



Federação das Indústrias do Estado da Bahia

CENTRO UNIVERSITÁRIO SENAI CIMATEC

Engenharia Elétrica

Trabalho de Conclusão do Curso

Desenvolvimento do robô de inspeção.

Apresentada por: Carlos Alberto Pereira
Cleber Couto Filho
Davi Costa
Ícaro Nascimento

Orientador: Prof. Marco Reis, M.Eng.

Setembro de 2018

Carlos Alberto Pereira
Cleber Couto Filho
Davi Costa
Ícaro Nascimento

Desenvolvimento do robô de inspeção.

Trabalho de Conclusão do Curso apresentada
ao , Curso de Engenharia Elétrica do Centro
Universitário SENAI CIMATEC, como requi-
sito parcial para a obtenção do título de **Ba-
charel em Engenharia.**

Área de conhecimento: Interdisciplinar

Orientador: Prof. Marco Reis, M.Eng.

Salvador
Centro Universitário SENAI CIMATEC
2016

Resumo

Este documento contempla a descrição das etapas do desenvolvimento do projeto de Theoprax de Conclusão de curso, ELIR, robô autônomo de inspeção de linhas de transmissão, atendendo aos objetivos gerais e específicos e aos requisitos estabelecidos pelo cliente, tendo em vista a necessidade do projeto num cenário tanto comercial quanto acadêmico. Durante o desenvolvimento do projeto foi necessário realizar o estudo de conceitos de robótica, bem como os softwares necessários para implementação das funcionalidades, também estudadas e definidas pelo grupo. Em paralelo ao desenvolvimento das funcionalidades diversos testes foram realizados para validação dos conceitos e verificação de erros, em etapas de testes individuais partindo para a etapa de testes integrados. Os conceitos estudados e desenvolvidos pelo grupo durante todo o projeto fazem parte de uma grande contribuição tecnológica para a área de robótica e engenharia, sendo um projeto enriquecedor tanto para a equipe envolvida no desenvolvimento quanto para as gerações futuras interessadas no desenvolvimento tecnológico em robótica.

Palavras-chave: Robô de Inspeção, Linhas de Transmissão, Navegação, Cinemática Inversa, Manipuladores

Abstract

This document contains a description of the development stages of the Theoprax end of course project, the ELIR, an autonomous robot for inspection transmission lines, meeting the general and specific objectives and the requirements established by the client, considering the need of the project in a scenario both commercial and academic. During the development of the project, it was necessary to carry out the study of robotics concepts, as well as the software required to implement the functionalities, also studied and defined by the group. In parallel with the development of the functionalities, several tests were carried out to validate the concepts and verify errors, in individual test stages, starting from the integrated testing stage. The concepts studied and developed by the group throughout the project are part of a great technological contribution to the area of robotics and engineering, being a project enriching both for the team involved in its development and for the team involved in development and for future generations interested in the technological development in robotics.

Keywords: Inspection Robot, Transmission Lines, Navigation, Inverse Kinematics, Manipulators

Sumário

1	Introdução	1
1.1	Objetivos	2
1.1.1	Objetivos Específicos	2
1.2	Justificativa	2
1.3	Organização do Trabalho de Conclusão do Curso	2
2	Fundamentação Teórica	4
2.1	Cinemática	4
2.1.1	Cinemática Direta	4
2.1.2	Cinemática Direta	4
2.2	Modelagem Cinemática de um Braço Planar	4
2.3	Desenvolvimento de Robôs	4
2.3.1	<i>Framework</i>	4
2.3.2	Simulação	4
2.3.3	Odometria	4
2.3.4	Gestão de Energia	4
2.3.5	Conceito de segurança e Integridade	4
2.3.6	Comunicação em sistemas robóticos	4
3	Metodologia	5
3.1	Especificação dos componentes	5
3.1.1	Lista de componentes	5
3.1.1.1	Servomotores	5
3.1.1.2	Placa de Gerenciamento de Energia (Gerenciamento de Energia)	6
3.1.1.3	<i>ROS</i> (Robot Operating System)	7
3.1.1.4	<i>MoveIt!</i>	8
3.1.1.5	<i>Gazebo</i>	9
3.1.1.6	<i>Visual Studio Code</i>	9
3.1.1.7	<i>PlatformIO</i>	10
3.2	Diagramas elétricos do sistema de Movimentação	10
3.2.1	Esquemas eletrônicos	11
3.3	Especificação das funcionalidades	11
3.3.1	Planejamento de Movimento	11
3.3.1.1	Dependências	12
3.3.1.2	Premissas Necessárias	13
3.3.1.3	Descrição da Funcionalidade	13
3.3.1.4	Saídas	13
3.3.2	Atuação	14
3.3.2.1	Dependências	14
3.3.2.2	Premissas Necessárias	15
3.3.2.3	Descrição da Funcionalidade	15
3.3.2.4	Saídas	15
3.3.3	Gerenciamento de Energia	16
3.3.3.1	Dependências	16

3.3.3.2	Premissas Necessárias	17
3.3.3.3	Descrição da Funcionalidade	17
3.3.3.4	Saídas	19
3.3.4	Sistema de Verificação da Integridade	19
3.3.4.1	Dependências	19
3.3.4.2	Premissas Necessárias	20
3.3.4.3	Descrição da Funcionalidade	20
3.3.4.4	Saídas	20
3.4	Simulação do sistema	20
4	Desenvolvimento e testes	23
4.1	Análise das Funcionalidades	23
4.1.1	Atuação	23
4.1.2	Planejamento de Movimento	23
4.1.3	Gerenciamento de Energia	23
4.1.4	Checação da Integridade do Sistema	23
4.2	Soluções Mecatrônicas para o sistema robótico	23
4.3	Estudo da Movimentação	23
4.4	Simulação	23
4.5	Testes de Movimentação Física	23
4.6	Integração com os subsistemas	23
4.7	Análise Preliminar	23
5	Conclusão	24
5.1	Considerações finais	24
A	QFD	25
B	Diagramas mecânicos	30
C	Diagramas eletro-eletrônicos	45
D	Logbook	46
E	Lista de componentes	69

Lista de Tabelas

3.1	Especificações Motor Robotis <i>Dynamixel</i> MX-28R	5
3.2	Especificações Motor Robotis <i>Dynamixel</i> MX-106R	6

Lista de Figuras

3.1	Motor Robotis <i>Dynamixel</i> MX-28R.	6
3.2	Motor Robotis <i>Dynamixel</i> MX-106R	6
3.3	Esquema das saídas do HUB.	10
3.4	Esquema das conexões do HUB para os motores.	11
3.5	Esquema HUB2.	11
3.6	Esquema HUB2.	11
3.7	Esquema HUB2.	12
3.8	Esquema HUB2.	12
3.9	Fluxograma de funcionamento da funcionalidade de Planejamento de Movimento	14
3.10	Fluxograma da funcionalidade Atuação	16
3.11	Fluxograma de funcionamento da funcionalidade de Gerenciamento de Energia	17
3.12	Fluxograma da rotina para checagem do sistema	21
3.13	Simulação do <i>ELIR</i> no <i>Gazebo</i>	22

Lista de Siglas

THEOPRAX

WWW World Wide Web

Lista de Simbolos

∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble
∂	Bla bla bla
Π	ble ble ble

Introdução

O Brasil apresenta uma matriz energética diferente da do resto do mundo, onde as fontes renováveis representam uma grande parte da geração da energia. Segundo a (??), em 2016, a matriz energética mundial contava com somente 14,1% da matriz energética constituída por fontes renováveis, enquanto o Brasil já apresentava 82% da sua matriz vinda de fontes renováveis, onde a geração hidrelétrica corresponde a 70% dessa geração.

A expectativa é de que a energia hidrelétrica continue sendo cada vez mais utilizada no país, devido ao crescimento previsto da demanda energética brasileira, onde segundo o (??) o consumo atual é de 405 TWh e a demanda esperada em 2030 é de 950 e 1.250 TWh/ano (??). Mesmo com a grande participação da geração hidrelétrica, somente 23% dos 260 GW totais de potencial hidrelétrico são aproveitados (??).

