um objeto não-cooperativo. O trabalho proposto por (CARIUS et al., 2018) propõe a confecção de um manipulador de 6 graus de liberdade e o desenvolvimento de uma arquitetura de planejamento e controle para a realização de missões autônomas, acoplado em uma plataforma móvel, utilizando controle servo-visual para manipular objetos de acordo com a especificação. A utilização de um manipulador assistivo iARM (DYNAMICS, 2012) de 6 graus de liberdade, para controle servo-visual com o objetivo de transportar e manusear um copo é detalhada por

(CHUNG; WANG; COOPER, 2013).

(PETREA et al., 2018) utiliza um manipulador de 6 graus de liberdade acoplado em uma plataforma móvel, e é utilizado no manuseio de peças no controle de qualidade através do controle servo-visual. O trabalho proposto por (SHI et al., 2012) visa uma aplicação na robótica espacial, um sistema de captura autônoma para satélite, que utiliza controle servo-visual para manipular um alvo cooperativo. Uma aplicação para manipulação autônoma de válvulas de controle usando um robô Baxter (SPECTRUM, 2018a), composto de dois manipuladores e um sistema de detecção visual, é proposto por (FARIA et al., 2015).

Marcos fiduciais também são utilizados por alguns autores, como implementado por (FENG et al., 2015), que desenvolveram aplicações implementadas em um manipulador autônomo KUKA KR 100 (KUKA, 2005) composto de 7 graus de liberdade, em tarefas de construção civil, utilizando uma câmera para a estimativa de pose. Um manipulador composto de 6 juntas e uma ferramenta foi utilizado por (ZHANG et al., 2019) para aplicação em operações em ambientes aquáticos, utilizando ArUco para estimativa da pose e calculando a atitude com base na forma e na cor do objeto alvo. Uma aplicação envolvendo a obtenção da pose de um manipulador industrial KUKA KR 210 (KUKA, 2002) de 6 graus de liberdade, utilizando uma, e em seguida, múltiplas AprilTags em um ambiente de produção, é descrita por (NISSLER et al., 2016). O trabalho proposto por (NARVÁEZ et al., 2017) visa a utilização de um manipulador Kinova Mico (KINOVA, 2018) com 6 graus de liberdade, fixo em uma plataforma móvel, que realiza a busca e resgate de um quadrotor através de técnicas de controle servo-visual por detecção de uma AprilTag fixada no drone.

(BURRELL et al., 2018) propõe um sistema de corte de tubos para descomissionamento nuclear através de detecção de tags ArUco, utilizando dois manipuladores hidráulicos. Um manipulador WAM (SPECTRUM, 2018b) de 7 graus de liberdade é utilizado por (HAMNER et al., 2010) fixado em uma plataforma móvel para executar tarefas de montagem no domínio da fabricação automotiva utilizando a detecção de ARTags para auxiliar na localização. (DU et al., 2017) trata do manejo de objetos por manipuladores robóticos em ambientes dinâmicos, visando a detecção e evasão dos manipuladores em relação aos objetos. No experimento é utilizado um manipulador Katana (FRÜH, 2003) com 5 graus de liberdade e códigos de barras, nos objetos e obstáculos.

Os trabalhos citados contribuíram para o aprofundamento do estudo da cinemática e a sua aplicação em manipuladores, visando a compreensão das técnicas utilizadas e da forma de obtenção dos resultados. Houve contribuições para o conhecimento de métodos utilizados na concepção e desenvolvimento de manipuladores de 5 graus de liberdade, escopo deste projeto. Também foi aproveitado o estudo de técnicas de visão computacional, com o objetivo de compreender o funcionamento e comportamento de manipuladores que utilizam detecção de um referencial visual (objetos e marcos fiduciais) visando o manuseio

de objetos.

Embora a aplicação (pressãoamento de uma botoeira) seja semelhante ao trabalho de (TIYU et al., 2018), o projeto descrito neste relatório diferencia-se deste trabalho principalmente com relação à estrutura do manipulador (apenas um manipulador fixo) e à utilização de marcos fiduciais para o referenciamento espacial do objeto alvo. O projeto diferencia-se, em relação aos demais trabalhos citados, pela aplicação final e também pela estrutura, uso da concepção e fabricação de peças, graus de liberdade, ou framework e técnicas utilizadas no desenvolvimento da aplicação.

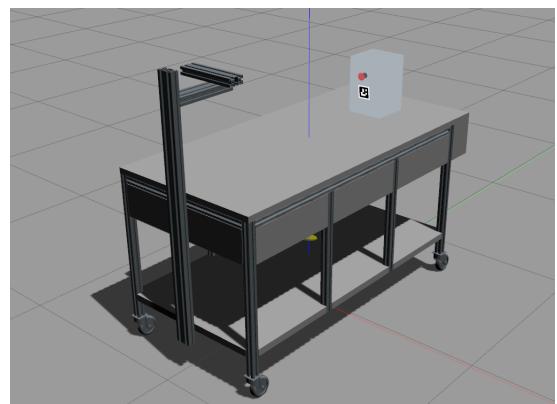
2.1.5 Ambiente de operação

O ambiente de operação do manipulador robótico é uma bancada com rodas travantes, com dimensões de aproximadamente 1,7 x 0,8 x 0,9 m, construída em perfil de alumínio e compensado, com tampo emborrachado. À sua lateral, foi fixado um suporte especialmente fabricado para o manipulador, de modo que haja o espaço esperado para posicionar o painel elétrico contendo a botoeira de acionamento da lâmpada, conforme detalhes demonstrados na Figura 1.

Figura 1: Ambiente de operação



(a) Real



(b) Simulado

Fonte: Autores.

O ambiente de operação conta com tomadas elétricas monofásicas com tensão eficaz de 220 V com terra, do tipo tomada de embutir com tampa, com referências normativas NBR 14136 (ABNT, 2013) e NBR NM 60884-1 (ABNT, 2010). A bancada é ligada à essas tomadas através de um plugue fixo de embutir, e provém ligações do tipo fixa de 10 A, com referência normativa NBR 14136/02, através de filtro de linha genérico.

A luminosidade do ambiente é um importante parâmetro a levar em consideração na tarefa de detecção visual. A cor final de um objeto depende do material e da luz que incide nele. Enquanto a cor percebida na luz representa a intensidade da luz emitida, a

cor percebida em um material representa a porcentagem de luz que um objeto reflete ([GAZEBO, 2020](#)). O mesmo material pode ter cores diferentes, dependendo da intensidade da luz que o atinge.

A iluminação do ambiente real é realizada com lâmpadas brancas com temperatura de cor típica superior a 6000 K com distribuição tipicamente uniforme.

Na simulação, são utilizados os componentes difuso e especular. O componente difuso é a cor de um objeto sob uma luz branca pura e é calculado usando a direção da luz e a superfície normal onde a luz atinge. Já o componente especular é o realce especular que ocorre quando a luz atinge a superfície de um objeto e reflete em direção à câmera ([GAZEBO, 2020](#)). A iluminação utilizada na simulação no Gazebo está situada na posição $z = 10$, tem valor difuso de RGBA(204, 204, 204, 255) e especular de RGBA(51, 51, 51, 255), com atenuação com valor constante de 0,9 e fator linear de atenuação de 0,1.

2.1.6 Normas utilizadas

A operação do manipulador robótico possui algumas características definidas, que permeiam a verificação de aspectos capazes de influenciar na tomada de medidas de segurança, na concepção e na operação do manipulador:

- capacidade de movimentação com muita energia em um grande espaço de operação;
- o início do movimento e a rota do manipulador são difíceis de prever e podem variar;
- o espaço de trabalho do manipulador pode conflitar com a posição de outros equipamentos;
- operadores podem precisar trabalhar próximos do manipulador durante a sua movimentação.

Para este projeto, alguns cuidados devem ser tomados de modo a garantir a segurança do operador e evitar acidentes durante a montagem e operação do manipulador. Alguns perigos em potencial são abordados, pois este equipamento possui ligação à rede de energia e também possui um dispositivo mecânico funcionando em conjunto com dispositivos eletrônicos, logo, algumas recomendações de segurança são fornecidas.

A Norma Regulamentadora NR 12 ([REGULAMENTADORA, 1978](#)) define referências técnicas, princípios fundamentais e medidas de proteção para garantir a integridade física dos trabalhadores e estabelece requisitos mínimos para a prevenção de acidentes nas fases de projeto e de utilização de máquinas e equipamentos.

A norma aborda aspectos referentes ao arranjo físico e aos sistemas de segurança recomendados que foram utilizados no projeto do manipulador, relacionados à arranjos físicos e instalações; instalações e dispositivos elétricos; e dispositivos de partida, acionamento e parada.

Com relação aos dispositivos de partida, acionamento e parada, os seguintes aspectos devem ser considerados:

- Nos locais de instalação de máquinas e equipamentos, as áreas de circulação devem ser devidamente demarcadas;
- As áreas de circulação devem ser mantidas permanentemente desobstruídas;
- Os materiais em utilização no processo produtivo devem ser alocados em áreas específicas de armazenamento;
- Os espaços ao redor das máquinas e equipamentos devem ser adequados de forma a prevenir a ocorrência de acidentes relacionados ao trabalho;
- A distância mínima entre máquinas, em conformidade com suas características e aplicações, deve garantir a segurança dos trabalhadores durante sua operação, manutenção, ajuste, limpeza e inspeção, e permitir a movimentação dos segmentos corporais, em face da natureza da tarefa;
- Os pisos dos locais de trabalho onde se instalaram máquinas e equipamentos e das áreas de circulação devem:
 - ser mantidos limpos e livres de objetos, ferramentas e quaisquer materiais que ofereçam riscos de acidentes;
 - ter características de modo a prevenir riscos provenientes de graxas, óleos e outras substâncias e materiais que os tornem escorregadios;
 - ser nivelados e resistentes às cargas a que estão sujeitos.
- As ferramentas utilizadas no processo produtivo devem ser organizadas e armazenadas ou dispostas em locais específicos para essa finalidade.

Os aspectos relativos às instalações e dispositivos elétricos são descritos a seguir:

- As instalações elétricas das máquinas e equipamentos devem ser projetadas e mantidas de modo a prevenir os perigos de choque elétrico, incêndio, explosão e outros tipos de acidentes;
- Devem ser aterradas as instalações, carcaças, invólucros, blindagens ou partes condutoras das máquinas e equipamentos que não façam parte dos circuitos elétricos, mas que possam ficar sob tensão;
- Os condutores de alimentação elétrica das máquinas e equipamentos devem atender aos seguintes requisitos mínimos de segurança:
 - oferecer resistência mecânica compatível com a sua utilização;
 - possuir proteção contra a possibilidade de rompimento mecânico, de contatos abrasivos e de contato com lubrificantes, combustíveis e calor;

- localização de forma que nenhum segmento fique em contato com as partes móveis ou cantos vivos;
- facilitar e não impedir o trânsito de pessoas e materiais ou a operação das máquinas;
- não oferecer quaisquer outros tipos de riscos na sua localização;
- ser constituídos de materiais que não propaguem o fogo, ou seja, autoextinguíveis, e não emitirem substâncias tóxicas em caso de aquecimento.
- As instalações elétricas das máquinas e equipamentos que utilizem energia elétrica fornecida por fonte externa devem possuir dispositivo protetor contra sobrecorrente, dimensionado conforme a demanda de consumo do circuito.

Os aspectos relativos aos dispositivos de partida, acionamento e parada são descritos a seguir:

- Os dispositivos de partida, acionamento e parada das máquinas devem ser projetados, selecionados e instalados de modo que:
 - não se localizem em suas zonas perigosas;
 - possam ser acionados ou desligados em caso de emergência por outra pessoa que não seja o operador;
 - impeçam acionamento ou desligamento involuntário pelo operador ou por qualquer outra forma accidental;
 - não acarretem riscos adicionais;
 - não possam ser burlados.
- Os comandos de partida ou acionamento das máquinas devem possuir dispositivos que impeçam seu funcionamento automático ao serem energizadas.

A norma ABNT NBR ISO 10218 ([ABNT, 2018](#)) foi criada devido aos riscos particulares presentes nos sistemas robotizados. Fornece orientações sobre as considerações de uso, projeto de segurança, construção, programação, operação, manutenção e reparo de robôs industriais e sistemas robóticos. Para reduzir os riscos, a norma utiliza algumas diretrizes:

- estabelecer a operação do sistema robótico de fora da área protegida;
- estabelecer um espaço protegido e restrito;
- definir dispositivos de segurança que consigam intervir no caso de uma ocorrência imprevista dentro do espaço protegido.

Segundo a norma, os seguintes fatores devem ser levados em conta durante a elaboração do projeto:

- estabelecer os limites físicos em três dimensões;
- definir espaço de trabalho;
- identificar o alcance máximo, estabelecendo a área restrita e de operação;
- identificar os espaços livres necessários para a operação, manutenção e limpeza;
- traçar rotas para passagem de pessoas e materiais identificando cabos e outros riscos.

3 DESENVOLVIMENTO DO SISTEMA

Nesta seção é abordada a caracterização geral do sistema, especificando a descrição do manipulador robótico. A listagem das funcionalidades é retratada através da especificação funcional, incluindo as premissas necessárias, dependências e saídas. Por fim, é abordada a etapa de simulação do sistema, incluindo as plataformas e pacotes utilizados, bem como os métodos utilizados e os requerimentos necessários para a execução das funcionalidades.

3.1 Descrição do sistema

No âmbito do manipulador robótico seus sistemas se dividem em elementos como: sistema mecânico, **EE** (ou “efetuador”, a ferramenta concedida ao robô para a realização do seu trabalho), sistema de acionamento e controle dos atuadores, plataforma de integração e um sistema de visão computacional.

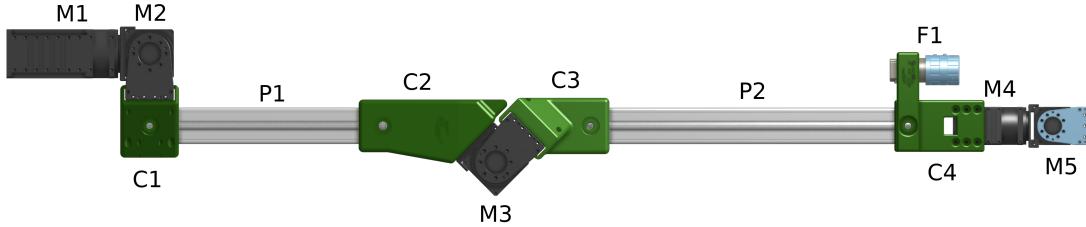
O posicionamento e orientação das juntas (do inglês *joints*) num manipulador robótico determinam os seus movimentos possíveis. A quantidade de juntas condiciona os *Degrees of Freedom* (*DoF*), ou graus de liberdade de um manipulador. O sistema mecânico depende principalmente das relações entre suas articulações, ou juntas, e elos (do inglês *links*), que são as partes fixas adjuntas às juntas. A definição da movimentação e das particularidades do manipulador robótico, sem considerar as forças e torques atuantes, é realizado através do estudo de sua cinemática.

A cinemática de robôs é dividida entre cinemática direta e cinemática inversa. Na cinemática direta, o objetivo é obter a posição e velocidade da ferramenta, **EE**, dada as posições das juntas. Na cinemática inversa, deseja-se obter as posições e velocidades das juntas a partir da posição da ferramenta. A combinação de ambos os métodos e sua integração com algoritmos de planejamento de trajetória são necessárias para o correto funcionamento do manipulador.

O manipulador robótico, apresentado nas Figuras 2 e 3, possui 5 graus de liberdade, 5 elos, 5 juntas de revolução, 5 motores e 1 **EE**, que é uma superfície planar do elo fixado ao eixo do motor último motor e movimentada por ele. Tal ferramenta é suficiente para a aplicação pretendida que consiste no acionamento da botoeira. Os nomes dos componentes na Figura 2 são encontrados de forma mais detalhada na Tabela 3.

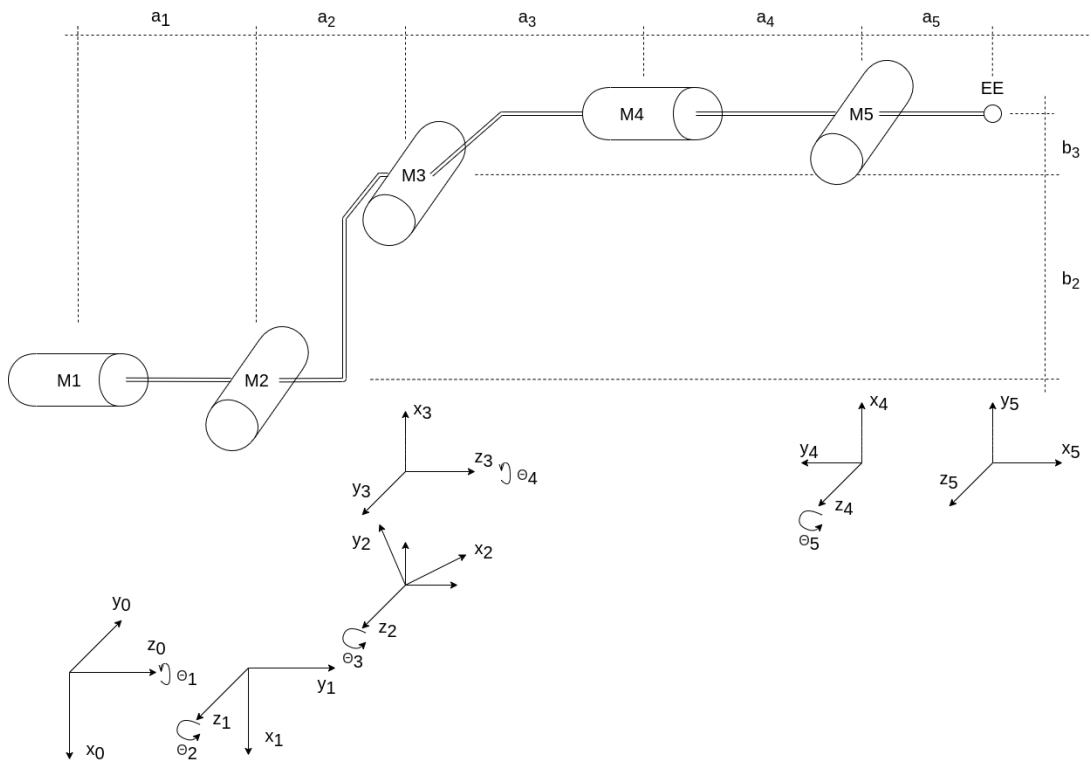
O manipulador proposto é composto por um conjunto de atuadores da linha DYNAMIXEL da ROBOTIS que, além de possuirem um sistema de controle integrado, são programáveis, de alto desempenho e apresentam recursos que facilitam a comunicação e integração em projetos robóticos. O conjunto de atuadores é apresentado na Figura 2 e representado nas juntas de revolução da Figura 3. A estrutura dos elos é composta principalmente por perfis estruturais de alumínio, acoplados ao motores através de conec-

Figura 2: Componentes do manipulador.



Fonte: Autores.

Figura 3: Diagrama dos motores.



Fonte: Autores.

tores desenvolvidos e impressos em plástico ABS. A câmera Basler acA4600 é o sensor que adquire dados para o sistema de detecção e visão computacional e o suporte para seu acoplamento também foi impresso em plástico ABS. A lista completa dos componentes utilizados está disponível na Seção 3.1.2 e suas especificações são apresentadas no Anexo A.

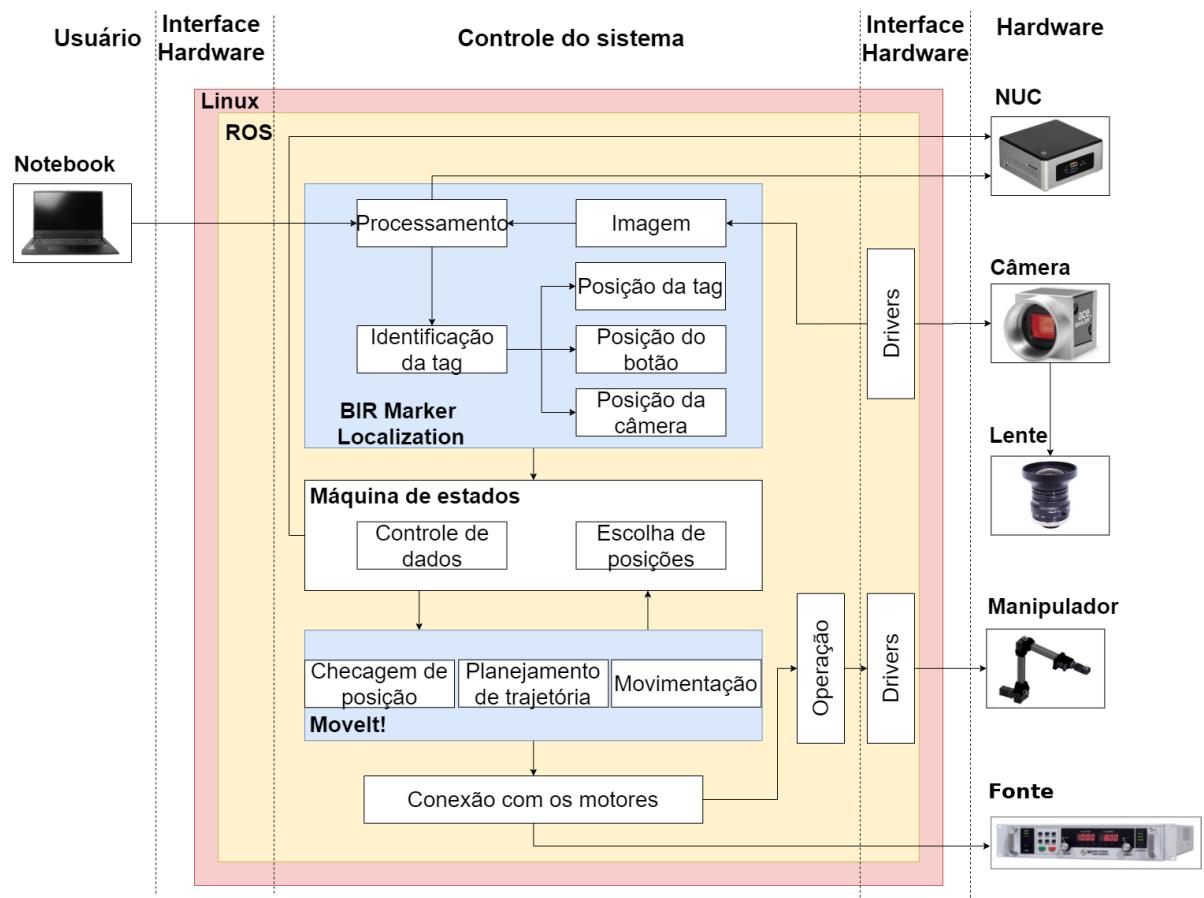
Para o desenvolvimento deste projeto, utilizou-se o *framework* de código aberto, ROS, que permite a criação de softwares e aplicações em robótica, e pode ser empregado em conjunto com o ambiente de simulação computacional Gazebo. Além destes, destacam-se o MoveIt - *Motion Planning Framework*, utilizado para o planejamento de trajetórias do manipulador; o *Open Source Computer Vision* (OpenCV), uma biblioteca de programas amplamente empregada em visão computacional e o pacote *bir_marker_localization*,

utilizado na localização espacial através da detecção e estimativa da pose de marcos fiduciais, disponibilizado pelo Laboratório de Robótica e Sistemas Autônomos do SENAI - CIMATEC.

3.1.1 Arquitetura geral

A arquitetura geral do sistema é a representação do sistema na forma mais ampla, esquematizando as funcionalidades dos componentes de hardware e software, e a interação entre eles. A arquitetura geral foi implementada e descrita após estudos realizados e a verificação da abrangência dos componentes e funções do manipulador robótico, como apresentado na Figura 4.

Figura 4: Arquitetura geral do sistema.



Fonte: Autores.

3.1.2 Especificação técnica

As principais características e componentes utilizados no manipulador robótico desenvolvido são apresentados respectivamente nas Tabelas 1, 2 e 3. As especificações completas dos componentes utilizados estão dispostas no Anexo A.

Tabela 1: Especificações dos motores

Especificação do Motor	Referência	Torque (N.m)	Tensão (V)	Corrente (A)	Velocidade de Rotação (rpm)
Dynamixel Pro 200W	PH54-200-S500-R	44.7	24.0	9.3	29.0
Dynamixel Pro 100W	PH54-100-S500-R	25.3	24.0	5.5	29.2
Dynamixel Pro 20W	PH42-020-S300-R	5.1	24.0	1.5	29.2

Fonte: Autores.

Tabela 2: Especificações do manipulador Borg Arm

Item	Borg Arm
DOF	5
Carga útil	1,5 kg
Extensão	1,2 m
Massa	6,5 kg
Tensão de operação	24 Vdc
Comunicação	RS 485 57600 baud com conversor USB
Especificação das juntas	Junta 1: $[-180, 180]$ °
	Junta 2: $[-90, 90]$ °
	Junta 3: $[-90, 90]$ °
	Junta 4: $[-180, 180]$ °
	Junta 5: $[-90, 90]$ °

Fonte: Autores.

Tabela 3: Lista de componentes utilizados.

Nome	Descrição	Fabricante	Referência	Qty
Câmera Basler ace (F1)	Câmera RGB	Basler	ace aca4600-10uc	1
Lente Kowa (F1)	Lente	Kowa	Kowa LM8HC	1
Dynamixel Pro 20W (M4 e M5)	Servo motor M4 e M5	Robotis	PH42-020-S300-R	2
Dynamixel Pro 100W (M3)	Servo motor M3	Robotis	PH54-100-S500-R	1
Dynamixel Pro 200W (M1 e M2)	Servo motor M1 e M2	Robotis	PH54-200-S500-R	2
Dynamixel Pro Cable Set	Cabos	Robotis	Robot Cable-X4P 180mm 10pcs	1
Conversor USB	Conversor USB - RS485	Robotis	U2D2	1
Conector M2 - Perfil 1 (C1)	Conector impresso em ABS	Autores		1
Conector M3 - Perfil 1 (C2)	Conector impresso em ABS	Autores		1
Conector M3 - Perfil 2 (C3)	Conector impresso em ABS	Autores		1
Conector M4 - Perfil 2 (C4)	Conector impresso em ABS	Autores		1
Dynamixel Pro 54: Adaptor Set	Conjunto mecânico	Robotis	FRP54-A110K Set	2
Dynamixel Pro 42: Adaptor Set	Conjunto mecânico	Robotis	FRP42-A110K Set	1
Dynamixel Pro 54: Idler Bearing Set	Conjunto mecânico	Robotis	FRP54-I110K Set	2
Dynamixel Pro 42-20: Idler Bearing Set	Conjunto mecânico	Robotis	FRP42-I110K Set	1
Dynamixel Pro 54-200: Single Hinge Set	Conjunto mecânico	Robotis	FRP54-H221K Set	1
Dynamixel Pro 54-100: Single Hinge Set	Conjunto mecânico	Robotis	FRP54-H121K Set	1
Dynamixel Pro 42-20: Single Hinge Set	Conjunto mecânico	Robotis	FRP42-H121K Set	1
Perfil 1 (P1)	Perfil de alumínio			1
Perfil 2 (P2)	Perfil de alumínio			1

Fonte: Autores.

Em posse das especificações dos limites das juntas, em conjunto com as informações a respeito da dimensão dos elos e motores, é possível calcular a região de trabalho (do inglês *workspace*) do manipulador. A região de trabalho de um robô é o volume total varrido pelo EE, à medida que o manipulador executa todos os movimentos possíveis. O volume e a

Figura 5: Simulação da área de trabalho revelada pelo uso da cinemática direta do modelo desenvolvido.

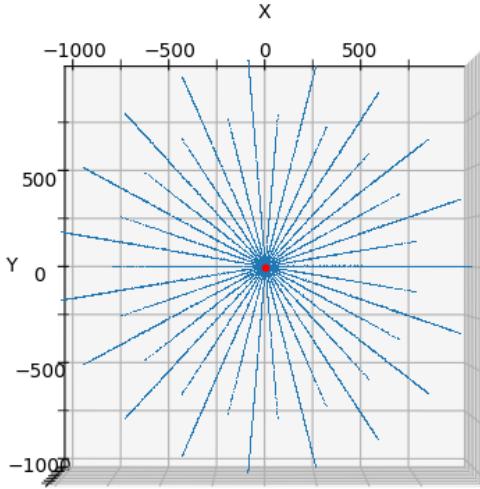


Figura 6: Projeção no plano XY.

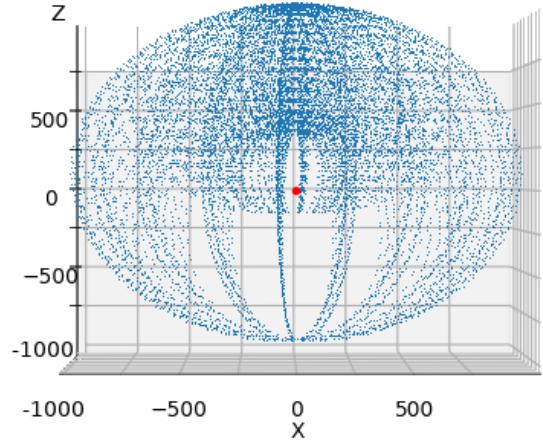


Figura 7: Projeção no plano XZ.

Fonte: Autores.

forma da região de trabalho determinam as aplicações e cenários nos quais o manipulador poderá ser utilizado. As Figuras 6 e 7 apresentam projeções planares da região de trabalho do manipulador desenvolvido. Tais projeções foram obtidas a através da simulação da cinemática direta do manipulador através de um *script* Python explicado no Apêndice C.

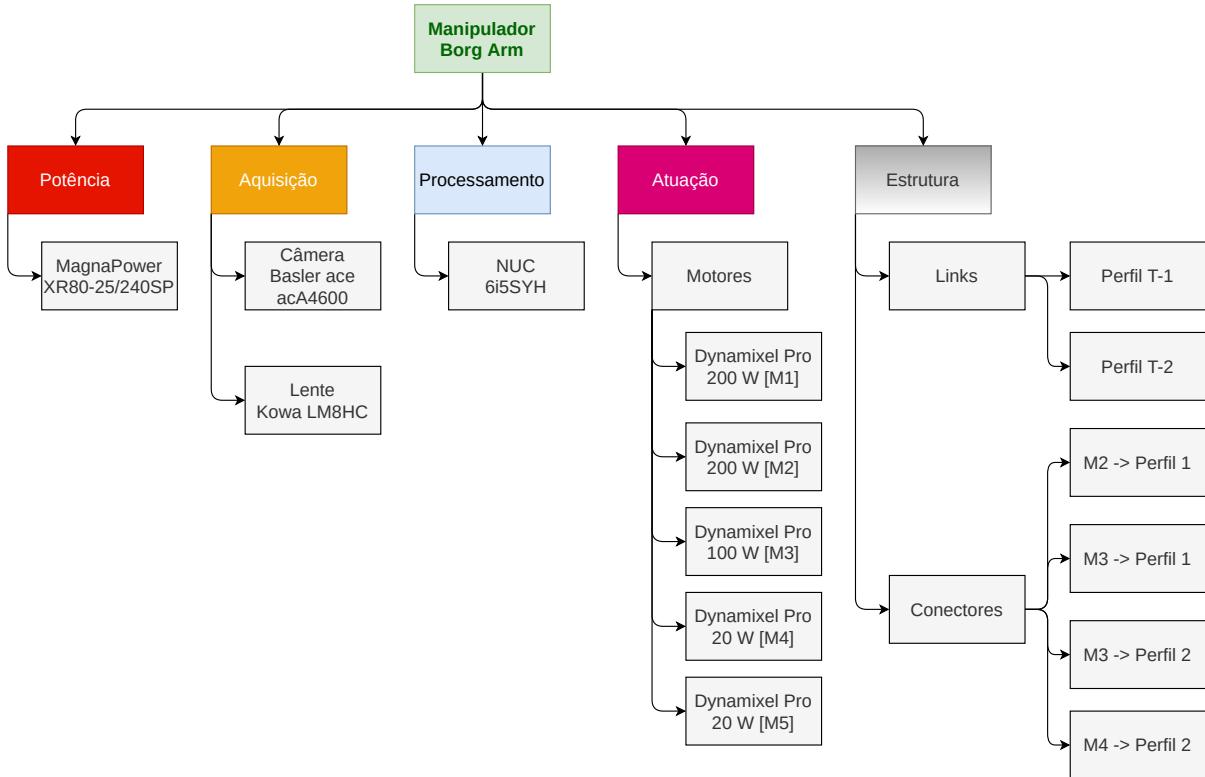
3.1.3 Estrutura analítica do protótipo

A estrutura analítica do protótipo é uma ferramenta usada para ajudar a descrever o escopo do projeto, e é descrita na Figura 8. Esta estrutura demonstra a estratificação do projeto em divisões e subdivisões, disposta de forma hierárquica, com a finalidade de simplificar e direcionar as atividades feitas para a conclusão do protótipo.

3.2 Especificação funcional

Nessa seção serão apresentadas as principais funcionalidades que permitem ao manipulador robótico executar a tarefa de acionamento da botoeira. Estas funcionalidades são Escaneamento, Identificação e Acionamento. Escaneamento é caracterizado pela ação de varredura do espaço no seu entorno para buscar possíveis ângulos que permitam identificar o alvo. Identificação é caracterizada pela rotina específica que visa processar imagens obtidas pela câmera e ao identificar o alvo procurado, publicar a pose do mesmo num tópico e as atualizações necessárias na *tf-tree*. Por fim, o Acionamento é caracterizado por mover o EE até posições definidas para pressionar a botoeira a partir da localização do alvo e da atualização da *tf-tree*.

Figura 8: Estrutura analítica do protótipo



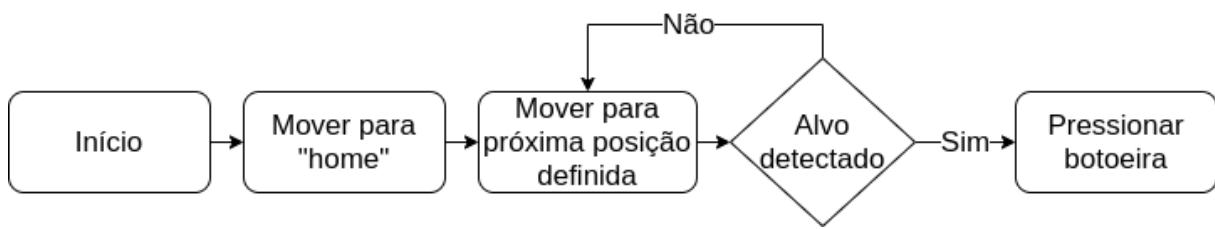
Fonte: Autores.

3.2.1 Escaneamento

O manipulador robótico é capaz de realizar o escaneamento dentro da região de trabalho em busca de um objeto de interesse. No caso da tarefa de acionamento da botoeira, a máquina de estados apresentada na Figura 4 é responsável pela rotina de busca do marco fiducial. O sistema executa um conjunto de movimentos pré-estabelecidos e retorna para a posição inicial caso o marco fiducial não tenha sido identificado.

3.2.1.1 Descrição

Figura 9: Fluxograma Escaneamento.



Fonte: Autores.

3.2.1.2 Premissas necessárias

- Câmera calibrada.
- Ativação da função de busca no manipulador.
- Definição de uma posição inicial.
- Definição do volume de trabalho do manipulador.
- Definição das regiões de colisão na bancada de trabalho.

3.2.1.3 Dependências

- Pacote *bir_marker_localization* instalado.
- Pacote *ros_control* instalado.
- MoveIt instalado.

3.2.1.4 Saídas

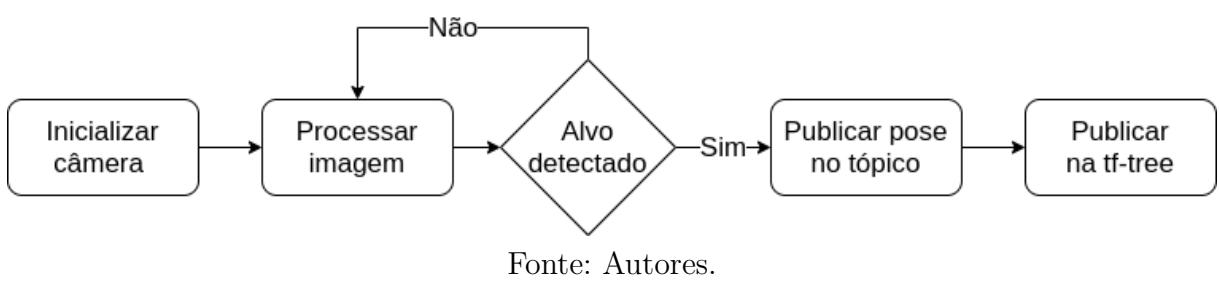
O escaneamento permite, ao longo de sua execução, a identificação do marco fiducial e consequente avanço nas etapas da máquina de estados.

3.2.2 Identificação

O processo de identificação do marco fiducial é realizado em paralelo com a etapa de escaneamento. As imagens provenientes da câmera são processadas com o auxílio do [OpenCV](#) e do pacote *bir_marker_localization* afim de identificar o marco fiducial do tipo *Augmented reality library from the University of Córdoba* ([ArUco](#)). O [ArUco](#) é um tipo específico de marco fiducial, com dimensões físicas e formatos previamente conhecidos e padronizados, que armazena dados em sua própria imagem. As informações são interpretadas pelo sistema de visão computacional afim de estimar a posição e orientação da câmera em relação ao [ArUco](#).

3.2.2.1 Descrição

Figura 10: Fluxograma Identificação.



Fonte: Autores.

3.2.2.2 Premissas necessárias

- Câmera calibrada.
- Definição dos parâmetros do marco fiducial, como ID, dicionário utilizado, tamanho e espaçamento de borda.
- Execução da rotina de identificação de marcos fiduciais.

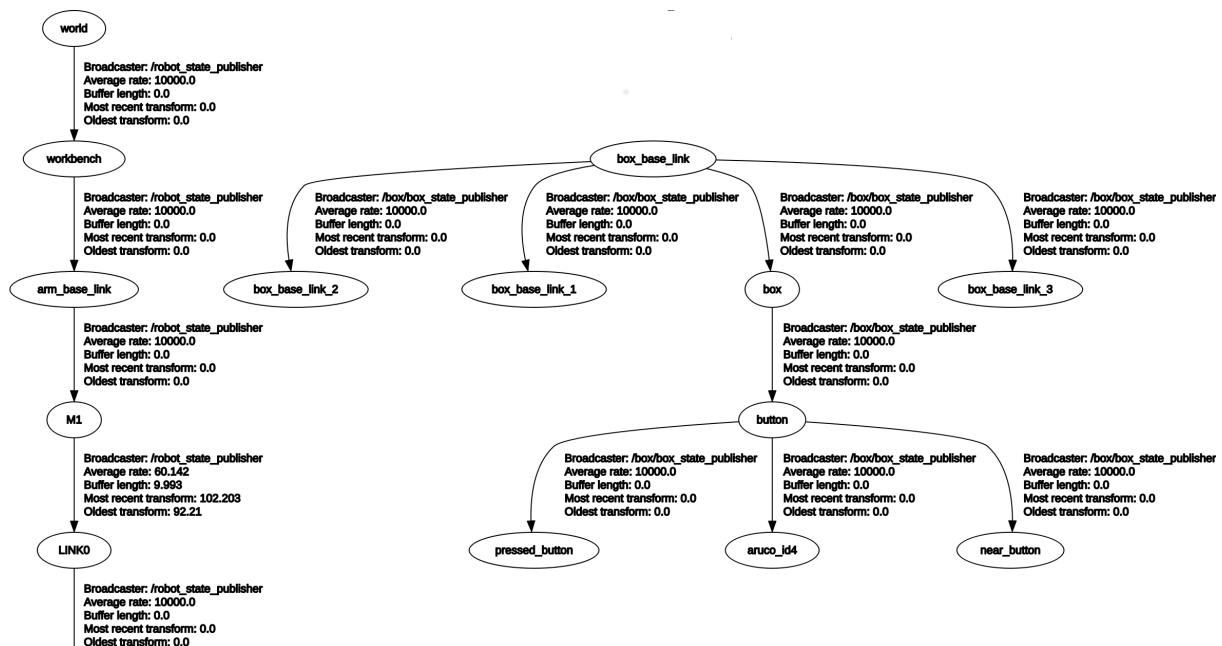
3.2.2.3 Dependências

- Biblioteca [OpenCV](#) 3.3.1 instalada e funcional.
- Pacote *bir_marker_localization* instalado.
- Pacote *pylon-ros-camera* instalado.
- Pacote *dragandbot_common* instalado.

3.2.2.4 Saídas

A partir da estimativa da posição e orientação do [ArUco](#), é obtida a transformação de coordenadas que relaciona as posições e orientações entre a câmera e o marco fiducial. Com essa relação é possível realizar a conexão entre as árvores de transformadas, apresentada na Figura 11. A primeira, que armazena as relações entre os componentes do manipulador, iniciando-se na transformada “world”, da bancada e do mundo. E a segunda, iniciando-se na transformada “box_base_link”, que armazena as relações entre o [ArUco](#), caixa e botoeira.