A concentração de demanda energética no Brasil está concentrada principalmente na região Sudeste devido a densidade populacional e elevada industrialização, isso faz com que dois terços do total da capacidade instalada estejam localizadas na Bacia do Rio Paraná que é a bacia mais próxima da região, enquanto as bacias com potencial menos aproveitado são as localizadas no norte e nordeste do país (??).

Com desenvolvimento do país é esperado um aumento na demanda de energia elétrica e consequentemente um aumento na geração de energia hidrelétrica, isso faz com que seja esperado um crescimento considerável na quantidade das linhas de transmissão, de acordo com (??), em setembro de 2018 o sistema elétrico brasileiro já atingiu 144.828 km de linhas de transmissão. Esse aumento na quantidade de linhas tende a ser amplificado pela tendência à exploração da geração de energia na região Norte, assim sendo necessário a construção de novas linhas para distribuir essa energia para as outras partes do País.

Quanto mais linhas de transmissão e maiores distâncias entre os centros geradores, maiores tendem a ser as perdas. Isso faz com que seja necessária um controle da qualidade dessa transmissão, o que se dá por meio de inspeções. A estrutura já existente apresenta precariedade em alguns aspectos, onde segundo (??) “no Brasil, há uma quantidade considerável de linhas de transmissão que já ultrapassou os 40 anos de idade. Com o envelhecimento das linhas de transmissão, a manutenção preventiva é um fator de extrema relevância para garantir o perfeito funcionamento dos sistemas.” A necessidade da constante manutenção e a alta periculosidade que os operadores são expostos faz com novas alternativas e tecnologias sejam aplicadas para a manutenção, o uso de Drones pilotados

remotamente, com câmeras e sensores já é uma realidade em alguns países do mundo. O desenvolvimento de um robô próprio para inspeção de linha configura uma dessas novas alternativas, e possibilita uma expansão dos horizontes para as tecnologias aplicadas.

1.1 *Objetivos*

O objetivo do trabalho é implementar o sistema de movimentação do robô ELIR (*Electrical Line Inspection Robot*). Onde esse sistema é complementar aos outros existentes no robô, onde o conjunto dessas soluções busca fundamentar a implementação de uma Inspeção autônoma da linha.

1.1.1 *Objetivos Específicos*

Para o desenvolvimento do sistema é necessário realizar o estudo da movimentação, gestão de energia, controle e elaboração de trajetória para sistemas robóticos. A operação na linha faz com que seja necessário a gestão de energia do robô, assim como a integração com os outros subsistemas já desenvolvidos. De forma a garantir a operação na linha, os dispositivos e ferramentas utilizadas devem estar integradas no ROS (*Robot Operating System*), onde é necessário também a integração com outros pacotes já desenvolvidos para o Robô.

1.2 *Justificativa*

1.3 *Organização do Trabalho de Conclusão do Curso*

O documento está organizado em cinco capítulos, seguindo a seguinte estrutura:

Capítulo 1 - Introdução: Faz a do âmbito no qual a pesquisa proposta está inserida. Apresenta, portanto, a problemática, objetivos e como este projeto theoprax de conclusão de curso está estruturado.

Capítulo 2 - Referencial Teórico: Apresenta a base teórica necessária para o desenvolvimento do projeto.

Capítulo 3 - Metodologia: Mostra o método adotado para o desenvolvimento do projeto, explicitando seu fluxo de atividades e entradas fornecidas pelo cliente.

Capítulo 4 - Desenvolvimento e testes: Mostra os resultados obtidos durante o desenvolvimento das fases do projeto.

Capítulo 5 - Conclusão: Apresenta as conclusões, contribuições e algumas sugestões de atividades de pesquisa a serem desenvolvidas no futuro.

Fundamentação Teórica

Nada é tão maravilhoso que não possa existir,
se admitido pelas leis da Natureza.

(Michael Faraday)

2.1 Cinemática

2.1.1 Cinemática Direta

2.1.2 Cinemática Direta

2.2 Modelagem Cinemática de um Braço Planar

2.3 Desenvolvimento de Robôs

2.3.1 Framework

2.3.2 Simulação

2.3.3 Odometria

2.3.4 Gestão de Energia

2.3.5 Conceito de segurança e Integridade

2.3.6 Comunicação em sistemas robóticos

Metodologia

Esta seção destaca o que é necessário para construção do projeto, contendo a lista de materiais, especificação dos componentes, funcionalidades, modelo esquemático de comunicação e de alimentação.

3.1 Especificação dos componentes

Serão detalhados os componentes utilizados para confecção do protótipo, sendo eles físicos ou digitais.

3.1.1 Lista de componentes

3.1.1.1 Servomotores

Os servomotores são responsáveis pela atuação do robô, realizando os movimentos das juntas dos braços e das garras, além de atuarem as rodas que deslocam o robô na linha. São utilizados os servomotores da Robotis, *Dynamixel* MX-106R e MX-28. Esses motores foram escolhidos pois apresentam drivers prontos para o *ROS*, que possibilitam uma integração mais fácil com as ferramentas de controle, apresentando bom torque, peso reduzido e fácil integração para controle conjunto.

Robotis Dynamixel MX-28R	
Peso (g)	153
Dimensões (mm)	40.2 x 65.1 x 46
Torque (N.m)	8.0 (em 11.1V), 8.4 (em 12V) e 10.0 (em 14.4V)
Temperatura de operação (°C)	-5 até +80
Tensão de operação (V)	10 até 14.8 (Tensão recomendada: 12V)
Baud rate	8000bps até 4.5Mbps
Protocolo de comunicação	RS485
Resolução	0.088°
ID	254 ID (0 até 253)
Feedback	Posição, temperatura, carga, tensão de alimentação, etc.

Tabela 3.1: Especificações Motor Robotis *Dynamixel* MX-28R

Figura 3.1: Motor Robotis *Dynamixel* MX-28R.

Fonte: (??)

Figura 3.2: Motor Robotis *Dynamixel* MX-106R

Fonte: (??)

Robotis Dynamixel MX-28R	
Peso (g)	153
Dimensões (mm)	40.2 x 65.1 x 46
Torque (N.m)	8.0 (em 11.1V), 8.4 (em 12V) e 10.0 (em 14.4V)
Temperatura de operação (°C)	-5 até +80
Tensão de operação (V)	10 até 14.8 (Tensão recomendada: 12V)
Baud rate	8000bps até 4.5Mbps
Protocolo de comunicação	RS485
Resolução	0.088°
ID	254 ID (0 até 253)
Feedback	Posição, temperatura, carga, tensão de alimentação, etc.

Tabela 3.2: Especificações Motor Robotis *Dynamixel* MX-106R

3.1.1.2 Placa de Gerenciamento de Energia (Gerenciamento de Energia)

A placa de gerenciamento de energia é responsável pela distribuição de corrente e de tensão para todos os componentes elétricos e eletrônicos do robô, além de monitorar os níveis de tensão e corrente demandados durante a operação.

Além de realizar o monitoramento do consumo em cada porta individualmente, a placa possui um sistema de proteção, cortando a alimentação em casos de surto de corrente. A placa funciona através da alimentação de 14.4 Volts provenientes da placa multiplexadora, responsável por transmitir a carga de duas baterias *Li-Íon* de 14.4 Volts e 6Ah.

Na placa de *Gerenciamento de Energia* existem conversores DC/DC responsáveis por fazer a conversão dos níveis de 14.4 Volts para 12 Volts em cada porta de saída da placa. As 7 portas de saída possuem sensores de tensão e corrente individuais, feitos com amplificadores de instrumentação INA226. Existem duas portas de saída que disponibilizam tensões em valores menores, de 5 Volts.

O monitoramento dos níveis de tensão e corrente se dá principalmente pela inteligência do sistema, um firmware embarcado em um microcontrolador Atmega32u4, responsável por fazer as leituras dos parâmetros em cada uma das portas, verificando se os seus níveis estão de acordo com os limites configurados, cortando a alimentação via relés digitais caso esses valores sejam ultrapassados.

3.1.1.3 ROS (*Robot Operating System*)

Framework, no ambiente de programação, é um espaço onde compatibiliza códigos comuns a fim de otimizar o trabalho e tempo, muito utilizado na área de desenvolvimento. A abstração de hardware, códigos de baixo nível, drivers de sensores, simuladores, etc - são as grandes vantagens de se utilizar essa aplicação, podendo assim, fazer com que o desenvolvedor foque somente nas soluções de problemas específicos do seu projeto.

Foi utilizado durante todo o desenvolvimento do *ELIR* o *framework ROS*, já que reúne uma série de ferramentas importantes para o desenvolvimento de um robô. “O Sistema Operacional de Robótica é um flexível framework para escrita de softwares para robótica. É uma coleção de ferramentas, bibliotecas e convenções que serve para simplificar a tarefa de criar complexos e robustos comportamentos de robôs diante a uma variedade de plataformas (??).”

A grosso modo, cada câmera, motor ou periférico ligado ao *ROS*, estão associados ao um nó. A comunicação entre os nós se dá através de tópicos ou de serviços, a diferença é que o primeiro a informação é trocada de forma constante com certo intervalo de tempo e o segundo somente quando solicitado.

Assim é feita toda a comunicação e interligação entre os periféricos no *ROS*, forma simples de integração dos componentes.

3.1.1.4 *MoveIt!*

Durante a inspeção de linha, é necessário que o robô realize a ultrapassem dos diferentes tipos de obstáculos que existem nas linhas de transmissão. O *MoveIt!* é um ferramenta que funciona de forma integrada com o *ROS*, apresentando funcionalidades de planejamento de movimento, percepção 3D, controle, manipulação e cinemática inversa.

A cinemática é o estudo do movimento, no âmbito da robótica designa o estudo do controle da posição do robô no espaço. Esse controle pode representar do robô como um todo, sua posição geográfica, ou controle de alguma parte sua em específico, como seu braço e a posição relativa desse braço e o robô. A cinemática direta é o cálculo onde se encontra a posição do robô para determinado valor de velocidade ou posição de suas juntas. Analogamente, na cinemática inversa, se encontra os valores de velocidade ou posição das juntas para uma posição no espaço, onde essa posição é denominada *end-effector*, geralmente sendo definido como a parte do robô que interage com o mundo, como por exemplo a garra no caso de manipuladores. O cálculo da cinemática inversa envolve equações complexas e retorna diferentes soluções, assim sendo necessário encontrar a solução que melhor atende às diretrizes do movimento, o *MoveIt!* já realiza esse cálculo e fornece uma trajetória otimizada baseada em parâmetros do usuário.

Outra das suas vantagens é utilizar o modelo *URDF* do robô. *URDF* é uma sigla para Unified Robot Description Format, e designa um arquivo com extensão *.urdf* e sintaxe em XML. É um dos tipos de modelos mais utilizados na robótica atual, sendo escolhido pois apresenta uma sintaxe simples e dinâmica, proporcionando conversões em outros formatos de forma fácil. Define o robô como um conjunto de partes, chamadas de *links* onde a união entre essas partes é uma junta. Onde cada *link* vai ter um *link* pai, que é determinado pela definição da junta, assim o modelo apresenta uma estrutura em árvore, onde todos os *links* vão ter um pai até chegar ao *link* da raiz, essa definição é importante pois a partir disso é feita a cinemática inversa do robô. Um erro no modelo *URDF* acarreta em uma mudança no comando que é mandado para o robô original.

Durante o desenvolvimento do projeto, foi possível realizar a integração do *MoveIt!* com o robô real, sendo possível realizar movimentos físicos utilizando a ferramenta de visualização para posicionamento de end-effector pelo usuário. Porém, ao tentar se enviar um comando com o valor de posição para o end-effector se mover, o programa falhou em encontrar soluções para a cinemática. Mesmo se utilizando os diversos solucionadores providos e enviando valores possíveis de se calcular, o programa sempre estava falhando em encontrar uma solução.

O *MoveIt!* é designado para funcionar com robôs de 6 graus de liberdade, onde cada grau de liberdade indica uma coordenada que o end-effector pode se mover e um

dos 3 eixos de referência (x,y,z) que ele pode girar. A grande quantidade de graus de liberdade faz com que os solucionadores utilizem formas de cálculos complexas, assim robôs que possuem menos que 6 graus de liberdade precisam ser compatibilizados, já que a solução leva em consideração todas as direções e giros. O Elir possui somente 2 graus de liberdade, e as soluções que antes funcionavam não estavam se utilizando especificamente da solução de cinemática inversa provida pelo *MoveIt!*. Assim, decidiu-se realizar o cálculo da cinemática inversa por meio de um código python, que utiliza a equação de cinemática inversa específica para o tipo de braço do robô e fornece os ângulos de junta necessários para o end-effector especificado.

Com o ângulo de junta em mãos, é feita a integração do robô com a ferramenta, de forma que o planejamento de movimento ocorre, só que com o software recebendo um ângulo desejado. Conhecendo os possíveis valores de end-effector, o que pode ser encontrado pela ferramenta de visualização, é possível realizar o movimento no robô enviando somente um comando de coordenadas.

Implementar o controle de movimento nessa plataforma possibilita uma série de implementações futuras que aumentam a autonomia do robô e robustez do sistema, como odometria, ferramentas de percepção e mapeamento.

3.1.1.5 *Gazebo*

O software *Gazebo* é software utilizado para simulação de robôs. Tem uma licença de uso livre e apresenta diversas formas de integração com o *ROS*, sendo o principal simulador utilizado em conjunto com essa plataforma, possibilitando a inserção de plugins como câmeras e sonares, que se comunicam com o *ROS* de forma fiel a dispositivos reais.

Nele é possível simular também o ambiente do robô, definindo parâmetros físicos como aceleração da gravidade e vento. Oferece suporte para a inserção de modelos 3D de softwares CAD, assim podendo ser inseridas diversas estruturas que já foram modeladas para outros propósitos no software.

3.1.1.6 *Visual Studio Code*

Para que todas as funcionalidades do robô sejam configuradas e desenvolvidas de forma correta a nível de software, é necessário o desenvolvimento de diversos códigos em diferentes linguagens para a configuração de aspectos específicos do projeto.

Foi utilizada durante o desenvolvimento do projeto a ferramenta *Visual Studio Code*. Trata-se de um editor de códigos open source desenvolvido pela *Microsoft* em 2015, sendo possível desenvolver códigos em diversas linguagens como C++, C, Python entre outros. Durante todo o desenvolvimento do *ELIR* a ferramenta foi utilizada para o desenvolvimento de arquivos nas extensões .py, .yaml, .launch e .urdf.

Por ter uma interface simples e amigável, o *VSCode* mostrou-se uma ferramenta extremamente útil para a escrita e desenvolvimento de códigos durante todas as fases do projeto.

3.1.1.7 PlatformIO

Na interface do *VS Code* existem diversas extensões que podem ser instaladas para adicionar novas funcionalidades na plataforma.

Uma das extensões utilizadas foi o *PlatformIO*, um ecossistema desenvolvido especificamente para o desenvolvimento de códigos e firmwares em plataformas microcontroladas, sendo extremamente versátil, tendo suporte para diversas plataformas como STM, MSP430, Arduino entre outras, tornando desnecessário o uso de uma IDE específica para se realizar a configuração e desenvolvimento de firmwares durante o projeto.

A necessidade de se utilizar essa extensão se deu principalmente pela necessidade de se embarcar o firmware de Gerenciamento de Energia na placa. O *PlatformIO* possui as funcionalidades de debug e gravação, sendo assim, todos os procedimentos necessários para atualizar o firmware são atendidos na extensão.

3.2 Diagramas elétricos do sistema de Movimentação

Devido à quantidade de motores presentes no robô e a forma com que eles estão distribuídos na estrutura, foram desenvolvidos dois modelos de hubs para a conexão dos motores na rede.

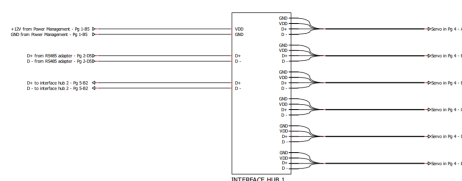


Figura 3.3: Esquema das saídas do HUB.