Figura 11: Árvores de transformadas antes da detecção do ArUco.

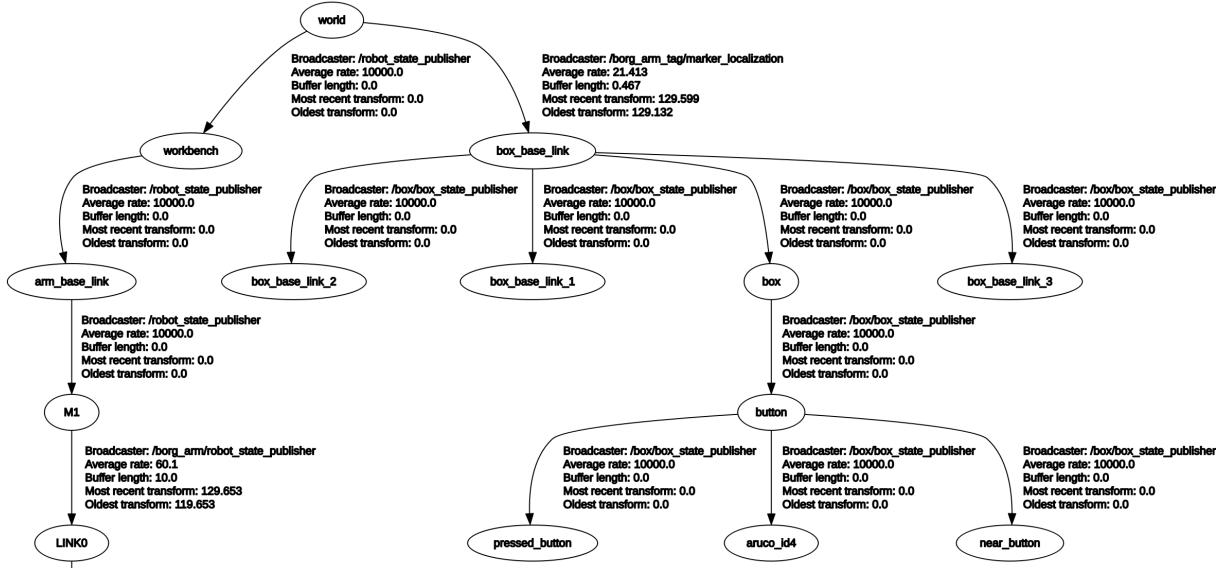


Fonte: Autores.

3.2.3 Acionamento

A etapa de acionamento é realizada após a identificação do ArUco. A conexão das árvores de transformadas, apresentada na Figura 12, permite a estimativa da posição e orientação que o EE deve atingir para o correto acionamento da botoeira. O acionamento é coordenado e executado através da máquina de estados e do MoveIt.

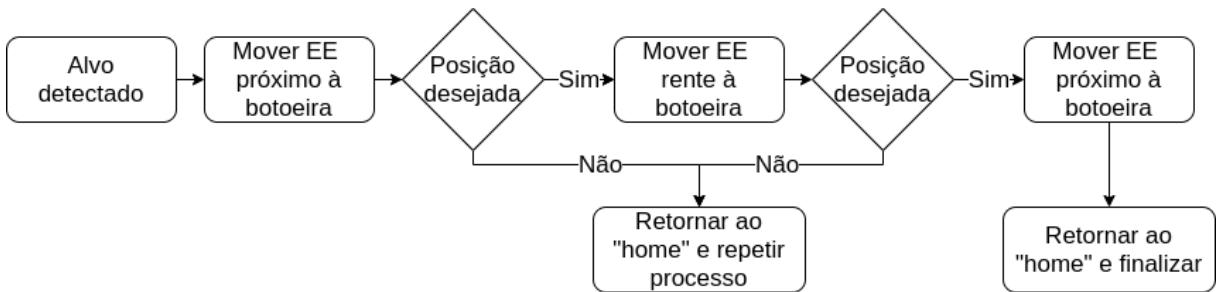
Figura 12: Conexão entre árvores de transformadas.



Fonte: Autores.

3.2.3.1 Descrição

Figura 13: Fluxograma Acionamento.



Fonte: Autores.

3.2.3.2 Premissas necessárias

- Estimação da pose do ArUco.
- Conhecimento prévio da relação entre a pose do ArUco e botoeira.
- Conexão entre árvores de transformadas.

- Definição das regiões de colisão na bancada de trabalho.
- Botoeira contida na região de trabalho do manipulador.

3.2.3.3 Dependências

- Pacote *bir_marker_localization* instalado.
- MoveIt instalado.

3.2.3.4 Saídas

Em posse da posição e orientação da botoeira, as mesmas são enviadas ao MoveIt e, caso a botoeira esteja dentro da região de trabalho, a trajetória é encontrada e executada. Após o execução da tarefa, o manipulador retorna a posição inicial.

3.3 Simulação do sistema

A etapa de simulação é essencial no processo de desenvolvimento de robôs. É possível realizar um estudo de sua estrutura, características, funções e limitações. Além disso, permite e acelera a validação de conceitos e algoritmos em diversos cenários sem por em risco o patrimônio ou a vida de pessoas próximas.

O processo de desenvolvimento e simulação do manipulador foi realizado na plataforma *ROS*, que é um *framework* de código aberto que permite a criação de softwares e aplicações em robótica, cujas funcionalidades podem ser implementadas de forma modular. A implementação do planejamento de trajetória, controle, manipulação, cinemática direta e inversa de manipuladores robóticos foi realizada com auxílio do MoveIt, que possui uma coleção de APIs que, em conjunto com o *ROS*, encapsulam as ferramentas necessárias. O algoritmo RRT-Connect, disponível no MoveIt através da biblioteca *Open Motion Planning Library* (*OMPL*), foi utilizado na criação de uma trajetória dada uma pose objetivo. De acordo com a trajetória encontrada, as posições das juntas e do *end-effector* são definidas com base na solução da cinemática inversa realizada através da biblioteca *TRAC-IK*. A diferença entre as posições atuais e desejadas são utilizadas como entrada nos controladores gerados pelo pacote *ros_control*, responsável pela geração dos sinais de controle e envio aos atuadores.

3.3.1 Estrutura de Pacotes

Para realização da simulação foram desenvolvidos alguns pacotes listados a seguir:

- *borg_arm_description*: O manipulador desenvolvido é descrito através de um arquivos *Extensible Markup Language (XML)* do tipo *Unified Robot Description Format (URDF)*. Neste pacote as relações entre os corpos rígidos e juntas é descrita de acordo com a localização dos componentes, tipos de juntas e propriedades físicas.

Tabela 4: Parâmetros de configuração dos motores ROBOTIS DYNAMIXEL

Motores	Fase de configuração		Fase de testes	
	Profile_Velocity	Profile_Acceleration	Profile_Velocity	Profile_Acceleration
M1 a M3	100	300	200	300
M4 e M5	500	500	500	500

Fonte: Autores.

- *borg_arm_gazebo*: Pacote responsável pela simulação do manipulador no Gazebo. Contém também a descrição dos componentes utilizados na simulação, como a bancada de trabalho e a caixa com boteira.
- *borg_arm_moveit_config*: Pacote responsável pela configuração do manipulador para utilização no *framework* do MoveIt.
- *borg_arm_tag*: Pacote responsável pela configuração e interação do manipulador com o API do *bir_marker_localization*.
- *borg_arm_controller*: Pacote responsável pela coordenação, integração e execução das atividades do manipulador, bem como integração com os controladores dos motores ROBOTIS DYNAMIXEL.

3.4 Integração

A integração entre a simulação e estrutura física ocorre principalmente por meio do pacote *dynamixel-workbench*. Com ele é possível enviar para os atuadores reais as mesmas coordenadas angulares recebidas pelos atuadores simulados. Dessa forma, uma trajetória planejada pelo MoveIt é executada também no mundo real, e desde que sejam estabelecidos limites na simulação, por meio de caixas e planos de colisão, por exemplo, uma movimentação livre de colisões será executada de modo análogo na realidade.

Cada atuador DYNAMIXEL teve seu *Operating_Mode* configurado com o valor “3” e recebeu parâmetros de configuração de velocidade e aceleração específicos. Esses valores visaram inicialmente a segurança das pessoas durante a utilização do manipulador na fase inicial de configuração mas também visaram um tempo viável para a execução de testes posteriores. Além disso, a limitação em velocidade e aceleração do manipulador contribui para que não ocorram esforços mecânicos excessivos sobre a estrutura, e limitam a potência demandada da fonte. Para configurar essas características, deve-se alterar os parâmetros *Profile_Velocity* e *Profile_Acceleration*.

Quando um atuador DYNAMIXEL tem o *Operating_Mode* = 3, os parâmetros de perfil de velocidade e aceleração mencionados anteriormente definem, respectivamente, o valor máximo de velocidade e o valor de aceleração desejados. Configurou-se os valores de velocidade e aceleração segundo a Tabela 4.

3.5 Testes realizados

Juntamente com a configuração dos perfis de velocidade e aceleração, cada um dos atuadores deve receber um número de identificação a fim de poder receber os sinais de dados provenientes da *Next Unit of Computing* (NUC). Há mais parâmetros na *tabela de controle* de cada atuador, porém os mencionados constituem alguns dos mínimos suficientes. Após a configuração dos atuadores individualmente, prosseguiu-se com a montagem do manipulador, cuja estrutura pode ser vista nas Figuras 14 e 15, onde está disposto nas posições de repouso e início de trabalho respectivamente.

Figura 14: Pose de repouso.



Fonte: Autores.

Figura 15: Pose início de trabalho.



Fonte: Autores.

O pressionamento de botão foi testado em posições distintas ao longo da bancada. A atuação bem sucedida foi registrada nas Figuras 16 e 17. Testes de repetibilidade foram realizados e as conclusões obtidas serão apresentadas mais detalhadamente em outro relatório.

Figura 16: Alvo à direita.



Fonte: Autores.

Figura 17: Alvo à esquerda.



Fonte: Autores.

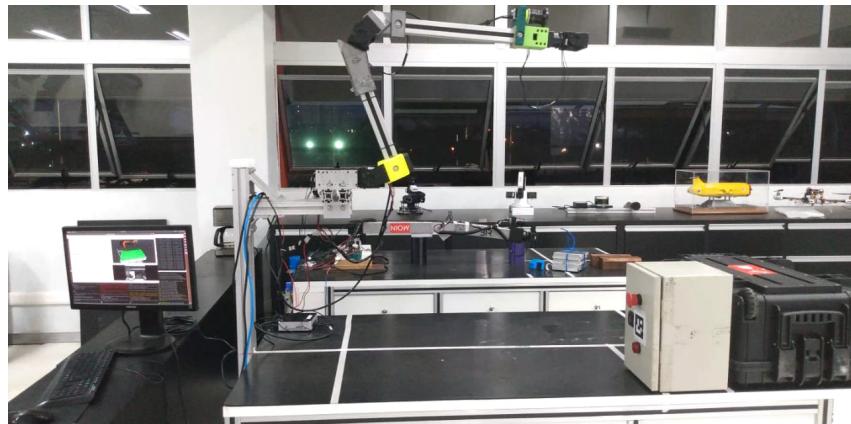
4 CONFIABILIDADE DO SISTEMA

Foram realizados testes para elaborar uma avaliação sobre a confiabilidade e repetibilidade do manipulador, utilizando experimentos para gerar uma base de dados de testes e em seguida empregar métodos estatísticos para mensurar e retirar informações úteis sobre os dados.

4.1 Experimentos

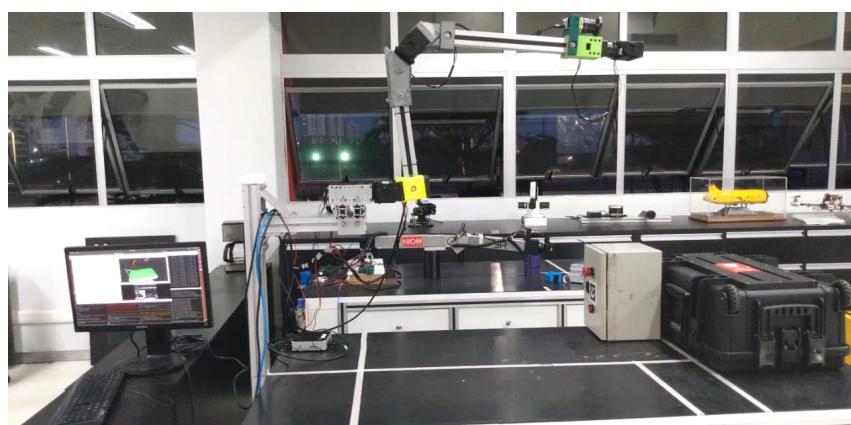
Os testes com o manipulador foram realizados posicionando o painel elétrico em duas posições distintas, Posição 1 (Figura 18) e Posição 2 (Figura 19), a fim de avaliar a confiabilidade do manipulador desenvolvido durante a execução da tarefa proposta em ambiente real.

Figura 18: Posição 1.



Fonte: Autores

Figura 19: Posição 2.



Fonte: Autores

Durante os testes, foram registrados o tempo de execução, tempo decorrido até a localização da *tag* e do pressionamento do botão, posição e orientação do *end effector* (valor esperado e real), de modo a verificar a repetibilidade do manipulador, a eficácia e os possíveis erros decorridos.

Nos testes com o manipulador, foram variadas a velocidade e posição, de modo a verificar os comportamentos do manipulador para cada situação. Foram realizados diversos testes para garantir a coleta suficiente de dados e posterior análise com cada uma destas variáveis. A Velocidade 1 (Tabela 5) e a Velocidade 2 (Tabela 6) são utilizadas conforme as configurações dos motores mostradas a seguir.

Tabela 5: Valores de configuração dos motores referentes à Velocidade 1.

Motor	Return_Delay_Time	Operating_Mode	Goal_Velocity	Profile_Acceleration	Profile_Velocity
M5	0	3	500	500	500
M4	0	3	500	500	500
M3	0	3	100	300	100
M2	0	3	100	300	100
M1	0	3	100	300	100

Fonte: Autores

Tabela 6: Valores de configuração dos motores referentes à Velocidade 2.

Motor	Return_Delay_Time	Operating_Mode	Goal_Velocity	Profile_Acceleration	Profile_Velocity
M5	0	3	500	500	500
M4	0	3	500	500	500
M3	0	3	200	300	200
M2	0	3	200	300	200
M1	0	3	200	300	200

Fonte: Autores

As posições do painel elétrico na bancada são mostrados na Tabela 7.

Tabela 7: Valores de configuração dos motores referentes à posição.

Posição	Distância da base (eixo x) [m]	Distância da base (eixo y) [m]
1	0,3	1
2	-0,3	1

Fonte: Autores

O experimento engloba a realização de 60 testes, onde são variadas as posições do painel elétrico e as velocidades de atuação do manipulador. Dentre os 60 testes, 30 são realizados com o painel elétrico na Posição 1 e 30 com o painel elétrico na Posição 2. Onde em cada posição distinta são executados 15 testes na Velocidade 1, e 15 testes utilizando a Velocidade 2. As configurações da quantidade de testes em relação à variação de velocidade e posição são mostradas na Figura 20

Figura 20: Quantidade dos testes em relação à variação da posição e velocidade.

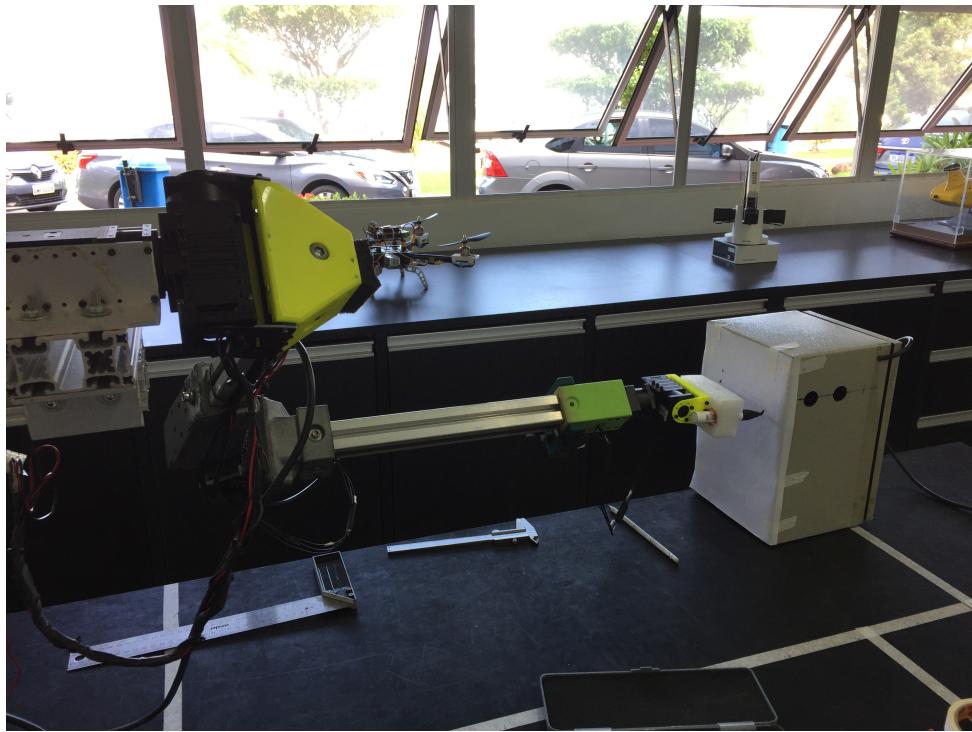
	Posição 1	Posição 2
Velocidade 1	15 testes	15 testes
Velocidade 2	15 testes	15 testes

Fonte: Autores

4.1.1 Teste de repetibilidade

Foi realizado um teste a fim de avaliar a repetibilidade do manipulador no planejamento e execução de um mesmo objetivo. Nesse teste o manipulador recebeu apenas a posição e orientação fixas e não utilizou as informações fornecidas pela câmera. Um marcador foi fixado na extremidade do *end effector* e uma folha A4 foi colocada na superfície da caixa como apresentado nas Figuras 21 e 22.

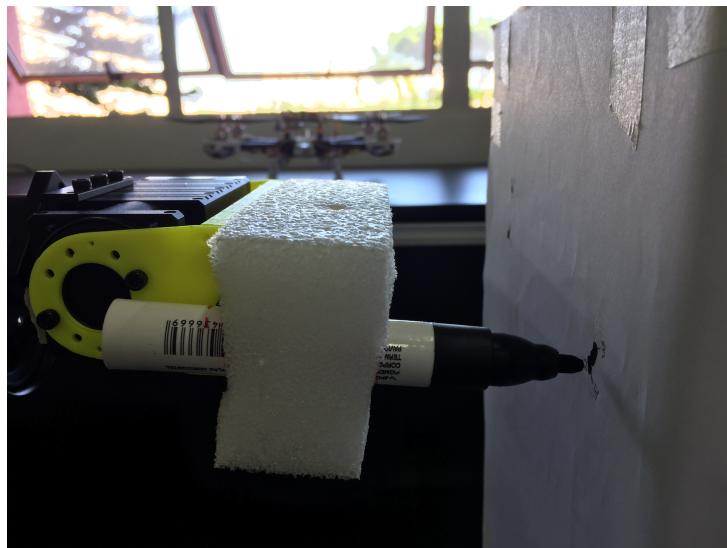
Figura 21: Configuração utilizada no teste de repetibilidade.



Fonte: Autores

A pose desejada foi obtida através da movimentação manual das juntas do manipulador até a posições e orientações de interesse. A partir dessas informações a caixa foi posicionada de forma que o marcador tocasse a superfície da folha A4. O primeiro ponto foi definido como referência para os testes subsequentes.

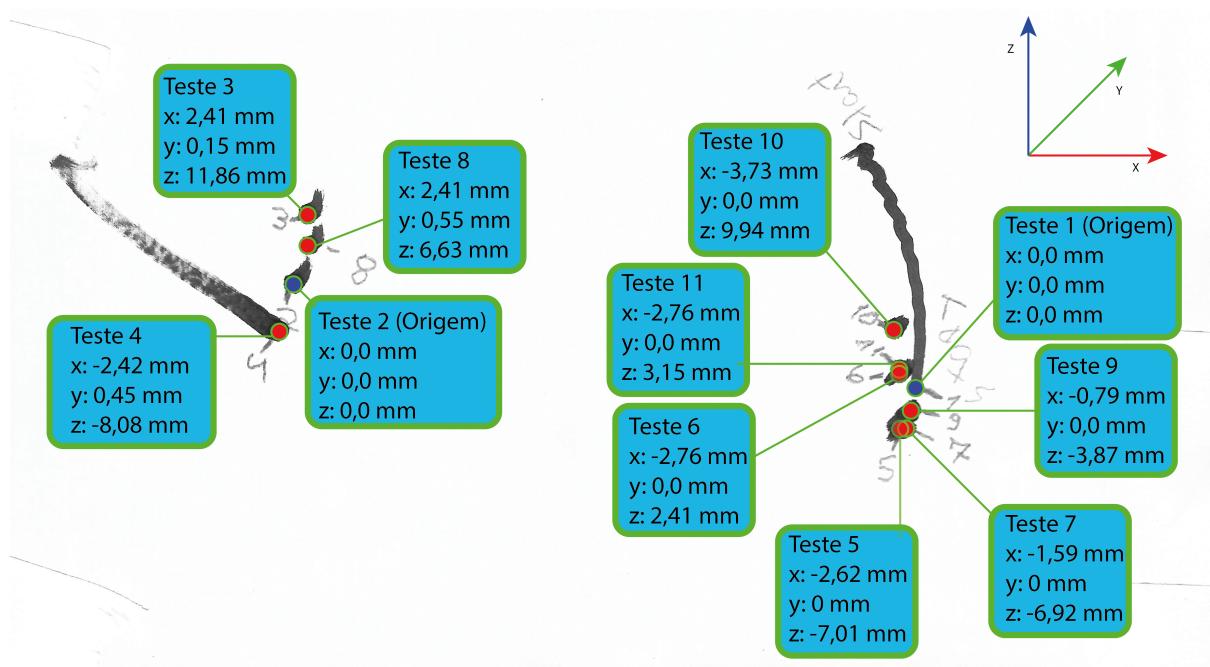
Figura 22: Marcador acoplado no manipulador.



Fonte: Autores

O planejamento e execução da trajetória foi realizado 10 vezes e em cada uma das execuções um ponto foi marcado no folha A4 e a profundidade, obtida através do deslocamento do marcador na espuma, foi registrada. O pontos registrados são apresentados na Figura 23.

Figura 23: Quantidade dos testes em relação à variação da posição e velocidade.



Fonte: Autores

Ao longo dos testes o algoritmo de planejamento de trajetória encontrou algumas

soluções nas quais o *end effector* apresentou variação, em relação a orientação desejada, de 180° em torno do eixo Y. Essa variação ocorreu nos testes 2,3,4 e 8 dando origem a pontos na extremidade oposta da folha. O primeiro ponto marcado com essa variação foi definido como ponto de referência dessa região. A posição de cada um dos pontos, em relação ao ponto de origem de cada região, e a profundida alcançada foram registrados na Figura 23.

4.2 Resultados dos testes de desempenho

A avaliação do desempenho foi feita utilizando análise estatística, a fim de estudar a frequência comportamental sobre 60 amostragens realizadas, utilizando 2 posições e 2 velocidades distintas, como demonstrado na tabela 8.

Foram adicionadas as variáveis referentes aos erros RMS da orientação e posição do *end effector* na tarefa de pressionar a botoeira. O RMS ou erro quadrático médio da raiz, é uma medida utilizada para estimar as diferenças entre os valores previstos por um modelo e os valores observados. Estes erros foram calculados para estimar a diferença entre os valores obtidos e os valores desejados de orientação e posição.

Tabela 8: Resultados referentes à execução do objetivo.

	Velocity	Time_scan	Time_button	rms_p_x	rms_p_y	rms_p_z	rms_x	rms_y	rms_z
max	2	111.427	131.087	0.3409685	0.249408	1.09048	2.220445	0.1230813	3.117041
min	1	29.3741	38.2763	0.3246036	0.2490637	0.970826	0.005041739	0.004337038	2.097779
mean	1.5	63.5085	80.10702	0.3332	0.249319	0.998241	0.9709479	0.05601647	2.670326
sd	0.5042195	30.62565	30.84238	0.007379236	8.596217e-05	0.02348372	1.09273	0.02981596	0.4461282
shapiro	6.36482e-11	8.287944e-07	8.56173e-05	4.387477e-10	1.565957e-07	4.011388e-10	6.680283e-11	0.02275034	1.375838e-09
ks	1.99929e-06	0.01084468	0.118196	4.676078e-06	0.008630634	0.0004384842	8.889616e-08	0.1962118	1.559156e-05

Fonte: Autores

4.2.1 Teste Shapiro-Wilk e Kolmogorov-Smirnov

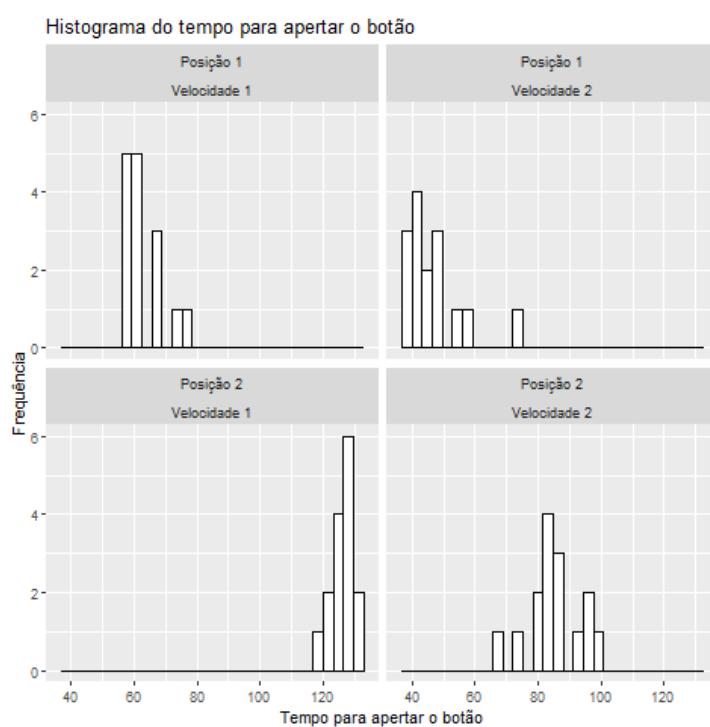
A hipótese nula do teste de Shapiro-Wilk (RAZALI; WAH et al., 2011) é que a população possui distribuição normal. Portanto, um valor de $p < 0.05$ indica que a hipótese nula foi rejeitada, ou seja, os dados não possuem distribuição normal. Todos os valores das variáveis seguem uma distribuição normal (Tabela 8).

Já o teste Kolmogorov-Smirnov é um teste não paramétrico utilizado para comparar uma amostra com uma distribuição de probabilidade de referência (RAZALI; WAH et al., 2011). A Tabela 8 também mostra que as variáveis com $p < 0.05$ não possuem distribuição normal. Segundo este teste, a variável que representa o tempo até pressionar o botão não segue uma distribuição normal, bem como a variável que representa o erro RMS do *end effector* na orientação y.

4.2.2 Histogramas

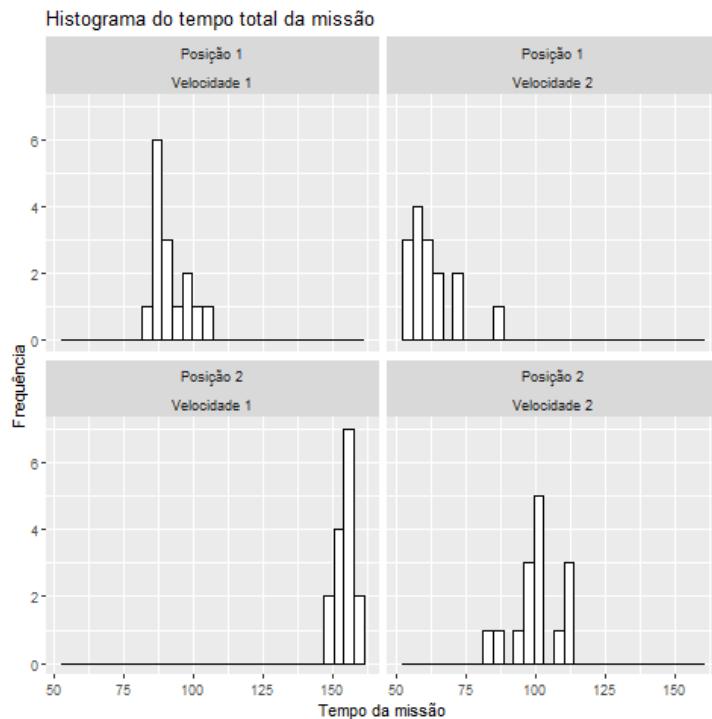
Os histogramas, os quais representam de modo gráfico os dados coletados de cada uma das posições, para analisar como as amostras de dados estão distribuídas, são apresentados nas Figuras 24a 32. Foram analisados os seguintes dados: tempo de *scan*, tempo até o pressionamento do botão, tempo total da missão, e erros RMS da orientação e posição, bem como as análises de desvio padrão, teste de normalidade *Shapiro-Wilk* e *Kolmogorov-Smirnov* (*p-value*).

Figura 24: Histograma do tempo até o pressionamento do botão.



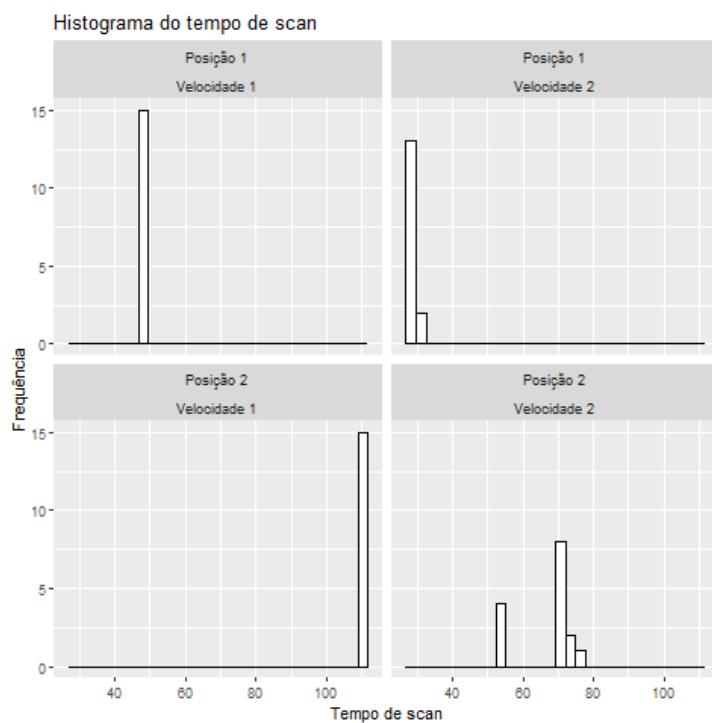
Fonte: Autores

Figura 25: Histograma do tempo da missão.



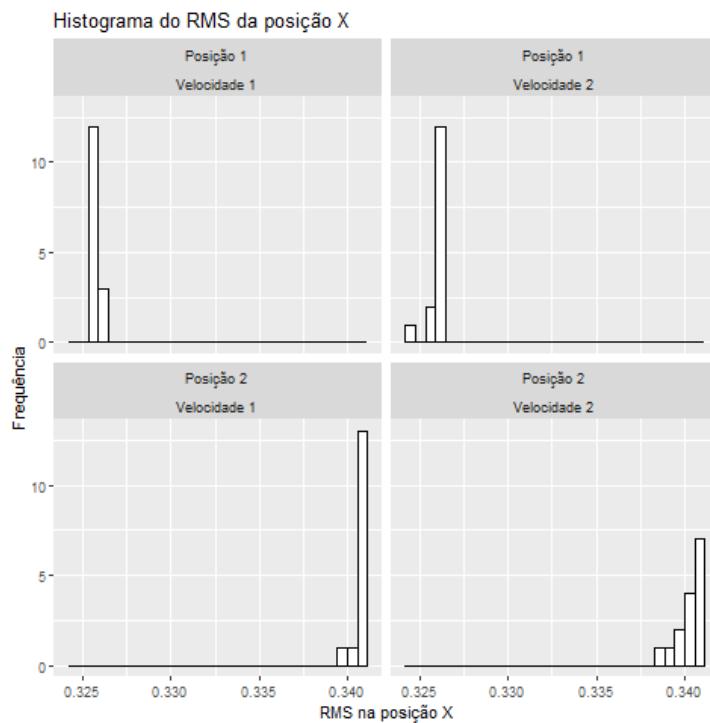
Fonte: Autores

Figura 26: Histograma do tempo de *scan*.



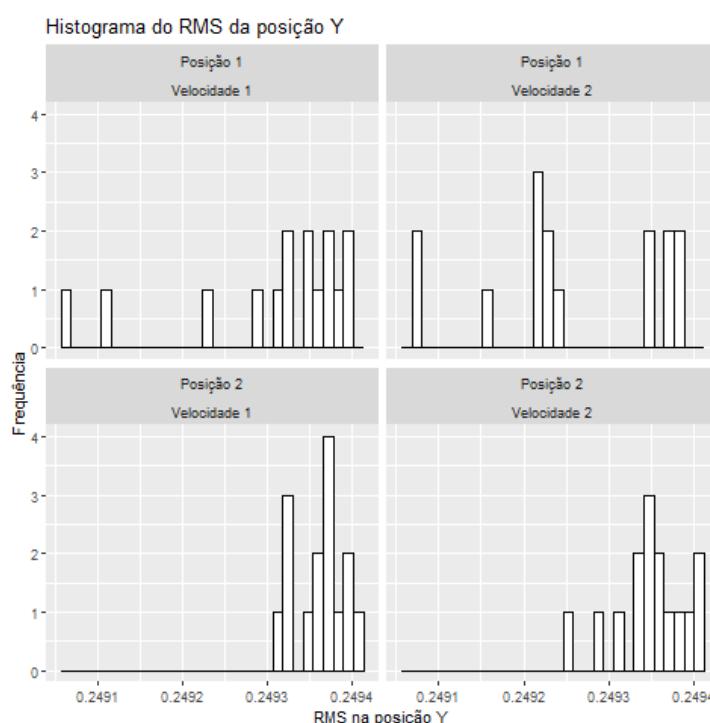
Fonte: Autores

Figura 27: Histograma do RMS da posição X.



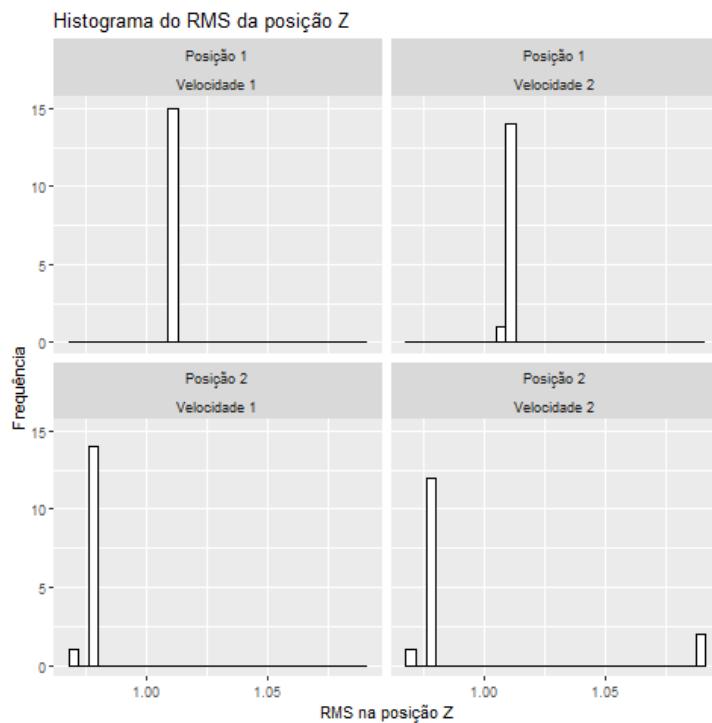
Fonte: Autores

Figura 28: Histograma do RMS da posição Y.



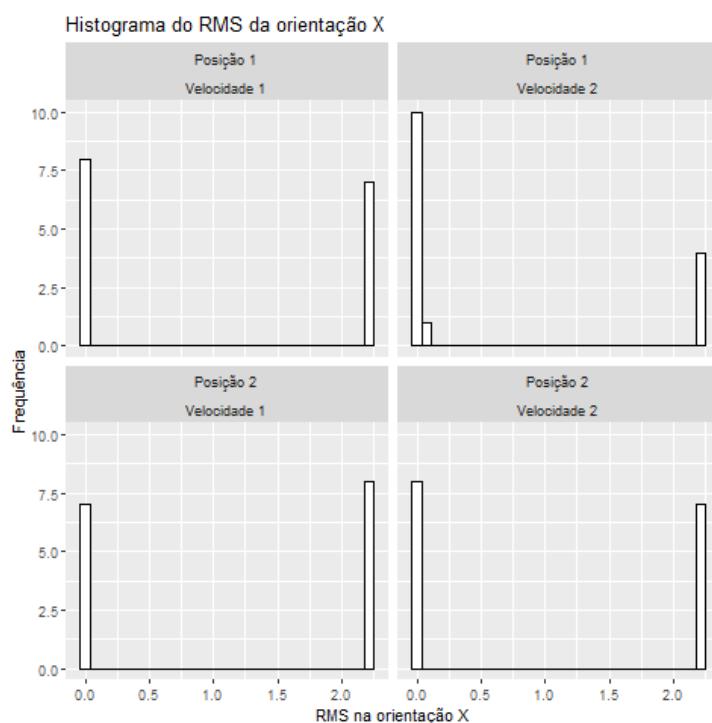
Fonte: Autores

Figura 29: Histograma do RMS da posição Z.



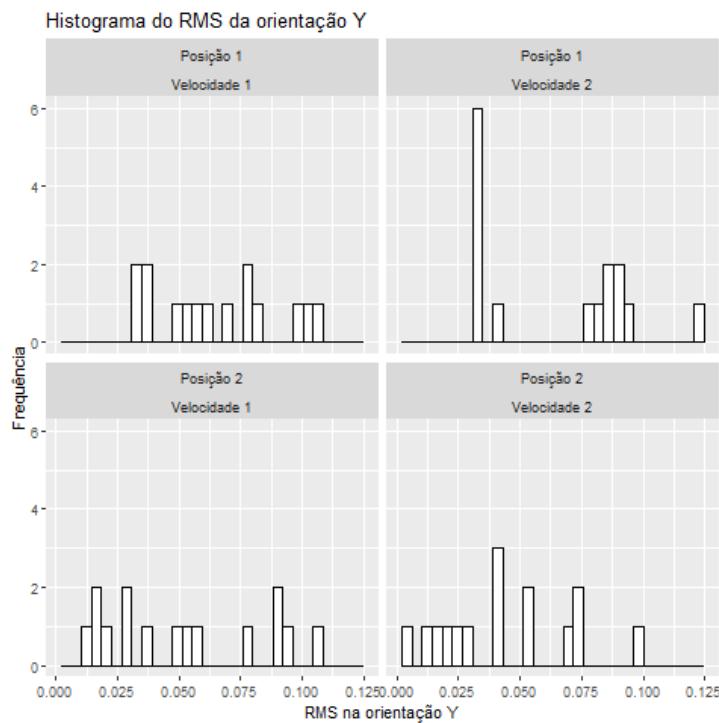
Fonte: Autores

Figura 30: Histograma do RMS da orientação X.



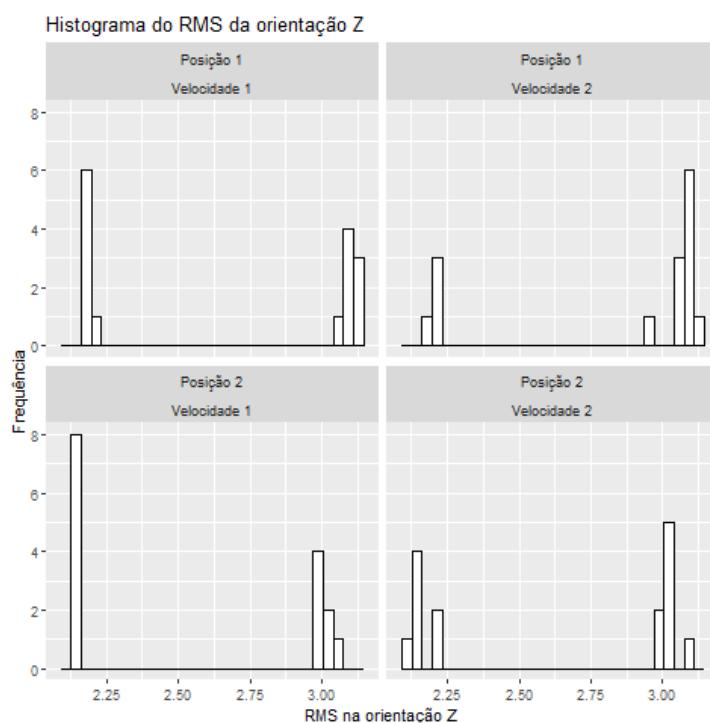
Fonte: Autores

Figura 31: Histograma do RMS da orientação Y.



Fonte: Autores

Figura 32: Histograma do RMS da orientação Z.



Fonte: Autores

4.2.3 ANOVA

Com o objetivo de analisar o sistema de medição utilizado na coleta de dados, o teste realizado será o de Análise de Variância, da sigla em Inglês ANOVA, e tem por intuito verificar a repetibilidade e a reproduzibilidade de um sistema quando o mesmo é submetido a condições distintas.

Este teste gera um conjunto de seis gráficos que são acompanhados de outros resultados numéricos correlacionados e arranjados em tabelas. O primeiro destes relaciona os Componentes de Variação (*Components of Variation*), o qual mostra através de um gráfico de barras o percentual de contribuição para a variação geral, azul, e no estudo, rosa, de cada um dos seguintes casos: Sistema de Medição (*Gage R & R*), Repetibilidade, Reproduzibilidade e de Peça-para-Peça (*Part-to-Part*) ([Qualidade e Produtividade, 2019](#)).

O segundo e o terceiro gráficos são denominados Carta de Amplitude por Operador (*R Chart by Operator*) e Carta de Média por Operador (*Chart by Operator*) respectivamente ([Qualidade e Produtividade, 2019](#)). Eles mostram a amplitude e a média amostral das medidas realizadas no eixo das ordenadas em função de cada componente testado no eixo das abscissas. Os diferentes sistemas de medição são delimitados pelas diferentes colunas mostradas. Estes permitem a comparação visual dos valores obtidos de forma a identificar possíveis problemas com o sistema de medição avaliado ou com as peças em estudo, tais problemas poderiam surgir como uma grande amplitude nos valores obtidos para uma peça em todas as medidas realizadas, indicando um problema com a peça, ou na variação com um sistema específico.

O quarto e o quinto gráficos mostram os valores das medidas obtidas em função do componente ou do operador respectivamente, assim são denominados Medida por Componente (*Measurement by Part*) e Medida por Operador (*Measurement by Operator*) ([Qualidade e Produtividade, 2019](#)). A dispersão dos valores obtidos representados por pontos e a variação das suas médias representados pelas linhas nestes gráficos podem indicar também problemas existentes em peças ou nos sistemas de medição. Por fim, o sexto gráfico de Interação entre Componente e Operador (*Part:Operator Interaction*) visa mostrar possíveis interações entre os valores obtidos para peças e operadores específicos ([Qualidade e Produtividade, 2019](#)).

Através da análise destes gráficos e das tabelas que os acompanham, espera-se que possíveis problemas com os componentes testados possam ser encontrados e posteriormente resolvidos. Desta forma, serão apresentados nas próximas seções o método utilizado para execução deste teste, seguido da discussão dos resultados obtidos. Por fim, o algoritmo utilizado na confecção deste teste foi feito em linguagem R e está disposto no Apêndice.

A partir da aplicação do teste ANOVA foi verificado que o Número de Categorias Distintas (NDC), que estima quantos grupos separados de peças o sistema consegue

distinguir, obteve um valor de 2. Esse resultado indica que para o sistema de medição, pode-se dividir a variação do processo em duas faixas, como alto e baixo por exemplo.

A contribuição de variância dos processos de medidas se mostraram elevados, mas a repetibilidade se mostrou numa faixa aceitável, como mostrado na Tabela 9.

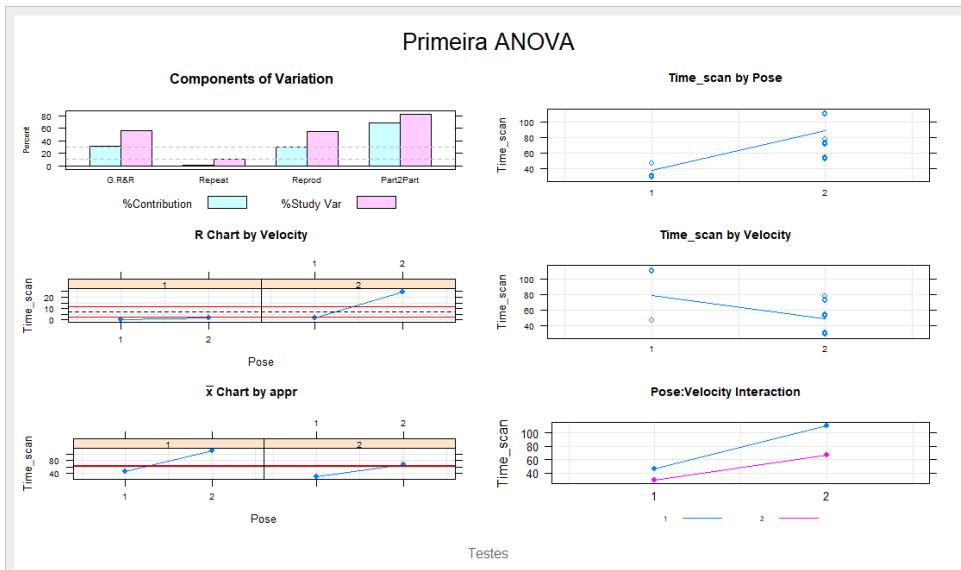
Tabela 9: Teste ANOVA estudo de contribuição para o manipulador.

	VarComp	%Contrib
Total Gage R&R	547.93002	31.41
Repeatability	19.97357	1.14
Reproducibility	527.95645	30.26
Velocidade	366.60727	21.01
Pose:Velocity	161.34919	9.25
Part-To-Part	1196.68446	68.59
Total Variation	1744.61449	100.00

Fonte: Autores.

A variação do estudo percentual também é mostrada na Figura 33. A variação do estudo percentual (*%Study Var*) é utilizada para comparar a variação sistema de medição com a variação total. Nos valores de *Gage R & R* e Repetibilidade, a *%Study Var* é menor do que 30% da variação do processo, mantendo-se numa faixa aceitável.

Figura 33: Figura representando o teste ANOVA.



Fonte: Autores

Devido aos resultados obtidos em *%Contribution* em *%Study Var* e no gráfico de medição, os resultados demonstraram-se satisfatórios para o teste ANOVA realizado apenas no quesito repetibilidade.

4.2.4 T-teste

Os testes t são testes de hipótese utilizados para comparação de médias. Pode-se comparar as médias de dois grupos com um teste t para duas amostras (RUXTON, 2006). Nos testes realizados, é colocada a hipótese alternativa, na qual a verdadeira diferença nas médias não é igual a 0. Isso significa que há uma diferença estatística entre as duas médias, ou seja, há uma diferença real quando varia-se a velocidade e posição (Tabela 10).

Quando ocorre uma variação da posição, há uma diferença das médias nas variáveis RMS das orientações X e Z. Assim como ocorre uma mudança na velocidade, as médias do RMS das posições e orientações X, Y e Z variam, indicando que este é um fator decisivo para a variação deste erro.

Tabela 10: T-teste.

	T - test								
	Tempo			rms da posição			rms da orientação		
	scan	apertar o botão	missão	x	y	z	x	y	z
p-value posição	8.626e-14	9.599e-15	<2.2e-16	<2.2e-16	0.0008898	5.371e-05	0.3307	0.0279	0.1164
p-value velocidade	7.817e-05	0.0002016	3.89e-07	0.9571	0.1579	0.2339	0.3008	0.5643	0.3023

Fonte: Autores

4.2.5 F-teste

Os testes F são uma razão de duas variâncias (SANDERSON; WINDMEIJER, 2016). As variâncias são uma medida de dispersão, ou até que ponto os dados estão dispersos em relação à sua média. Valores maiores representam maior dispersão (Tabela 11). Quando o p valor é maior que o nível de significância de 0,05, não há diferença significativa entre as duas variâncias. Os p valores do RMS de todas as orientações são maiores que 0.05 quando se altera a posição. Isso também ocorre com os RMS das posições X e Y, e com o RMS de todas as orientações, o que significa que as variâncias diferem quando altera a velocidade.

Tabela 11: F-teste.

	F - test								
	Tempo			rms da posição			rms da orientação		
	scan	apertar o botão	missão	x	y	z	x	y	z
p-value posição	2.025e-06	0.0005775	0.005817	0.006671	1.866e-07	<2.2e-16	0.8387	0.7795	0.9493
p-value velocidade	0.01443	0.02737	0.02705	0.7827	0.3543	0.003821	0.8315	0.8044	0.8049

Fonte: Autores

4.3 Estudo dos modos e efeitos de falhas

Realizou-se o *Failure Mode Effects and Criticality Analysis* (FMECA), estudo que analisa todos os componentes com o objetivo de identificar possíveis falhas, suas causas

e efeitos, os quais são comumente demonstrados em tabela com o objetivo de qualificar sua severidade, ocorrência, detecção e identificar qual o problema representa maior risco de ocorrer em cada componente do sistema, oriundo do produto entre os Números de Prioridade de Risco (NPR) demonstrado na tabela 12.

Foi feito um *FMEA* para cada componente constituinte do sistema de um manipulador robótico, encontrado nas Tabelas 13 a 17.

Tabela 12: Coeficiente de Prioridade de Risco

		SEVERIDADE			OCORRÊNCIA			DETECÇÃO		
Índice	Severidade	Critério		Ocorrência	Proporção		Índice	Detectão	Critério	
1	Mínima	O cliente mal percebe que a falha ocorre		1	Remota	1:1.000.000				
2	Pequena	Ligeira deterior. no desemp. com leve descontent. do cliente		2	Pequena	1:20.000 1:4.000 1:1000				
3	Moderada	Deterior. signific. do desemp. de um sistema com descontentamento do cliente		3	Moderada	1:400				
4	Alta	Sistema deixa de funcionar e é grande o descontentamento do cliente		4	Alta	1:40 1:20				
5	Muito Alta	Idem ao anterior porém afeta a segurança/ambiente		5	Muito Alta	1: 8 1: 2				

Fonte: Autores

Tabela 13: Estudo dos modos e efeitos de falha: Motor

Função(s)	Modo(s) de falha em potencial	S	Efeito(s) potencial(is) da falha(s)	Motor O D	NPR (SxOxD)	Causa(s) potenciais	Ação(es) recomendada(s)	Competência
Movimentação do manipulador	Rotação incorreta	4	Perda da orientação do manipulador	2 4	32	Isolamento ineficiente	Realizar inspeção visual do motor	Elétrica
	Superaquecimento	2	Alto consumo de energia	3 5	30	Emperramento	Realizar inspeção visual do motor	Mecânica
	Danificação da carcaça	1	Deformação de partes	3 3	9	Choque mecânico	Realizar inspeção visual do motor	Mecânica
	Curto circuito	5	Parada do motor	2 1	10	Avarias no sistema elétrico	Medir isolamento	Elétrica
	Sobrecarga	5	Parada do motor	2 1	10	Sobrecarga mecânica	Verificar as forças atuantes nos motores	Mecânica
	Perda de comunicação	3	Parada do motor	4 1	12	Mau contato elétrico	Verificar os contatos dos conectores	Elétrica
Exercer força para apertar o botão	Perda de torque	3	Perda de desempenho	3 4	36	Sobrecarga mecânica	Certificar se o manipulador está carregando peso extra	Mecânica
	Resposta imprecisa do controlador	2	Perda da orientação do manipulador	2 4	16	Calibração incorreta	Ajustar o equipamento e inspecionar os dados recebidos	Mecânica

Fonte: Autores

Tabela 14: Estudo dos modos e efeitos de falha: Câmera

Função(es)	Modo(s) de falha em potencial	S	Efeito(s) potencial(is) da falha(s)	Câmera RGB			Causa(s) potenciais	Ação(es) recomendada(s)	Competência	
				O	D	NPR (SxOxD)				
Auxiliar a localização do botão	Incapacidade de localizar o marco	4	Perda da capacidade de detecção	3	2	24	Obstrução no campo de visão	Executar a limpeza da mesa de trabalho	Operador	
	Ruptura da estrutura da câmera	3		3	2	18	Variação na luminosidade	Controle da luminosidade do ambiente	Operador	
	Coleta de dados inconsistentes	3		2	4	24	Ausência da referência fiducial	Posicionar a tag em local visível	Operador	
	Perda da calibração	3		2	4	24	Danificação por colisão no manipulador	Verificar as condições do equipamento antes das missões	Mecânica	
	Iluminação insuficiente	3		3	3	27	Obstrução parcial no campo de visão	Executar a limpeza da mesa de trabalho	Operador	
	Posicionamento da câmera no local incorreto	3		3	3	27	Falha no canal de comunicação	Ajustes ou troca de cabos	Operador	
	Variação de iluminação	3		3	3	27	Calibração incorreta da câmera	Recalibração da câmera	Operador	
									Operador	

Fonte: Autores

Tabela 15: Estudo dos modos e efeitos de falha: NUC

Função(es)	Modo(s) de falha em potencial	S	Efeito(s) potencial(is) da falha(s)	NUC			Causa(s) potenciais	Ação(es) recomendada(s)	Competência
				O	D	NPR (SxOxD)			
Processamento do sistema principal do manipulador	Não processamento do sistema principal do manipulador	4	Não funcionamento do Manipulador	2	3	24	Falha de alimentação	Inspeções periódicas	Eletrônica
							Falha na porta de comunicação	Executar rotinas de envio e recebimento de mensagens comunicação com a NUC (PING)	Software
							Falta de espaço do HD	Rotina de limpeza e verificação do espaço do HD	Software
							Danificação do HD	Rotina de verificação da condição de operação do HD	Eletrônica
							Queima/danificação de componentes	Inspeção de bancada	Eletrônica
							Falta de memória	Verificar o uso de memória e adapta-lá ao projeto	Software

Fonte: Autores

Tabela 16: Estudo dos modos e efeitos de falha: Conector Mecânico

Função(es):	Modo(s) de falha em potencial	S	Efeito(s) potencial(is) da falha(s)	Conector Mecânico			Causa(s) potenciais	Ação(es) recomendada(s)	Competência
				O	D	NPR (SxOxD)			
Fazer a conexão entre o atuador e o link	Defeito estrutural	3	Modificar as dimensões estruturais ou de ruptura	2	5	30	Colisão da estrutura	Calcular um plano de colisão	Mecânica
							Danificação por reação química	Manter longe de materiais reagentes	Mecânica

Fonte: Autores

Tabela 17: Estudo dos modos e efeitos de falha: Cabo

Função(es)	Modo(s) de falha em potencial	S	Efeito(s) potencial(is) da falha(s)	Cabo			Causa(s) potenciais	Ação(es) recomendada(s)	Competência
				O	D	NPR (SxOxD)			
Fazer a conexão entre o NUC e os motores	Falha da comunicação	4	Incapacidade de conexão entre os componentes	2	3	24	Entradas e saídas não funcionais	Teste de bancada dos conectores	Elétrica
							Entradas e saídas com mau contato	Teste de bancada dos conectores	Elétrica

Fonte: Autores

5 GESTÃO DO CONHECIMENTO

A Gestão do Conhecimento tem por objetivo apoiar e orientar de maneira eficiente a busca do conhecimento organizacional. Para tanto, reúne a função de organizar, disseminar, avaliar, mensurar e capturar os diversos conhecimentos organizacionais gerados pelo projeto. ([STRAUHS et al., 2012](#)).

- Organizar o conhecimento: identificar os conhecimentos importantes e relevantes para criar um repositório com todos os conhecimentos gerados a partir da realização do projeto.
- Disseminar o conhecimento: através do compartilhamento de conhecimento, somados a troca de experiências, ocorrerá o aprimoramento do aprendizado.
- Avaliar o conhecimento: utilizado para o aprimoramento contínuo das idéias aprendidas, com o objetivo de colocação em prática, para agilizar implementações posteriores de todo o conhecimento obtido e compartilhado.
- Mensurar o conhecimento: conjunto de medidas que permitem quantificar e qualificar os conhecimentos obtidos com o intuito de auxiliar a melhora e a evolução na implementação de projetos iguais os semelhantes.
- Criar conhecimento: expandir e adquirir idéias, a partir de conhecimento obtido, somado a experiências semelhantes, visando a implementação de outros novos conhecimentos, para o aprimoramento de futuros projetos.

5.1 Lições aprendidas

As lições aprendidas podem ser definidas como um conjunto de aprendizagens obtidas no processo de realização do projeto ([PMBOK, 2008](#)). A coleta e análise de lições aprendidas são uma forma de transferir todo o conhecimento gerado durante a fase de execução de um projeto ([VERONESE, 2014](#)).

O processo de lições aprendidas necessita do envolvimento de todos os participantes, pois os portadores do conhecimento são as pessoas envolvidas na realização do projeto. Na Tabela [18](#) é mostrada uma relação das lições aprendidas durante o projeto, descrevendo o caminho percorrido para finalização desse desafio.

Tabela 18: Lições aprendidas

Tema	Fase	Impacto	Lições aprendidas			Recomendações para os próximos projetos
			O que ocorreu?	Como resolveu?	Resultados	
Gestão	Planejamento	Negativo	Falta de comunicação entre os membros para compartilhamento de informações	Reunião diária em horários específicos, organização e compartilhamento de informações no Trello	Melhor disposição do tempo gasto para realizar tarefas e troca de informações de modo mais eficiente entre os membros	Divida o tempo diário em pelo menos duas reuniões para organizar e atualizar suas atividades para esclarecimentos
Técnico	Execução	Negativo	Dificuldade de integração do manipulador com o ROS	Testes em vários pacotes, de modo a verificar a possibilidade de solução do problema	Resolução parcial do problema com a utilização de uma branch descontinuada do pacote Dynamixel workbench	Verificar com antecedência a existência de pacotes que possibilitem a integração do protótipo ao ROS
Técnico	Execução	Negativo	Motores de modelos distintos não funcionaram em conjunto, mesmo sendo do mesmo fabricante	Troca dos motores, possibilitando a utilização do mesmo modelo para todo o projeto	Comunicação adequada entre os motores e garantia da sincronicidade	Listar os equipamentos disponíveis e verificar com antecedência a compatibilidade entre eles
Técnico	Execução	Negativo	Impressão de peça 3d de materiais diferentes ABS, PLA, PETG, dimensões diferentes quando finalizadas, demandando tempo extra para ajustes finais	Acabamento final na peça para um encaixe adequado sem danificar o equipamento	Encaixe adequado do equipamento	Imprimir peças pequenas de partes específicas como modelo, toda vez que se ajustar ou substituir algum material na impressora 3d
Aquisição	Execução	Negativo	Compra de materiais para confecção de partes do manipulador, o qual sua finalização não foi satisfatória, acarretando em retrabalho e gastos financeiros	Modelagem e impressão 3d da peça	Melhor acabamento, funcionalidade e redução de peso total da estrutura	Fazer uma análise geral com a equipe da necessidade do material e procurar outras opções de fabricar o equipamento
Gestão	Teste	Negativo	O conversor USB foi utilizado em testes nos motores e houve uma sobrecarga quando testado em um determinado motor com configurações diferentes	Substituição dos componentes danificados	Possibilidade de testes nos motores restantes	Organizar a bancada e sinalizar a diferença entre os equipamentos similares
Técnico	Execução	Negativo	Problemas com a repetibilidade de movimentos gerados pelo planejador de trajetória	Aumento da tolerância na movimentação das juntas para execução do movimento	Capacidade de realizar movimentos que antes não eram possíveis, mas ocasionou uma orientação indesejada do end effector na posição final	Fazer uma análise do MoveIt! para verificar a possibilidade de contornar estes erros na movimentação, caso não seja possível, verificar outras estratégias para o planejamento de trajetória
Técnico	Execução	Negativo	Problemas com a detecção do marcador ArUco em determinadas posições	Calibração da câmera, redução da resolução da imagem e melhoria no algoritmo de busca do painel elétrico	Melhoria na precisão da detecção	Utilização de técnicas de visão computacional para garantir redundância nas informações coletadas e elaborar lógica para realizar coleta de informações do ArUco constantemente, ao passo que a câmera se aproxima do marcador

Fonte: Autores

5.2 Guia de uso

Este projeto foi desenvolvido e testado no Ubuntu 18.04 LTS usando [ROS](#) Melodic Morenia. Embora este *software* possa funcionar em outras plataformas, não se recomenda o seu uso em plataformas com configurações diferentes das mencionadas anteriormente.

5.2.1 Exigências

O correto funcionamento do manipulador exige a instalação dos pacotes e programas listados a seguir.

5.2.1.1 Pacotes

- Rosdep: é uma ferramenta de linha de comando para instalar pacotes ROS ou dependências do sistema.
- MoveIt Versão completa: é uma plataforma para o desenvolvimento de aplicações de robótica utilizando manipuladores, esse pacote engloba o planejamento de movimento, manipulação, percepção 3D, cinemática, controle e navegação.
- MoveIt *Visual Tools*: pacote utilizado para exibir e depurar dados do MoveIt no Rviz por meio de marcadores publicados, trajetórias e objetos de colisão do MoveIt.
- TRAC-IK *Kinematics Solver*: TRAC-IK é um solucionador de cinemática inversa desenvolvido pela TRACLabs que combina duas implementações IK para obter soluções mais confiáveis do que os solucionadores IK de código aberto comuns disponíveis.
- Pacote Gazebo *ROS control*: um conjunto de pacotes que inclui interfaces de controladores, gerenciadores de controladores, transmissões e interfaces de hardware.
- Pacote *ROS position controller*: efetua o controle de uma posição desejada para a junta.
- Pacote *ROS effort controller*: efetua o controle de torque no caso de juntas rotativas.
- Pacote *ROS joint*: contém uma ferramenta para definir e publicar valores dos estado das juntas.
- Pacote DYNAMIXEL SDK: é um kit de desenvolvimento de software que fornece funções de controle DYNAMIXEL usando comunicação por pacote.
- Pacote DYNAMIXEL *Workbench*: é uma solução baseada no DYNAMIXEL SDK para ROS. Este metapacote permite que você altere facilmente o ID, a taxa de transmissão e o modo de operação do DYNAMIXEL, objetivando um simples e fácil uso de qualquer motor DYNAMIXEL.
- Pacote DYNAMIXEL *Workbench Messages*: este pacote inclui mensagens ROS e serviços para o DYNAMIXEL *Workbench*.
- Pacote Pylon *ROS Camera*: o driver pylon ROS oficial, utilizado para câmeras Basler GigE Vision e USB3 Vision.

5.2.1.2 Pacote Externo

- Bir_marker_localization <https://github.com/Brazilian-Institute-of-Robotics/bir-marker_localization/tree/final_settings> (*branch final settings*). O qual requer a instalação do OpenCV 3.3.1 para o seu funcionamento. A instalação deste é explicada no guia de utilização deste pacote.

5.2.1.3 Programas

- Pylon Viewer<<https://docs.baslerweb.com/overview-of-the-pylon-viewer>>: é um pacote de software composto por um SDK, drivers e ferramentas utilizadas operar câmeras Basler.

5.2.2 Como executar

Para executar a inicialização da rotina do manipulador no equipamento físico, depois da instalação de todos os pacotes e programas listados anteriormente, deve-se primeiro executar o programa *Pylon Viewer* com a câmera conectada. Em seguida, deve-se ir na aba *Image Format Control* e no campo *Scaling* deve-se preencher o valor 0,4.

Em seguida, em uma área de trabalho do ambiente *ROS-catkin*, deve-se clonar o nosso repositório <https://github.com/Brazilian-Institute-of-Robotics/borg_arm> ou extrair o arquivo ZIP baixado do mesmo, logo depois seguir os comandos abaixo para executar o movimento do manipulador real:

- No terminal, na raíz da área de trabalho *catkin* contendo os arquivos do manipulador, executar
 - catkin_make
- Executar os próximos comandos na sua área de trabalho em terminais diferentes
 - Iniciar o pacote da câmera, os controladores dos motores e o Moveit
 - roslaunch borg_arm_controller real_arm_1.launch
 - Iniciar a conexão do MoveIt com os motores, inicia o painel elétrico no cenário, e o pacote *bir_marker_localization* com nosso arquivo de configuração
 - roslaunch borg_arm_controller real_arm_2.launch
 - Iniciar o *borg_arm_controller* específico para o manipulador real
 - roslaunch borg_arm_controller real_arm_3.launch

- E, para finalizar, após todos os comandos anteriores, deve-se executar comando a seguir para que o manipulador initialize a rotina para apertar a botoeira.
 - `rosservice call /StartScanService "{}"`

Por outro lado, para efetuar a simulação do manipulador robótico em ambiente Gazebo, deve-se executar os seguintes comandos:

- No terminal, na raíz da área de trabalho *catkin* contendo os arquivos do manipulador, executar
 - `catkin_make`
- Executar os próximos comandos na sua área de trabalho em terminais diferentes
- Iniciar todos os pacotes necessários para a simulação
 - `roslaunch borg_arm_controller simulation_arm_startup.launch.launch`
- Inicializar o *borg_arm_controller* específico para a simulação.
 - `roslaunch borg_arm_controller simulation_borg_arm_controller.launch.launch`
- E, para finalizar, após todos os comandos anteriores, deve-se executar comando a seguir para que o manipulador initialize a rotina para apertar a botoeira.
 - `rosservice call /StartScanService "{}"`

6 CONCLUSÃO

O projeto descrito neste relatório apresentou resultados obtidos no estudo, concepção, e desenvolvimento de um manipulador robótico de 5 graus de liberdade denominado **The Borg Arm**. A finalidade proposta neste relatório foi realizar o pressionamento de um botão integrado em um painel elétrico, utilizando a detecção de um marco fiducial ArUco presente no painel para auxiliar na localização do botão.

No desenvolvimento deste sistema foram utilizados o [ROS](#) na versão Melodic Morenia, o Gazebo na versão 9 e o MoveIt, bem como outros algoritmos existentes nestes *frameworks* e *softwares*. Tais como o RRT-Connect, disponível no MoveIt através da biblioteca [OMPL](#), foi utilizado na criação de uma trajetória dada uma pose de interesse, e o TRAC-IK que realiza a solução da cinemática inversa necessária para a movimentação do manipulador. As trajetórias desejadas para o manipulador, isto é, a posição de cada uma das suas juntas no tempo, são enviadas como entrada nos controladores gerados pelo pacote ROS Control, responsável pela geração dos sinais de controle e envio aos atuadores durante a simulação. Por sua vez, o pacote Dynamixel Workbench assume este papel durante a execução no manipulador real construído.

Uma câmera acoplada ao manipulador é responsável pela captação de imagens, que são processadas pelo pacote Bir Marker Localization, capaz de detectar o marco fiducial ArUco e fornecer a posição e orientação da câmera em relação ao marco.

Os estudos e procedimentos realizados neste projeto obtiveram um resultado parcialmente satisfatório na tarefa de utilizar um robô manipulador em conjunto com uma câmera, para acionamento da botoeira. Os experimentos feitos mostram que o manipulador desenvolvido pode ser utilizado em projetos futuros, a depender da adequação do mesmo para a atividade proposta ou implementação de algumas melhorias.

Durante o desenvolvimento, notou-se que a posição de referência encontrada para o marco fiducial apresentava desvios maiores do que o desejado, dificultando na precisão da aproximação e pressionamento do botão. Após falhas recorrentes na execução da tarefa antes e durante a demonstração do manipulador, foram realizados pequenas modificações no código do manipulador. Durante os testes de confiabilidade, constatou-se que a fixação da coordenada y na pose do marco fiducial possibilitou a correta execução de 65 dos 67 testes realizados. Ressalta-se que esta coordenada está na direção normal ao marco fiducial.

Desta forma, sugere-se um estudo de melhorias na detecção do marco fiducial, seja ele na aplicação de mais marcos fiduciais simultaneamente no entorno da botoeira, na aplicação de outras técnicas de sensoriamento e estimativa de distância relativa entre o manipulador e o painel elétrico, ou até mesmo na análise dos motivos causadores da maior imprecisão no eixo normal ao ArUco.

REFERÊNCIAS

- ABB. *Ágil e flexível O novo robô IRB 120 da ABB e o seu controlador IRC5 Compact.* 2014. Disponível em: <<https://search-ext.abb.com/library/Download.aspx?DocumentID=9AKK106103A3760&LanguageCode=pt&DocumentPartId=&Action=Launch>>. Acesso em: 22 de abr. de 2020. Citado na página 21.
- ABNT, N. Nm 60884-1. *Plugues e tomadas para uso doméstico e análogo.*, 2010. Citado na página 23.
- ABNT, N. 14136. *Plugues e tomadas para uso doméstico e análogo até 20 A - 250V AC,* 2013. Citado na página 23.
- ABNT, N. Abnt nbr iso 10218-1:2018. *Robôs e dispositivos robóticos — Requisitos de segurança para robôs industriais.*, 2018. Citado na página 26.
- ANDALUZ, V. H. et al. Passivity-based visual feedback control with dynamic compensation of mobile manipulators: Stability and l2-gain performance analysis. *Robotics and Autonomous Systems*, Elsevier, v. 66, p. 64–74, 2015. Citado na página 21.
- BURRELL, T. et al. Towards a cooperative robotic system for autonomous pipe cutting in nuclear decommissioning. In: IEEE. *2018 UKACC 12th International Conference on Control (CONTROL)*. [S.l.], 2018. p. 283–288. Citado na página 22.
- CARIUS, J. et al. Deployment of an autonomous mobile manipulator at mbzirc. *Journal of Field Robotics*, Wiley Online Library, v. 35, n. 8, p. 1342–1357, 2018. Citado na página 21.
- CHEN, C. et al. Analysis and simulation of kinematics of 5-dof nuclear power station robot manipulator. In: IEEE. *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*. [S.l.], 2014. p. 2025–2030. Citado na página 20.
- CHEN, Z. et al. Kinematics analysis and application of 5-dof manipulator with special joint. In: IEEE. *2017 Chinese Automation Congress (CAC)*. [S.l.], 2017. p. 7421–7426. Citado na página 21.
- CHENG, H.; CHEN, H.; LIU, Y. Object handling using autonomous industrial mobile manipulator. In: IEEE. *2013 IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems*. [S.l.], 2013. p. 36–41. Citado na página 21.
- CHUNG, C.-S.; WANG, H.; COOPER, R. A. Autonomous function of wheelchair-mounted robotic manipulators to perform daily activities. In: IEEE. *2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR)*. [S.l.], 2013. p. 1–6. Citado na página 22.
- CRAIG, J. J. Introduction to robotics. 2005. Citado 2 vezes nas páginas 15 e 87.
- CUI, J. et al. A review of teleoperation system control. In: FLORIDA ATLANTIC UNIVERSITY BOCA RATON, FL. *Proceedings of the Florida Conference on Recent Advances in Robotics*. [S.l.], 2003. p. 1–12. Citado na página 16.
- DEROOS, B. G.; PRICE, J. D.; REIDY, J. J. Law enforcement robot technology assessment. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Enabling Technologies for Law Enforcement and Security*. [S.l.], 2001. v. 4232, p. 409–420. Citado na página 16.

DIS, I. 8373: 2012. robots and robotic devices—vocabulary. *International Standardization Organization (ISO)*, 2012. Citado na página 19.

DONG, G.; ZHU, Z. H. Incremental visual servo control of robotic manipulator for autonomous capture of non-cooperative target. *Advanced Robotics*, Taylor & Francis, v. 30, n. 22, p. 1458–1465, 2016. Citado na página 21.

DONG, G.; ZHU, Z. H. Predictive visual servo kinematic control for autonomous robotic capture of non-cooperative space target. *Acta Astronautica*, Elsevier, v. 151, p. 173–181, 2018. Citado na página 21.

DU, S. et al. Moving obstacle avoidance of a 5-dof robot manipulator by using repulsive vector. In: IEEE. *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. [S.I.], 2017. p. 688–693. Citado na página 22.

DYNAMICS, E. *iARM*. 2012. Disponível em: <<http://www.exactdynamics.nl/site/?page=iarm>>. Acesso em: 22 de abr. de 2020. Citado na página 21.

FARIA, R. O. et al. A methodology for autonomous robotic manipulation of valves using visual sensing. *IFAC-PapersOnLine*, Elsevier, v. 48, n. 6, p. 221–228, 2015. Citado na página 22.

FENG, C. et al. Vision guided autonomous robotic assembly and as-built scanning on unstructured construction sites. *Automation in Construction*, Elsevier, v. 59, p. 128–138, 2015. Citado na página 22.

FRÜH, H. Neuronics ag: Handling robots equipped with highly redundant sensors. *CHIMIA International Journal for Chemistry*, Swiss Chemical Society, v. 57, n. 6, p. 340–341, 2003. Citado na página 22.

GAZEBO. *Color And Texture Models*. 2020. Disponível em: <http://gazebosim.org/tutorials?tut=color_model>. Acesso em: 23 de abr. de 2020. Citado na página 24.

HAMNER, B. et al. An autonomous mobile manipulator for assembly tasks. *Autonomous Robots*, Springer, v. 28, n. 1, p. 131, 2010. Citado na página 22.

HAO, L.; RU, L. An open 5-dof manipulator platform and its control applications. In: IEEE. *Proceedings 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)*. [S.I.], 2013. p. 824–827. Citado na página 21.

HUANG, G.-S.; TSENG, P.-H. Development and analysis of 5-dof manipulator kinematics. In: IEEE. *2016 International Conference on System Science and Engineering (ICSSE)*. [S.I.], 2016. p. 1–4. Citado na página 20.

HVILSHØJ, M.; BØGH, S. “little helper”—an autonomous industrial mobile manipulator concept. *International Journal of Advanced Robotic Systems*, SAGE Publications Sage UK: London, England, v. 8, n. 2, p. 15, 2011. Citado na página 21.

ILIUKHIN, V. et al. The modeling of inverse kinematics for 5 dof manipulator. *Procedia Engineering*, Elsevier, v. 176, p. 498–505, 2017. Citado na página 21.

ISSA, A. et al. 5-dof robot manipulator modelling, development and automation using lab-view, vision assistant and arduino. In: IEEE. *2019 International Conference on Promising Electronic Technologies (ICPET)*. [S.I.], 2019. p. 124–129. Citado na página 21.

KIM, H. W.; CHEN, H.; LEE, J. M. Path planning of 5-dof manipulator. In: IEEE. *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. [S.l.], 2014. p. 877–881. Citado na página 21.

KINOVA. *KINOVA MICO Robotic arm*. 2018. Disponível em: <<https://www.kinovarobotics.com/en/resources/technical-resources-library>>. Acesso em: 22 de abr. de 2020. Citado na página 22.

KUKA. *The Series 2000 shelf-mounted robots for shorter cycle times*. 2002. Disponível em: <https://www.kuka.com/-/media/kuka-downloads/imported/6b77eecacfe542d3b736af377562ecaa/pf0031_kr_2102_k_en.pdf?rev=58d7be9330204d8cbf0c4d23a81339d9>. Acesso em: 22 de abr. de 2020. Citado na página 22.

KUKA. *The press-to-press robot for the high payload category*. 2005. Disponível em: <https://www.kuka.com/-/media/kuka-downloads/imported/6b77eecacfe542d3b736af377562ecaa/pf0053_kr_1002_p_kr_1202_p_en.pdf>. Acesso em: 22 de abr. de 2020. Citado na página 22.

LAROCHE, B. P.; ZHU, Z. H. Autonomous robotic capture of non-cooperative target using visual servoing and motion predictive control. *Autonomous Robots*, Springer, v. 37, n. 2, p. 157–167, 2014. Citado na página 21.

LI, F. et al. Kinematics modeling and workspace analysis of a 5-dof hydraulic manipulator. In: IEEE. *2019 5th International Conference on Control, Automation and Robotics (ICCAR)*. [S.l.], 2019. p. 713–718. Citado na página 20.

LU, J.; XU, D.; WANG, P. A kinematics analysis for a 5-dof manipulator. In: IEEE. *The 26th Chinese Control and Decision Conference (2014 CCDC)*. [S.l.], 2014. p. 1495–1499. Citado na página 20.

LUNDBERG, C.; REINHOLD, R.; CHRISTENSEN, H. I. Evaluation of robot deployment in live missions with the military, police, and fire brigade. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense VI*. [S.l.], 2007. v. 6538, p. 65380R. Citado na página 16.

MELLODGE, P. Chapter 2 - system modeling. In: MELLODGE, P. (Ed.). *A Practical Approach to Dynamical Systems for Engineers*. Woodhead Publishing, 2016. p. 17 – 145. ISBN 978-0-08-100202-5. Disponível em: <<http://www.sciencedirect.com/science/article/pii/B9780081002025000024>>. Citado na página 93.

MURZAKAEV, R. T.; POLIAKOV, A. N.; ARTEMEV, V. V. Solution of kinematics problem of 5-dof manipulator emitter for non-destructive defectoscopy. In: IEEE. *2018 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM)*. [S.l.], 2018. p. 1–5. Citado na página 20.

NARVÁEZ, E. et al. Vision based autonomous docking of vtol uav using a mobile robot manipulator. In: IEEE. *2017 IEEE/SICE International Symposium on System Integration (SII)*. [S.l.], 2017. p. 157–163. Citado na página 22.

NISSLER, C. et al. Evaluation and improvement of global pose estimation with multiple apriltags for industrial manipulators. In: IEEE. *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*. [S.l.], 2016. p. 1–8. Citado na página 22.

PAUL, G. et al. Autonomous robot manipulator-based exploration and mapping system for bridge maintenance. *Robotics and Autonomous Systems*, Elsevier, v. 59, n. 7-8, p. 543–554, 2011. Citado na página 21.

PETREA, G. et al. Visual servoing systems based control of complex autonomous systems serving a p/rml. In: IEEE. *2018 22nd International Conference on System Theory, Control and Computing (ICSTCC)*. [S.l.], 2018. p. 323–328. Citado na página 22.

PMBOK, U. G. d. C. d. *Conhecimentos em Gerenciamentos de Projetos: Guia PMBOK®*. [S.l.]: Pennsylvania: Four Campus Boulevard, 2008. Citado na página 57.

Qualidade e Produtividade. *Gage R e R*. 2019. <<https://sites.google.com/site/qualidadeeprodutividade/six-sigma/measure/2-1-2-3-analise-de-sistemas-de-medicao-msa/2-1-2-3-2-gage-r-r---repetibilidade-e-reproducibilidade>>, note = Accessed: 2020-07-20. Citado na página 51.

QUANSER. *Joint Control Robot - 6 DOF D*. 2017. Disponível em: <<https://www.quanser.com/products/joint-control-robot-6-dof-d>>. Acesso em: 22 de abr. de 2020. Citado na página 21.

RAZALI, N. M.; WAH, Y. B. et al. Power comparisons of shapiro wilk, kolmogorov smirnov, lilliefors and anderson darling tests. *Journal of statistical modeling and analytics*, v. 2, n. 1, p. 21–33, 2011. Citado na página 45.

REGULAMENTADORA, N. Nr 12 segurança no trabalho em máquinas e equipamentos. *Referências Elaboração Portaria*, v. 3, 1978. Citado na página 24.

ROBAI. *Cyton Alpha 7D 1G*. 2008. Disponível em: <<http://vigin.missouri.edu/~gdesouza/Research/MobileRobotics/Software/EnergidArm/cytonusermanual.pdf>>. Acesso em: 22 de abr. de 2020. Citado na página 21.

ROBOT, D. *Hawk Wireless Networked Autonomous Humanoid Mobile Robot with Dual Arms*. 2009. Disponível em: <http://www.drrobot.com/products_hawk.asp>. Acesso em: 22 de abr. de 2020. Citado na página 21.

ROBOTICS, U. *Braço de robô colaborativo UR5e - Um braço robótico flexível e leve*. 2008. Disponível em: <<https://www.universal-robots.com/products/ur5-robot/>>. Acesso em: 22 de abr. de 2020. Citado na página 21.

RUXTON, G. D. The unequal variance t-test is an underused alternative to student's t-test and the mann-whitney u test. *Behavioral Ecology*, Oxford University Press, v. 17, n. 4, p. 688–690, 2006. Citado na página 53.

SANDERSON, E.; WINDMEIJER, F. A weak instrument f-test in linear iv models with multiple endogenous variables. *Journal of Econometrics*, Elsevier, v. 190, n. 2, p. 212–221, 2016. Citado na página 53.

SHARP, A. et al. Semiautonomous dual-arm mobile manipulator system with intuitive supervisory user interfaces. In: IEEE. *2017 IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO)*. [S.l.], 2017. p. 1–6. Citado na página 21.

SHI, Y. et al. Study on intelligent visual servoing of space robot for cooperative target capturing. In: IEEE. *2012 IEEE International Conference on Information and Automation*. [S.l.], 2012. p. 733–738. Citado na página 22.

SPECTRUM, I. *Baxter*. 2018. Disponível em: <<https://robots.ieee.org/robots/baxter/>>. Citado na página 22.