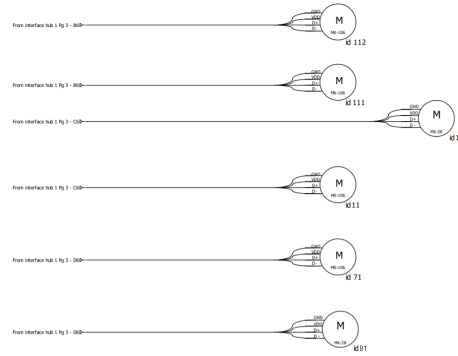


Figura 3.4: Esquema das conexões do HUB para os motores.

3.2.1 Esquemas eletrônicos

Nas unidades de tração do robô, os hubs contam com um conector de alimentação, um para a entrada de dados e 6 de saída para os motores.

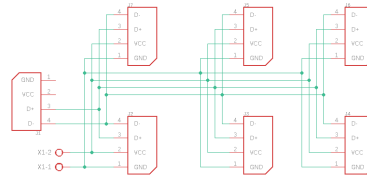


Figura 3.5: Esquema HUB2.

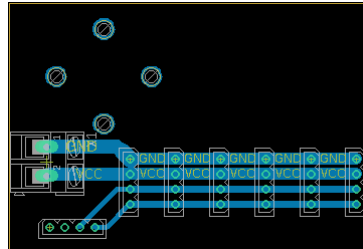


Figura 3.6: Esquema HUB2.

Já na unidade central, o hub além de conectar os seis motores ali presentes, também é responsável pela conexão dos hubs das unidades de tração e do conversor rs485 que está ligado à *NUC*.

3.3 Especificação das funcionalidades

3.3.1 Planejamento de Movimento

A funcionalidade de *Planejamento de Movimento* é responsável por realizar o planejamento da trajetória do Robô, utilizando o software *MoveIt!* que realiza o cálculo da cinemática inversa para encontrar a melhor forma de ultrapassar os obstáculos.

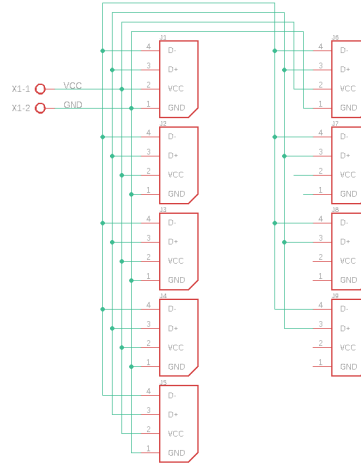


Figura 3.7: Esquema HUB2.

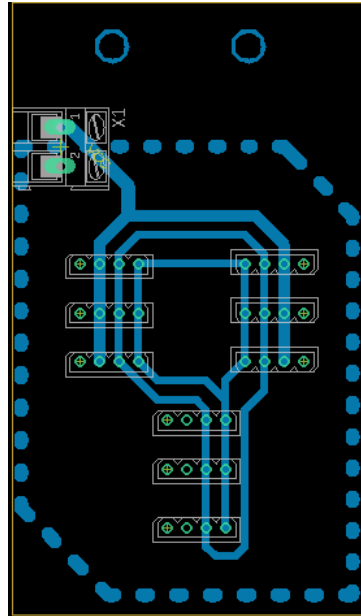


Figura 3.8: Esquema HUB2.

3.3.1.1 Dependências

O software *MoveIT!* pode utilizar o modelo matemático da cinemática inversa do robô ou um arquivo do tipo *URDF*. O nome *URDF* é uma sigla para *Unified Robot Description Format*, esse arquivo é uma especificação em XML utilizada para descrever robôs. Modelos em *URDF* apresentam uma simplicidade na descrição do robô, e para o caso do Robô *Elir*, utilizar o modelo *URDF* possibilitará uma aproximação fiel ao modelo real do robô, assim para o cálculo da cinemática inversa será utilizado o seu modelo *URDF* e não o seu modelo matemático.

3.3.1.2 *Premissas Necessárias*

Para o correto funcionamento dessa funcionalidade as seguintes premissas são necessárias:

- A configuração dos limites de giro das juntas do robô estarão compatíveis com os comandos enviados
- O modelo *URDF* do robô estará adequado com o modelo físico
- O pacote gerado pelo *MoveIt! Setup Assistant* estará configurado adequadamente

3.3.1.3 *Descrição da Funcionalidade*

A movimentação do robô na linha acontecerá por movimentos de translação e transposição de obstáculos. A translação na linha será feita por controladores de torque nas rodas do robô, enquanto a transposição do obstáculos utilizará o *MoveIT!*. Por meio da ferramenta *MoveIt! Setup Assistant*, se utiliza o modelo do robô para criar um pacote do *ROS* com os principais arquivos pelo *MoveIT!*. A configuração correta do *MoveIT!* possibilita que se utilizem as funções da sua biblioteca para o cálculo da trajetória, levando em consideração também obstáculos no caminho.

O *MoveIT!* fornece uma *user interface* que recebe o end-effector, a nomenclatura atribuída ao node feito em python que recebe o *end-effector* é *moveit_commander*. O *node* responsável por fazer a integração da user interface com os parâmetros recebidos pelo *ROS Parameter Server* com o *end-effector* para fazer os cálculos é denominado *move_group*. O *node move_group* também pode receber parâmetros como leituras dos sensores do robô e nuvens de pontos.

3.3.1.4 *Saídas*

Por meio da compatibilização do *MoveIt!* com o *ROS*, a saída dessa funcionalidade são os comandos de velocidade, esforço e posição para cada junta do robô.

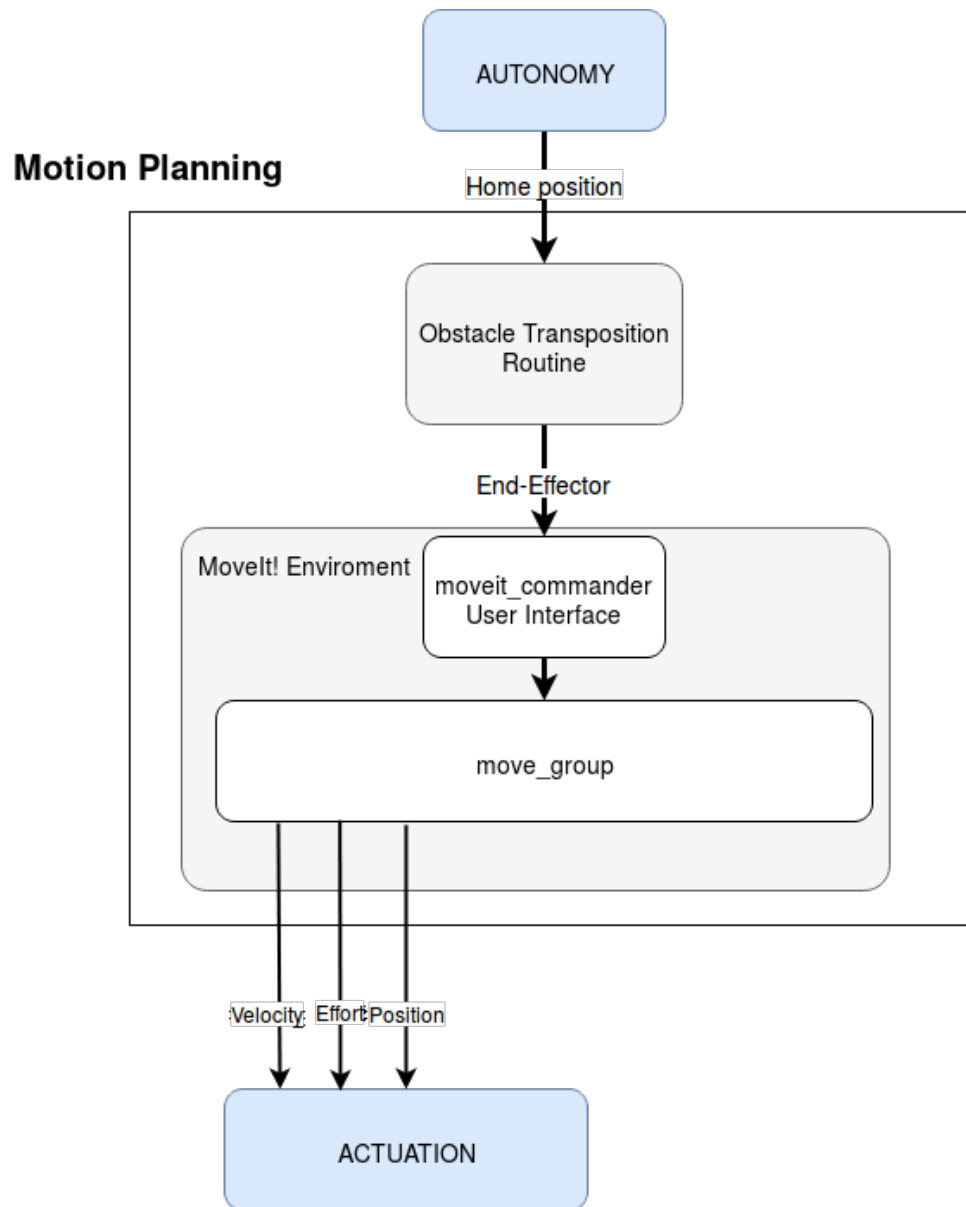


Figura 3.9: Fluxograma de funcionamento da funcionalidade de Planejamento de Movimento

Fonte: Própria

3.3.2 Atuação

A funcionalidade de Atuação tem como objetivo mover a estrutura física do robô, possibilitando o controle dos movimentos das juntas, garras e unidades de tração.

3.3.2.1 Dependências

Essa funcionalidade depende das funcionalidades de *Gerenciamento de Energia* e *Planejamento de Movimento*. O *Gerenciamento de Energia* será responsável por fazer

alimentação dos motores, possibilitando controlar a corrente máxima fornecida para cada grupo. A dependência em relação à funcionalidade de *Planejamento de Movimento* está atrelada principalmente com o software *MoveIt!*, que ao receber um *end-effector*, realiza o cálculo de trajetória e envia os comandos de velocidade, esforço e posição para os controladores das juntas, garras e unidades de tração.

3.3.2.2 *Premissas Necessárias*

Para o correto funcionamento desse módulo, devem ser consideradas as seguintes premissas:

- Os motores devem estar configurados de acordo com o padrão de ID determinado pela equipe, fazendo parte da mesma malha de controle;
- Os controladores das juntas, garras e unidades devem estar configurados de acordo com os comandos que serão recebidos pelo *MoveIt!*;
- Os 3 grupos de motores estarão em malhas de alimentação de 12V individuais.

3.3.2.3 *Descrição da Funcionalidade*

O *ROS* disponibiliza uma série de drivers para compatibilização dos motores dynamixel, possibilitando a criação de controladores específicos no seu ambiente. Serão criados os controladores referentes as juntas e unidades de tração do robô. Os controladores receberão comandos de *velocity* e *position* do *MoveIt!* junto com os comandos para movimentar o robô na linha. Após os comandos serem recebidos pelos controladores, eles serão enviados para o *hardware* do robô, de acordo do padrão de comunicação dos motores, por meio de comunicação serial.

3.3.2.4 *Saídas*

A saída desta funcionalidade é o movimento da estrutura física do robô, que estará de acordo com o planejamento de trajetória do *MoveIt!* e com as instruções para operação na linha

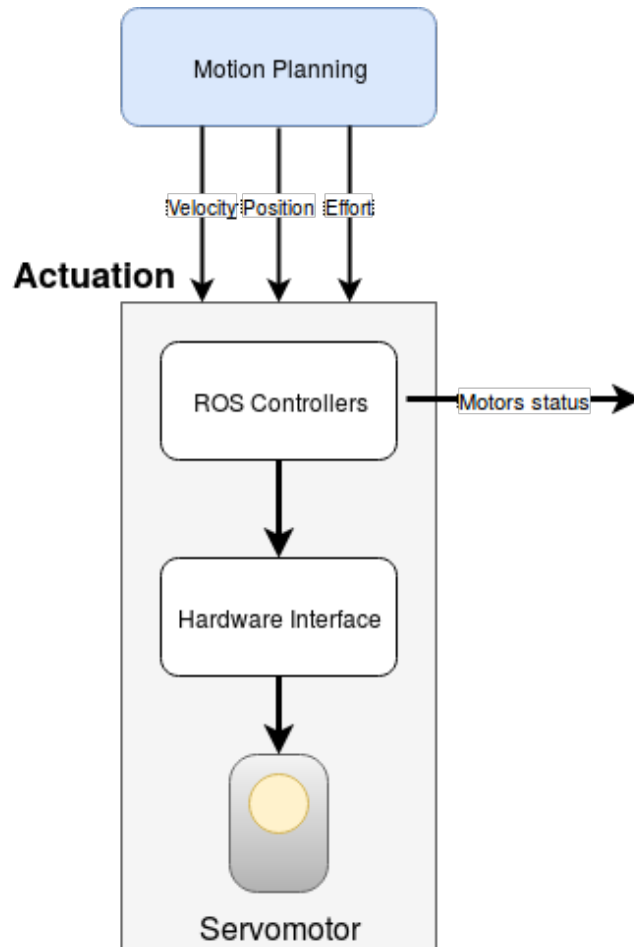


Figura 3.10: Fluxograma da funcionalidade Atuação

Fonte: Própria

3.3.3 Gerenciamento de Energia

A funcionalidade de *Gerenciamento de Energia* é responsável pelo gerenciamento de alimentação elétrica dos componentes elétricos e eletrônicos do robô, através da integração das funcionalidades de seu firmware no ambiente *ROS*.

3.3.3.1 Dependências

Essa funcionalidade depende da comunicação serial por meio da biblioteca **rosserial** para compatibilização e integração das funcionalidades de firmware no ambiente *ROS*. Operacionalização e customização do firmware embarcado no hardware de acordo com as necessidades do projeto e da alimentação fornecida pela placa multiplexadora, por meio de baterias Li-Ion NH2054 14.4 volts.

3.3.3.2 Premissas Necessárias

Para o correto funcionamento desse módulo de *Gerenciamento de Energia*, devem ser consideradas as seguintes premissas:

- A placa multiplexadora estará conectada diretamente ao módulo de *Gerenciamento de Energia*
- Todos os dispositivos estarão conectados nas suas respectivas entradas
- A placa deverá ser alimentada por 2 baterias de 14.4 Volts e 3 Amperes, totalizando um fornecimento de até 6 Amperes
- A placa estará conectada diretamente na NUC, por meio de uma USB

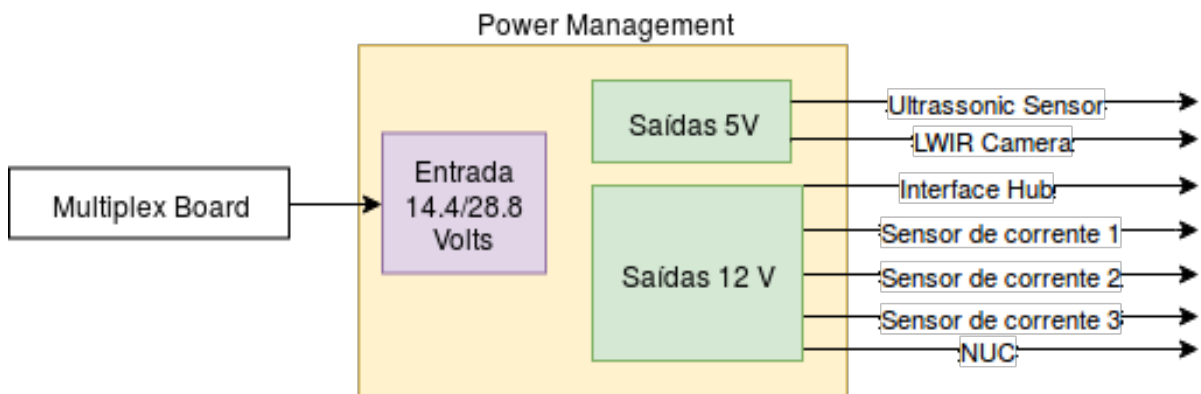


Figura 3.11: Fluxograma de funcionamento da funcionalidade de Gerenciamento de Energia

Fonte: Própria

3.3.3.3 Descrição da Funcionalidade

A funcionalidade *Gerenciamento de Energia* é responsável por fornecer diversos recursos em sua totalidade. O hardware utilizado (placa *Zord*) possui um sensor de corrente e tensão para cada porta de saída, permitindo o monitoramento individual de cada uma das portas. O microcontrolador utilizado Atmega32U4 possui um firmware embarcado onde toda a compatibilização com o ambiente *ROS* é realizada, o que torna essencial o uso do pacote *rosserial* para o seu funcionamento. O firmware é responsável pela ativação dos relés digitais em caso de surtos de corrente para proteção dos dispositivos elétricos.