SPECTRUM, I. *WAM*. 2018. Disponível em: <<https://robots.ieee.org/robots/wam/>>. Citado na página 22.

STRAUHS, F. d. R. et al. *Gestão do conhecimento nas organizações*. [S.l.]: Aymará Educação, 2012. Citado na página 57.

TIYU, F. et al. Positioning and pressing elevator button by binocular vision and robot manipulator. In: IEEE. *2018 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*. [S.l.], 2018. p. 120–133. Citado 2 vezes nas páginas 21 e 23.

USBDC. *Explosives Incident Report (EIR) 2016*. 2016. Disponivel em: <<https://www.atf.gov/file/136971/download>>. Acesso em: 27 de abr. de 2020. Citado na página 16.

VERONESE, G. S. Métodos para captura de lições aprendidas: em direção a melhoria contínua na gestão de projetos. *Revista de Gestão e Projetos*, v. 5, n. 1, p. 71–83, 2014. Citado na página 57.

ZHANG, Y.; LI, Y.; XIAO, X. A novel kinematics analysis for a 5-dof manipulator based on kuka youbot. In: IEEE. *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. [S.l.], 2015. p. 1477–1482. Citado na página 21.

ZHANG, Z. et al. Research on autonomous grasping control of underwater manipulator based on visual servo. In: IEEE. *2019 Chinese Automation Congress (CAC)*. [S.l.], 2019. p. 2904–2910. Citado na página 22.

APÊNDICE A

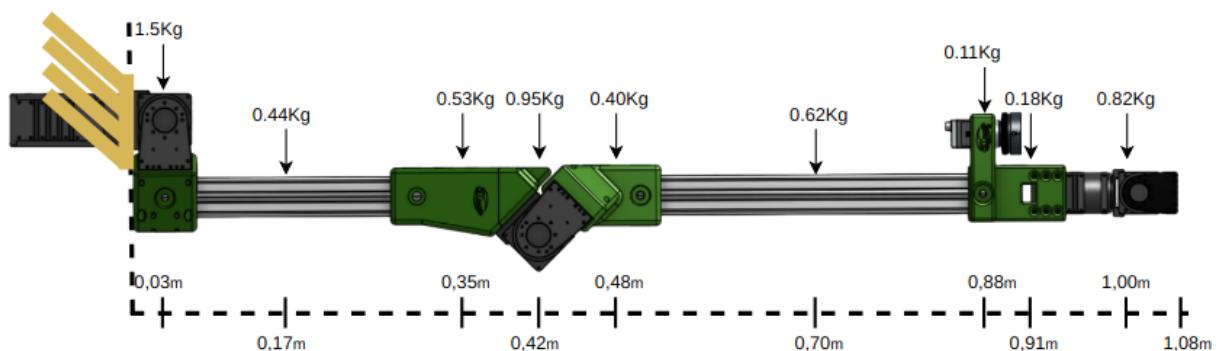
Projeto mecânico

Neste apêndice são demonstrados os resultados obtidos durante o projeto mecânico do manipulador desenvolvido. São demonstrados os resultados de análise estática do conceito proposto e dispostos os diversos desenhos mecânicos das peças construídas durante a execução deste projeto.

A.1 Análise estática

Os cálculos realizados assumem a situação estática, o qual o primeiro motor, M1, está totalmente fixo, como demonstrada na Figura 34, gerando uma condição de engaste no manipulador, para aplicar os esforços sofridos pelo manipulador, o qual cada parte da estrutura está com o seu respectivo peso aplicado no centro de massa.

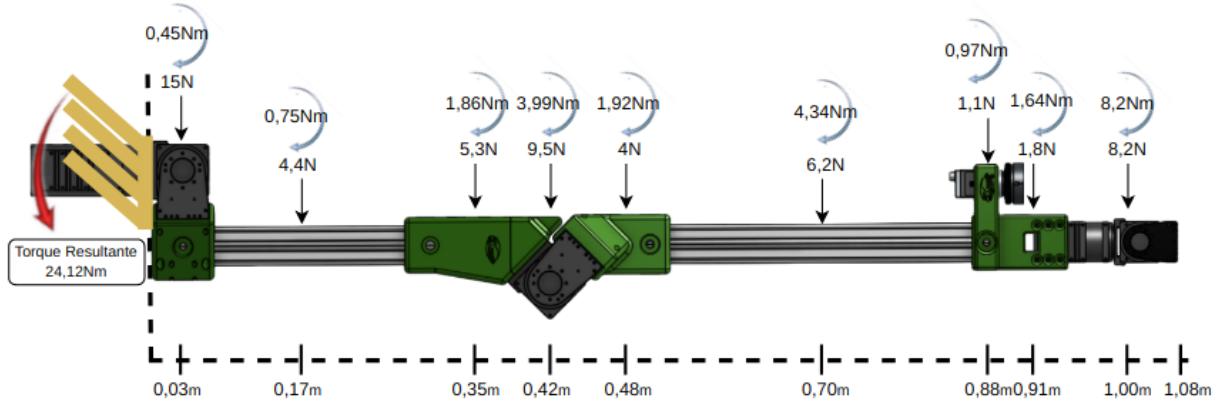
Figura 34: Condição Estática.



Fonte: Autores.

A atribuição das condição estática do manipulador geram os esforços demonstrado da figura 35.

Figura 35: Carga Estática.



Fonte: Autores.

Os diagramas de força cortante(V) e momento fletor(M), demonstrado nas figuras 36 e 37 e os cálculos feitos demonstrando nas equações A.2 e A.3, analisa a aplicação dos esforços que variam ao longo da extensão do manipulador, distância do engaste até o *end effector*, para que se conheça os valores de V e M em toda seção transversal.

$$\Sigma F = 0$$

$$V = 15 + 4,4 + 5,3 + 9,5 + 4 + 6,2 + 1,1 + 1,8 + 8,2 \quad (\text{A.1})$$

$$V = 54,5$$

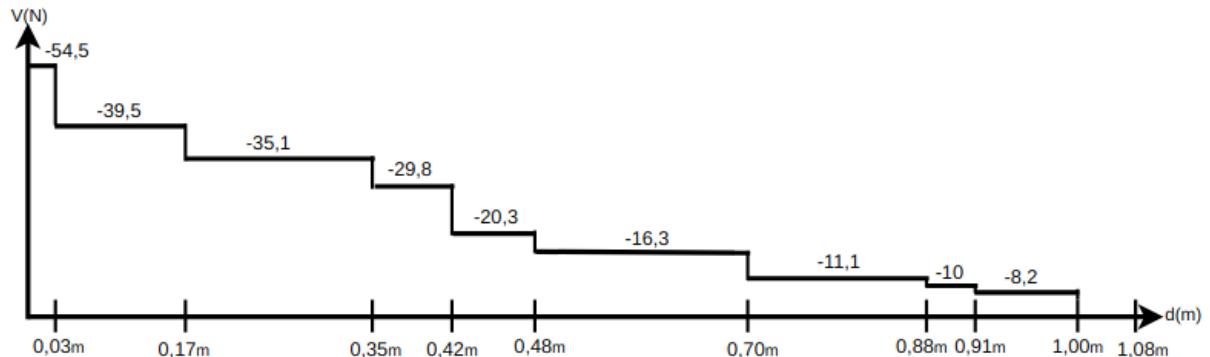
$$\Sigma M = 0$$

$$M = (15 \cdot 0,03) + (4,4 \cdot 0,17) + (5,3 \cdot 0,35) + (9,5 \cdot 0,42) + (9,5 \cdot 0,42) + (4 \cdot 0,48) + (6,2 \cdot 0,70) + (6,2 \cdot 0,88) + (1,8 \cdot 0,91) + (8,2 \cdot 1) \quad (\text{A.2})$$

$$M = 0,45 + 0,75 + 1,86 + 3,99 + 1,92 + 4,34 + 0,97 + 1,64 + 8,2$$

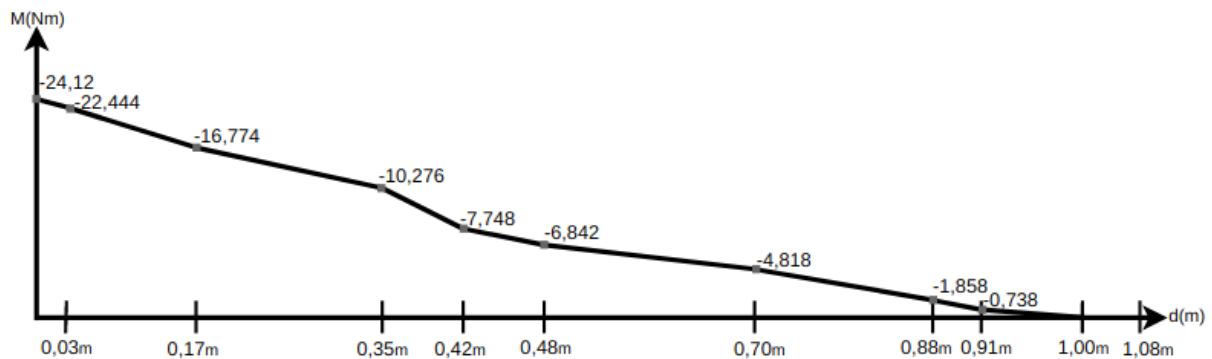
$$M = 24,12 \text{ Nm}$$

Figura 36: Forças cortantes.



Fonte: Autores.

Figura 37: Momentos fletores.



Fonte: Autores.

Calculando a carga útil (do inglês *payload*) do manipulador usando o cálculo de análise estática e usando o gráfico de performance do DYNAMIXEL PH54-200-S500-R demonstrado *datasheets* no Anexo A.4. *Payload* foi calculado com a diferença entre torque máximo do motor M1 com melhor eficiência subtraído pelo torque máximo em que o motor está aplicado em situação estática demonstrado na equação A.3.

Torque

$$Pl = 44,12 - 24,12$$

$$Pl = 20 \text{ Nm} \quad (\text{A.3})$$

Força calculada pela divisão do torque da equação A.3 pelo comprimento máximo do manipulador robótico na equação A.4.

Força

$$Pl = 20/1,08$$

$$Pl = 18,5N \quad (A.4)$$

Peso suportado na extremidade do manipulado adquirido pela divisão da força suportada da equação A.4 dividido pela gravidade demonstrada na equação A.5.

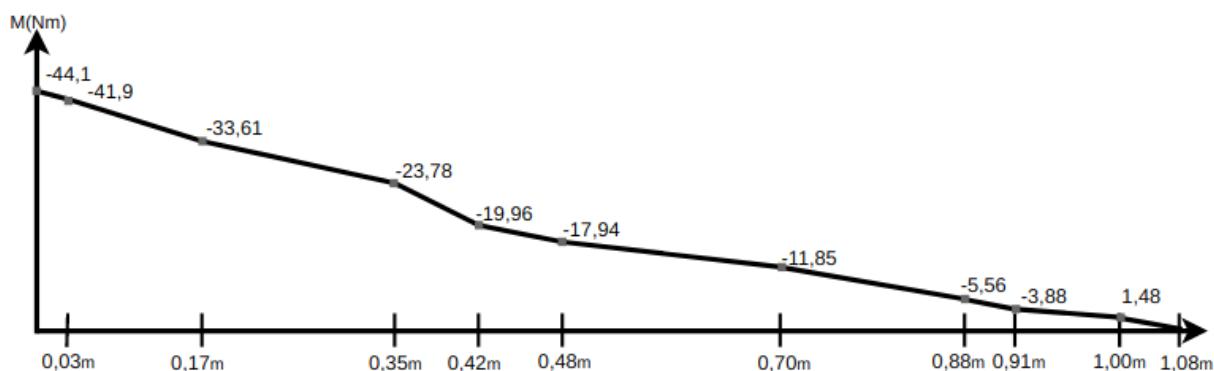
Peso

$$Pl = 18,5/10$$

$$Pl = 1,85kg \quad (A.5)$$

O diagrama de momento fletor foi analisado com o peso de 1,85kg referente ao *payload* para analisar o torque em toda a estrutura demonstrada na figura 38.

Figura 38: Momentos fletores com a carga do *payload*.



Fonte: Autores.

A.2 Desenho técnico mecânico

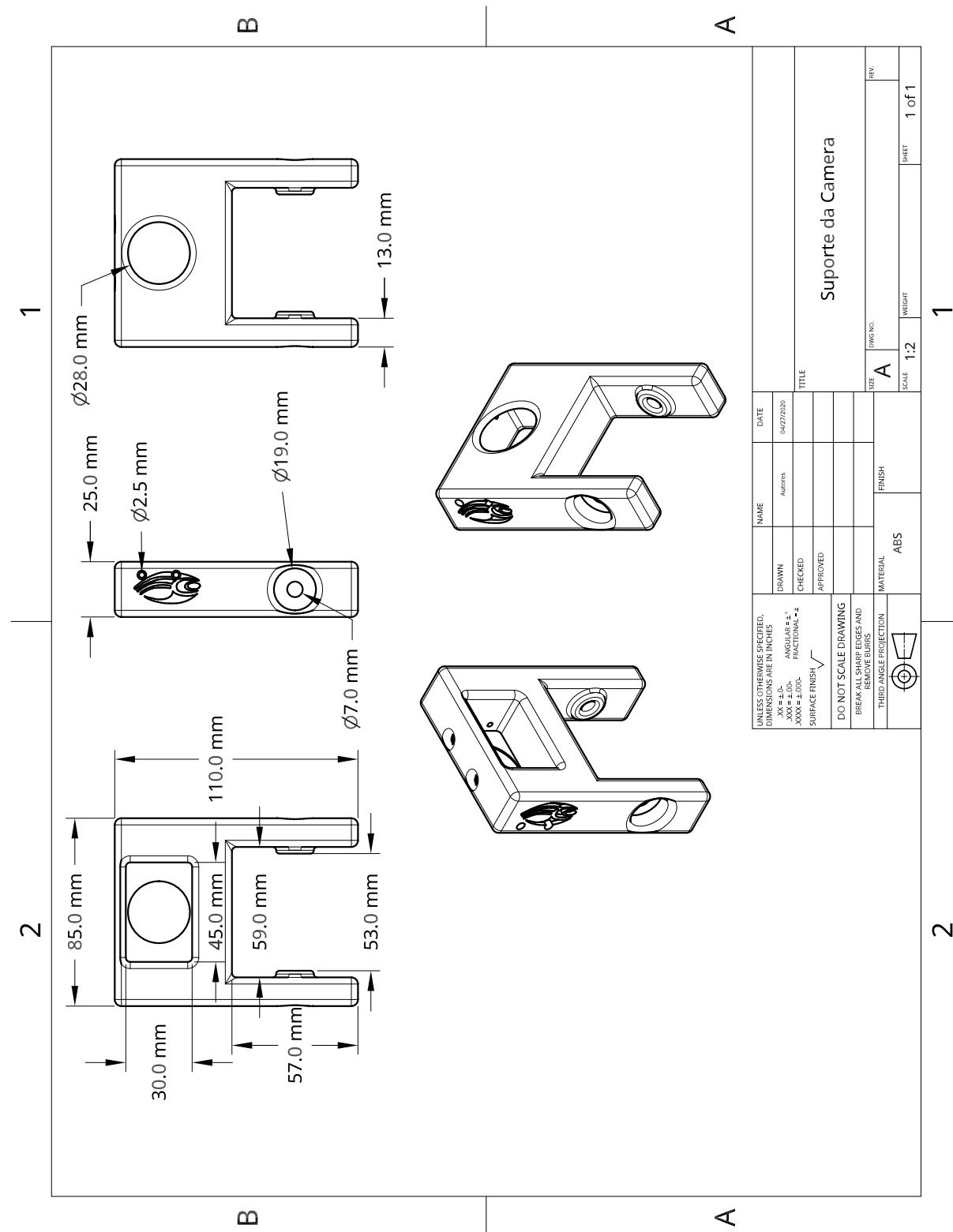
O desenho técnico mecânico é amplamente utilizado na elaboração de trabalhos e diagramas detalhados de máquinas e peças, assim como na construção de projetos mecânicos. É importante para documentação e elaboração de relatórios, além de outras formas de documentação textual.

Um desenho técnico deve conter vistas que demonstrem todos os detalhes necessários para a execução do projeto, as quais, incluem, vista lateral esquerda e direita, frontal e superior.

Nos tópicos seguintes demonstraremos os desenhos técnicos de partes importantes do manipulador robótico, projetados e construídos neste projeto.

A.2.1 Suporte da câmera

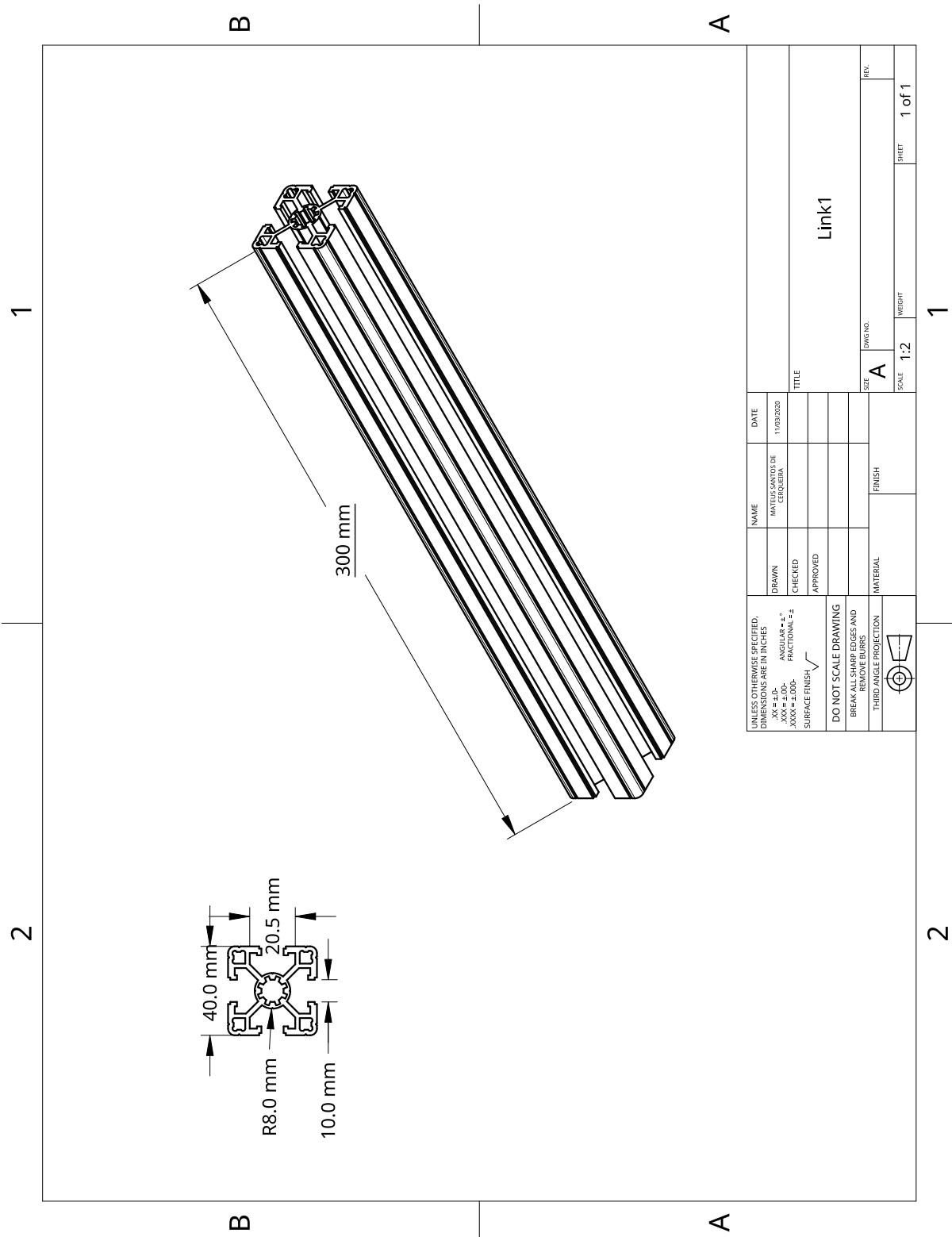
Figura 39: Desenho técnico do suporte da camera.



Fonte: Autores.

A.2.2 Perfil de alumínio 1

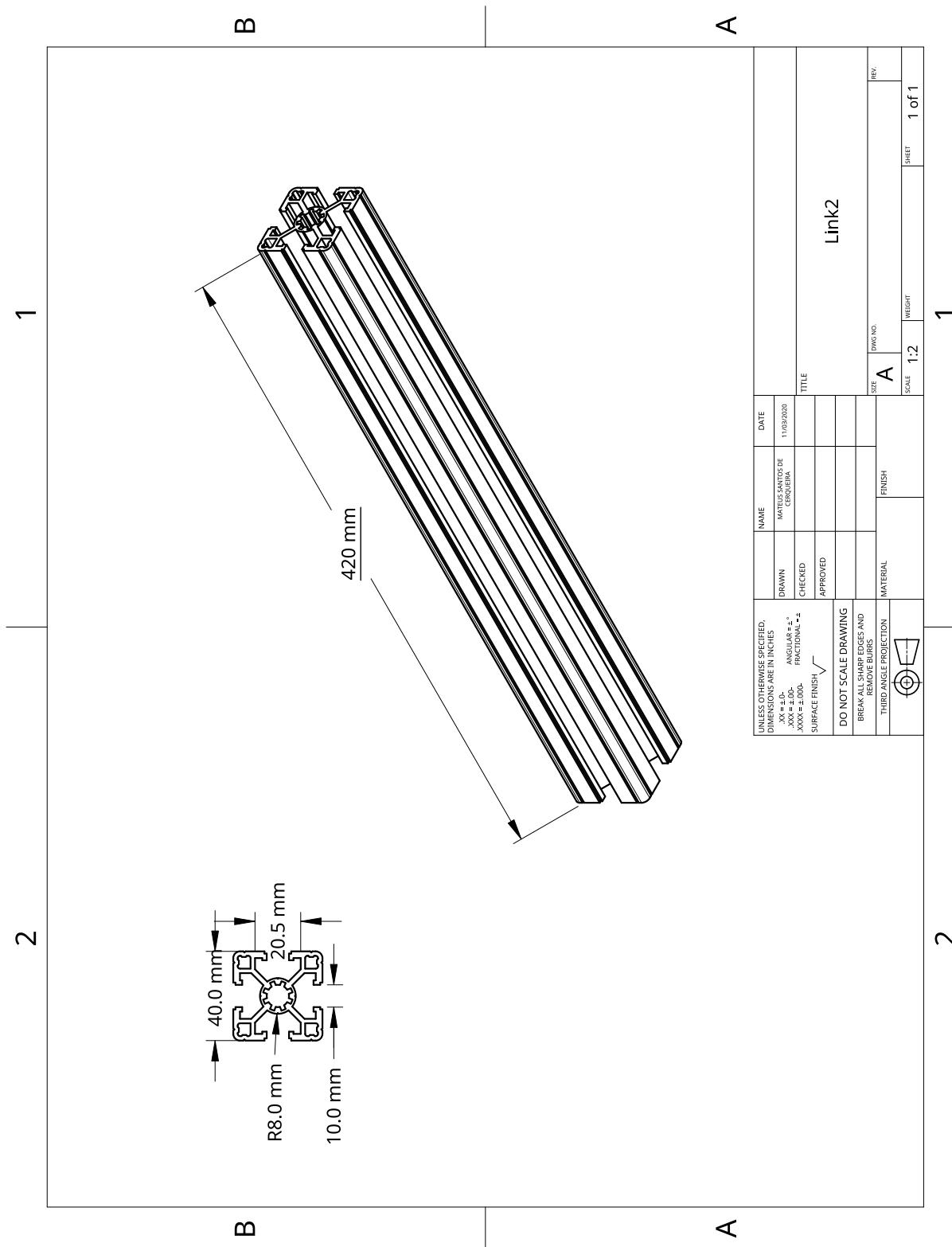
Figura 40: Desenho técnico perfil de alumínio.



Fonte: Autores.

A.2.3 Perfil de alumínio 2

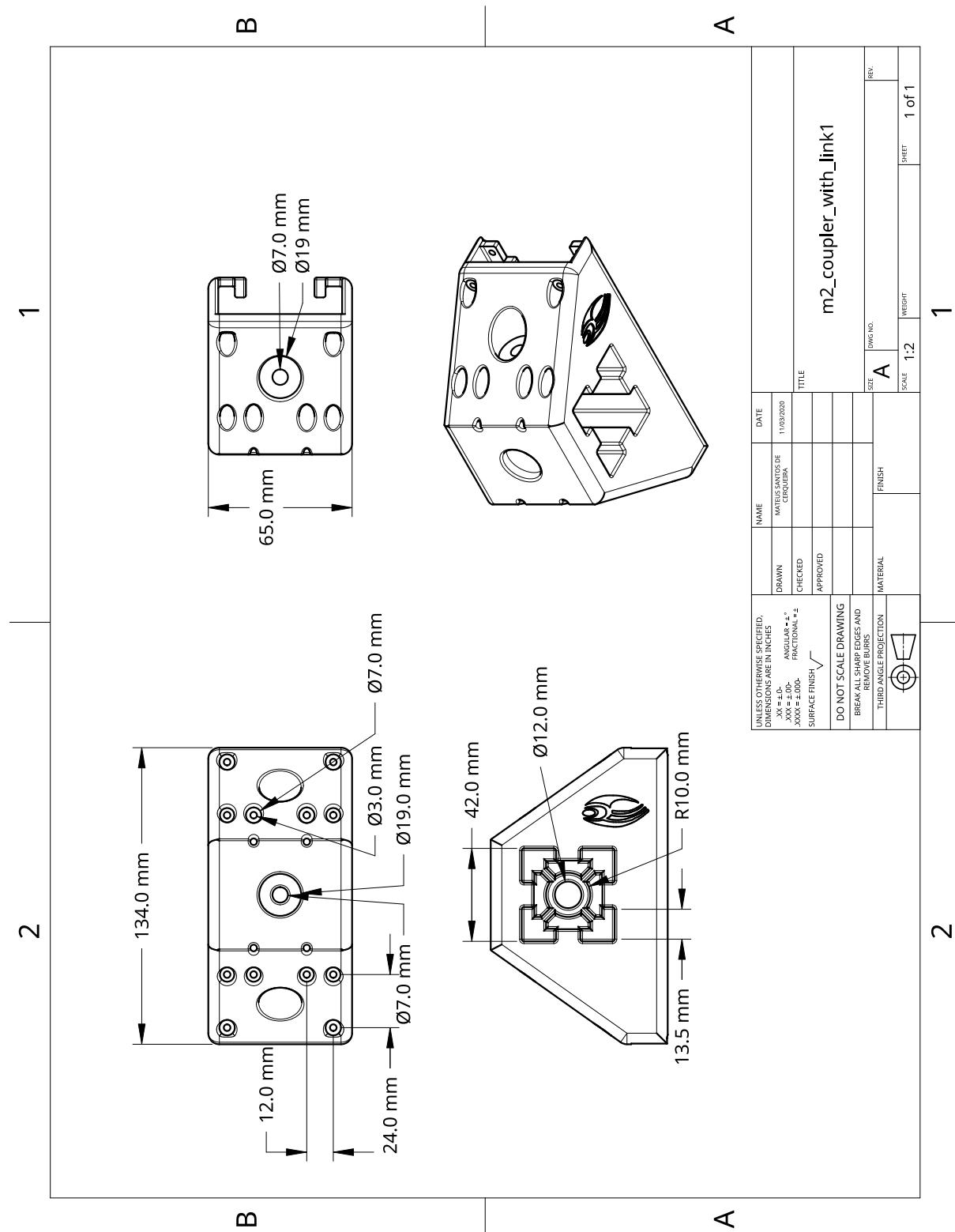
Figura 41: Desenho técnico perfil de alumínio.



Fonte: Autores.

A.2.4 Conector do motor 2 com perfil de alumínio 1

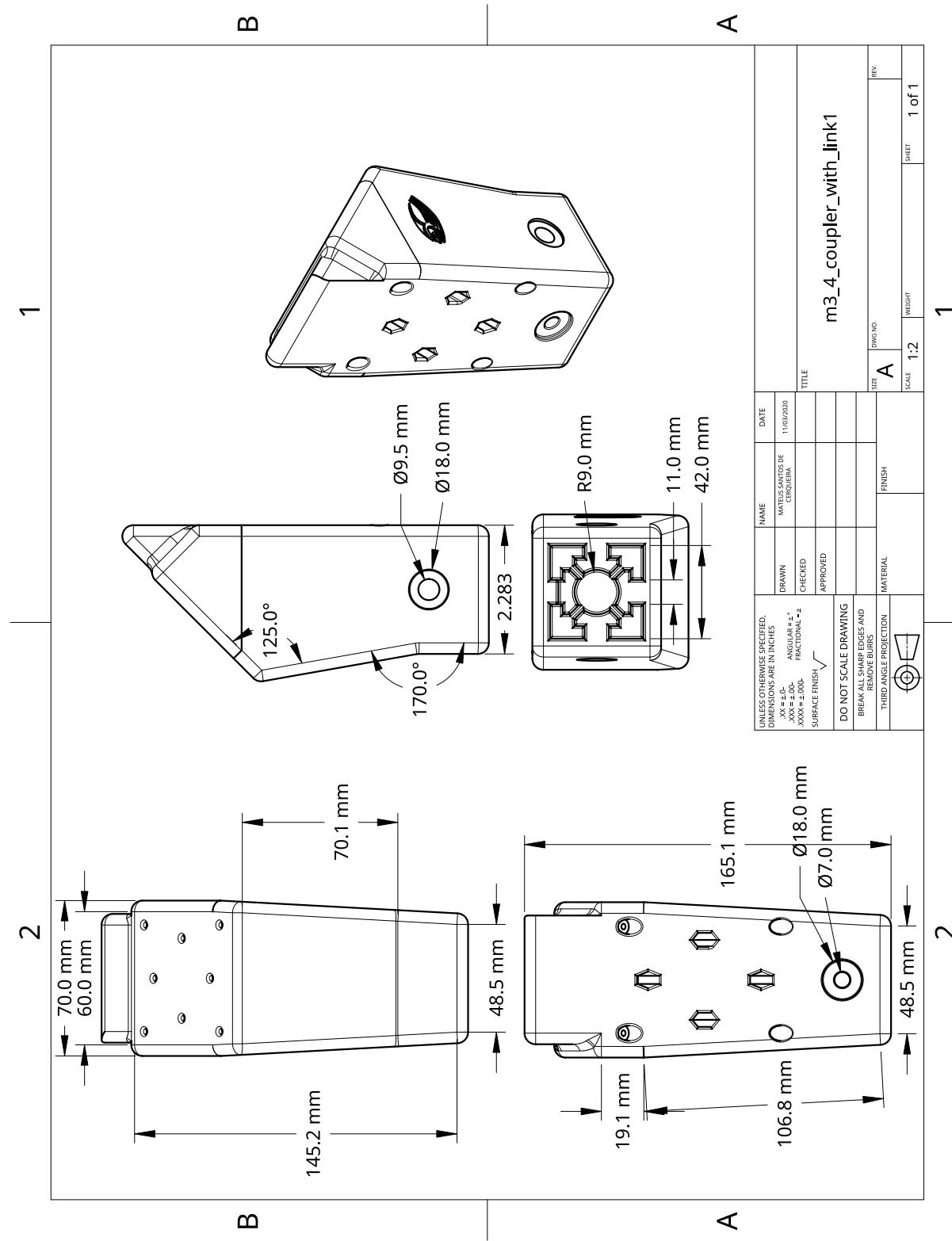
Figura 42: Desenho técnico conector do motor 2 com o perfil de alumínio 1.



Fonte: Autores.

A.2.5 Conector do motor 3 com perfil de alumínio 1

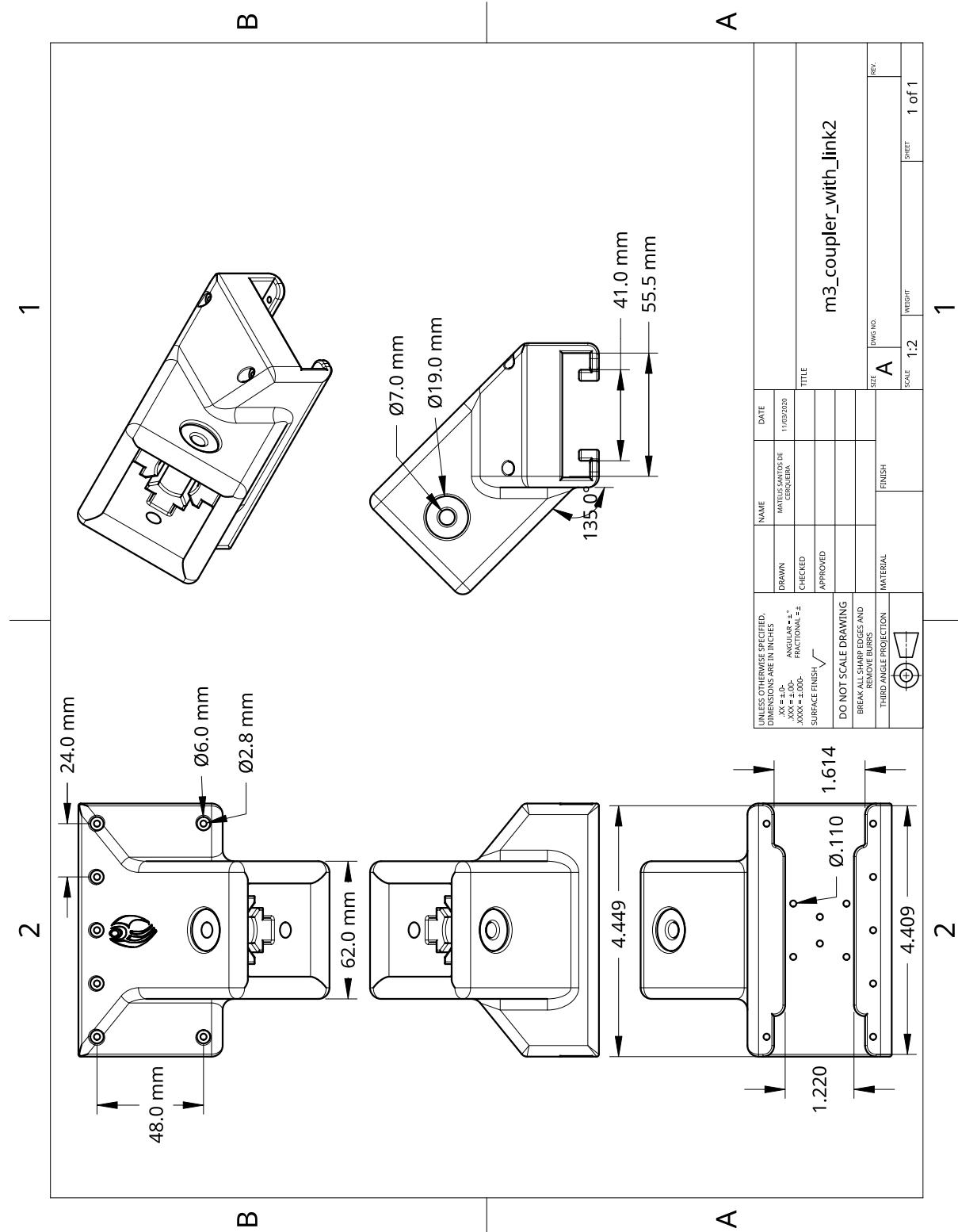
Figura 43: Desenho técnico conector do motor 3 com o perfil de alumínio 1.



Fonte: Autores.

A.2.6 Conector dos motor 3 com perfil de alumínio 2

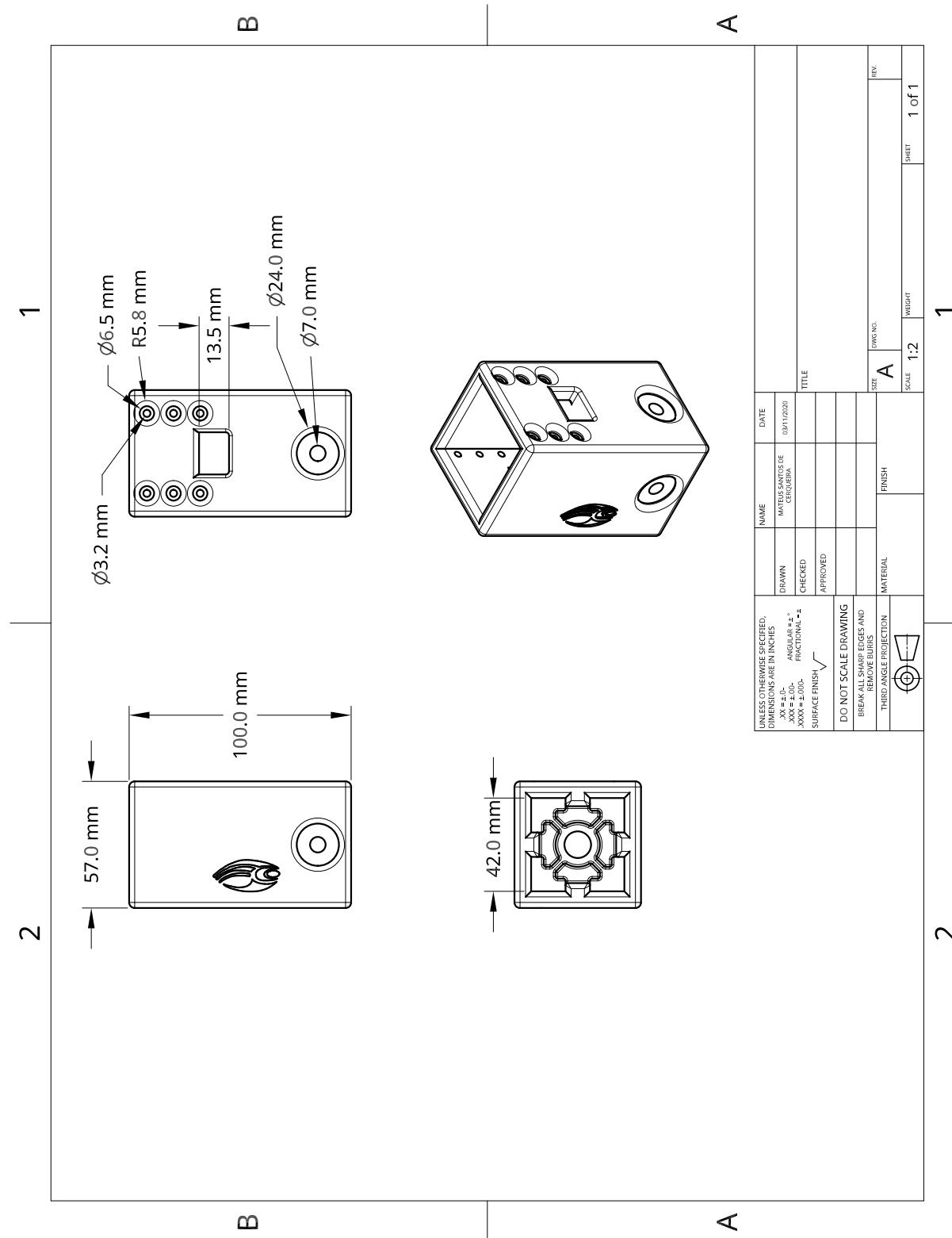
Figura 44: Desenho técnico conector do motor 3 com o perfil de alumínio 2.



Fonte: Autores.

A.2.7 Conector do motor 4 com perfil de alumínio 2

Figura 45: Desenho técnico conector do motor 4 com o perfil de alumínio 2.



Fonte: Autores.