Os limites nos valores de corrente funcionam justamente para que o hardware interrompa a alimentação em um possível caso de surto de corrente. Todos os aspectos importantes para o funcionamento do sistema de gerenciamento de energia pode ser configurado tanto via *ROS*, por meio das configurações dos serviços, ou por meio do firmware,

modificando os parâmetros do tempo de duração dos picos de corrente. Os principais serviços e tópicos criados pela funcionalidade Gerenciamento de Energia no *ROS* são:

- *Tópicos*
 - *PowerOutput* Este tópico disponibiliza os valores de tensão e corrente de todas as portas da placa em tempo real.
 - *TakeStatus* Disponibiliza o estado de cada porta da placa, informando os eventos ocorridos e a porcentagem de corrente demandada durante a ocorrência do evento.
- *Serviços*
 - *GetCurrentLimitCommand* Este comando retorna o valor de corrente máxima de saída configurado para a porta escolhida
 - *SetCurrentLimitCommand* Este comando realiza a configuração do valor máximo de corrente de saída em uma determinada porta
 - *PowerOnOffCommand* Este comando realiza a ação de ativação ou desligamento de uma determinada porta.

A placa de Gerenciamento de energia irá receber a carga das baterias pela placa multiplexadora e irá realiza o controle de alimentação dos seguintes componentes:

- Grupos de servo motores
- Grupo de sensores de corrente
- NUC
- Interface HUB
- Câmera LWIR
- Sensor ultrassônico
- Phidgets
- STM Nucleo
- Módulo GPS

3.3.3.4 *Saídas*

A funcionalidade irá disponibilizar a energia para o robô e as seguintes estruturas no ambiente *ROS*:

- Tópicos com informações de tensão e corrente nas portas
- Tópico para aviso de sobre-corrente
- Tópico para informar disponibilidade da placa
- Serviços para ler e configurar limite de corrente das portas
- Serviço para ligar ou desligar energia em uma porta

3.3.4 *Sistema de Verificação da Integridade*

É a funcionalidade responsável por checar a integridade do sistema antes do início da missão, verificando os subsistemas e suas variáveis.

3.3.4.1 *Dependências*

A funcionalidade receberá informações dos seguintes componentes

- Sensor de Temperatura
- Servomotores
- Câmera IR
- Câmera Stéreo
- IMU
- Sensor de Proximidade
- Placa de Power Management
- Sonar
- Baterias

Todas as informações serão enviadas por meio do ambiente *ROS*, na forma de *Services* ou *Publishers*.

3.3.4.2 *Premissas Necessárias*

As premissas necessárias para o funcionamento dessa funcionalidade são:

- Os subsistemas do robô irão disponibilizar o seu status no ambiente *ROS* por meio de tópicos ou serviços
- A checagem fará parte do planejamento de missão

3.3.4.3 *Descrição da Funcionalidade*

A checagem da integridade do sistema é uma funcionalidade essencial para garantir o sucesso da missão e preservar a integridade do robô. O *ROS* facilita essa comunicação entre os subsistemas, possibilitando que seja criada uma rotina de checagem antes de cada missão.

Será disponibilizado no sistema uma rotina para iniciar a missão. Ao receber o comando para início de missão, os sistemas serão checados sequencialmente, utilizando estrutura de *Services* e *Publishers* do *ROS*. Caso algum sistema apresente falha, a missão não se iniciará e o erro será mostrado no *terminal* e registrado no arquivo de `log`. Se todos os sistemas estiverem em funcionamento, se iniciará a missão. O fluxograma da funcionalidade está ilustrado na figura 3.12.

3.3.4.4 *Saídas*

No início da rotina de inspeção, a funcionalidade será responsável por enviar o sinal inicia a missão. Caso todos os sistemas checados estejam funcionando, a inspeção ocorrerá normalmente, se algum sistema apresentar defeitos, o defeito será mostrado no *terminal*, registrado em `log` e a missão será abortada.

3.4 *Simulação do sistema*

A simulação de sistemas robóticos consistem em um dos pilares para o desenvolvimento de projetos. Com a simulação é possível testar aplicações sem a necessidade de adquirir componentes, os membros da equipe de projeto conseguem trabalhar de forma simultânea no robô enquanto o protótipo físico fica reservado para testes específicos.

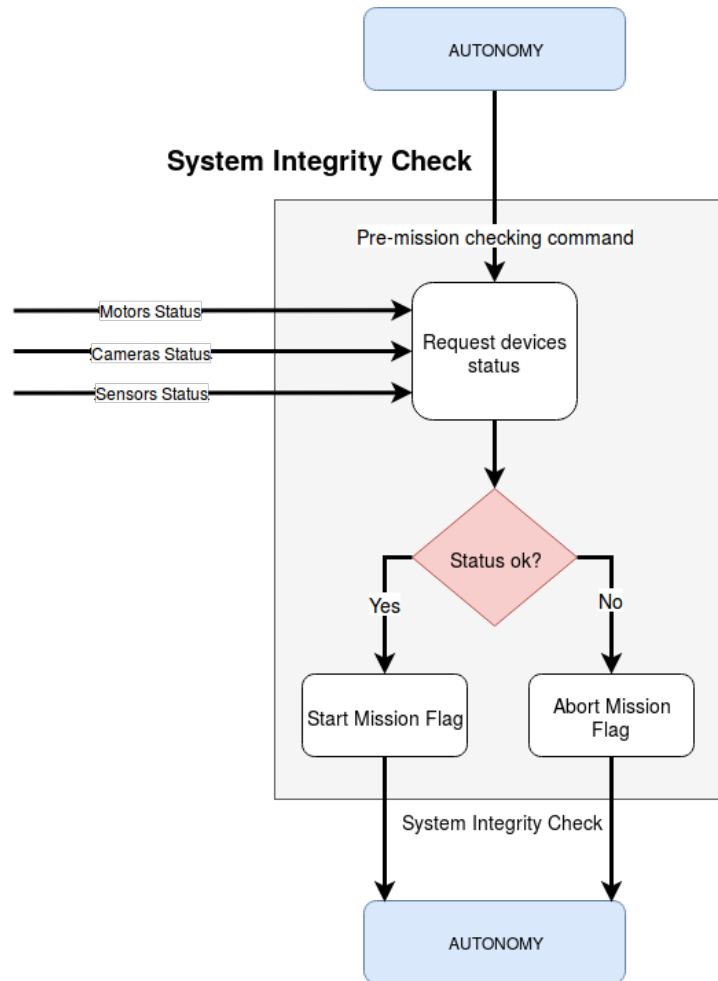


Figura 3.12: Fluxograma da rotina para checagem do sistema

Fonte: Própria

O *ROS* oferece ferramentas de visualização já integradas no seu sistema, o *Rviz*, que possibilita o usuário visualizar os modelos do robô e também administrar plugins, como de mapeamento e planejamento de movimento, que é o caso do *MoveIt!*.

Para a simulação do robô no ambiente aberto, é utilizado o software *Gazebo*. A integração entre *ROS* e *Gazebo* consegue fazer com que o modelo *URDF*, por mais que não seja o nativo do *Gazebo*, seja aceito na simulação. Parâmetros do mundo podem ser ajustados e a integração de plugins como câmeras e sensores faz com que a simulação consiga ser utilizada em diferentes estudos. Algoritmos de imagem podem ser testados com os plugins de câmera já implementados, proporcionando um auxílio para demonstrar conceitos e teorias de funcionamentos

A simulação fornecida possui os controladores de juntas já implementados, fazendo com que testes de códigos de movimentação e testes de controles já pudessem ser previamente testados, poupando riscos de dano ao protótipo e possibilitando trabalho simultâneo.

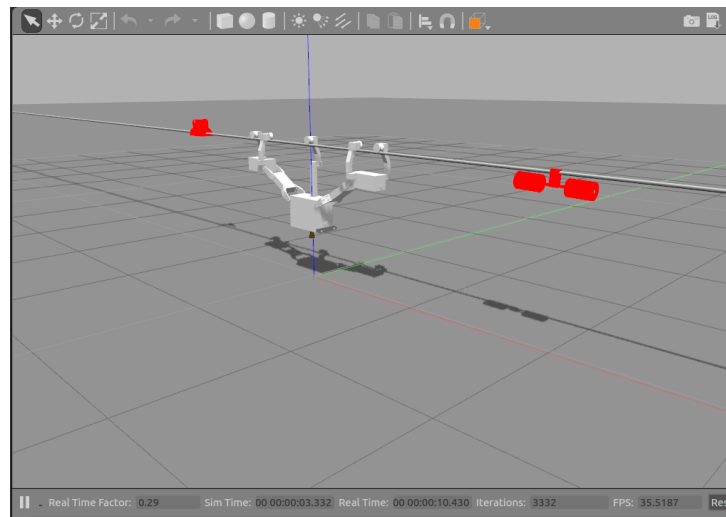


Figura 3.13: Simulação do *ELIR* no *Gazebo*.

Desenvolvimento e testes

4.1 Análise das Funcionalidades

4.1.1 Atuação

4.1.2 Planejamento de Movimento

4.1.3 Gerenciamento de Energia

4.1.4 Checagem da Integridade do Sistema

4.2 Soluções Mecatrônicas para o sistema robótico

4.3 Estudo da Movimentação

4.4 Simulação

4.5 Testes de Movimentação Física

4.6 Integração com os subsistemas

4.7 Análise Preliminar

Conclusão

O resultado do projeto alcançou as expectativas. Os problemas que aconteceram conseguiram ser contornados e o tempo gasto para sua solução conseguiu se adequar ao esperado pelo cronograma das tarefas. O material produzido atende às demandas do cliente e os pacotes produzidos foram organizados buscando facilitar o uso por terceiros.

5.1 *Considerações finais*

A gestão do projeto do robô como um todo, fez com que o resultado produzido alcançasse as expectativas. O nível de desenvolvimento aumentou progressivamente de forma que o projeto foi conduzido, adicionando paulatinamente diversos conhecimentos específicos que não seriam vistos normalmente durante a graduação, mas também fortalecendo conhecimentos já formados.

O que foi produzido para o projeto estará disponível para futuras consultas, o que impulsiona o desenvolvimento de projetos semelhantes. Novos estudos podem ser iniciados como trabalhos de graduação, ou pós-graduação, realizando provas de conceito e abrindo oportunidades para novas tecnologias.

O início do projeto se deu de forma lenta, por apresentar uma área do conhecimento nova para maior parte da equipe. A robótica necessita da integração de diversas áreas diferentes, e para a engenharia elétrica, o conhecimento de diversas camadas de abstração. Com a experiência e o desenvolver das atividades, os conhecimentos adquiridos possibilitaram atividades em paralelo e o aumento da versatilidade dos integrantes.

A experiência como um todo foi muito enriquecedora, adicionando conhecimentos que serão necessários no futuro e proporcionando um crescimento para todos os participantes.

QFD

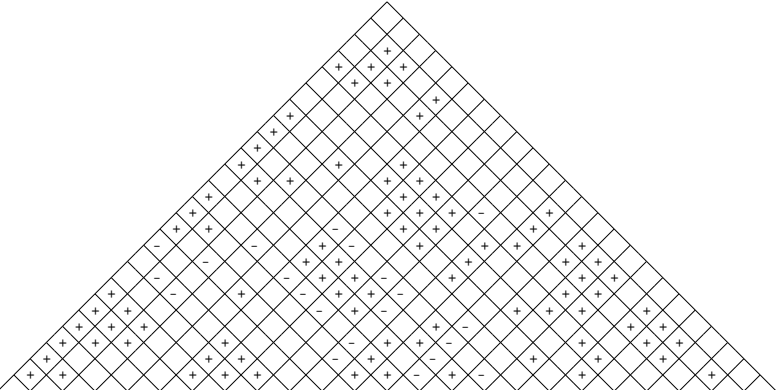
Correlations	
Positive	+
Negative	-
No Correlation	

Relationships	
Strong	●
Moderate	○
Weak	▽

Direction of Improvement	
Maximise	▲
Target	◇
Minimise	▼

Row #	Weight Chart	Relative Weight	Customer Importance	Maximum Relationship
1	<div><div></div></div>	12%	9	9
2	<div><div></div></div>	9%	7	9
3	<div><div></div></div>	6%	5	9
4	<div><div></div></div>	4%	3	9
5	<div><div></div></div>	9%	7	9
6	<div><div></div></div>	12%	9	9
7	<div><div></div></div>	6%	5	9
8	<div><div></div></div>	12%	9	9
9	<div><div></div></div>	12%	9	9
10	<div><div></div></div>	12%	9	9
11	<div><div></div></div>	8%	6	9

Customer Requirements (Explicit and Implicit)		Direction of Improvement																							
Technical Requirements		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Confabilidade		●	●	●	●	▲	▲	▲	▲	▲	◇	◇	▲	▲	▲	▲	▲	▲	◇	◇	▲	◇	◇	◇	◇
Monitoramento de consumo energético		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Sistema regulador de tensão para alimentar a bateria		●	○	▽	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Georeferenciamento		●	●	●	▽	▽	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Interface Teve e simplisten. para processar e transmitir dados visuais (UI)		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Câmera espectro visível		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Câmera IR		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
resolução das câmeras		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Luminosidade		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
massa do robot		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
armazenamento de dados da missão		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
desempenho de deslocamento		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
velocidade de deslocamento		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Classificação de obstáculos		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
detectar e ultrapassar obstáculos		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
autonomia de potência		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
medição da taxa do cabo ao obstáculo ao chão		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
registro de ocorrências		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
testar obstáculos em visada direta		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
testar obstáculos dos obstáculos		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
garantir e monitorar o posicionamento das garras na posição de trabalho		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Interface operacional + Framework		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Back-end		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●



Customer Competitive Assessment											
0 1 2 3 4 5											Row #
5	0	0	0	0	0	0	●				1
4	0	0	0	0	0	0	●				2
5	0	0	0	0	0	0	●				3
4	0	0	0	0	0	0	●				4
5	0	0	0	0	0	0	●				5
5	0	0	0	0	0	0	●				6
5	0	0	0	0	0	0	●				7
4	0	0	0	0	0	0	●				8
5	0	0	0	0	0	0	●				9
5	0	0	0	0	0	0	●				10
5	0	0	0	0	0	0	●				11
							●~PiRo	○~Fu	○~LineScout	●~Expliner	
							●~Phillips	—ELIR			