APÊNDICE B

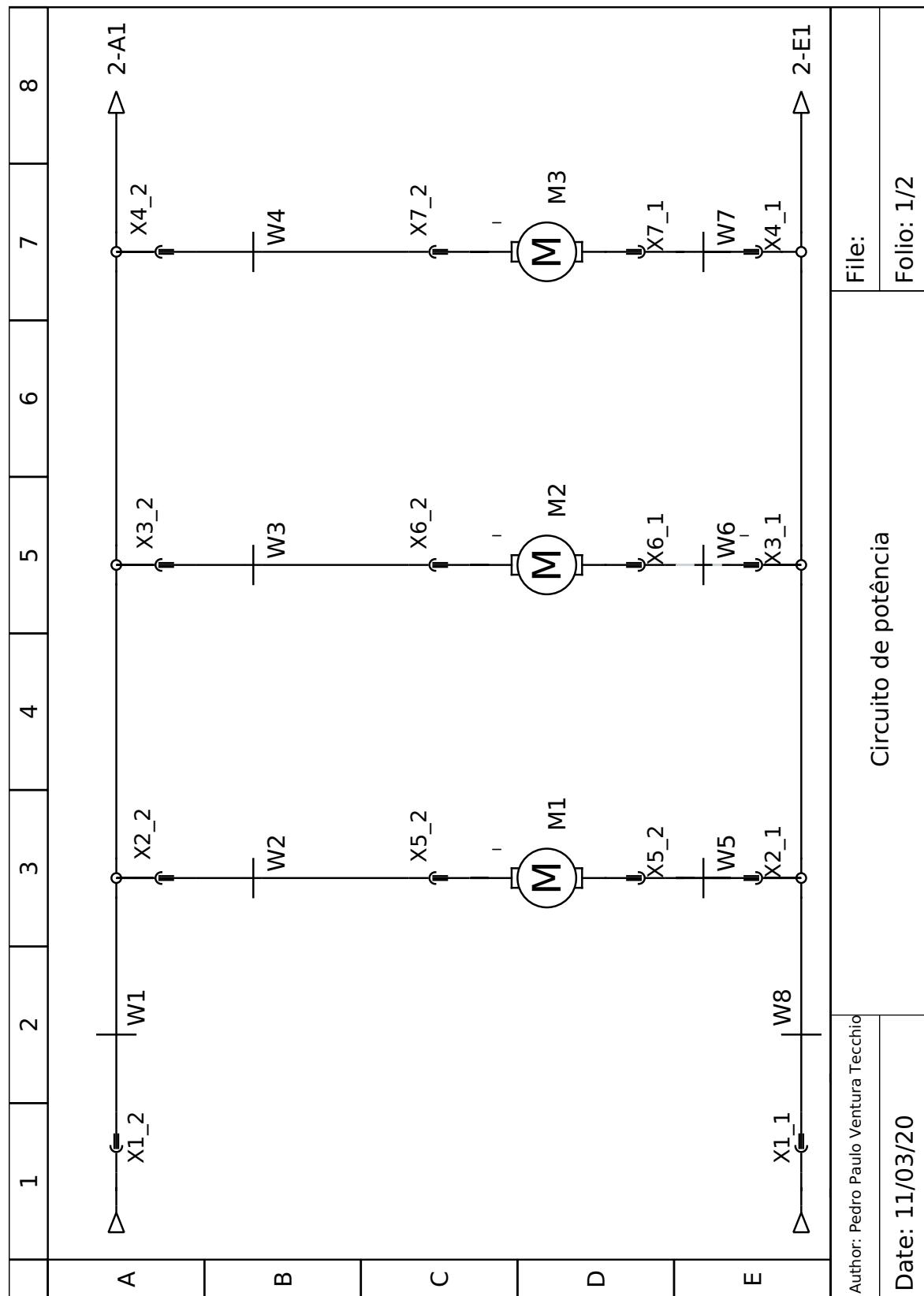
Projeto elétrico

Neste apêndice são apresentados os esquemas do sistema elétrico de potência do manipulador proposto e o cálculo da potência demandada para a utilização do manipulador projetado e apresentado neste relatório.

B.1 Esquema elétrico de potência

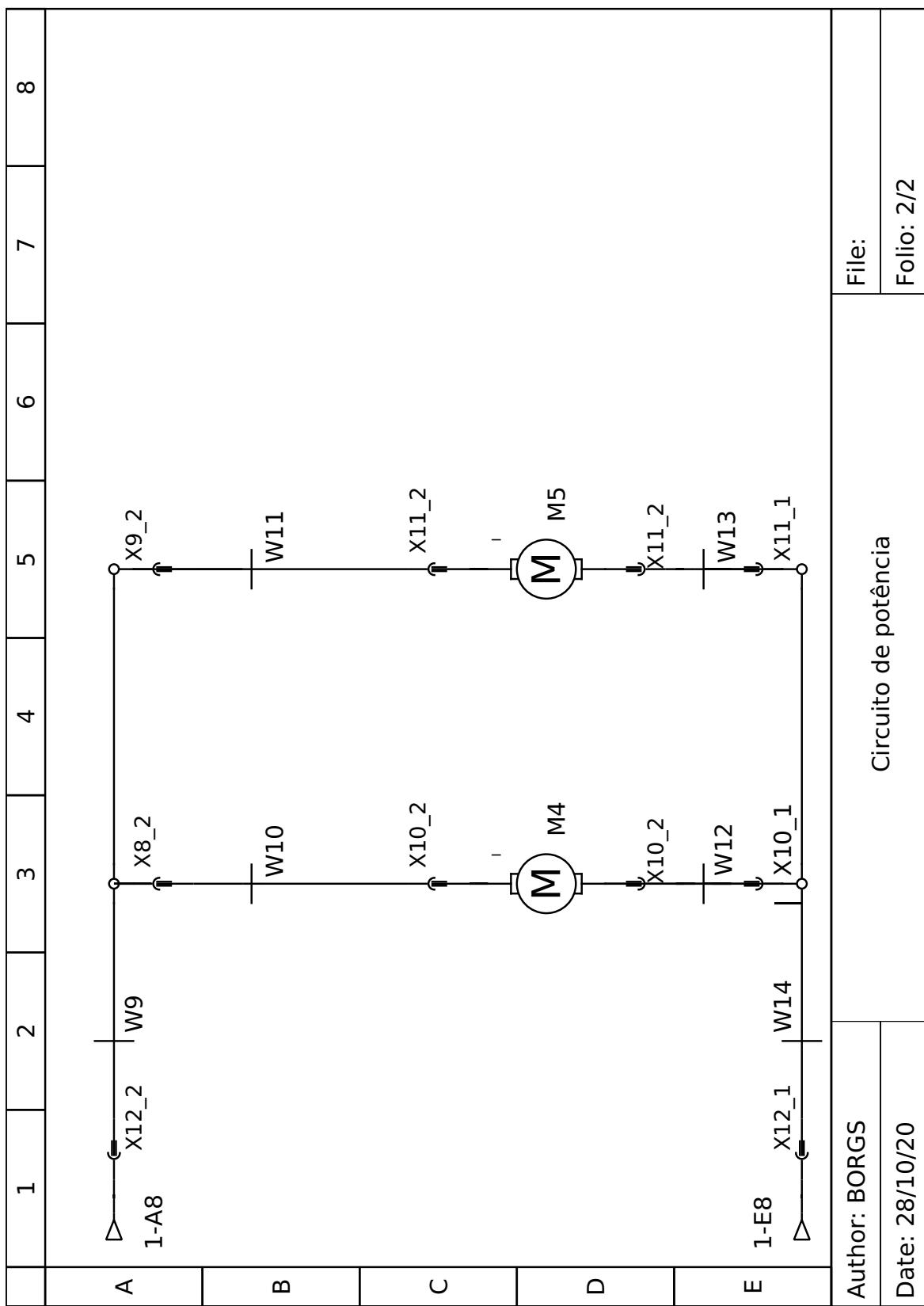
O esquema elétrico de potência projetado é exibido nas Figuras [46](#) e [47](#) respectivamente. Os motores são M1 (PH54-200-S500-R), M2 (PH54-200-S500-R), M3 (PH54-100-S500-R), M4(PH42-020-S300-R) e M5 (PH42-020-S300-R) conectados ao barramento de 24 V. Os motores do tipo “PH54-200” e “PH54-100” deverão ser ligados com sua própria conexão em um barramento, sendo os restantes energizados através de cabos adquiridos diretamente da ROBOTIS. As especificações dos motores ROBOTIS estão disponíveis no Anexo [A.4](#).

Figura 46: Esquema elétrico de potência para motores 1 a 3



Fonte: Autores.

Figura 47: Esquema elétrico de potência para motores 4 a 5



Fonte: Autores.

B.2 Potência demandada

A potência necessária para os atuadores do manipulador projetado pode ser estimada através do método de análise nodal e das Leis de Kirchhoff, em especial a primeira lei que especifica que a soma das correntes em um nó de um circuito elétrico é igual a zero. Aplicando este conceito para o circuito ilustrado nas Figuras 46 e 47, temos que a corrente demandada da fonte externa, i_F , é a soma das corrente que passa pelos motores, $i_{M_j}, j \in \{1, 2, 3, 4, 5\}$:

$$i_F = \sum_{j=1}^5 i_{M_j} \quad (\text{B.1})$$

Por conseguinte, a potência demandada da fonte externa, P_F , é igual a

$$P_F = 24 * i_F = 24 * (i_{M_1} + i_{M_2} + i_{M_3} + i_{M_4} + i_{M_5}). \quad (\text{B.2})$$

Portanto, a potência necessária para o acionamento do manipulador, P_F , pode ser calculada sabendo-se apenas os valores de corrente demandados por cada motor. Tais correntes dependem da quantidade de torque necessário para a movimentação do manipulador e estão sujeitas à variação da sua carga mecânica. No Anexo A.4 podem ser encontrados os *datasheets* contendo as curvas de corrente por torque de cada um dos motores utilizados, enquanto que o Apêndice A apresenta os cálculos de torque necessários para a movimentação do manipulador. Logo, uma estimativa das correntes necessárias para o funcionamento do manipulador pode ser estabelecida. Devido à imprecisão da tomada de valores pelos gráficos de corrente versus torque presentes nos *datasheets* mencionados, optou-se por considerar valores de corrente arredondados para cima, sempre que necessário.

No caso em vazio, sem carga aplicada sobre o manipulador, o maior consumo de energia ocorre quando o manipulador apresenta uma configuração totalmente estendida, Figura 37. Nesta configuração, os motores 1, 2, 3, 4 e 5 usam aproximadamente 5,0, 4,5, 2,4, 0,6 e 0,6 A respectivamente. Logo, a potência demandada é de aproximadamente 314,4 W.

Já no caso, onde todos os motores estão operando com a carga máxima especificada para o *payload*, Figura 38, a potência total demandada pelo manipulador é de 537,6 W. Considerando-se que os motores 1, 2, 3, 4 e 5 usam 8,2, 7,8, 4,8, 0,8 e 0,8 A respectivamente.

APÊNDICE C

Aplicação da teoria de manipuladores robóticos em experimentos simulados

Através do uso da teoria de Denavit-Hartenberg ([CRAIG, 2005](#)) e tendo-se como base a estrutura mecânica do manipulador projetado, pode-se construir uma sequência de eixos Cartesianos para cada junta de operação do manipulador tal como demonstrado na Figura 3. A partir destes eixos, construiu-se a matriz DH - Denavit-Hartenberg exposta na Tabela 19.

Tabela 19: Matriz Denavit-Hartenberg para o manipulador proposto.

	$\theta[\text{rad}]$	$\alpha[\text{rad}]$	$r[\text{mm}]$	$d[\text{mm}]$
1	θ_1	$\frac{\pi}{2}$	0	96
2	$\theta_2 + \frac{\pi}{2} + \text{atan2}(384.834, 119.792)$	0	403.048	0
3	$\theta_3 + \frac{\pi}{2} - \text{atan2}(384.834, 119.792)$	$\frac{\pi}{2}$	40.213	0
4	θ_4	$-\frac{\pi}{2}$	0	619.32
5	$\theta_5 - \frac{\pi}{2}$	0	39	0

Fonte: Autores.

Com o intuito de investigar e entender os algoritmos básicos envolvidos na utilização de manipuladores robóticos, foi desenvolvido um *script* em linguagem Python3. Este script é constituído de uma classe denominada *Manipulator*, cujos métodos podem resolver o problema de cinemática direta e inversa (simplificada, apenas origem dos eixos de coordenadas). Nas subseções a seguir, serão discutidos em maiores detalhes os métodos implementados nessa classe.

C.1 Inicialização da classe

A classe *Manipulator* é inicializada através da passagem de vetores contendo os valores dos ângulos θ e α e deslocamentos d e r dispostos na matriz Denavit-Hartenberg, juntamente com variáveis simbólicas e seus valores numéricos relacionando os deslocamentos nos eixos longitudinal e transversal aos frames dos motores.

```
import numpy as np
import sympy as sp
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt

class Manipulator:
```

```

def __init__(self, _t, _a, _d, _r, _x, _y, _xn, _yn, _limits):
    self.x_ = sp.Array(_x)
    self.y_ = sp.Array(_y)
    self.xn_ = _xn
    self.yn_ = _yn
    self.n_joints_ = min(len(_t), len(_a), len(_d), len(_r))
    self.symbolic_DH_matrix_ = self.setSymbolicDHMatrix(_t, _a, _d, _r)
    self.homogeneous_transforms_ = None
    self.frames_origin_ = None
    self.numeric_frames_origin_ = None
    self.jacobian_ = None
    self.numeric_jacobian_ = None
    self.forwardTransformFunction = None
    self.jacobianFunction = None
    self.limits_ = _limits

```

Durante a inicialização, a matriz DH do manipulador é construída através do método disposto a seguir:

```

def setSymbolicDHMatrix(self, _t, _a, _d, _r):
    t = sp.Array(sp.symbols('t:{}'.format(self.n_joints_)))
    _t = sp.Array(_t)
    _d = sp.Array(_d)
    _r = sp.Array(_r)
    t = t + _t
    symbolic_DH_matrix = sp.Matrix([t, _a, _d, _r]).transpose()
    return symbolic_DH_matrix

```

C.2 Obtenção das matrizes de transformação homogênea

Utilizando a matriz DH e os conceitos de transformadas homogêneas, pode-se obter as matrizes de transformação homogêneas entre cada um dos sistemas de coordenadas adjacentes. Através dos parâmetros da matriz DH pode-se usar o seguinte método para se obter a transformada homogênea simbólica:

```

def computeSingleHomogeneousTransform(self, _t, _a, _d, _r):
    ct = sp.cos(_t)
    st = sp.sin(_t)
    ca = sp.cos(_a)
    sa = sp.sin(_a)
    ht = sp.Matrix(
        [[ct, -st * ca, st * sa, ct * _r],
         [st, ct * ca, -ct * sa, st * _r],
         [0, sa, ca, _d],

```

```

        [ 0,           0,           0,           1 ]
)
return sp.simplify(ht)

```

Contudo, para se obter a matriz de transformação entre sistemas não adjacentes, há necessidade de multiplicar as respectivas matrizes na sequência apropriada. A matriz resultante de transformação do **EE** até o sistema de referência adotado na base do manipulador pode se tornar razoavelmente complexa. O seguinte método é capaz de encontrar todas as matrizes de transformação homogênea entre o sistema de origem e o eixo de coordenadas de cada junta do manipulador.

```

def computeCompleteSetHomogeneousTransform(self):
    ht = sp.eye(4)
    self.homogeneous_transforms_ = []
    for i in range(self.n_joints_):
        t = self.symbolic_DH_matrix_[i,0]
        a = self.symbolic_DH_matrix_[i,1]
        d = self.symbolic_DH_matrix_[i,2]
        r = self.symbolic_DH_matrix_[i,3]
        ht = sp.simplify(ht * self.computeSingleHomogeneousTransform(t, ...
            a, d, r))
        self.homogeneous_transforms_.append(ht)
    # print('ComputeCompleteSetHomogeneousTransform')
    # print(ht)

```

C.3 Cálculo da cinemática direta

Com o intuito de se obter a função de cinemática direta do manipulador em estudo limitada apenas à origem do eixo cartesiano do **EE**, foram criados os seguintes métodos, que devem ser invocados de maneira sequencial: *computeFramesOrigin()*, *computeNumericFramesOrigin()* e *computeForwardTransformFunction()*.

O primeiro e segundo métodos são responsáveis por encontrar a origem dos sistemas de coordenadas de cada junta do manipulador de forma simbólica e numérica respectivamente.

```

def computeFramesOrigin(self):
    origin = sp.Matrix([0, 0, 0, 1])
    frames_origin = []
    for ht in self.homogeneous_transforms_:
        frames_origin.append(ht * origin)
    self.frames_origin_ = frames_origin
    # print('computeFramesOrigin')
    # print(self.frames_origin_)

```

```

def computeNumericFramesOrigin(self):
    self.numeric_frames_origin_ = []
    subs_vars = []
    for i in range(self.n_joints_):
        subs_vars.append((self.x_[i],self.xn_[i]))
        subs_vars.append((self.y_[i],self.yn_[i]))
    for eqs in self.frames_origin_:
        numeric_eqs = []
        for i in range(3):
            numeric_eqs.append(eqs[i].subs(subs_vars))
        self.numeric_frames_origin_.append(numeric_eqs)
    # print('ComputeNumericFrameOrigin')
    # print(self.numeric_frames_origin_)

```

Por sua vez, o terceiro método gera uma função a ser utilizada em conjunto com a biblioteca *numpy* de forma a acelerar o cálculo numérico dos valores de posição do EE a partir dos ângulos de cada junta.

```

def computeForwardTransformFunction(self):
    t = sp.Array(sp.symbols('t:{}'.format(self.n_joints_)))
    self.forwardTransformFunction = sp.lambdify(t, ...
                                                self.numeric_frames_origin_, 'numpy')

```

C.4 Estimação do espaço de trabalho do manipulador

Um algoritmo simplificado utilizando optimizações da biblioteca *numpy* foi implementado para obter a região de trabalho do manipulador num espaço tri-dimensional. Esse método recebe um valor ou um conjunto de valores inteiros que representam a quantidade de pontos a serem utilizados na divisão dos limites máximos e mínimos das juntas do manipulador, anteriormente indicados na inicialização da classe.

```

def computeWorkspace(self, _steps):
    Δ_theta = []
    if isinstance(_steps, int):
        steps = np.ones((self.n_joints_,1))
        steps *= _steps
        print(steps)
    elif len(_steps) == len(self.limits_):
        steps = _steps
    else:
        print('Number of _steps is not compatible')
    for i in range(self.n_joints_):
        lim = self.limits_[i]

```

```

    Δ_theta.append(np.linspace(lim[0], lim[1], int(steps[i])))

mesh = np.meshgrid(Δ_theta[0], Δ_theta[1], Δ_theta[2], Δ_theta[3], Δ...
    _theta[4])

output = self.forwardTransformFunction(mesh[0], mesh[1], mesh[2], ...
    mesh[3], mesh[4])

x = output[4][0].flatten()
y = output[4][1].flatten()
z = output[4][2].flatten()

print('min(x) {} max(x) {}'.format(min(x), max(x)))
print('min(y) {} max(y) {}'.format(min(y), max(y)))
print('min(z) {} max(z) {}'.format(min(z), max(z)))
self.drawWorkspace(x, y, z)

```

O resultado deste método é indireto, uma vez que outro método é iniciado para o desenho da área de trabalho num sistema de eixos tri-dimensional, através do método *drawWorkspace()*, disposto a seguir.

```

def drawWorkspace(self, _x, _y, _z):
    fig = plt.figure()
    ax = fig.add_subplot(111, projection = '3d', aspect = 'equal')
    x = _y
    y = _z
    z = _x

    ax.set_xlabel('x')
    ax.set_ylabel('y')
    ax.set_zlabel('z')
    ax.set_xlim([-1500,1500])
    ax.set_ylim([-1500,1500])
    ax.set_zlim([-1500,1500])
    # Major ticks every 20, minor ticks every 5
    major_ticks = np.arange(-1500, 1500, 500)
    minor_ticks = np.arange(-1500, 1500, 100)

    ax.set_xticks(major_ticks)
    ax.set_xticks(minor_ticks, minor=True)
    ax.set_yticks(major_ticks)
    ax.set_yticks(minor_ticks, minor=True)
    ax.set_zticks(major_ticks)
    ax.set_zticks(minor_ticks, minor=True)

    # And a corresponding grid

```

```

ax.grid(which='both')
ax.plot(x, y, z, 'o')
ax.plot([0.0], [0.0], [0.0], 'ro')
# ax.plot([-1500.0, 1500.0], [0.0], [123.0, 123.0], 'k-')
plt.show()

```

C.5 Obtenção da matriz Jacobiana do manipulador

O algoritmo anterior funciona de acordo com o princípio conhecido como Cinemática Direta ou do inglês *Forward Kinematics*. Neste caso, um conjunto de valores são especificados para as juntas do manipulador, de forma que o movimento é completado antes do conhecimento da posição final do **EE**. No entanto, a movimentação efetiva do manipulador geralmente ocorre através da especificação de *poses* específicas para o **EE** no sistema de coordenadas de referência na base do manipulador, sem que haja conhecimento dos valores adotados para cada junta.

Logo, para que haja um controle efetivo do manipulador em conjunto com o planejamento de sua trajetória, faz-se necessário encontrar os valores necessários de atuação sobre cada junta de forma que o **EE** seja posicionado da acordo com o desejado. A teoria que descreve esta funcionalidade é conhecida como Cinemática Inversa, ou do inglês *Inverse Kinematics*. Com o intuito de desenvolver uma implementação desse algoritmo, é necessário o cálculo da matriz Jacobiana da função de Cinemática Direta, o qual é realizado através do *script* a seguir.

```

def computeJacobian(self):
    t = sp.Array(sp.symbols('t:{}'.format(self.n_joints_)))
    self.jacobian_ = sp.zeros(3, self.n_joints_)
    for i in range(3):
        for j in range(self.n_joints_):
            self.jacobian_[i, j] = sp.diff(self.frames_origin_[-1][i], t[j])
    self.jacobian_.simplify()
    # print('ComputeJacobian')
    # print(self.jacobian_)

```

O método anterior obtém a matriz Jacobiana da Cinemática Direta de forma simbólica, no entanto de forma a acelerar o processo de cálculo de valores numéricos através de uma função *lambda* associada à biblioteca *numpy*, foram implementados dois métodos, o primeiro *computeNumericJacobian()* converte todas as variáveis simbólicas conhecidas para seus valores numéricos, enquanto o segundo método *computeJacobianFunction()* cria uma função *lambda* a partir do resultado do primeiro método.

```

def computeNumericJacobian(self):
    self.numeric_jacobian_ = []
    subs_vars = []
    for i in range(self.n_joints_):
        subs_vars.append((self.x_[i], self.xn_[i]))
        subs_vars.append((self.y_[i], self.yn_[i]))

    self.numeric_jacobian_ = self.jacobian_.subs(subs_vars)
    # print('ComputeNumericJacobian')
    # print(self.numeric_jacobian_)

```

```

def computeJacobianFunction(self):
    t = sp.Array(sp.symbols('t:{}'.format(self.n_joints_)))
    self.jacobianFunction = sp.lambdify(t, self.numeric_jacobian_, 'numpy')

```

C.6 Um algoritmo para a Cinemática Inversa

Utilizando os conhecimentos obtidos nos Apêndices C.2 e C.5, pode-se implementar um método para o cálculo da cinemática inversa tal como disposto a seguir. Nesse algoritmo, a *pose* do EE é obtida utilizando o método de Newton-Raphson modificado de acordo com o descrito em (MELLODGE, 2016).

```

def inverseKinematics(self, _init_angles, _goal_pose, _n_iter, ...
                      _tolerance):
    pose = []
    angles = np.array(_init_angles)

    for i in range(int(_n_iter)):
        temp = self.forwardTransformFunction(angles[0], angles[1], angles[2],
                                             angles[3], angles[4])
        pose = np.array(temp[4])
        jacobian = self.jacobianFunction(angles[0], angles[1], angles[2],
                                         angles[3], angles[4])
        e = _goal_pose - pose
        if np.linalg.norm(e) < _tolerance:
            return angles, pose, e, i
        u, s, vh = np.linalg.svd(jacobian, full_matrices=False)
        s = np.diag(s)
        s_inv = np.linalg.pinv(s)
        matrix = np.dot(vh.transpose(), np.dot(s_inv, u.transpose()))
        delta_theta = np.dot(matrix, e)
        angles = angles + delta_theta

```

C.7 Exemplo de uso da classe *Manipulator*

A classe *Manipulator* pode ser corretamente utilizada tal como o descrito na função *main()* disposta a seguir.

```
def main():
    # Manipulador mechanical v3
    xn = np.array([96.0, 119.792, 550.359, 68.961, 39.0])
    yn = np.array([ 0.0, 384.834, 40.213, 0.0, 0.0])
    x = sp.symbols('x:5')
    y = sp.symbols('y:5')
    h = sp.sqrt((x[1] ** 2) + (y[1] ** 2))
    gamma = sp.atan2(y[1], x[1])
    theta = [0.0, sp.pi/2 + gamma, sp.pi/2 - gamma, 0.0, -sp.pi/2]
    alpha = [sp.pi/2, 0.0, sp.pi/2, -sp.pi/2, 0.0]
    d = [x[0], 0.0, 0.0, x[2] + x[3], 0.0]
    r = [0.0, h, y[2], 0.0, x[4]]

    # # Manipulador mechanical v1
    # xn = np.array([95.97, 435.75, 409.995, 96.499, 36.00])
    # yn = np.array([ 0.0, 0.0, 0.0, 0.0, 0.0])
    # x = sp.symbols('x:5')
    # y = sp.symbols('y:5')
    # # h = sp.sqrt((x[1] ** 2) + (y[1] ** 2))
    # # gamma = sp.atan2(y[1], x[1])
    # theta = [0.0, sp.pi/2,-sp.pi/2, 0.0, sp.pi/2]
    # alpha = [sp.pi/2, 0.0, -sp.pi/2, sp.pi/2, 0.0]
    # d = [x[0], 0.0, 0.0, x[2] + x[3], 0.0]
    # r = [0.0, x[1], 0.0, 0.0, x[4]]

    pi = np.pi
    pi_2 = pi / 2

    limits = np.array([
        [-pi, pi],
        [-pi_2, pi_2],
        [-pi_2, pi_2],
        [-pi, pi],
        [-pi_2, pi_2]
    ])

    manipulator = Manipulator(theta, alpha, d, r, x, y, xn, yn, limits)
    manipulator.computeCompleteSetHomogeneousTransform()
    manipulator.computeFramesOrigin()
    manipulator.computeNumericFramesOrigin()
    manipulator.computeForwardTransformFunction()
```

```

manipulator.computeJacobian()
manipulator.computeNumericJacobian()
manipulator.computeJacobianFunction()
print(manipulator.forwardTransformFunction(0,0,0,0,0))
print(manipulator.jacobianFunction(0,0,0,0,0))

manipulator.computeWorkspace([5,10,10,5,3])

goal_angles = [0.5, -0.5, 0.0, 0.5, 0.0]
pose = manipulator.forwardTransformFunction(
    goal_angles[0],goal_angles[1], goal_angles[2],
    goal_angles[3],goal_angles[4])
goal_pose = pose[4]
init_angles = [0,0,0,0,0]

angle, pose, error, n_iter = manipulator.inverseKinematics(
    init_angles, goal_pose, 1e5, 1e-6)
print('ik goal_pose {} goal_angle {} error {} n_iter {}'.format(pose,
    angle, error, n_iter))

if __name__ == "__main__":
    main()

```


APÊNDICE D

Dados coletados durante experimentos

Durante os experimentos de repetibilidade realizados para a tarefa de pressionamento da boteira, foram coletados dados relacionados aos tempos de execução e de precisão do posicionamento do [EE](#) em relação à posição do botão. Estes dados estão dispostos nas Tabelas [20](#) e [21](#) e são utilizados durante a realização dos testes de confiabilidade, Capítulo [4](#).

Tabela 20: Dados obtidos com a realização dos experimentos parte 1.

Nº	Pose	Vel	Time			Target						
			Scan	Button	End	p_x	p_y	p_z	x	y	z	
1	1	1	46,8556	58,7439	87,2438	-0,325561	0,249401	1,01072	-0,0206522	0,0226497	0,0226892	0,999273
2	1	1	46,9092	59,577	88,077	-0,325504	0,249401	1,01079	-0,0213556	0,0229463	0,0216646	0,999274
3	1	1	46,7219	72,9514	101,95	-0,325588	0,249401	1,01087	-0,0195401	0,0228043	0,0221787	0,999303
4	1	1	46,6925	75,821	104,421	-0,325625	0,249401	1,01083	-0,0187503	0,0224045	0,0198616	0,999376
5	1	1	46,9883	61,5615	91,5567	-0,325508	0,249401	1,01077	-0,0216689	0,0228548	0,0234031	0,99923
6	1	1	46,8381	57,4659	86,766	-0,325595	0,249401	1,01083	-0,018859	0,0225386	0,0210059	0,999347
7	1	1	46,8066	58,3332	87,0332	-0,325662	0,249401	1,01082	-0,0203163	0,0224146	0,0225252	0,999288
8	1	1	46,7619	66,69	95,3902	-0,325722	0,249401	1,01085	-0,0170121	0,0221361	0,0205472	0,999399
9	1	1	46,9941	56,5624	85,062	-0,325756	0,249401	1,01089	-0,0154136	0,0220484	0,0195682	0,999447
10	1	1	46,7828	59,4413	88,1413	-0,32563	0,249401	1,01085	-0,0174462	0,022266	0,0202515	0,999396
11	1	1	46,7258	56,2842	84,7844	-0,32565	0,249401	1,01081	-0,0187716	0,0222654	0,020496	0,999366
12	1	1	46,7525	67,9413	96,5408	-0,325606	0,249401	1,01083	-0,0177162	0,0225966	0,0203686	0,99938
13	1	1	46,799	68,0972	97,7973	-0,325596	0,249401	1,01079	-0,0206555	0,0224878	0,0210247	0,999313
14	1	1	46,8506	60,7092	89,4088	-0,325791	0,249401	1,01092	-0,013201	0,0218874	0,0181126	0,999509
15	1	1	46,7947	60,7829	89,3827	-0,325585	0,249401	1,01091	-0,0177849	0,022538	0,0212748	0,999361
16	1	2	29,4767	42,5951	58,495	-0,325754	0,249401	1,01098	-0,0220106	0,0218734	0,0205023	0,999308
17	1	2	29,4709	39,1993	55,299	-0,326007	0,249401	1,0107	-0,0174041	0,0223767	0,0210457	0,999377
18	1	2	29,492	40,2204	56,62	-0,326022	0,249401	1,01077	-0,0185955	0,0224197	0,021293	0,999349
19	1	2	29,42	56,4173	72,3177	-0,326041	0,249401	1,01082	-0,0172329	0,0224542	0,0200156	0,999399
20	1	2	31,1155	72,0136	87,9138	-0,326013	0,249401	1,01074	-0,0148427	0,0224012	0,0188505	0,999461
21	1	2	29,4783	38,2763	54,5761	-0,325992	0,249401	1,01085	-0,0148559	0,0219498	0,0205774	0,999437
22	1	2	29,3741	40,9319	57,232	-0,326069	0,249401	1,01082	-0,0138114	0,0219948	0,0179659	0,999501
23	1	2	29,5066	46,9048	62,6048	-0,325984	0,249401	1,01077	-0,0184191	0,0224356	0,0210267	0,999357
24	1	2	29,5631	45,0514	60,9514	-0,32566	0,249401	1,01099	-0,0147681	0,0223855	0,0179803	0,999479
25	1	2	29,5558	42,6843	58,7839	-0,326178	0,249401	1,01078	-0,0148877	0,022359	0,0170654	0,999493
26	1	2	29,6273	52,9955	70,9953	-0,325745	0,249401	1,01091	-0,0167894	0,0223545	0,020169	0,999406
27	1	2	29,5748	47,5729	63,6731	-0,325951	0,249401	1,01091	-0,0144268	0,0222243	0,0199542	0,99945
28	1	2	29,9229	45,1212	61,1211	-0,326183	0,249401	1,01083	-0,0126964	0,0219844	0,0178747	0,999518
29	1	2	29,5192	47,8172	63,3172	-0,324388	0,249401	1,00505	-0,0497225	0,0231272	0,0281077	0,9981
30	1	2	29,5233	39,4912	55,1914	-0,326105	0,249401	1,01092	-0,0176062	0,0227547	0,019843	0,999389
31	2	2	53,7322	83,986	99,986	0,339806	0,249401	0,979121	-0,0601294	0,00904195	-0,00080706	0,998149
32	2	2	52,7632	78,812	94,612	0,340299	0,249401	0,979298	-0,063012	0,0085775	2,22E-05	0,997976
33	2	2	52,7332	68,7699	82,8696	0,338873	0,249401	1,09049	0,00496171	0,00305798	-0,0127888	0,999901
34	2	2	71,7979	91,5866	107,386	0,340107	0,249401	0,979478	-0,0574433	0,00822387	-0,00848154	0,998279
35	2	2	54,175	74,3658	88,4662	0,338696	0,249401	1,09008	0,00519601	0,00294867	-0,0142098	0,999881
36	2	2	71,931	86,5589	102,559	0,340511	0,249401	0,979309	-0,0575402	0,00868692	-0,00248087	0,998302
37	2	2	77,4071	97,6656	113,566	0,340592	0,249401	0,979246	-0,05823	0,00938354	-0,00149669	0,998258
38	2	2	73,0868	83,3447	99,1446	0,340457	0,249401	0,979008	-0,0633082	0,00889062	0,00205317	0,997952
39	2	2	71,6256	97,2139	112,914	0,340425	0,249401	0,979095	-0,0597356	0,00862026	-0,000108307	0,998177
40	2	2	71,811	82,4692	98,169	0,340048	0,249401	0,97079	-0,0929786	0,0116922	-0,00496059	0,995587
41	2	2	71,759	81,0874	97,9871	0,340846	0,249401	0,979314	-0,0585229	0,00889822	-0,000978916	0,998246
42	2	2	72,0034	84,6415	100,242	0,340757	0,249401	0,979251	-0,0575109	0,00909783	-0,00256477	0,9983
43	2	2	71,7325	85,621	101,721	0,340806	0,249401	0,97885	-0,067112	0,00883382	-0,00120211	0,997706
44	2	2	73,079	85,177	100,977	0,340769	0,249401	0,97913	-0,0616579	0,00855015	-0,00252495	0,998058
45	2	2	72,0249	97,0832	113,083	0,340779	0,249401	0,979117	-0,0625642	0,00862044	-0,000803735	0,998003
46	2	1	109,914	128,873	157,273	0,340871	0,249401	0,978964	-0,0627238	0,00807842	-0,00180294	0,997991
47	2	1	110,149	127,217	155,617	0,34087	0,249401	0,979234	-0,0573721	0,00890731	-4,50E-05	0,998313
48	2	1	111,427	124,355	153,255	0,340914	0,249401	0,978946	-0,0643339	0,00904728	-0,00166166	0,997886
49	2	1	109,992	127,19	156,19	0,340962	0,249401	0,979115	-0,0594875	0,00891095	-0,00145788	0,998188
50	2	1	110,019	124,917	153,617	0,340877	0,249401	0,978957	-0,0642161	0,00858547	-0,0011151	0,997898
51	2	1	109,967	127,165	155,765	0,340913	0,249401	0,979005	-0,0623219	0,0087427	-0,000417701	0,998018
52	2	1	109,92	119,749	148,349	0,34091	0,249401	0,978919	-0,0641507	0,00886618	-0,00215085	0,997899
53	2	1	110,706	126,905	155,605	0,34095	0,249401	0,979045	-0,0617054	0,00905468	0,000116497	0,998053
54	2	1	109,919	127,447	155,747	0,340932	0,249401	0,979016	-0,0616343	0,00902773	0,000234014	0,998058
55	2	1	109,935	123,964	152,464	0,340865	0,249401	0,978887	-0,0637482	0,00882454	-0,00301194	0,997922
56	2	1	109,865	125,693	154,094	0,340915	0,249401	0,978834	-0,06557	0,00885732	-0,00176912	0,997807
57	2	1	109,899	121,558	149,658	0,340395	0,249401	0,970686	-0,0624759	0,0113105	0,00500949	0,99797
58	2	1	109,888	122,714	151,214	0,340949	0,249401	0,978961	-0,0627518	0,00897761	-0,003852	0,997981
59	2	1	109,922	130,95	159,45	0,340856	0,249401	0,979048	-0,0607407	0,0087597	-0,00233689	0,998112
60	2	1	110,432	131,087	159,487	0,339546	0,249401	0,978809	-0,0599549	0,00825934	-0,00107395	0,998166

Fonte: Autores

Tabela 21: Dados obtidos com a realização dos experimentos parte 2.

Pose	Vel	Reached_p_x	Reached_p_y	Reached_p_z	Reached_x	Reached_y	Reached_z	Reached_w
1	1	-0,325618	0,249335	1,01063	-0,00399829	-0,999092	-0,00138563	0,0424021
1	1	-0,325813	0,249184	1,01057	-0,00841398	0,0358289	-0,00382191	0,999315
1	1	-0,325745	0,249297	1,01077	-0,00279084	-0,997324	0,0286128	0,0672194
1	1	-0,325763	0,24933	1,01075	0,0137007	0,0517191	-0,00407589	0,998559
1	1	-0,325568	0,24937	1,01078	0,0142709	-0,0141018	-0,00371745	0,999792
1	1	-0,325765	0,249298	1,01066	-0,00321307	-0,998036	0,0379353	0,049741
1	1	-0,325919	0,249221	1,01064	0,00402281	-0,000526414	-0,00394919	0,999984
1	1	-0,325831	0,249307	1,0107	-0,0606106	-0,01448	-0,00697803	0,998032
1	1	-0,326601	0,248726	1,01015	0,0032181	0,999871	0,0141552	0,00690963
1	1	-0,325654	0,249385	1,01086	-0,0247274	0,0305934	-0,00408197	0,999218
1	1	-0,32582	0,249253	1,01061	-0,00428454	-0,997454	0,0656338	0,0275398
1	1	-0,325891	0,24924	1,01063	-0,00394671	-0,999446	0,00321287	0,0328765
1	1	-0,325621	0,2494	1,01083	-0,0434803	0,0723817	-0,00196101	0,996427
1	1	-0,326252	0,249068	1,0105	-0,0169504	0,0528796	-0,00318183	0,998452
1	1	-0,326354	0,248815	1,01027	-0,0019171	-0,996823	0,0436233	0,0666165
1	2	-0,325806	0,249357	1,01095	-0,00484861	-0,998335	0,0540504	0,0195483
1	2	-0,326126	0,249301	1,01065	0,0517238	-0,00988035	-0,00188624	0,998611
1	2	-0,326722	0,248736	1,01018	-0,0643058	0,00452636	-0,00528383	0,997906
1	2	-0,326091	0,24937	1,01083	-0,00686891	-0,00204409	-0,00430001	0,999965
1	2	-0,326115	0,249332	1,01061	-0,0187477	0,0488307	-0,00331809	0,998626
1	2	-0,326647	0,248757	1,01027	-0,00326479	-0,99838	0,00523034	0,0565634
1	2	-0,326527	0,24903	1,0104	-0,0361886	0,0586766	-0,00238321	0,997618
1	2	-0,326494	0,249033	1,01041	-0,0478179	0,0545733	-0,00231104	0,997361
1	2	-0,326082	0,249032	1,01062	0,0159269	0,0605617	-0,0040481	0,998029
1	2	-0,326238	0,249338	1,01071	-0,0437972	-0,00550323	-0,00558978	0,99901
1	2	-0,326225	0,249072	1,01056	-0,0156995	0,00850052	-0,00423298	0,999832
1	2	-0,326405	0,249052	1,01054	0,00369144	0,999965	0,00743665	0,000262038
1	2	-0,326347	0,249299	1,01072	-0,00336862	-0,99828	0,00953211	0,0577497
1	2	-0,324819	0,248921	1,00474	-0,131618	0,0856605	0,00539544	0,987578
1	2	-0,326591	0,24905	1,01053	-0,0299585	0,0620667	-0,00244573	0,997619
2	2	0,339742	0,249415	0,979232	-0,098751	0,0146804	-0,0103277	0,99495
2	2	0,340334	0,249297	0,979289	-0,0111095	-0,99519	0,0968761	0,00945548
2	2	0,33898	0,249358	1,09047	0,0298153	-0,998783	0,0392406	0,00183513
2	2	0,340188	0,249328	0,97937	-0,0584724	0,048022	-0,0110513	0,997072
2	2	0,338519	0,249222	1,09016	-0,0294457	0,99876	-0,0396709	0,00612399
2	2	0,340615	0,249299	0,979232	-0,0696975	-0,00870781	-0,012961	0,997446
2	2	0,340683	0,249318	0,97922	-0,0116467	-0,99728	0,0675556	0,0270867
2	2	0,340506	0,24933	0,979026	-0,0681038	0,0512884	-0,0104839	0,996304
2	2	0,340602	0,24918	0,978957	-0,0913176	0,0506899	-0,00877289	0,994492
2	2	0,339544	0,249405	0,971311	-0,0112317	-0,99129	0,128934	0,0243926
2	2	0,340934	0,249282	0,979303	-0,0129619	-0,998546	0,0494112	0,0172314
2	2	0,340773	0,249303	0,9793	-0,00537728	-0,0272314	-0,0139422	0,999517
2	2	0,340776	0,249382	0,978833	-0,0541855	0,068311	-0,0112887	0,996128
2	2	0,340828	0,249103	0,979104	-0,00997422	-0,995327	0,0886579	0,0369452
2	2	0,340761	0,249261	0,979087	-0,047942	0,0347911	-0,0125395	0,998165
2	1	0,341019	0,249312	0,9789	-0,00925464	-0,994189	0,0994381	0,0401905
2	1	0,340981	0,249353	0,979182	-0,0529813	-0,00732227	-0,0134425	0,998478
2	1	0,340864	0,24939	0,978958	0,013517	0,998585	-0,0512941	0,00370827
2	1	0,340855	0,24935	0,979272	0,0139512	0,99878	-0,0407121	0,0242062
2	1	0,340573	0,249227	0,979296	-0,0134258	-0,99878	0,032561	0,0346146
2	1	0,34089	0,249242	0,979029	-0,00638793	-0,991764	0,111003	0,0635708
2	1	0,340726	0,249253	0,979049	0,0126716	0,99282	-0,118216	0,0131295
2	1	0,340964	0,249345	0,979014	-0,0244475	0,037563	-0,0141614	0,998895
2	1	0,34082	0,249235	0,979156	0,0124706	0,993765	-0,110388	0,00946593
2	1	0,340857	0,249397	0,97893	-0,00595846	-0,99104	0,118349	0,0616315
2	1	0,340499	0,249333	0,979254	-0,0739419	0,0737246	-0,00882536	0,994495
2	1	0,340131	0,249413	0,970966	-0,0983465	0,0520785	-0,0112358	0,993725
2	1	0,340988	0,24937	0,97897	-0,100418	-0,018253	-0,0133547	0,994688
2	1	0,340963	0,249321	0,978939	-0,0723731	0,0650142	-0,00976294	0,995209
2	1	0,339664	0,249283	0,978722	-0,0775956	-0,0193467	-0,0131168	0,996711

Fonte: Autores

APÊNDICE E

Aplicação de testes estatísticos nos dados obtidos

Construção de uma rotina de testes estatísticos de forma a avaliar o desempenho do manipulador através de diversas métricas para verificar o desempenho, eficiência e repetibilidade.

```
path <-("~/SENAI/Estatistica/Desafio 2/")

# Transforma o diretorio raiz na pasta que contem os arquivos
setwd(path)
getwd()

# Le arquivo
library(readxl)
dataset <- read_excel("Dados - Testes Finais.xlsx",
sheet = "Formatado")

library(RSpinCalc)
# Converte para rotacao em euler - angulos target
euler_target <- apply(dataset[,c('Target_x', 'Target_y', 'Target_z', ...
    'Target_w')], 1, function(x) Q2EA(x, EulerOrder='xyz') )
euler_target <- as.data.frame(t(euler_target))
colnames(euler_target) <- c("euler_x_target", "euler_y_target", ...
    "euler_z_target")

# Converte para rotacao em euler - angulos desired
euler_desired <- apply(dataset[,c('Desired_x', 'Desired_y', ...
    'Desired_z', 'Desired_w')], 1, function(x) Q2EA(x, EulerOrder='xyz') )
euler_desired <- as.data.frame(t(euler_desired))
colnames(euler_desired) <- c("euler_x_desired", "euler_y_desired", ...
    "euler_z_desired")

# Da um merge nos dados e nos angulos de euler
dataset2 <-cbind(dataset, euler_target, euler_desired)

# RMS
library(seewave)
rms_p_x <- apply(dataset2[,c('Target_p_x', 'Desired_p_x')], 1, ...
    function(x) rms(x) )
rms_p_x <- as.data.frame(rms_p_x)

rms_p_y <- apply(dataset2[,c('Target_p_y', 'Desired_p_y')], 1, ...)
```

```

    function(x) rms(x) )
rms_p_y <- as.data.frame(rms_p_y)

rms_p_z <- apply(dataset2[,c('Target_p_z', 'Desired_p_z')], 1, ...
  function(x) rms(x) )
rms_p_z <- as.data.frame(rms_p_z)

rms_x <- apply(dataset2[,c('euler_x_target', 'euler_x_desired')], 1, ...
  function(x) rms(x) )
rms_x <- as.data.frame(rms_x)

rms_y <- apply(dataset2[,c('euler_y_target', 'euler_y_desired')], 1, ...
  function(x) rms(x) )
rms_y <- as.data.frame(rms_y)

rms_z <- apply(dataset2[,c('euler_z_target', 'euler_z_desired')], 1, ...
  function(x) rms(x) )
rms_z <- as.data.frame(rms_z)

dataset3 <- cbind(dataset2[, c(1, 2, 3, 4, 5)], rms_p_x, rms_p_y, ...
  rms_p_z, rms_x, rms_y, rms_z)

# Computa varias metricas de variaveis do dataset 3
library("SixSigma")

f <- function(x) {
  list(max(x), min(x), mean(x), sd(x), shapiro.test(x)$p.value, ...
    ks.test(x, "pnorm", mean(x),sd(x))$p.value)
}
fun <- sapply(dataset3, f)
#rownames(fun) <- c("max", "min", "mean", "sd", "shapiro", "ks")

# Aplica teste gage rr

a <- ss.rr(var = Time_scan, part = Pose, appr = Velocity, data = ...
  dataset, sub = "Testes", main = "Primeira ANOVA")

library(tikzDevice)
beamer.parms = list(paperwidth = 364.19536/72, # converts pt to in
  paperheight = 273.14662/72,
  textwidth = 307.28987/72,
  textheight = 269.14662/72)

tikz("velocidade4.tex", width = beamer.parms$textwidth, height = ...
  beamer.parms$textwidth)

# Histograma

```

```

library(ggplot2)
png(filename= paste(path, "/scan.png", sep = ""))
ggplot(data = dataset3) +
  geom_histogram(aes(x = Time_scan), bins = 30, colour = "black", fill ...
                 = "white") +
  facet_wrap(~ Pose + Velocity, labeller = labeller(Pose =
    c("1" = "Posicao 1", "2" = "Posicao 2"), Velocity = c("1" = ...
      "Velocidade 1", "2" = "Velocidade 2")))) +
  ggtitle("Histograma do tempo de scan") + # for the main title
  xlab("Tempo de scan") + # for the x axis label
  ylab("Frequencia")

dev.off()

png(filename= paste(path, "/button.png", sep = ""))
ggplot(data = dataset3) +
  geom_histogram(aes(x = Time_button), bins = 30, colour = "black", ...
                 fill = "white") +
  facet_wrap(~ Pose + Velocity, labeller = labeller(Pose = c("1" = ...
    "Posicao 1", "2" = "Posicao 2"), Velocity = c("1" = "Velocidade ...
      1", "2" = "Velocidade 2")))) +
  ggtitle("Histograma do tempo para apertar o botao") + # for the main title
  xlab("Tempo para apertar o botao") + # for the x axis label
  ylab("Frequencia")

dev.off()

png(filename= paste(path, "/end.png", sep = ""))
ggplot(data = dataset3) +
  geom_histogram(aes(x = Time_end), bins = 30, colour = "black", fill = ...
                 "white") +
  facet_wrap(~ Pose + Velocity, labeller = labeller(Pose = c("1" = ...
    "Posicao 1", "2" = "Posicao 2"), Velocity = c("1" = "Velocidade ...
      1", "2" = "Velocidade 2")))) +
  ggtitle("Histograma do tempo total da missao") + # for the main title
  xlab("Tempo da missao") + # for the x axis label
  ylab("Frequencia")

dev.off()

png(filename= paste(path, "/rmspx.png", sep = ""))
ggplot(data = dataset3) +
  geom_histogram(aes(x = rms_p_x), bins = 30, colour = "black", fill = ...

```

```

    "white") +
facet_wrap(~ Pose + Velocity, labeller = labeller(Pose = c("1" = ...
"Posicao 1", "2" = "Posicao 2"), Velocity = c("1" = "Velocidade ...
1", "2" = "Velocidade 2")))+
ggtitle("Histograma do RMS da posicao X")+ # for the main title
xlab("RMS na posicao X")+ # for the x axis label
ylab("Frequencia")

dev.off()

png(filename= paste(path, "/rmspy.png", sep = ""))
ggplot(data = dataset3) +
geom_histogram(aes(x = rms_p_y), bins = 30, colour = "black", fill = ...
"white") +
facet_wrap(~ Pose + Velocity, labeller = labeller(Pose = c("1" = ...
"Posicao 1", "2" = "Posicao 2"), Velocity = c("1" = "Velocidade ...
1", "2" = "Velocidade 2")))+
ggtitle("Histograma do RMS da posicao Y")+ # for the main title
xlab("RMS na posicao Y")+ # for the x axis label
ylab("Frequencia")

dev.off()

png(filename= paste(path, "/rmspz.png", sep = ""))
ggplot(data = dataset3) +
geom_histogram(aes(x = rms_p_z), bins = 30, colour = "black", fill = ...
"white") +
facet_wrap(~ Pose + Velocity, labeller = labeller(Pose = c("1" = ...
"Posicao 1", "2" = "Posicao 2"), Velocity = c("1" = "Velocidade ...
1", "2" = "Velocidade 2")))+
ggtitle("Histograma do RMS da posicao Z")+ # for the main title
xlab("RMS na posicao Z")+ # for the x axis label
ylab("Frequencia")

dev.off()

png(filename= paste(path, "/rmsx.png", sep = ""))
ggplot(data = dataset3) +
geom_histogram(aes(x = rms_x), bins = 30, colour = "black", fill = ...
"white") +
facet_wrap(~ Pose + Velocity, labeller = labeller(Pose = c("1" = ...
"Posicao 1", "2" = "Posicao 2"), Velocity = c("1" = "Velocidade ...
1", "2" = "Velocidade 2")))+
ggtitle("Histograma do RMS da orientacao X")+ # for the main title

```

```

xlab("RMS na orientacao X") + # for the x axis label
ylab("Frequencia")

dev.off()

png(filename= paste(path, "/rmsy.png", sep = ""))
ggplot(data = dataset3) +
  geom_histogram(aes(x = rms_y), bins = 30, colour = "black", fill = ...
                 "white") +
  facet_wrap(~ Pose + Velocity, labeller = labeller(Pose = c("1" = ...
               "Posicao 1", "2" = "Posicao 2"), Velocity = c("1" = "Velocidade ...
               1", "2" = "Velocidade 2")) +
  ggtitle("Histograma do RMS da orientacao Y") + # for the main title
  xlab("RMS na orientacao Y") + # for the x axis label
  ylab("Frequencia")

dev.off()

png(filename= paste(path, "/rmsz.png", sep = ""))
ggplot(data = dataset3) +
  geom_histogram(aes(x = rms_z), bins = 30, colour = "black", fill = ...
                 "white") +
  facet_wrap(~ Pose + Velocity, labeller = labeller(Pose = c("1" = ...
               "Posicao 1", "2" = "Posicao 2"), Velocity = c("1" = "Velocidade ...
               1", "2" = "Velocidade 2")) +
  ggtitle("Histograma do RMS da orientacao Z") + # for the main title
  xlab("RMS na orientacao Z") + # for the x axis label
  ylab("Frequencia")

dev.off()

# Aplica t test (verifica se medias de dois grupos sao iguais entre si)
t.test(dataset3$Time_scan[dataset3$Pose==1], ...
       dataset3$Time_scan[dataset3$Pose==2])
t.test(dataset3$Time_scan[dataset3$Velocity==1], ...
       dataset3$Time_scan[dataset3$Velocity==2])
t.test(dataset3$Time_button[dataset3$Pose==1], ...
       dataset3$Time_button[dataset3$Pose==2])
t.test(dataset3$Time_button[dataset3$Velocity==1], ...
       dataset3$Time_button[dataset3$Velocity==2])
t.test(dataset3$Time_end[dataset3$Pose==1], dataset3$end[dataset3$Pose==2])
t.test(dataset3$Time_end[dataset3$Velocity==1], ...
       dataset3$Time_end[dataset3$Velocity==2])
t.test(dataset3$rms_p_x[dataset3$Pose==1], ...

```

```

dataset3$rms_p_x[dataset3$Pose==2])
t.test(dataset3$rms_p_x[dataset3$Velocity==1], ...
       dataset3$rms_p_x[dataset3$Velocity==2])
t.test(dataset3$rms_p_y[dataset3$Pose==1], ...
       dataset3$rms_p_y[dataset3$Pose==2])
t.test(dataset3$rms_p_y[dataset3$Velocity==1], ...
       dataset3$rms_p_y[dataset3$Velocity==2])
t.test(dataset3$rms_p_z[dataset3$Pose==1], ...
       dataset3$rms_p_z[dataset3$Pose==2])
t.test(dataset3$rms_p_z[dataset3$Velocity==1], ...
       dataset3$rms_p_z[dataset3$Velocity==2])
t.test(dataset3$rms_x[dataset3$Pose==1], dataset3$rms_x[dataset3$Pose==2])
t.test(dataset3$rms_x[dataset3$Velocity==1], ...
       dataset3$rms_x[dataset3$Velocity==2])
t.test(dataset3$rms_y[dataset3$Pose==1], dataset3$rms_y[dataset3$Pose==2])
t.test(dataset3$rms_y[dataset3$Velocity==1], ...
       dataset3$rms_y[dataset3$Velocity==2])
t.test(dataset3$rms_z[dataset3$Pose==1], dataset3$rms_z[dataset3$Pose==2])
t.test(dataset3$rms_z[dataset3$Velocity==1], ...
       dataset3$rms_z[dataset3$Velocity==2])

# Aplica f test comparando proporcao de duas variancias
# Aplica t test (verifica se medias de dois grupos sao iguais entre si)
var.test(dataset3$Time_scan[dataset3$Pose==1], ...
          dataset3$Time_scan[dataset3$Pose==2])
var.test(dataset3$Time_scan[dataset3$Velocity==1], ...
          dataset3$Time_scan[dataset3$Velocity==2])
var.test(dataset3$Time_button[dataset3$Pose==1], ...
          dataset3$Time_button[dataset3$Pose==2])
var.test(dataset3$Time_button[dataset3$Velocity==1], ...
          dataset3$Time_button[dataset3$Velocity==2])
var.test(dataset3$Time_end[dataset3$Pose==1], ...
          dataset3$Time_end[dataset3$Pose==2])
var.test(dataset3$Time_end[dataset3$Velocity==1], ...
          dataset3$Time_end[dataset3$Velocity==2])
var.test(dataset3$rms_p_x[dataset3$Pose==1], ...
          dataset3$rms_p_x[dataset3$Pose==2])
var.test(dataset3$rms_p_x[dataset3$Velocity==1], ...
          dataset3$rms_p_x[dataset3$Velocity==2])
var.test(dataset3$rms_p_y[dataset3$Pose==1], ...
          dataset3$rms_p_y[dataset3$Pose==2])
var.test(dataset3$rms_p_y[dataset3$Velocity==1], ...
          dataset3$rms_p_y[dataset3$Velocity==2])
var.test(dataset3$rms_p_z[dataset3$Pose==1], ...
          dataset3$rms_p_z[dataset3$Pose==2])
var.test(dataset3$rms_p_z[dataset3$Velocity==1], ...
          dataset3$rms_p_z[dataset3$Velocity==2])

```

```

var.test(dataset3$rms_x[dataset3$Pose==1], ...
  dataset3$rms_x[dataset3$Pose==2])
var.test(dataset3$rms_x[dataset3$Velocity==1], ...
  dataset3$rms_x[dataset3$Velocity==2])
var.test(dataset3$rms_y[dataset3$Pose==1], ...
  dataset3$rms_y[dataset3$Pose==2])
var.test(dataset3$rms_y[dataset3$Velocity==1], ...
  dataset3$rms_y[dataset3$Velocity==2])
var.test(dataset3$rms_z[dataset3$Pose==1], ...
  dataset3$rms_z[dataset3$Pose==2])
var.test(dataset3$rms_z[dataset3$Velocity==1], ...
  dataset3$rms_z[dataset3$Velocity==2])

library(animation)
library(scatterplot3d)
saveHTML({
  for (i in 1:nrow(dataset)) {
    pt<-dataset[i,]
    scatterplot3d(pt[6],pt[7],pt[8], xlim=range(dataset$Target_p_x), ...
      ylim=range(dataset$Target_p_y), zlim=range(dataset$Target_p_z))
  }
})

##### Repetibilidade (ainda verificando) #####
library(rptR)
rep <- rpt(Time_scan ~ (1 | Pose) + (1 | Velocity), grname = c("Pose", ...
  "Velocity"), data = dataset, datatype = "Gaussian", nboot = 1000,
  npermut = 0)

plot(rep, grname = "Pose", type = "boot", cex.main = 0.8)
plot(rep, grname = "Velocity", type = "boot", cex.main = 0.8)

table(dataset$Pose)
table(dataset$Velocity)

summary(rep)

hist(dataset$Velocity)

```


ANEXO A

Especificação de componentes adquiridos

A.1 Computador Intel NUC

A ficha resumo contendo as principais especificações do computador Intel NUC6i5SYB utilizado está disposta na Tabela 22. Maiores informações deste componente estão dispostos no *databrief* anexado a seguir.

Tabela 22: Especificações do computador Intel NUC 6i5SYB.

Item	Especificação
Coleção	Intel NUC Kit 6º geração
Sistema operacional	Ubuntu 18.04 LTS
Modelo da placa	NUC6i5SYB
Formato da placa	UCFF (4 x 4 [polegadas])
Processador	Intel Core i5-6260U
Processador gráfico	Intel Iris Graphics 540
Memória de sistema	2x8GB DDR4-2133 1.2V SO-DIMM
Disco de armazenamento de dados	Samsung SSD 950 PRO 512GB
Periféricos	Intel Gigabit LAN
	4 USB 3.0
	2 USB 2.0
	Intel Dual Band Wireless-AC 8260
	Bluetooth 4.1
Tensão de operação	19 [V]
Temperatura de operação	0 ~40 [°C]
Temperatura de armazenagem	-20 ~60 [°C]
Dimensões (L x A x P)	115 x 32 x 111 [mm]
Potência especificada da fonte	65 [W]

Fonte: Autores, adaptado de Intel.

GO SMALL. GET BIG.

Highlighted Features

- 1 6th generation Intel® Core™ i5-6260U processor
- 2 Intel® Iris™ Graphics 540
- 3 Two DDR4 SO-DIMM sockets
(up to 32 GB, 2133 MHz)
- 4 M.2 slot with flexible support for a 42 or 80 mm
M.2 SSD, 1x SATA3 port for connection to
2.5" HDD or SSD (NUC6i5SYH)
- 5 Intel® Dual Band Wireless-AC 8260 and Bluetooth® 4.1
- 6 Back panel DC power connector (12V - 19V)
- 7 One HDMI* 1.4b display port supporting
8 channel audio (7.1 surround sound)
- 8 Intel® Gigabit LAN
- 9 2x USB 3.0 ports on the back panel
- 10 One Mini DisplayPort® version 1.2 supporting
8 channel digital audio (7.1 surround sound)
- 11 Kensington lock support
- 12 Support for user-replaceable third-party lids
- 13 SD card slot
- 14 2x USB 3.0 ports (including one charging port)
- 15 Intel® HD Audio¹ via Headphone/Microphone jack
- 16 Consumer infrared sensor



NUC6i5SYK



INTEL® NUC KITS NUC6i5SYK AND NUC6i5SYH

Technical Specifications

Processor

- Intel® Core™ i5-6260U processor
(1.8 GHz up to 2.9 GHz Turbo, Dual Core, 4 MB Cache, 15W TDP)

Graphics

- Intel® Iris™ Graphics 540
- One HDMI* 1.4b port
- One Mini DisplayPort* 1.2

System Memory

- Two DDR4 SO-DIMM sockets
(up to 32 GB, 2133 MHz), 1.2V

Storage Capabilities

NUC6i5SYH

- SDXC slot with UHS-I support on the side
- One M.2 Type M connector supporting 22x42 or 22x80 M.2 SSD
- One SATA3 port for connection to 2.5" HDD or SSD (up to 9.5 mm thickness)

NUC6i5SYK

- SDXC slot with UHS-I support on the side
- One M.2 Type M connector supporting 22x42 or 22x80 M.2 SSD

Peripheral Connectivity

- Intel® Gigabit LAN
- Four Super Hi-Speed USB 3.0 ports
(two back panel ports and two front ports including one charging port)
- Two additional Hi-Speed USB 2.0 ports via internal headers
- Intel® Dual Band Wireless-AC 8260 (802.11ac), 1x1, up to 867 Mbps
- Dual Mode Bluetooth® 4.1

System Bios

- 64 Mb Flash EEPROM with Intel® Platform Innovation Framework for EFI Plug and Play
- Advanced configuration and power interface V3.0b, SMBIOS2.5
- Intel® Visual BIOS
- Intel® Express BIOS update support

Hardware Management Features

- Processor fan speed control
- Voltage and temperature sensing
- Fan sensor inputs used to monitor fan activity
- ACPI-compliant power management control

Expansion Capabilities

- One NFC header
- Two Internal USB 2.0 ports via two 1x4 headers (for replaceable lid support)
- One AUX_PWR header

Audio

- Up to 7.1 surround audio via HDMI* and Mini DisplayPort*
- Headphone/Microphone jack on the front panel

Front Panel Header

- Reset, HDD LED, Power LEDs, power on/off

Mechanical Chassis Size

NUC6i5SYH

- 4.53" x 4.37" x 1.89"
- 115 mm x 111 mm x 48 mm

NUC6i5SYK

- 4.53" x 4.37" x 1.26"
- 115 mm x 111 mm x 32 mm

Baseboard Power Requirements

- 19V, 65W wall-mount AC-DC power adapter
- Multi-country AC adapter

Environment Operating Temperature

- 0° C to +40° C

Storage Temperature

- -20° C to +60°

Safety Regulations and Standards

- IEC 60950-1
- UL 60950-1
- EN 60950-1
- CAN/CSA-C22.2 No. 60950-1

EMC (Class B)/RF Regulations and Standards

- CISPR 22
- FCC CFR Title 47, Chapter I, Part 15, Subparts A, B
- ICES-003
- EN 55022
- EN 55024
- ETSI EN 300 328
- ETSI EN 300 440-1
- ETSI EN 300 440-2
- ETSI EN 301 489-1
- ETSI EN 301 489-17
- ETSI EN 301 893
- EN 62311
- AS/NZS 2772.2
- AS/NZS 4268
- VCCI V-3, V-4
- KN-22
- KN-24
- CNS 13438

Environmental Regulations

- RoHS Directive 2011/65/EU
- WEEE Directive 2012/19/EU
- China RoHS MII Order #39

¹ Intel technologies may require enabled hardware, specific software, or services activation. Check with your system manufacturer or retailer.

Intel products are not intended for use in medical, life-saving, or life-sustaining applications. Intel may make changes to specifications and product descriptions at any time, without notice.

All products, dates, and figures specified are preliminary based on current expectations, and are subject to change without notice. Availability in different channels may vary.

Actual Intel® NUC kit may differ from the image shown.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY RELATING TO THE SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT, OR OTHER INTELLECTUAL PROPERTY RIGHT.

To go small, visit:
www.intel.com/NUC



A.2 Câmera Basler ace acA4600

A ficha resumo contendo as principais especificações da câmera fotográfica Basler ace acA4600 está disposta na Tabela ???. Maiores informações e desenho mecânico deste componente estão dispostos no *datasheet* anexado a seguir.

Tabela 23: Especificações da câmera Basler ace.

Item	Especificação
Especificação de câmera	acA4600
Resolução	4608 × 3288 [px]
Modelo do sensor	On-Semi MT9F002
Taxa de captura (padrão)	10 [fps]
Tamanho de pixel	1,4 × 1,4 μ m
Alcance dinâmico	62,1 [dB]
Círculo máximo de imagem	Formato óptico 1/2,3"
Encaixe de lente	C-Mount
Protocolo de comunicação	USB 3.0 (5 [Gbit/s])
Porta de comunicação	USB
Tensão de operação	5 [V]
Temperatura de operação	0 ~ 50 [°C]
Massa	< 80 [g]
Dimensões (L x A x P)	29,3 x 29 x 29 [mm ³]
Potência necessária	~ 2,5 W a 5 V _{DC} , máximo de 2,8 W

Fonte: Basler.

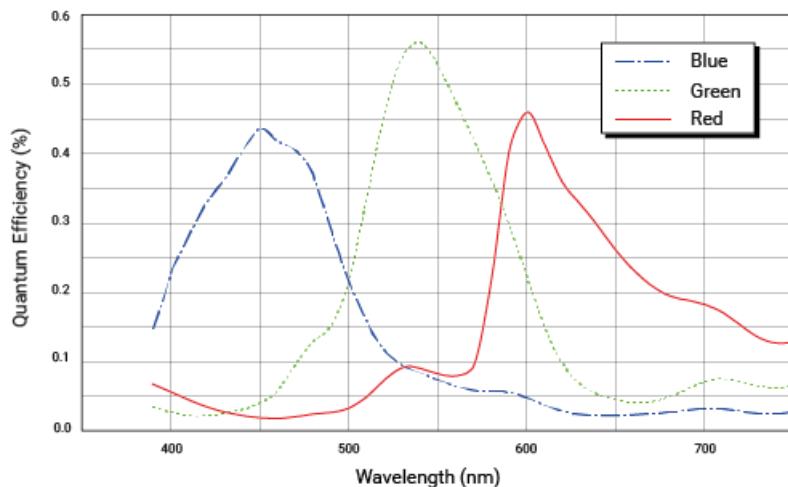
Specifications

General Specifications

Specification	acA4600-10uc
Resolution (H x V Pixels)	4608 x 3288
Sensor Type	ON Semiconductor MT9F002 Progressive scan CMOS Rolling shutter
Optical Size	1/2.3"
Effective Sensor Diagonal	8.0 mm
Pixel Size (H x V)	1.4 µm x 1.4 µm
Frame Rate (at Default Settings)	10 fps
Product Line	 ace classic
Mono / Color	Color
Image Data Interface	USB 3.0, nominal max. 5 Gbit/s (SuperSpeed)
Pixel Formats	See Pixel Format .
Synchronization	Via hardware trigger Via software trigger Via free run
Exposure Time Control	Programmable via the camera API
Camera Power Requirements	Nominal 5 VDC supplied via the camera's USB 3.0 port ≈2.5 W (typical) @ 5 VDC ≈2.8 W (max.)
I/O Lines	1 opto-coupled input line

	<p>1 opto-coupled output line 2 general purpose I/O (GPIO) lines</p>
Lens Mount	C-mount
Size (L x W x H)	29.3 mm x 29 mm x 29 mm (without lens mount or connectors) 48.2 mm x 29 mm x 29 mm (with lens mount and connectors)
Weight	<80 g
Conformity	CE (includes RoHS), UL Listed, FCC, GenICam 2.x (including PFNC 2.x and SFNC 2.x), IP30, USB3 Vision, REACH The EU Declaration of Conformity is available on the Basler website .
Software	Basler pylon Camera Software Suite (version 4.0 or higher) Available for Windows, Linux x86, Linux ARM, and OS X
Accessories	Cables for your camera model Lenses for your camera model Additional accessories for your camera model

Spectral Response



The spectral response curve excludes lens characteristics, light source characteristics, and IR cut filter characteristics.

IR Cut Filter

Color cameras are equipped with an IR cut filter. The filter is mounted in a filter holder inside the lens mount.

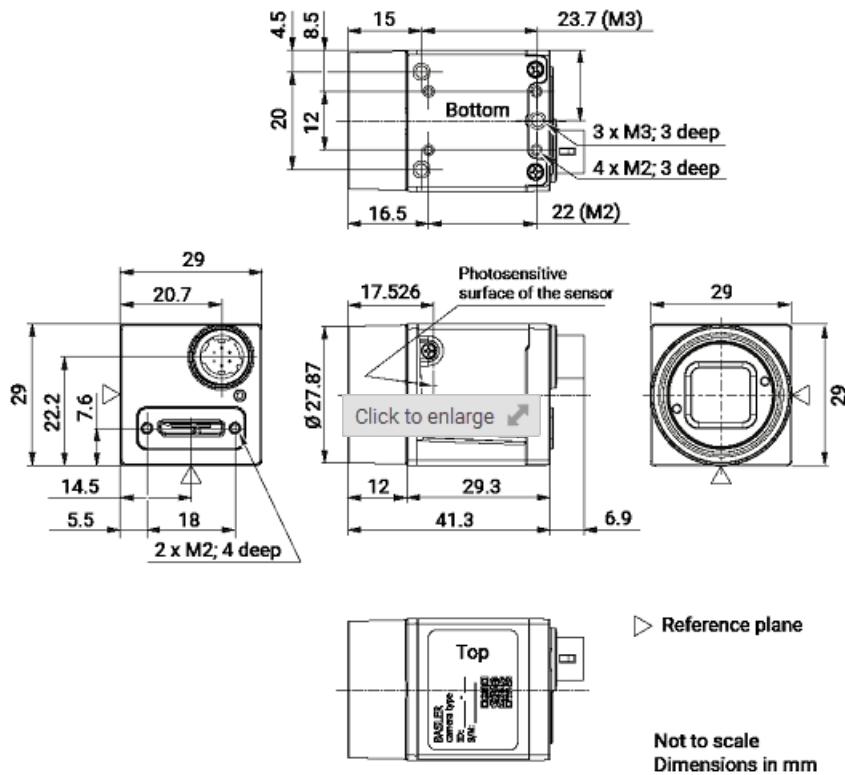
The IR cut filter has the following spectral characteristics:

Wavelength [nm]	Transmittance
450–610	$T_{\min} > 90 \%$
450–620	$T_{\text{avg}} > 93 \%$
645 ± 10	$T = 50 \%$
700–1070	$T_{\max} < 4 \%$
690–1070	$T_{\text{avg}} < 1 \%$

The filter holder can be removed. For more information, see the [ace IR Cut Filter Holder Removal Procedure](#) application note.

Mechanical Specifications

Camera Dimensions and Mounting Points



Maximum Allowed Lens Intrusion

→ See [Maximum Allowed Lens Intrusion](#).

Mounting Instructions

→ See [Mounting Instructions](#).

Stress Test Results

→ See [Stress Test Results](#).

Requirements

Environmental Requirements

Temperature and Humidity

Housing temperature during operation	0–50 °C (32–122 °F)
Humidity during operation	20–80 %, relative, non-condensing
Storage temperature	-20–80 °C (-4–176 °F)
Storage humidity	20–80 %, relative, non-condensing
Housing temperature according to UL 60950-1	max. 70 °C (158 °F)
Ambient temperature according to UL 60950-1	max. 30 °C (86 °F)

UL 60950-1 test conditions: no lens attached to camera; no heat dissipation measures; ambient temperature kept at 30 °C (86 °F).

Heat Dissipation

→ See [Providing Heat Dissipation](#).

Electrical Requirements

DANGER
Electric Shock Hazard
WARNING
Fire Hazard
NOTICE
Incorrect voltage can damage the camera.

Camera Power

You must supply camera power that complies with the Universal Serial Bus 3.0 specification.

The camera's nominal operating voltage is 5 VDC, effective on the camera's connector.

Opto-Coupled I/O Input Line

Voltage	Description
30 VDC	Absolute maximum. This voltage must never be exceeded. Doing so may damage the camera and voids the warranty.
0–24 VDC	Safe operating range.
0–1.4 VDC	Indicates a logical 0 (with inverter disabled).
>1.4–2.2 VDC	Region where the logic level transition occurs; the logical status is not defined in this region.
>2.2 VDC	Indicates a logical 1 (with inverter disabled).
<ul style="list-style-type: none">• Minimum current: 5 mA• Current draw: 5–15 mA	

Opto-Coupled I/O Output Line

Voltage	Description
30 VDC	Absolute maximum. This voltage must never be exceeded. Doing so may damage the camera and voids the warranty.
3.3–24 VDC	Safe operating range.
<3.3 VDC	Unreliable I/O output.
<ul style="list-style-type: none">• Leakage current: <60 μA. Actual leakage depends on operating temperature and production spread of electronic components.• Maximum load current: 50 mA• Minimum load current: Not specified. Consider the following:<ul style="list-style-type: none">• Leakage current will have a stronger effect when load currents are low.• Propagation delay of the output increases as load currents decrease.• Higher-impedance circuits tend to be more susceptible to EMI.• Higher currents cause higher voltage drops in long cables.	

General Purpose I/O Lines

NOTICE

Applying incorrect electrical signals to the camera's GPIO line can severely damage the camera.

Operation as Input

Voltage	Description
30 VDC	Absolute maximum. This voltage must never be exceeded. Doing so may damage the camera and voids the warranty.
0–5 VDC	Safe operating range. The minimum external pull-up voltage is 3.3 VDC.
0–0.8 VDC	Indicates a logical 0 (with inverter disabled).
>0.8–2.0 VDC	Region where the logic level transition occurs; the logical status is not defined in this region.
>2.0 VDC	Indicates a logical 1 (with inverter disabled).
<ul style="list-style-type: none">Current draw (high-level): <100 µASink current: Your application must be able to accept 2 mA sink current from the GPIO input line without exceeding 0.8 VDC.	

Operation as Output

Voltage	Description
30 VDC	Absolute maximum. This voltage must never be exceeded. Doing so may damage the camera and voids the warranty.
3.3–24 VDC	Safe operating range.
<3.3 VDC	Unreliable GPIO output.
<ul style="list-style-type: none">Internal pull-up resistor: $\approx 2\text{ k}\Omega$, with open collector. Many applications will have to provide an additional pull-up resistor.Residual voltage ("on" state): $\approx 0.4\text{ V}$ at 50 mA and 25 °C (77 °F) housing temperature. Actual residual voltage depends on operating temperature, load current, and production spread of electronic components.Leakage current: <60 µA. Actual leakage depends on operating temperature and production spread of electronic components.Maximum load current: 50 mAMinimum load current: Not specified. However, consider the following:<ul style="list-style-type: none">Leakage current will have a stronger effect when load currents are low.Propagation delay of the output increases as load currents decrease.Higher-impedance circuits tend to be more susceptible to EMI.	

- Higher currents cause higher voltage drops in long cables.

Circuit Diagrams

→ See [Circuit Diagrams for Basler ace Cameras](#).

Cable Requirements

USB 3.0 Cable

- Use a high-quality USB 3.0 cable with a Micro-B plug.
- To [avoid EMI](#), cables must be shielded, as specified in the USB 3.0 standard.
- Basler recommends using USB 3.0 cables from the [Basler Vision Components](#) range.

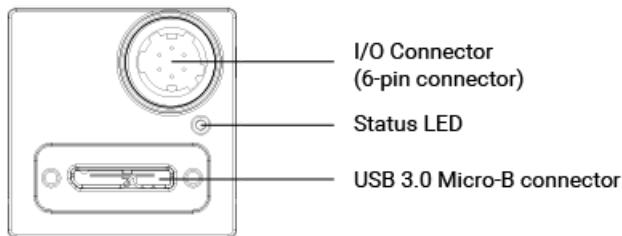
For more information about recommended USB 3.0 cables, see the [Recommended Accessories for Basler USB 3.0 Cameras](#) document.

I/O Cable

- The I/O cable must be shielded.
- The I/O cable must have a cross-section at least 0.14 mm^2 (close to AWG26).
- Use twisted pair wire cables.
- Maximum recommended cable length: 10 m
- Camera-side connector: Hirose micro plug (part number HR10A-7P-6S) or equivalent
- Close proximity to strong magnetic fields should be avoided.
- Basler recommends using I/O cables from the [Basler Vision Components](#) range:
 - [GPIO cable, 10 m](#) (yellow cable): For use with the [GPIO lines](#) of your camera. To avoid interferences due to crosstalk, the opto-coupled I/O lines are not connected.
 - [Opto-I/O cable, 10 m](#) (blue cable): For use with the [opto-coupled I/O lines](#) of your camera. To avoid interferences due to crosstalk, the GPIO lines are not connected.
 - [Opto-GPIO Y-cable, 2 x 10 m](#) (yellow-blue cable): Allows you to use the [GPIO lines](#) and the [opto-coupled I/O lines](#) simultaneously without interferences due to crosstalk. There are two separate wires to split both I/O types.

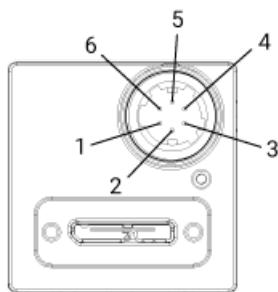
Physical Interface

Camera Connectors and Status LED



6-pin connector	Hirose micro receptacle (part number HR10A-7R-6PB) Recommended mating connector: Hirose micro plug (part number HR10A-7P-6S)
USB 3.0 Micro-B connector	Standard USB 3.0 Micro-B connector with screw lock Recommended mating connector: Standard connector with screws
Status LED	Indicates camera operation (LED lit = camera operating).

Connector Pin Numbering and Assignments



Pin	Line	Function
1	Line 3	General purpose I/O (GPIO) line
2	Line 1	Opto-coupled I/O input line
3	Line 4	General purpose I/O (GPIO) line
4	Line 2	Opto-coupled I/O output line
5	-	Ground for opto-coupled I/O lines
6	-	Ground for General Purpose I/O (GPIO) lines

Precautions

→ See [Safety Instructions for Basler ace Cameras](#).

Installation

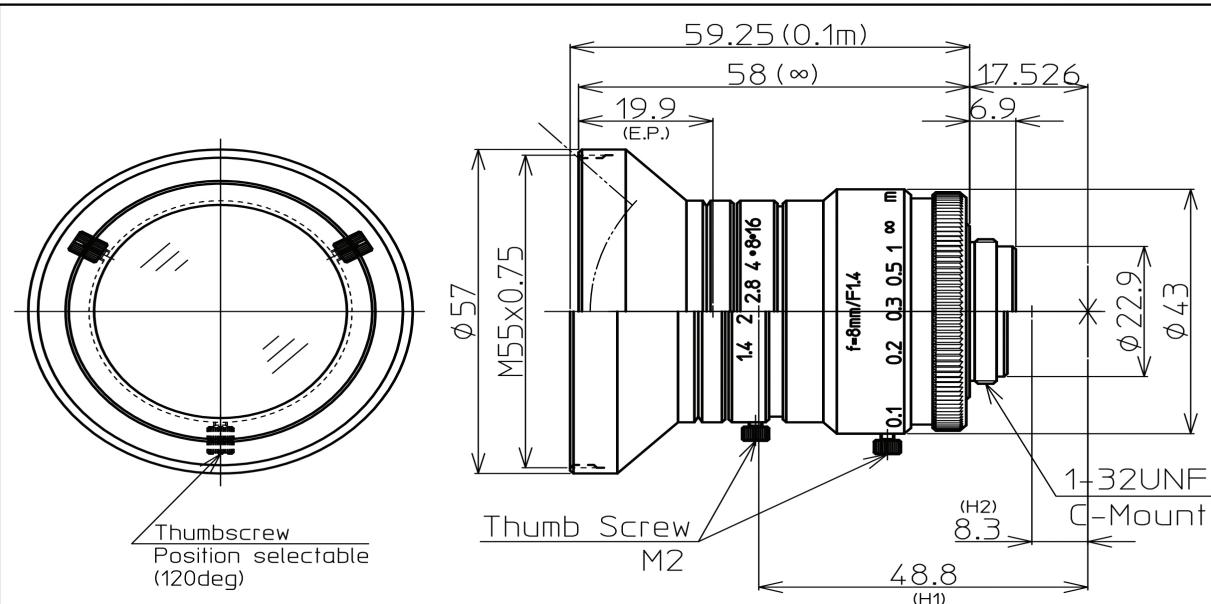
A.3 Lente Kowa LM8HC

A ficha resumo contendo as principais especificações da lente Kowa LM8HC utilizada em conjunto com a câmera fotográfica está disposta na Tabela 24. Maiores informações e desenho mecânico deste componente estão dispostos no *datasheet* anexado a seguir.

Tabela 24: Especificações da lente Kowa LM8HC para sensor de 1 polegada de diâmetro.

Item	Especificação
Comprimento focal	8 [mm]
Razão de diâmetro máxima	$F = 1 : 1,4 \sim 1,6$
Tamanho da imagem	9,6 x 12,8 [mm]
Ângulo de visão vertical	63,0 [°]
Ângulo de visão horizontal	79,4 [°]
Ângulo de visão diagonal	92,4 [°]
Distância focal mínima	0,1 [mm]
Tamanho do filtro	M55 x 0.75
Encaixe de lente	C-Mount
Temperatura de operação	-10 ~ 50 [°C]
Temperatura de armazenagem	-20 ~ 60 [°C]
Massa	205 [g]
Dimensões (L x A x P)	66,15 x 57 x 57 [mm]

Fonte: Autores, adaptado de Kowa American Corp.



1" 1.4 / 8mm C	1"	2/3"	1/1.8"
FOCAL LENGTH	f=8mm		
MAX. DIAMETER (RATIO)	F=1 : 1.4 ~ 16		
PICTURE SIZE	9.6×12.8mm	6.6×8.8mm	5.4×7.2mm
ANGLE			
VER. ANGLE	63.0°	44.7°	36.9°
HOR. ANGLE	79.4°	58.3°	48.5°
DIA. ANGLE	92.4°	70.7°	59.5°
DISTORTION (TV)	-1.2%		
SHOOTING RANGE AT	VER.	143.2mm	96.0mm
MINIMUM FOCUS	HOR.	196.0mm	130.4mm
DISTANCE	DIA.	247.0mm	166.2mm
MINIMUM FOCUS DISTANCE		0.1m	
FLANGE BACK		17.526mm in air	
BACK FOCUS		11.2mm in air	
FILTER SCREW SIZE	M55x0.75		
FRONT/REAR EFFECTIVE DIA.	FRONT Ø 37.4mm	REAR Ø 17.5mm	
MOUNT	C-MOUNT		
EXIT PUPIL	-45.9mm		
TEMPERATURE RANGE	-10~+50 °C		
STORAGE TEMPERATURE RANGE	-20~+60 °C		
RESOLUTION	CENTER 120lp/mm	CORNER 80lp/mm	
WEIGHT	205 g		

NOTE: Specifications and availability are subject to change without notice



	1/1	HR975NCN-3H
2015.7.30		1" F1.4 f=8mm
2014.12.25		LM8HC
2013.4.15		

A.4 Motores ROBOTIS

Neste apêndice estão dispostos as fichas resumo, *datasheets* e desenhos mecânicos dos motores fabricados pela ROBOTIS e utilizados na montagem do manipulador desenvolvido.

A.4.1 DYNAMIXEL PH42-020-S300-R

A ficha resumo contendo as principais especificações do motor ROBOTIS DYNAMIXEL PH42-020-S300-R está disposta na Tabela 25. O gráfico de torque versus velocidade, corrente e eficiência está disposto no *datasheet* anexado a seguir. O desenho mecânico deste motor esta disposto na Figura 48.

Tabela 25: Especificações do motor ROBOTIS DYNAMIXEL PH42-20-S300-R.

Item	Especificação
Microcontrolador	ARM CORTEX-M4 (168 [MHz], 32Bit)
Motor	Coreless (Maxon)
Taxa de transmissão	9,600 [bps] ~10.5 [Mbps]
Modos de operação	Torque Control Mode Velocity Control Mode Position Control Mode Extended Position Control Mode PWM Control Mode (Voltage Control Mode)
Massa	340 [g]
Dimensões (L x A x P)	42 x 84 x 42 [mm]
Resolução	607,500 [pulsos/rev]
Relação de transmissão	303.75:1
Folga	<6 [arcmin], 0.1 [°]
Carga radial	280 [N] (10 [mm] afastado da face do eixo)
Carga axial	100 [N]
Velocidade sem carga	32.7 [rev/min]
Corrente sem carga	0.57 [A]
Velocidade contínua	29.2 [rev/min]
Torque contínuo	5.1 [Nm]
Corrente contínua	1.5 [A]
Potência de saída	20 [W]
Temperatura de operação	5 ~55 [°C]
Tensão de operação	24.0 [V]
Sinal de comando	Pacotes digitais
Protocolo de comunicação	RS485 Comunicação serial assíncrona (8bit, 1 bit de parada, sem paridade)
Conexão física	RS485 Barramento multiponto
ID	253 ID (0 ~252)
Corrente de espera	80 [mA]

Fonte: Autores, adaptado de ROBOTIS.

[Edit on GitHub](#)

ToC

**H42-20-S300-R(A)**

1. Specifications

Item	Specifications
MCU	ARM CORTEX-M4 (168 [MHz], 32Bit)
Motor	Coreless(Maxon)
Baud Rate	9,600 [bps] ~ 10.5 [Mbps]
Operating Modes	Torque Control Mode Velocity Control Mode Position Control Mode Extended Position Control Mode PWM Control Mode(Voltage Control Mode)
Weight	340 [g]
Dimensions (W x H x D)	42 x 84 x 42 [mm]
Resolution	607,500 [pulse/rev]
Gear Ratio	303.75:1
Backlash	< 6 [arcmin], 0.1 [°]
Radial Load	280 [N] (10 [mm] away from the horn)
Axial Load	100 [N]
No Load Speed	32.7 [rev/min]
No Load Current	0.57 [A]
Continuous Speed	29.2 [rev/min]
Continuous Torque	5.1 [Nm]
Continuous Current	1.5 [A]
Output	20 [W]
Operating Temperature	5 ~ 55 [°C]



ToC

TOP

Operating Voltage	24.0 [V]
Command Signal	Digital Packet
Protocol Type	RS485 Asynchronous Serial Communication (8bit, 1stop, No Parity)
Physical Connection	RS485 Multidrop Bus
ID	253 ID (0 ~ 252)
Standby Current	80 [mA]

**DANGER**

(May cause serious injury or death)

- Never place items containing water, flammables, and solvents near product.
- Never place fingers, arms, toes, and other body parts near product during operation.
- Cut power off if product emits strange odors or smoke.
- Keep product out of reach of children.
- Check the power polarity before wiring.

**CAUTION**

(May cause injury or damage to product)

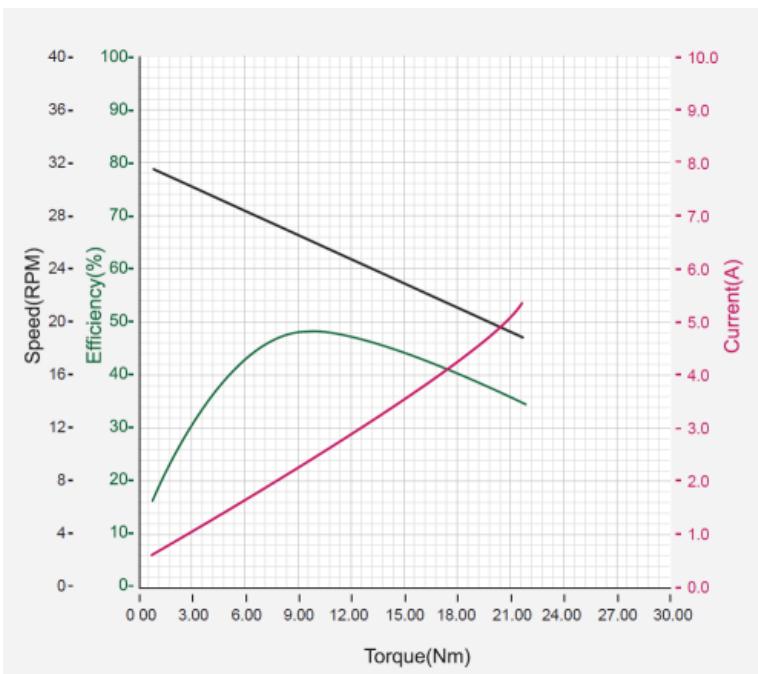
- Do not operate the product at a temperature exceeding 5 ~ 55 [°C] range.
- Do not insert sharp blades nor pins during product operation.

**ATTENTION**

(May cause injury or damage to product)

- Do not disassemble or modify product.
- Do not drop or apply strong shock to product.

1. 1. Performance Graph



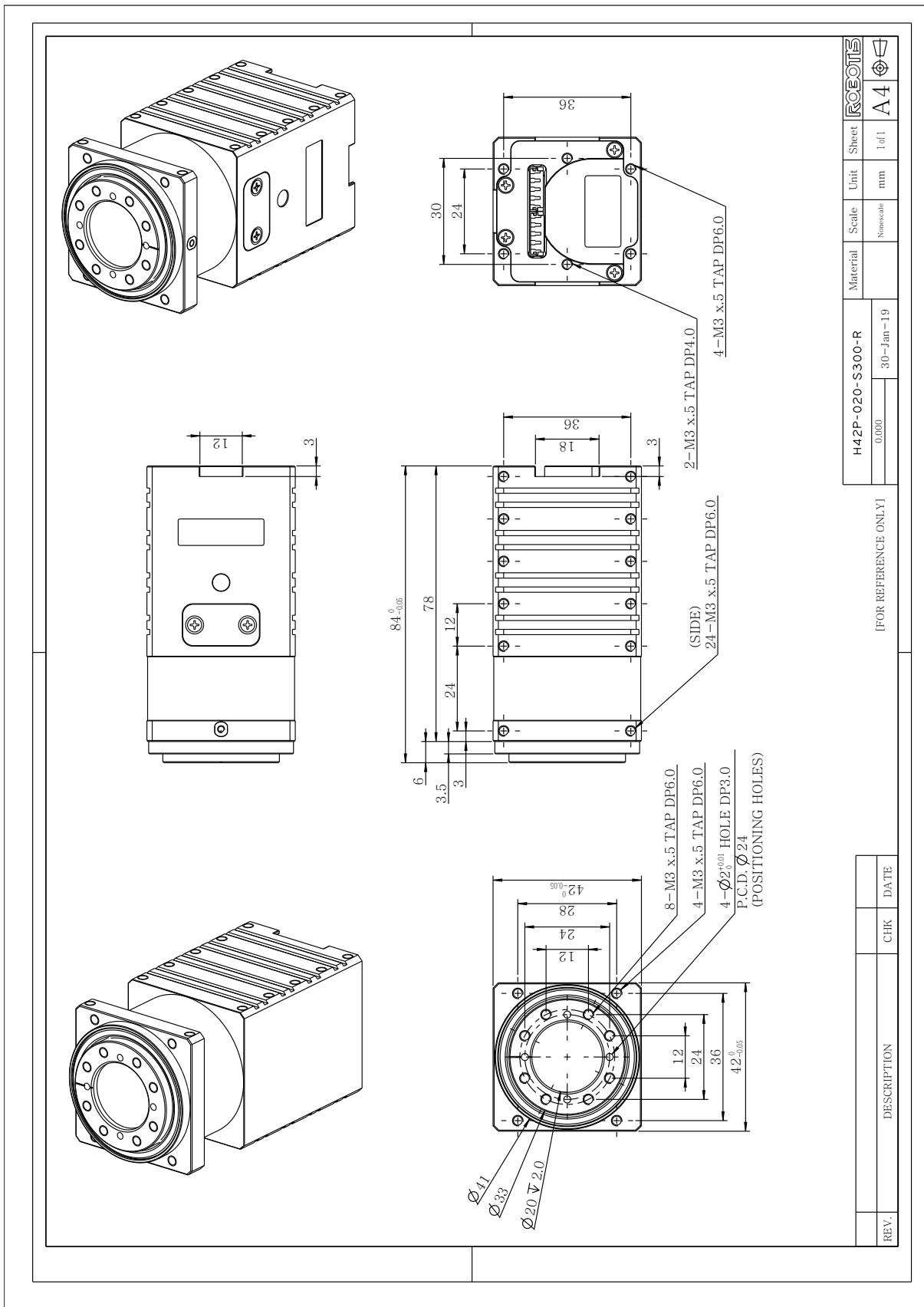
[Show Enlarged Graph](#)

NOTE : The Max Torque and the Stall Torque of Performance Graph are different in measurement methods. Stall torque is a measured value of the momentary torque that it can reach. This is generally how RC servos are measured. The Performance graph is also called as N-T curves, which is measured with the gradually increasing load. The actual motor operation environment is closer to the performance graph, not stall torque method. For this reason, the performance graph is broadly used in the industrial field. Generally, Max Torque of the Performance Graph is less than the Stall Torque.

CAUTION : When supplying power

- It is recommended using ROBOTIS controller or SMPS2DYNAMIXEL.
- Do not connect or disconnect DYNAMIXEL when power is being supplied.
- In case of DYNAMIXEL PRO and DYNAMIXEL-P series, please supply power through 24V power port.

Figura 48: Desenho mecânico do motor ROBOTIS DYNAMIXEL PH42-020-S300-R



Fonte: ROBOTIS.

A.4.2 DYNAMIXEL PH54-100-S500-R

A ficha resumo contendo as principais especificações do motor ROBOTIS DYNAMIXEL PH54-100-S500-R está disposta na Tabela 26. O gráfico de torque versus velocidade, corrente e eficiência está disposto no *datasheet* anexado a seguir. O desenho mecânico deste motor esta disposto na Figura 49.

Tabela 26: Especificações do motor ROBOTIS DYNAMIXEL PH54-100-S500-R.

Item	Especificação
Microcontrolador	ARM CORTEX-M4 (168 [MHz], 32Bit)
Motor	BLDC (Maxon)
Taxa de transmissão	9,600 [bps] ~10.5 [Mbps]
Modos de operação	Controle de torque Controle de velocidade Controle de posição Controle de posição extendida (mais do que uma revolução) Controle por PWM (Controle de tensão)
Massa	740 [g]
Dimensões (L x A x P)	54 x 108 x 54 [mm]
Resolução	1,003,846 [pulsos/rev]
Relação de transmissão	501.923 : 1
Folga	<6 [arcmin], 0.1 [°]
Carga radial	370 [N] (10 [mm] afastado da face do eixo)
Carga axial	130 [N]
Velocidade sem carga	33.3 [rev/min]
Corrente sem carga	1.13 [A]
Velocidade contínua	29.2 [rev/min]
Torque contínuo	25.3 [Nm]
Corrente contínua	5.5 [A]
Potência de saída	100 [W]
Temperatura de operação	5 ~55 [°C]
Tensão de operação	24.0 [V]
Sinal de comando	Pacotes digitais
Protocolo de comunicação	RS485 Comunicação serial assíncrona (8bit, 1 bit de parada, sem paridade)
Conexão física	RS485 Barramento multiponto
ID	253 ID (0 ~252)
Corrente de espera	40 [mA]

Fonte: Autores, adaptado de ROBOTIS.

[Edit on GitHub](#)

ToC



PH54-100-S500-R

1. Specifications

Item	Specifications
MCU	ARM CORTEX-M4 (168 [MHz], 32Bit)
Motor	BLDC (Maxon)
Baud Rate	9,600 [bps] ~ 10.5 [Mbps]
Operating Modes	Torque Control Mode Velocity Control Mode Position Control Mode Extended Position Control Mode PWM Control Mode(Voltage Control Mode)
Weight	740 [g]
Dimensions (W x H x D)	54 x 108 x 54 [mm]
Resolution	1,003,846 [pulse/rev]
Gear Ratio	501.923 : 1
Backlash	< 6 [arcmin], 0.1 [°]
Radial Load	370 [N] (10 [mm] away from the horn)
Axial Load	130 [N]
No Load Speed	33.3 [rev/min]
No Load Current	1.13 [A]
Continuous Speed	29.2 [rev/min]



ToC

TOP

¹	Continuous Torque	25.3 [Nm]
¹	Continuous Current	5.5 [A]
Output		100 [W]
Operating Temperature		-5 ~ 55 [°C]
Input Voltage		24.0 [V]
Command Signal		Digital Packet
Protocol Type		RS485 Asynchronous Serial Communication (8bit, 1stop, No Parity)
Physical Connection		RS485 Multidrop Bus
ID		253 ID (0 ~ 252)
Standby Current		40 [mA]

¹ These specifications are calculated based on the specifications of the core motor.
Please consult ROBOTIS for the long term use or special use, or else refer to the Performance Graph for general use.

**DANGER**

(May cause serious injury or death)

- Never place items containing water, flammables, and solvents near product.
- Never place fingers, arms, toes, and other body parts near product during operation.
- Cut power off if product emits strange odors or smoke.
- Keep product out of reach of children.
- Check the power polarity before wiring.

**CAUTION**

(May cause injury or damage to product)

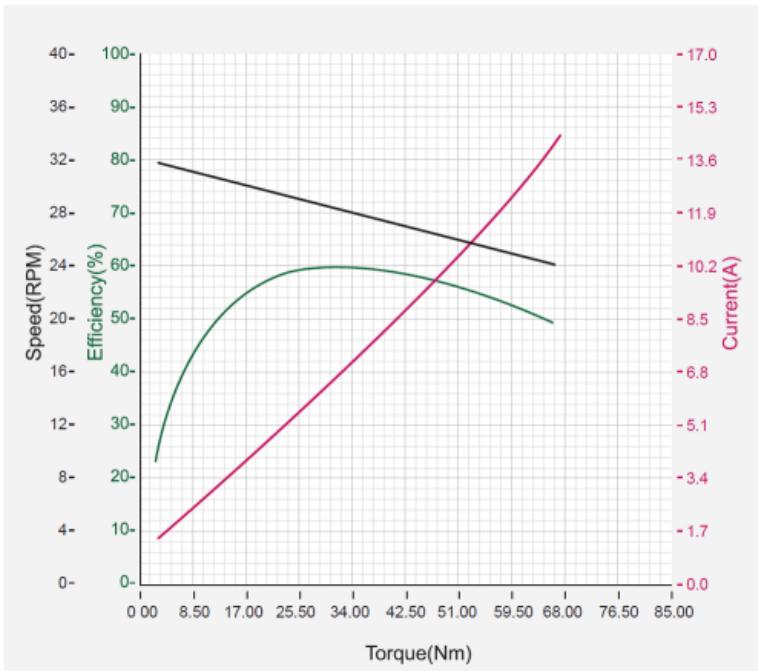
- Do not operate the product at a temperature exceeding -5 ~ 55 [°C] range.
- Do not insert sharp blades nor pins during product operation.

**ATTENTION**

(May cause injury or damage to product)

- Do not disassemble or modify product.
- Do not drop or apply strong shock to product.

1. 1. Performance Graph



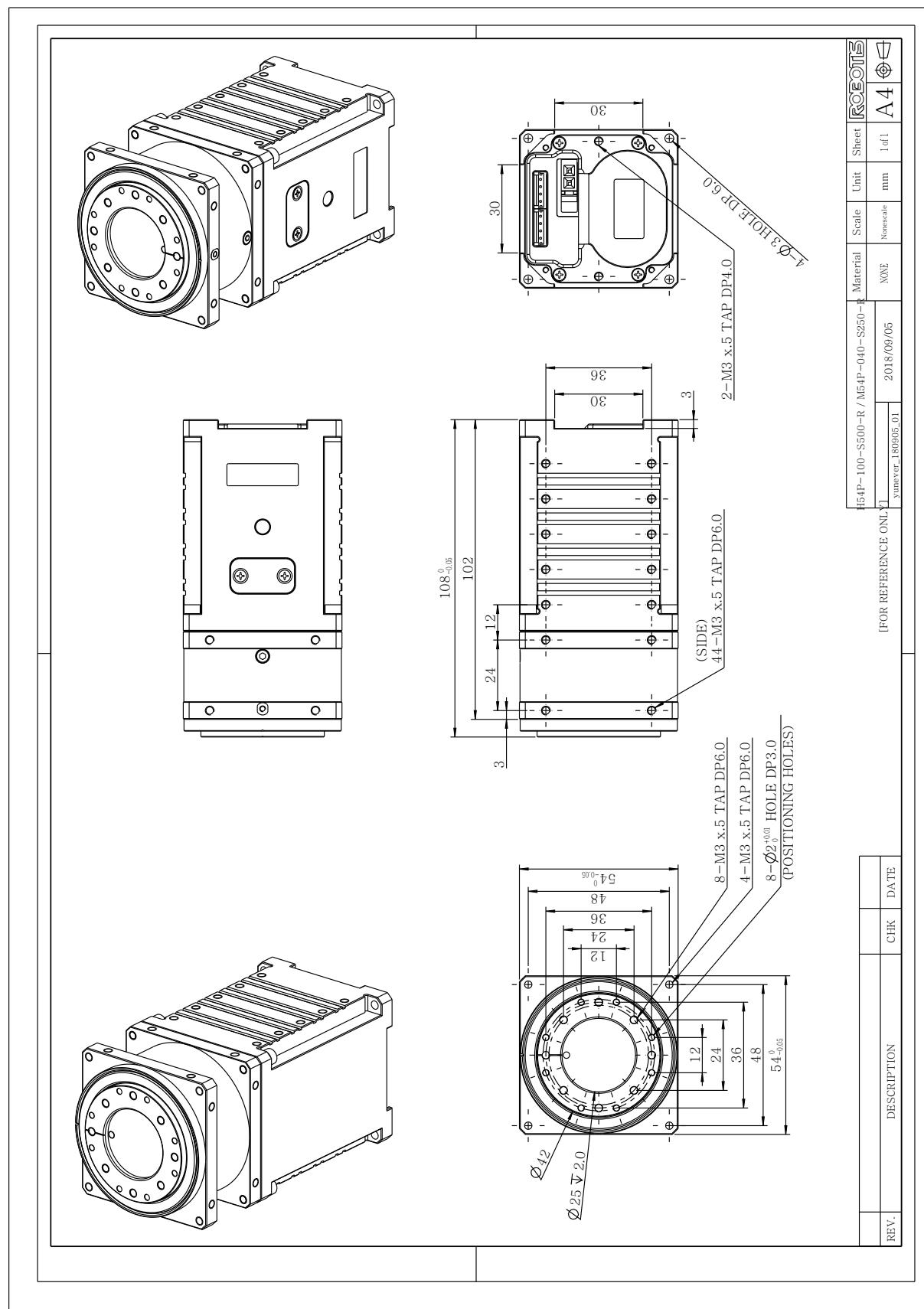
[Show Enlarged Graph](#)

NOTE : The Max Torque and the Stall Torque of Performance Graph are different in measurement methods. Stall torque is a measured value of the momentary torque that it can reach. This is generally how RC servos are measured. The Performance graph is also called as N-T curves, which is measured with the gradually increasing load. The actual motor operation environment is closer to the performance graph, not stall torque method. For this reason, the performance graph is broadly used in the industrial field. Generally, Max Torque of the Performance Graph is less than the Stall Torque.

CAUTION : When supplying power

- It is recommended using ROBOTIS controller or SMPS2DYNAMIXEL.
- Do not connect or disconnect DYNAMIXEL when power is being supplied.
- In case of DYNAMIXEL PRO and DYNAMIXEL-P series, please supply power through 24V power port.

Figura 49: Desenho mecânico do motor ROBOTIS DYNAMIXEL PH54-100-S500-R.



Fonte: ROBOTIS.

A.4.3 DYNAMIXEL PH54-200-S500-R

A ficha resumo contendo as principais especificações do motor ROBOTIS DYNAMIXEL PH54-200-S500-R está disposta na Tabela 27. O gráfico de torque versus velocidade, corrente e eficiência está disposto no *datasheet* anexado a seguir. O desenho mecânico deste motor esta disposto na Figura 50.

Tabela 27: Especificações do motor ROBOTIS DYNAMIXEL PH54-200-S500-R.

Item	Especificação
Microcontrolador	ARM CORTEX-M4 (168 [MHz], 32Bit)
Motor	BLDC (Maxon)
Taxa de transmissão	9,600 [bps] ~10.5 [Mbps]
Modos de operação	Controle de torque Controle de velocidade Controle de posição Controle de posição extendida (mais do que uma revolução) Controle por PWM (Controle de tensão)
Massa	855 [g]
Dimensões (L x A x P)	54 x 126 x 54 [mm]
Resolução	1,003,846 [pulsos/rev]
Relação de transmissão	501.923 : 1
Folga	<6 [arcmin], 0.1 [°]
Carga radial	370 [N] (10 [mm] afastado da face do eixo)
Carga axial	130 [N]
Velocidade sem carga	33.1 [rev/min]
Corrente sem carga	1.65 [A]
Velocidade contínua	29.0 [rev/min]
Torque contínuo	44.7 [Nm]
Corrente contínua	9.3 [A]
Potência de saída	200 [W]
Temperatura de operação	5 ~55 [°C]
Tensão de operação	24.0 [V]
Sinal de comando	Pacotes digitais
Protocolo de comunicação	RS485 Comunicação serial assíncrona (8bit, 1 bit de parada, sem paridade)
Conexão física	RS485 Barramento multiponto
ID	253 ID (0 ~252)
Corrente de espera	40 [mA]

Fonte: Autores, adaptado de ROBOTIS.

[Edit on GitHub](#)

ToC



TOP

PH54-200-S500-R

1. Specifications

Item	Specifications
MCU	ARM CORTEX-M4 (168 [MHz], 32Bit)
Motor	BLDC (Maxon)
Baud Rate	9,600 [bps] ~ 10.5 [Mbps]
Operating Modes	Torque Control Mode Velocity Control Mode Position Control Mode Extended Position Control Mode PWM Control Mode(Voltage Control Mode)
Weight	855 [g]
Dimensions (W x H x D)	54 x 126 x 54 [mm]
Resolution	1,003,846 [pulse/rev]
Gear Ratio	501.923 : 1
Backlash	< 6 [arcmin], 0.1 [°]
Radial Load	370 [N] (10 [mm] away from the horn)
Axial Load	130 [N]
No Load Speed	33.1 [rev/min]
No Load Current	1.65 [A]
Continuous Speed	29.0 [rev/min]



ToC

TOP

1	Continuous Torque	44.7 [Nm]
1	Continuous Current	9.3 [A]
Output	200 [W]	
Operating Temperature	-5 ~ 55 [°C]	
Input Voltage	24.0 [V]	
Command Signal	Digital Packet	
Protocol Type	RS485 Asynchronous Serial Communication (8bit, 1stop, No Parity)	
Physical Connection	RS485 Multidrop Bus	
ID	253 ID (0 ~ 252)	
Standby Current	40 [mA]	

1 These specifications are calculated based on the specifications of the core motor.

Please consult ROBOTIS for the long term use or special use, or else refer to the Performance Graph for general use.



DANGER

(May cause serious injury or death)

- Never place items containing water, flammables, and solvents near product.
- Never place fingers, arms, toes, and other body parts near product during operation.
- Cut power off if product emits strange odors or smoke.
- Keep product out of reach of children.
- Check the power polarity before wiring.



CAUTION

(May cause injury or damage to product)

- Do not operate the product at a temperature exceeding -5 ~ 55 [°C] range.
- Do not insert sharp blades nor pins during product operation.

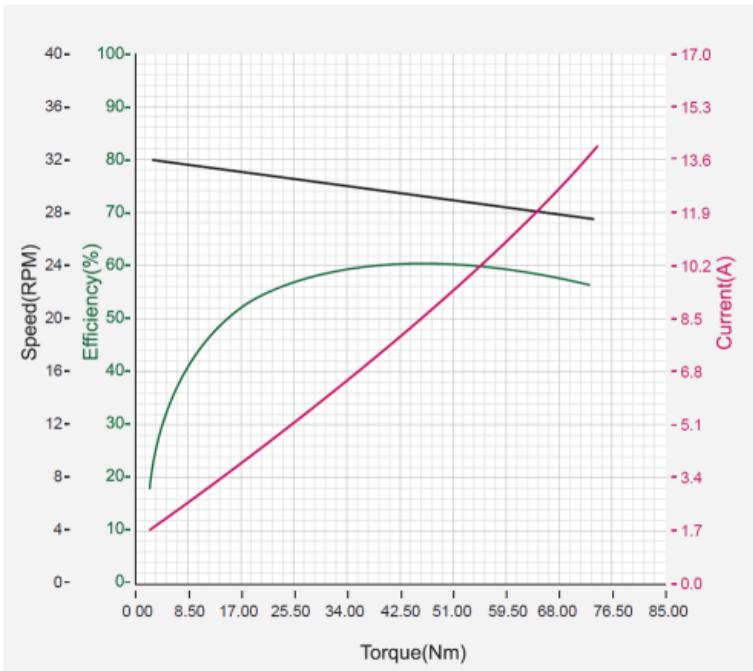


ATTENTION

(May cause injury or damage to product)

- Do not disassemble or modify product.
- Do not drop or apply strong shock to product.

1. 1. Performance Graph



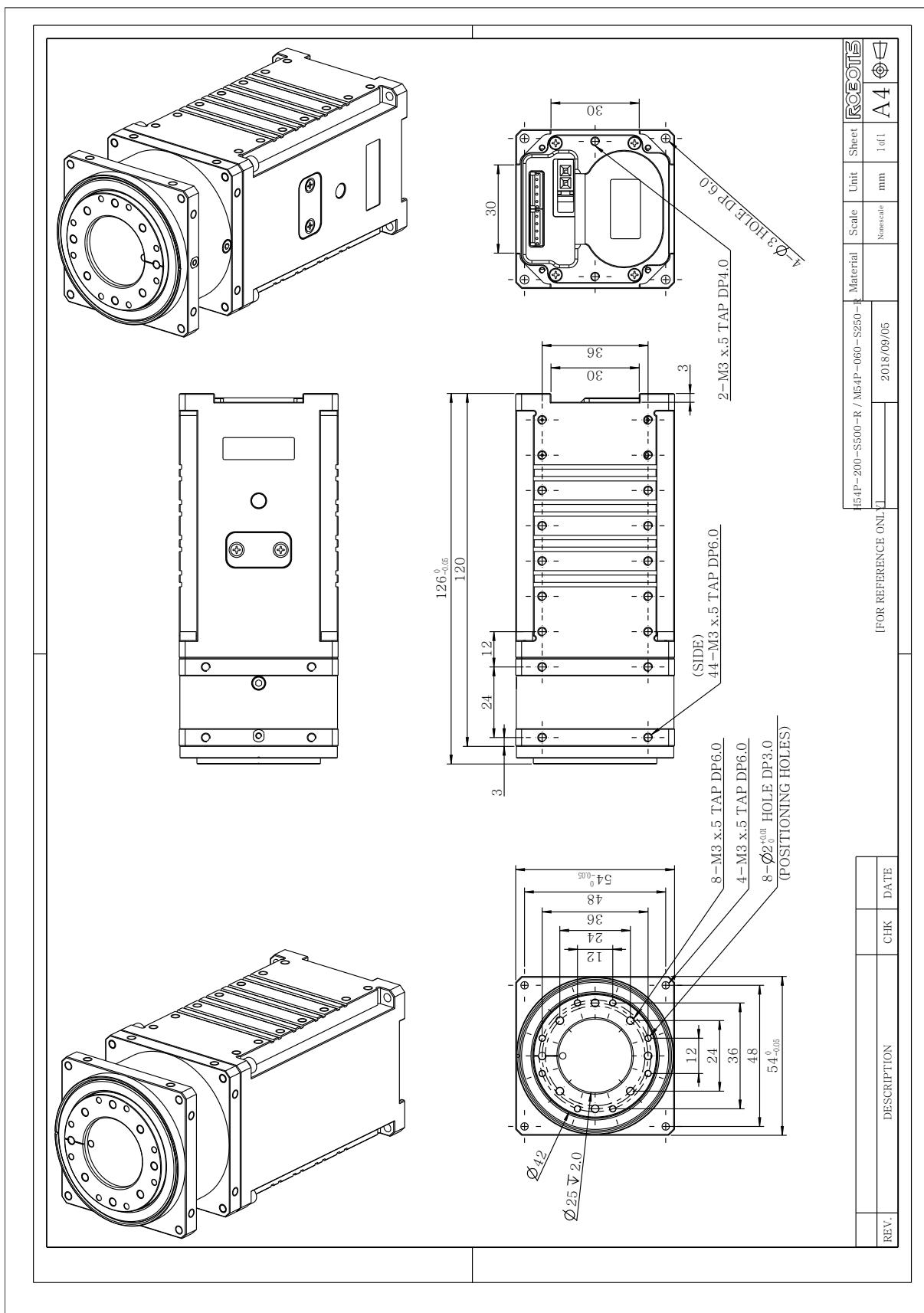
[Show Enlarged Graph](#)

NOTE : The Max Torque and the Stall Torque of Performance Graph are different in measurement methods. Stall torque is a measured value of the momentary torque that it can reach. This is generally how RC servos are measured. The Performance graph is also called as N-T curves, which is measured with the gradually increasing load. The actual motor operation environment is closer to the performance graph, not stall torque method. For this reason, the performance graph is broadly used in the industrial field. Generally, Max Torque of the Performance Graph is less than the Stall Torque.

CAUTION : When supplying power

- It is recommended using ROBOTIS controller or SMPS2DYNAMIXEL.
- Do not connect or disconnect DYNAMIXEL when power is being supplied.
- In case of DYNAMIXEL PRO and DYNAMIXEL-P series, please supply power through 24V power port.

Figura 50: Desenho mecânico do motor ROBOTIS DYNAMIXEL PH54-200-S500-R.



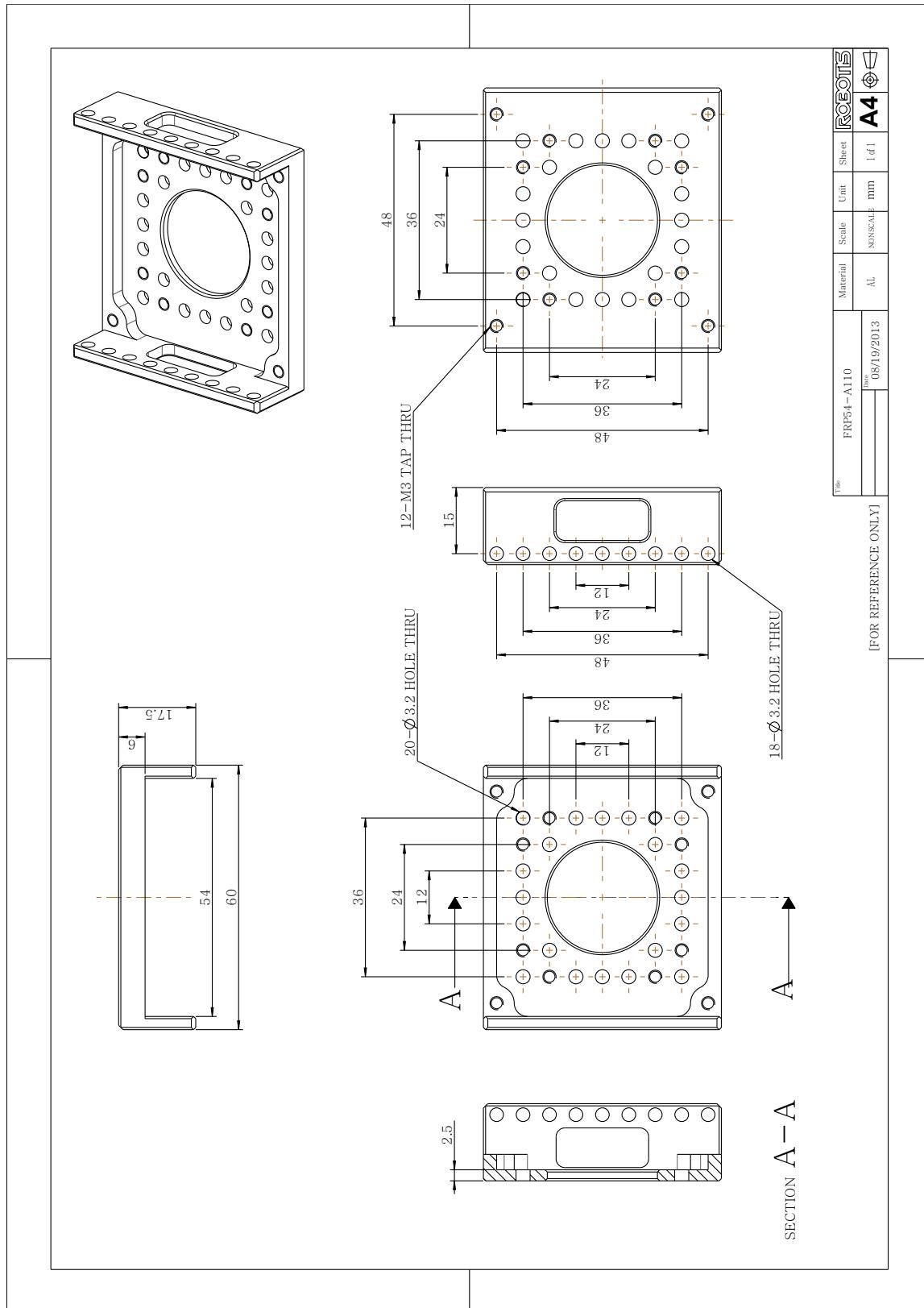
Fonte: ROBOTIS.

A.5 Estruturas ROBOTIS

Neste apêndice estão dispostos todos os desenhos mecânicos das estruturas fabricadas pela ROBOTIS e utilizadas na montagem do manipulador desenvolvido.

A.5.1 DYNAMIXEL Pro 54: *Adaptor Set* (FRP54-A110)

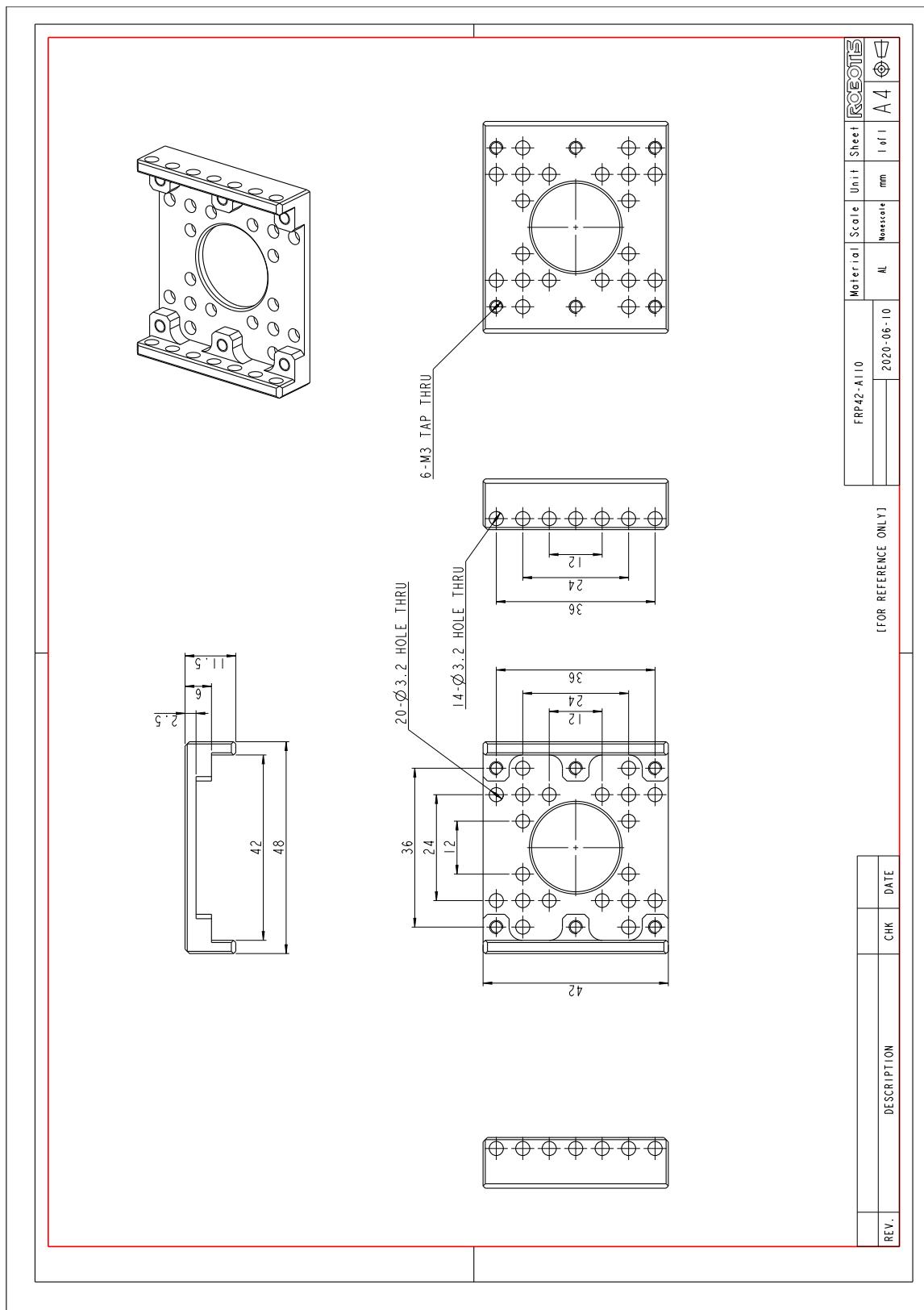
Figura 51: Desenho mecânico da estrutura ROBOTIS FRP54-A110.



Fonte: ROBOTIS.

A.5.2 DYNAMIXEL Pro 42: Adaptor Set (FRP42-A110K)

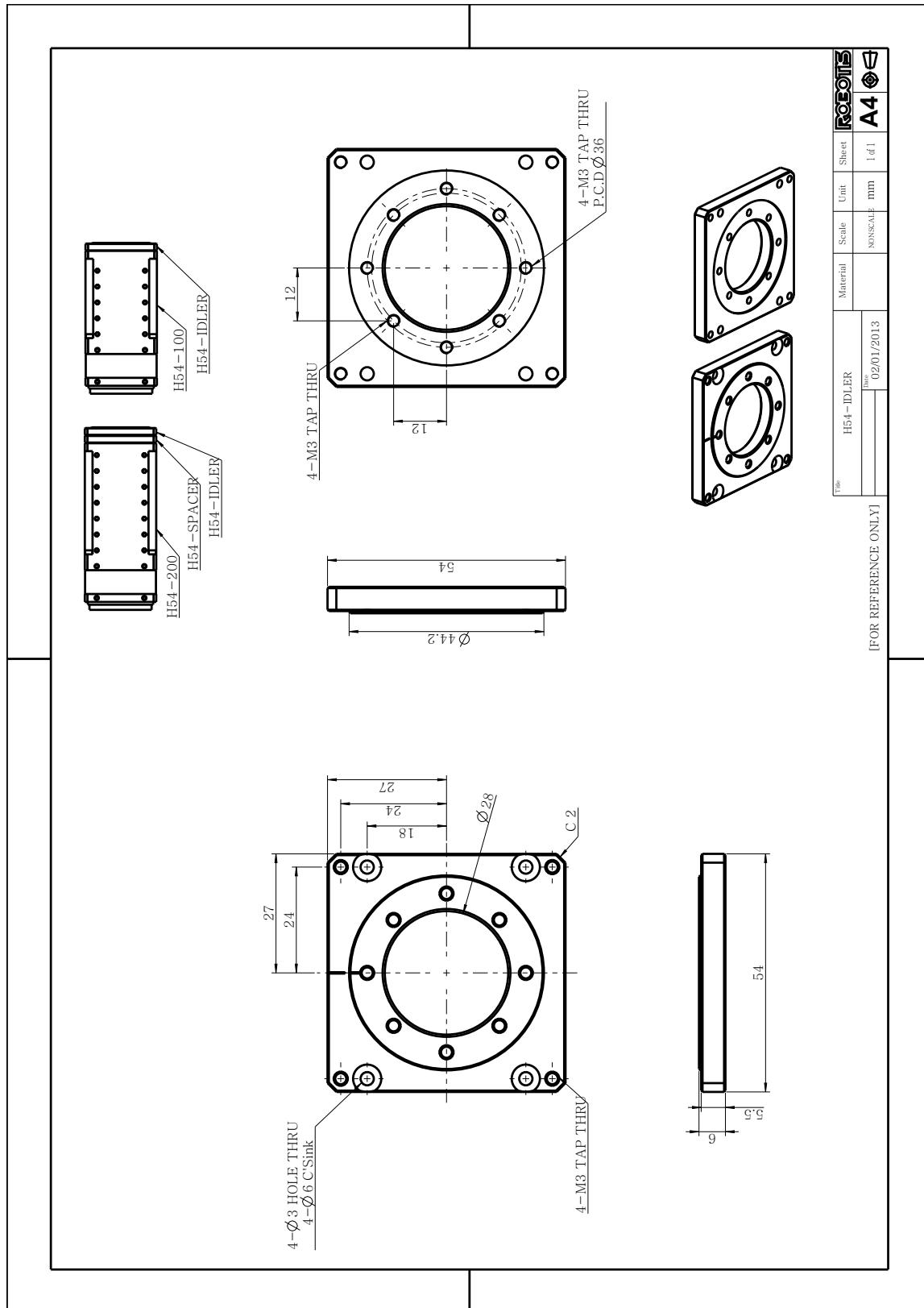
Figura 52: Desenho mecânico da estrutura ROBOTIS FRP42-A110.



Fonte: ROBOTIS.

A.5.3 DYNAMIXEL Pro 54: *Idler Bearing Set* (FRP54-I110K)

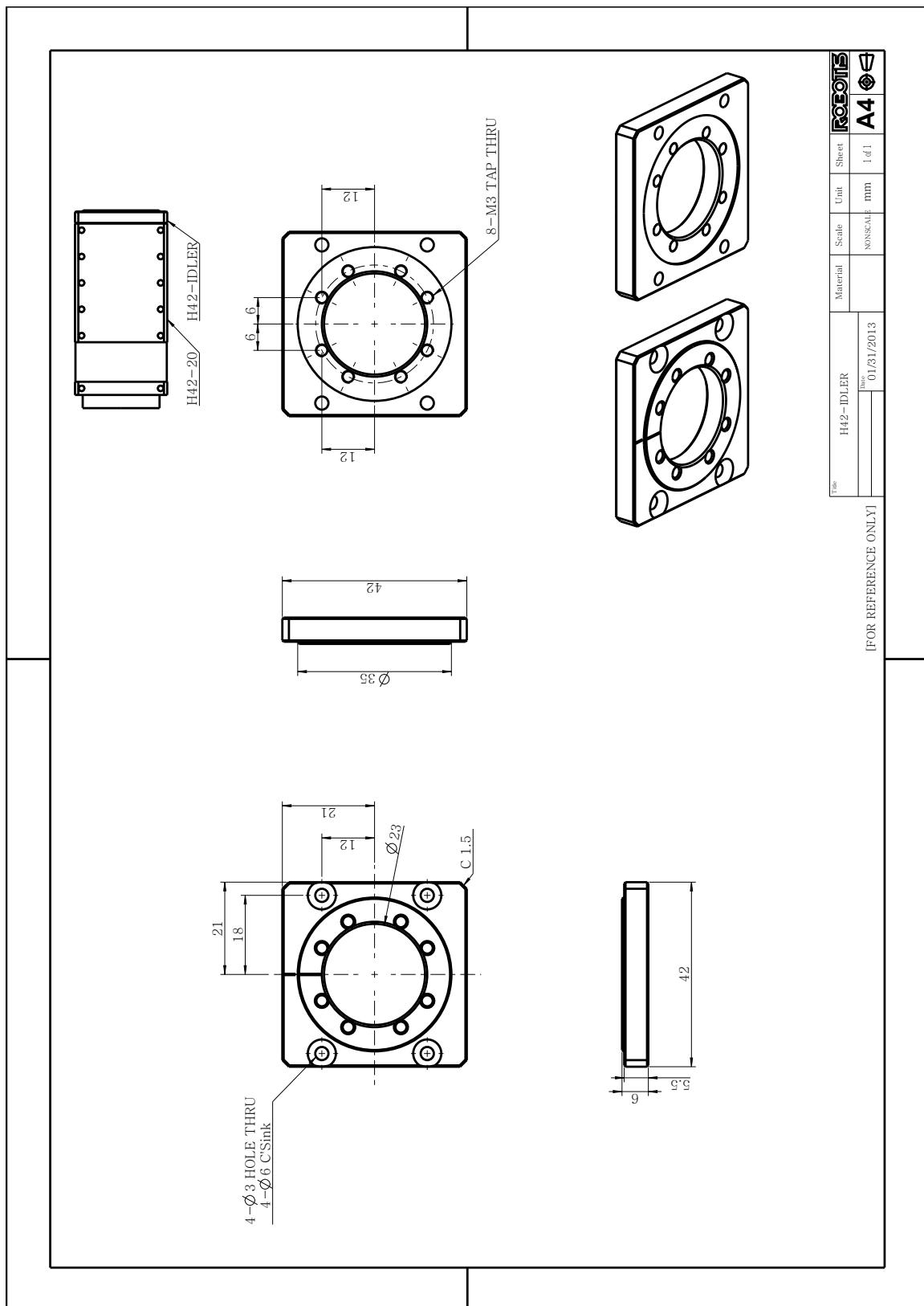
Figura 53: Desenho mecânico da estrutura ROBOTIS FRP54-I110K.



Fonte: ROBOTIS.

A.5.4 DYNAMIXEL Pro 42-20: *Idler Bearing Set* (FRP42-I110K)

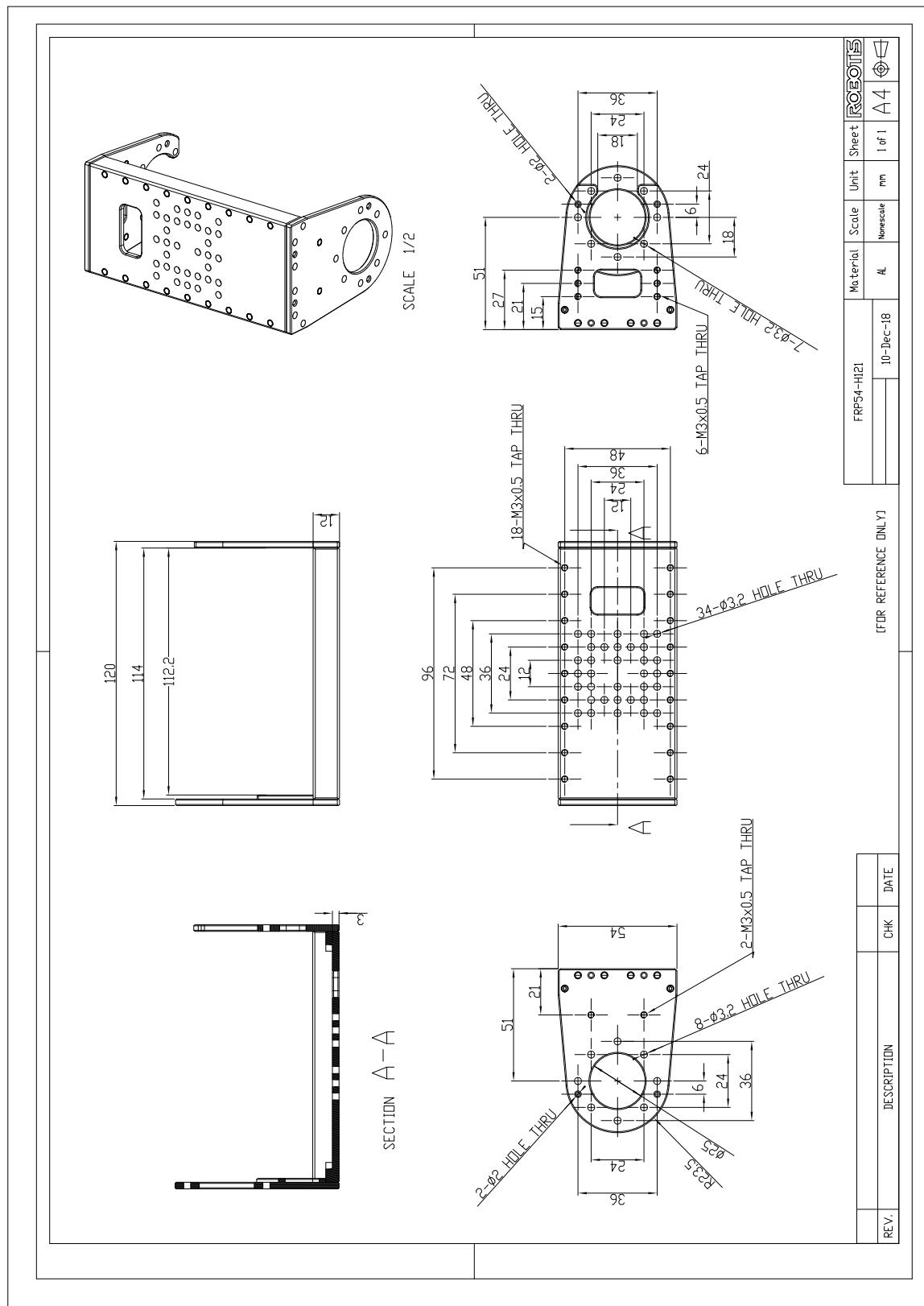
Figura 54: Desenho mecânico da estrutura ROBOTIS FRP42-I110K.



Fonte: ROBOTIS.

A.5.5 DYNAMIXEL Pro 54-100: *Single Hinge Set* (FRP54-H121K)

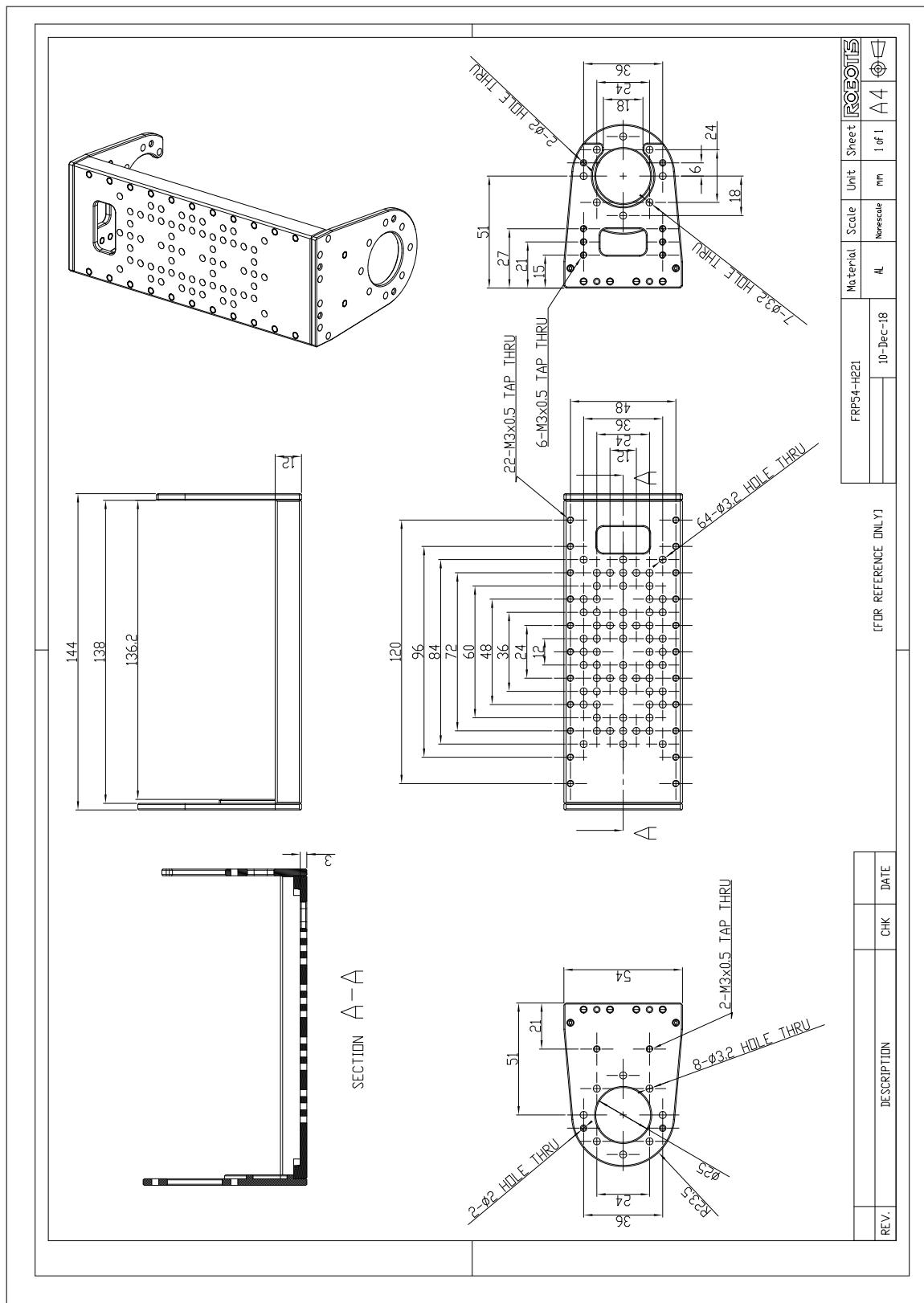
Figura 55: Desenho mecânico da estrutura ROBOTIS FRP54-H121K.



Fonte: ROBOTIS.

A.5.6 DYNAMIXEL Pro 54-200: *Single Hinge Set* (FRP54-H221K)

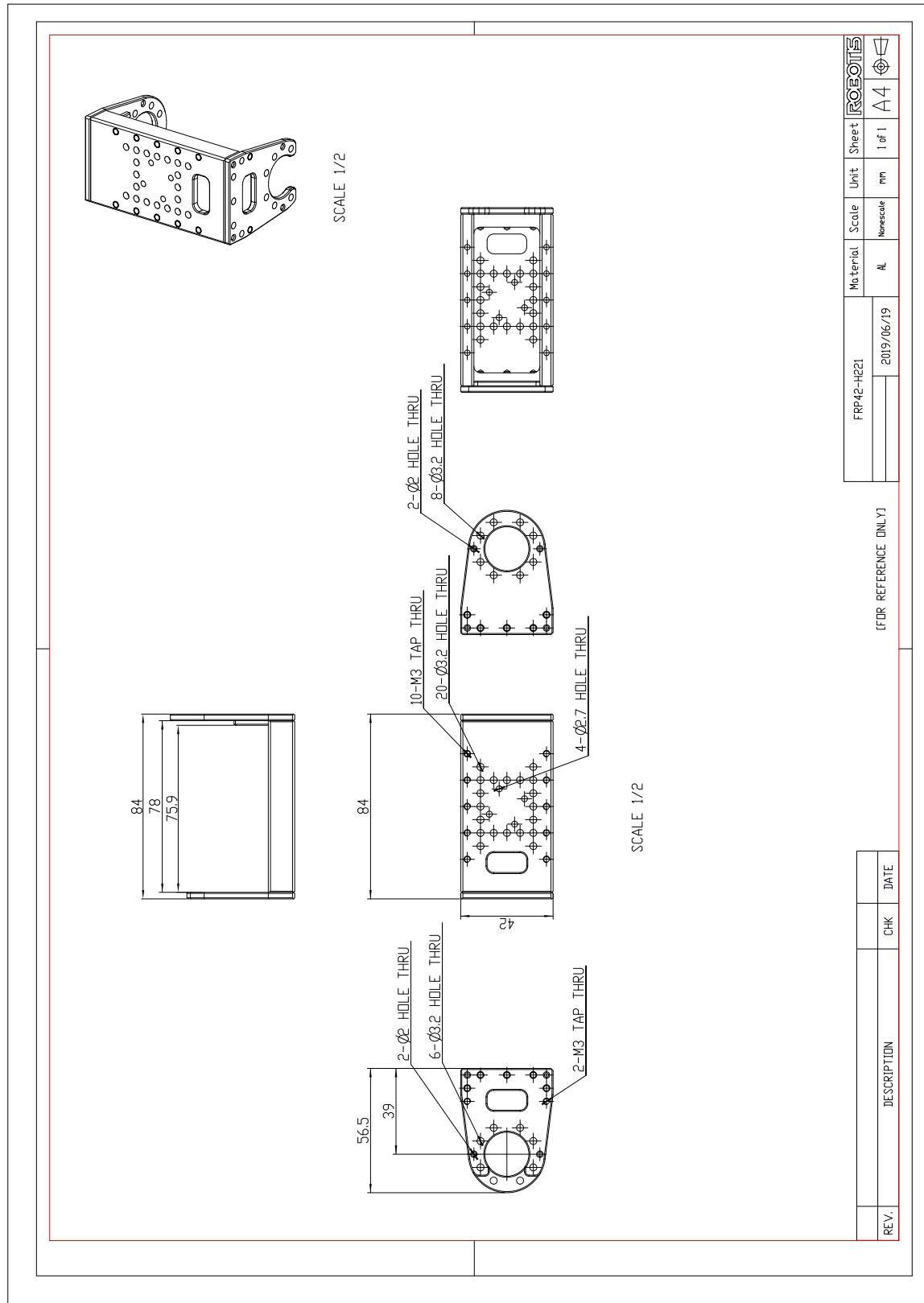
Figura 56: Desenho mecânico da estrutura ROBOTIS FRP54-H221K.



Fonte: ROBOTIS.

A.5.7 DYNAMIXEL Pro 42-20: Single Hinge Set (FRP42-H221K)

Figura 57: Desenho mecânico da estrutura ROBOTIS FRP42-H221K.



Fonte: ROBOTIS.

A.6 Conversor ROBOTIS U2D2 (RS485 - USB)

A ficha resumo contendo as principais especificações do conversor de sinais ROBOTIS U2D2 está disposta na Tabela 28. Maiores informações sobre sua utilização podem ser encontradas no *datasheet* anexado a seguir. O desenho mecânico deste conversor esta disposto na Figura 58.

Tabela 28: Especificações do conversor ROBOTIS U2D2 (RS485-USB).

Item	Especificação
Massa	9 [g]
Dimensões (L x A x P)	48 x 18 x 14.6 [mm]
Portas de comunicação com computador	USB
Portas de comunicação com periféricos	3 Pinos nível TTL 4 Pinos RS-485 4 Pinos UART
Taxa de transmissão	Máximo 6 [Mbps]

Fonte: Autores, adaptado de ROBOTIS.



1. Introduction

[Edit on GitHub](#)

U2D2

U2D2 is a small size USB communication converter that enables to control and operate DYNAMIXEL with PC.

2. How to Use

2. 1. PC to DYNAMIXEL

U2D2 can be connected to the USB port of the PC with the enclosed USB cable.

It supports both 3Pin TTL connector and 4Pin RS-485 connector to link up with various DYNAMIXEL's.

U2D2 does not supply power to DYNAMIXEL, therefore, an external power supply should provide power to DYNAMIXEL.

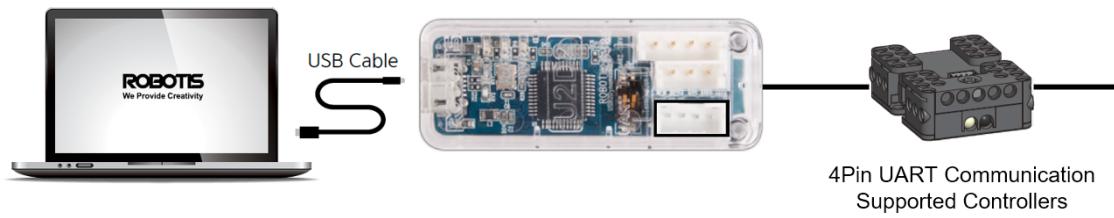
Be careful as recommended voltage for each DYNAMIXEL could be vary by model.

Please refer to e-Manual for recommended supply voltage and power connection with various power supplies.



2. 2. PC to UART

U2D2 is able to connect ROBOTIS controllers that support 4Pin UART communication such as OpenCM9.04, CM-150 and CM-200 to the PC with the enclosed USB cable.



4Pin UART Communication
Supported Controllers



ToC

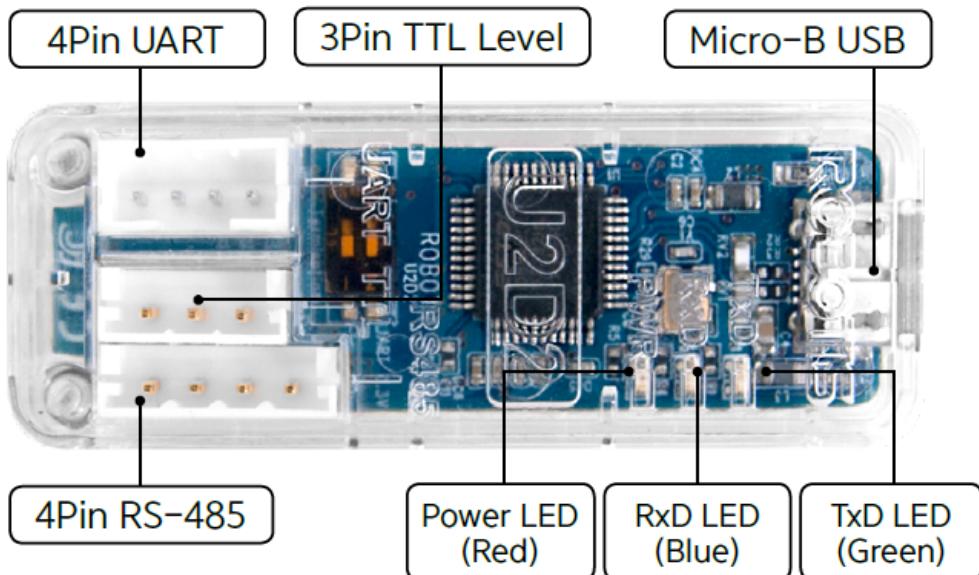
TOP

3. Specifications



Item	Description
Weight	9g
Dimensions	48mm x 18mm x 14.6mm
Ports	3Pin TTL Level(TTL Communication supported DYNAMIXEL) 4Pin RS-485(RS-485 Communication supported DYNAMIXEL) 4Pin UART(Controllers that support 4Pin UART(such as CM-150 and CM-200))
Baudrate	Maximum 6Mbps

4. Layout



Ports	Description
4Pin UART	Convert USB and UART
3Pin TTL Level	Connect to DYNAMIXEL with 3Pin TTL Level Communication
4Pin RS-485	Connect to DYNAMIXEL with 4Pin RS-485 Communication
Status LED	Display status of Power supply, TxD(Data write) and RxD(Data Read)
Micro-B USB	Connect to the PC with USB cable

5. Baudrate

Reference BPS	Actual BPS	Error(%)
9,600	9,600	0.00
57,600	57,588.4823	-0.02
115,200	115,246.0984	0.04
1,000,000	1,000,000	0.00
2,000,000	2,000,000	0.00
3,000,000	3,000,000	0.00
4,000,000	4,000,000	0.00
4,500,000	4,571,428.571	1.56
6,000,000	6,000,000	0.00

NOTE : Less than 3% of the baud rate error will not affect to the UART communication.

NOTE : For the stable communication with higher baudrate, configure USB Latency value to the lower.
[USB Latency Setting](#)

6. Supply Power

U2D2 does not supply power to DYNAMIXEL, therefore, an external power supply should provide power to DYNAMIXEL as below. Before connecting DYNAMIXEL to an external power supply, please check the recommended voltage for DYNAMIXEL.



ToC

TOP

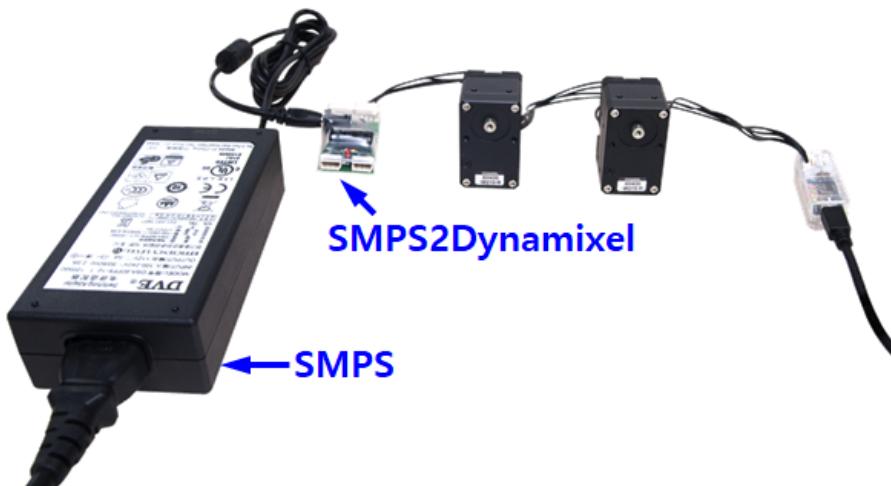
Caution for Power Supply

- For a stable power supply, it is recommended to use ROBOTIS controller or SMPS2Dynamixel.
- When connecting cable to DYNAMIXEL, please turn off the power.

6. 1. Using SMPS2Dynamixel

In order to operate DYNAMIXEL with SMPS2Dynamixel, please connect DYNAMIXEL to SMPS2Dynamixel, then connect SMPS to SMPS2Dynamixel as shown below image.

Please compare operating voltage of DYNAMIXEL with 12V 5A ROBOTIS SMPS before supplying power.



6. 2. Using ROBOTIS Controller

In order to operate DYNAMIXEL with OpenCM 485 Expansion board, please connect DYNAMIXEL to OpenCM485 Exp board, then connect battery or SMPS to the expansion board as shown below image.

Please compare operating voltage of DYNAMIXEL with battery or 12V 5A ROBOTIS SMPS before supplying power.

OpenCM485 EXP board will bypass the input power source to output power.



6. 3. Using Power Supply

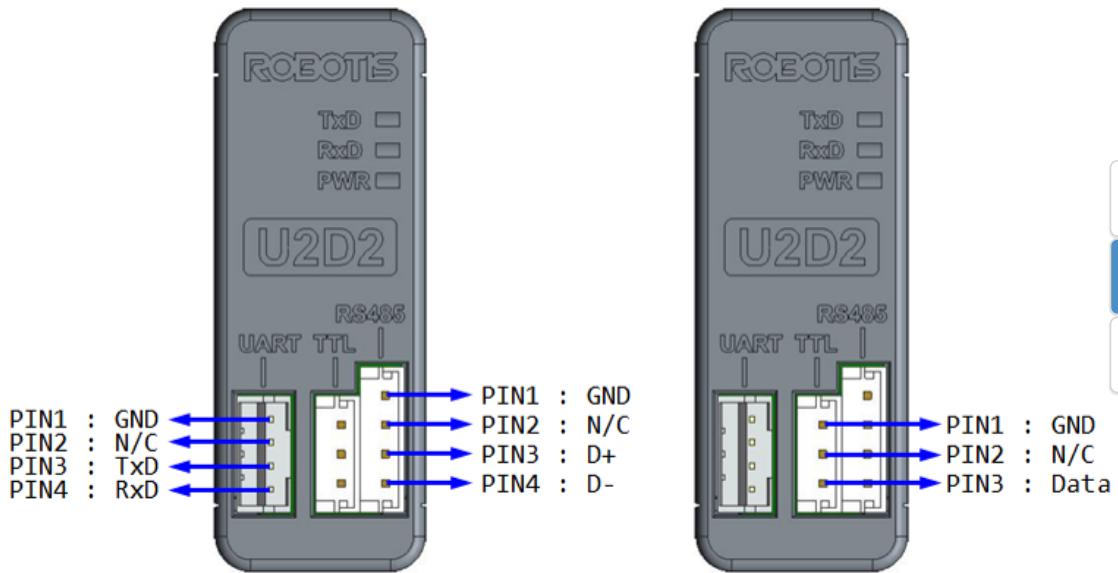
In order to operate DYNAMIXEL with Power Supply, please configure the power supply for DYNAMIXEL before connecting cable.

Please check the pinout of the cable to avoid connecting to wrong pins.

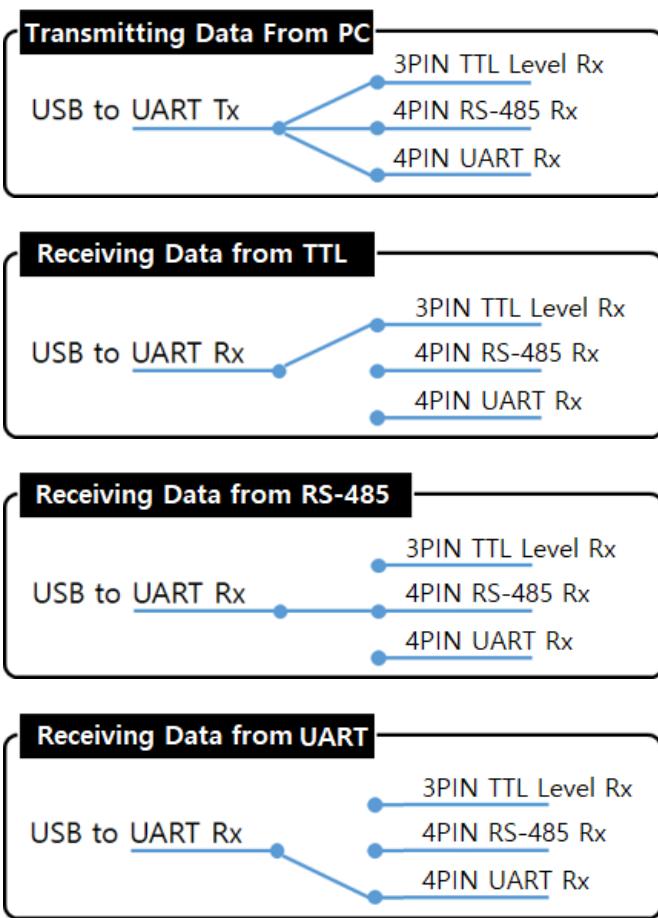
How to check whether the USB downloader(LN-101) driver is installed correctly.



7. Pinouts



8. Data Flow Control





ToC

TOP

When data is transmitted from the USB port in PC to peripheral devices(Tx), all connected device will receive an identical data.

On the other hand, when data is received from one of the peripheral devices(Rx), it will be automatically converted as USB communication and sent to the PC. During this process, received data will not be transmitted to other peripheral devices. If data from more than two peripheral devices are received, the received data can be corrupted.

Therefore, when connecting 3Pin TTL Level DYNAMIXEL and 4Pin RS-485 DYNAMIXEL at the same time, all connected DYNAMIXEL should have its unique ID in order to prevent receiving corrupt data. In addition, when using Bulk Read and Sync Read instructions that requires to receive data from multiple DYNAMIXEL's, relevant DYNAMIXEL's should be connected to the same communication channel or else the instruction will not work properly.

9. Softwares

1. RoboPlus

RoboPlus supports Windows environment to manage controllers and DYNAMIXEL's.

- R+ Task 2.0 : R+ Task 2.0 can be used when downloading Task Code to the controller.
- R+ Motion 2.0 : R+ Motion 2.0 can be used when downloading Motion file to the controller.
- R+ Manager 2.0 : R+ Manager 2.0 is used to connect controllers or DYNAMIXEL's and update firmware or test the control table..
- [ROBOTIS Homepage](#)

2. DYNMAIXEL Workbench

DYNMAIXEL Workbench supports Linux environment to test control table of DYNAMIXEL.

- [ROBOTIS GitHub](#)
- [DYNMAIXEL Workbench Manual](#)

3. DYNAMIXEL SDK

DYNAMIXEL SDK supports various programming languages such as C, C++, C#, MATLAB, LabVIEW, Python and Java in order to help developing DYNAMIXEL control software. DYNAMIXEL SDK provides a bundle of functions/methods that can be used for operating DYNAMIXEL with protocols in the form of digital packet.

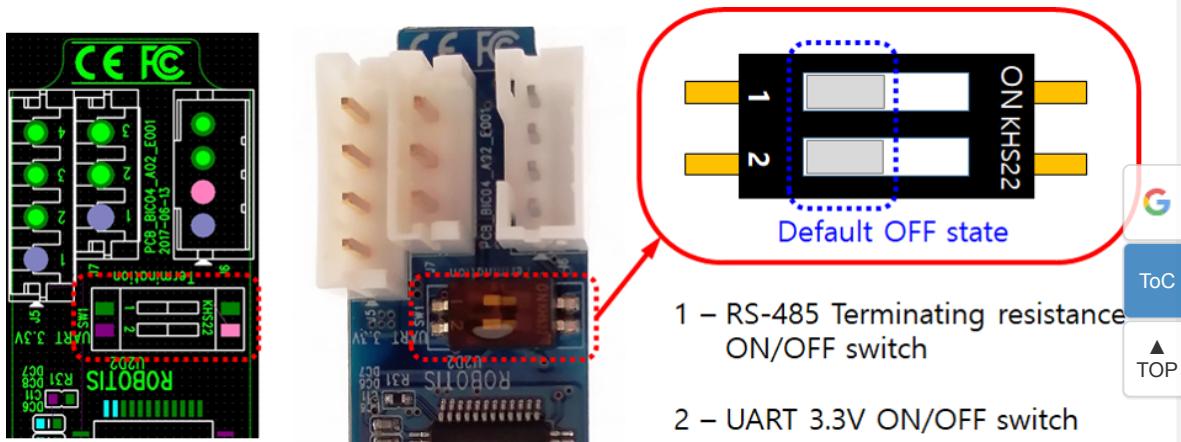
DYNAMIXEL SDK is free to download from GitHub.

- [ROBOTIS GitHub](#)
- [DYNAMIXEL SDK Manual](#)

10. Additional Features

10. 1. Internal DIP Switch

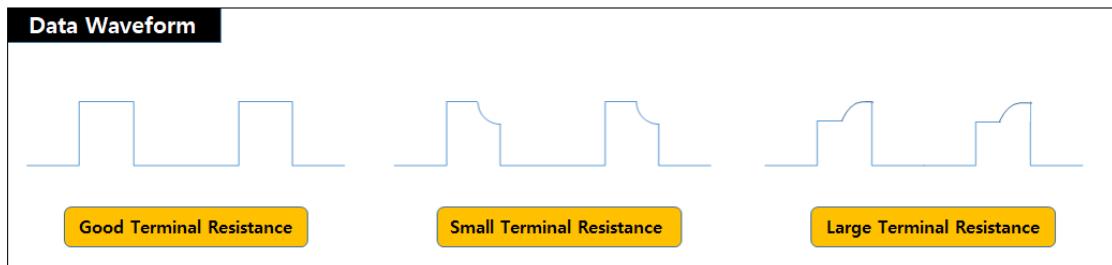
There is a two-pole switch inside of the plastic cover. This switch is set to OFF as a default. Under normal circumstances this default setting is recommended, therefore, the switch cannot be accessed without opening the case. Please open the case and set RS-485 termination resistor and UART 3.3V power switch only after thoroughly understand its usage.



10. 1. 1. RS-485 Termination Resistor(120Ω) Switch

Transmission devices are designed to have certain impedances based on design standard and signal reflection or interference can occur when impedance do not match over the transmission lines. The termination resistors at the end of transmission lines suppress such effect by adjusting impedance and allowing constant current flow. However, the termination resistor is not a mandatory, especially in a short RS-485 network with lower communication baud rate.

If communication is suffering from frequent errors, the termination resistor might be required. The regular 120Ω termination resistor will be applied on the transmission line by shifting the switch to ON. If termination resistor doesn't resolve the communication issue, the resistance value might need to be adjusted.



10. 1. 2. UART 3.3V Power Supply Switch

The peripheral device such as controllers that can be connected to the 4Pin UART usually have an external power source, therefore additional 3.3V power from the UART port #2 pin is not necessary.

However, when connecting BT-410 that does not have an external power source needs to be powered by the 4Pin UART port #2 pin. In this case, the UART 3.3V power switch need to be turned on.

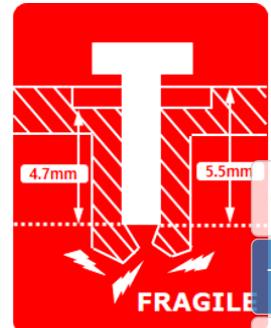
If the switch is turned on, second pin of the 4Pin UART connector will be able to supply 3.3V.

10. 2. Caution for Assembly



Mount Holes

The maximum depth of the available tap is 4.7mm. When tightening the bolts to a depth of more than 4.7mm may cause damage.



- Bolt for Mount Hole : M2 Tap bolt
- Mount Hole Depth : Valid depth 4.7mm, Depth from the surface 5.5mm

11. References

11.1. Certifications

Please inquire us for information regarding unlisted certifications.

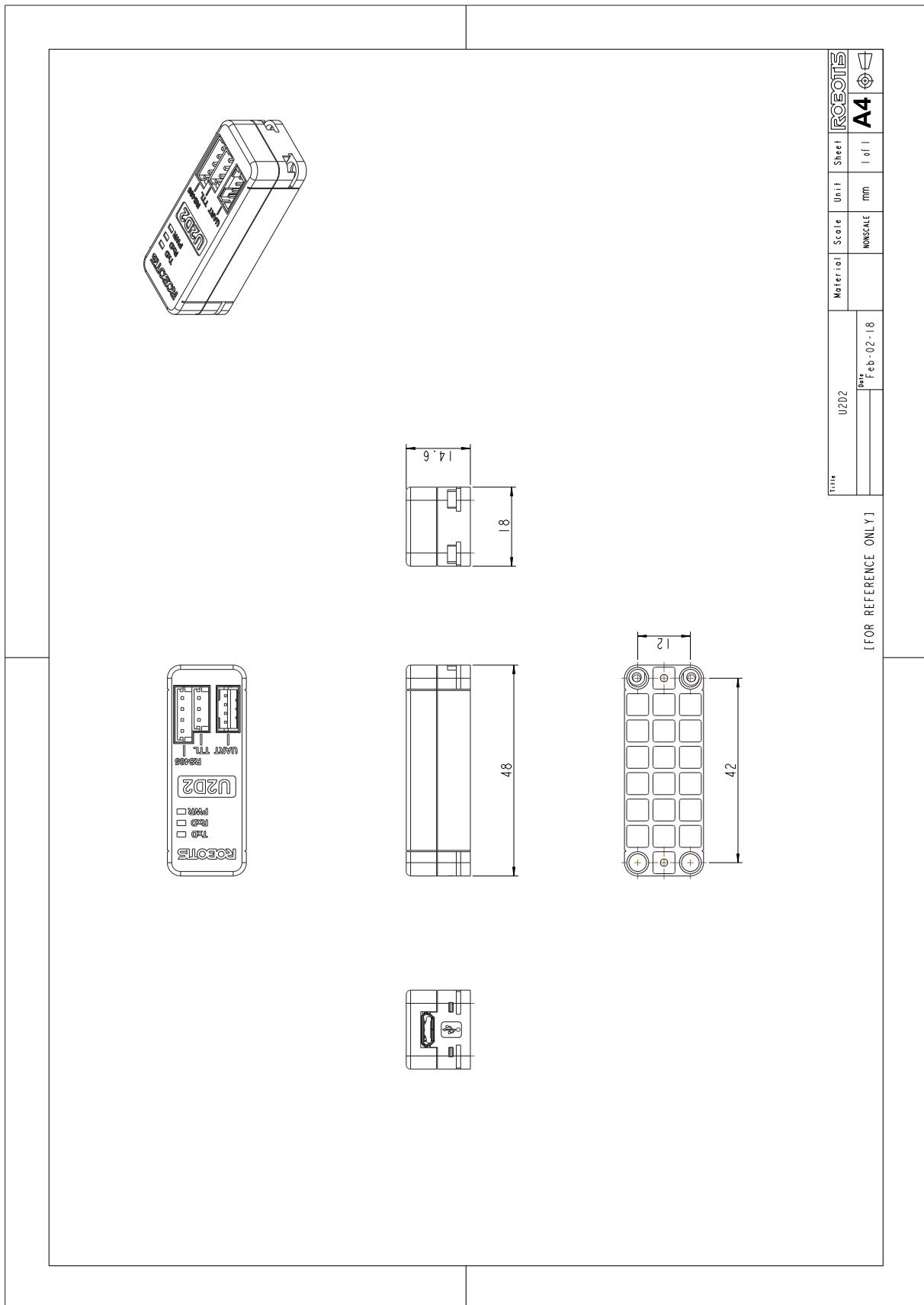
11.1.1. FCC

Note: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

WARNING

Any changes or modifications not expressly approved by the manufacturer could void the user's authority to operate the equipment.

Figura 58: Desenho mecânico do conversor ROBOTIS U2D2 (RS485 - USB).



Fonte: ROBOTIS.