- PIRo

○-Fu

○-LineScout

●-Expliner

●-Phillips

—ELIR
- PIRo

○-Fu

○-LineScout

●-Expliner

●-Phillips

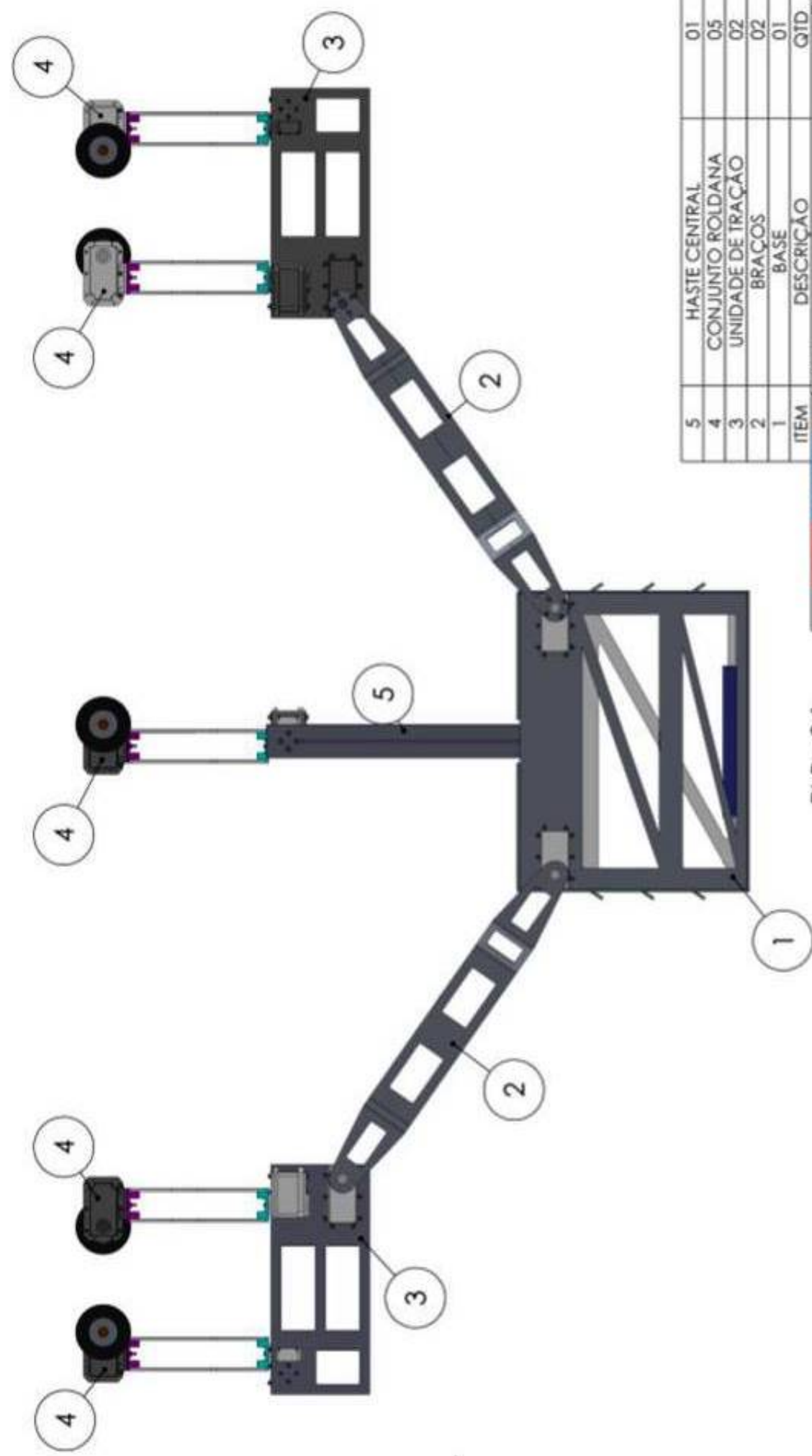
—ELIR

Correlations	
Positive	+
Negative	-
No Correlation	
Relationships	
Strong	●
Moderate	○
Weak	▽
Direction of Improvement	
Maximize	▲
Target	◇
Minimize	▼

					Column #	1	2	3	4	5	6	7	8	9	10	11	12
					Direction of Improvement	◇	◇	◇	▲	▲	▲	▲	◇	◇	▲	▲	12
Row #	Weight Chart	Relative Weight	Technical Importance	Maximum Relationship	Functionalities	SYSTEM INTEGRITY CHECK	POWER MANAGEMENT	ACQUISITION	LOCALIZATION	MOTION PLANNING	ACTUATION	CLASSIFICATION	DETECTION	USER INTERFACE	AUTONOMY	SIMULATION	
1	<div></div>	9%	762	9	Confiabilidade	●	●	●	●	●	●	●	●	▽	●	○	
2	<div></div>	3%	219	9	Monitoramento de consumo energético	▽	●	●	○		●				○		
3	<div></div>	2%	196	9	Sistema regulador de tensão para acoplar à bateria		●	○	○		○						
4	<div></div>	4%	373	9	Georreferenciamento				●				○	○			
5	<div></div>	5%	445	9	Interface leve e simplista, com poucos elementos visuais [UI]			▽						●		○	
6	<div></div>	6%	491	9	câmera espectro visível			●				○	○	●	○	○	
7	<div></div>	4%	369	9	câmera IR			●					●	●	○	○	
8	<div></div>	5%	385	9	resolução das câmeras			●				○	●	●			
9	<div></div>	3%	274	9	luminosidade							●	●		▽		
10	<div></div>	4%	312	9	massa do robot		○				●					○	
11	<div></div>	5%	450	9	Armazenamento de dados da missão		▽	●	○	○		●	●	●	●		
12	<div></div>	5%	390	9	desempenho de deslocamento	▽	○			▽	●		●		●		
13	<div></div>	4%	332	9	velocidade de deslocamento	●	○			●	●				○		
14	<div></div>	2%	208	9	Classificação de obstáculos			●		○	▽	●	●	○	●	○	
15	<div></div>	3%	235	9	detectar e ultrapassar obstáculos			▽		●	▽	●	●	○	●	○	
16	<div></div>	5%	418	9	autonomia de potência	○	●	○	○	●	○	○	○		●		
17	<div></div>	2%	132	9	medição da cota do cabo ao obstáculo ao chão	▽		●					●	○		○	
18	<div></div>	6%	508	9	registro de ocorrências	●	●	●	●	●	●	●	●	●	●		
19	<div></div>	2%	204	9	Exibir obstáculos em visada direta			○				○	●	●		▽	
20	<div></div>	2%	169	9	Exibir distâncias dos obstáculos			○				○	●	●		▽	
21	<div></div>	4%	377	9	garantir e monitorar o posicionamento das garras na linha de transmissão	●		●		▽					○	○	
22	<div></div>	6%	515	9	Sistema operacional + Framework	○	○	○	○	●	○	○	○	●	●	○	
23	<div></div>	9%	738	9	Back-end	○	▽	○	▽	●	○	●	●	●	●	○	
24		0%	0														
					Target	verificar a integridade física do sistema antes e durante a missão	gerenciar o fornecimento de energia	realizar a comunicação e a aquisição dos dados	informar o posicionamento e orientação do sistema quando solicitado	realizar o planejamento da trajetória	mover a estrutura física e transposição dos obstáculos	classificar os objetos encontrados na linha	identificar pontos quentes e objetos na linha e na faixa de serviço	disponibilizar de forma simplificada os dados mais relevantes	fornecer parâmetro de confiabilidade e da estratégia a ser adotada	realizar a simulação da missão antes do início da mesma	
					Max Relationship	9	9	9	9	9	9	9	9	9	9	3	
					Functional Importance Rating	277.1	291.2	500	246	403.6	338.1	413.1	574	494.9	513.4	166.1	0
					Relative Weight	7%	7%	12%	6%	10%	8%	10%	14%	12%	12%	4%	0%
					Weight Chart	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>	
					Column #	1	2	3	4	5	6	20	21	22	23	24	25

Diagramas mecânicos

REV	DESCRIPTION	DRAW	DATE
0	Drawing elaboration	Juliana Sartori	07/09/2017



ITEM	DESCRICO	QTD
5	HASTE CENTRAL	01
4	CONJUNTO ROLDANA	05
3	UNIDADE DE TRACAO	02
2	BRAÇOS	02
1	BASE	01

PI-Ro 2.1

Material: Alumínio

Author: Juliana Sartori

Approved: Marco Reis

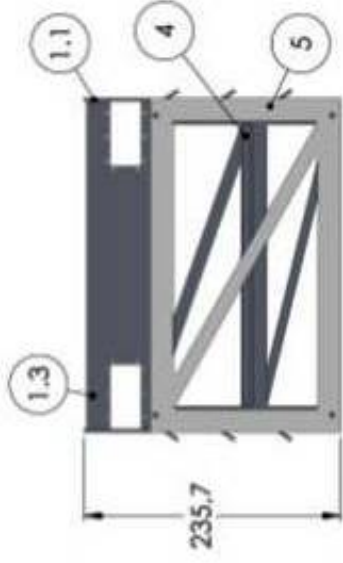
Date: 07/09/2017

Scale: 1:4

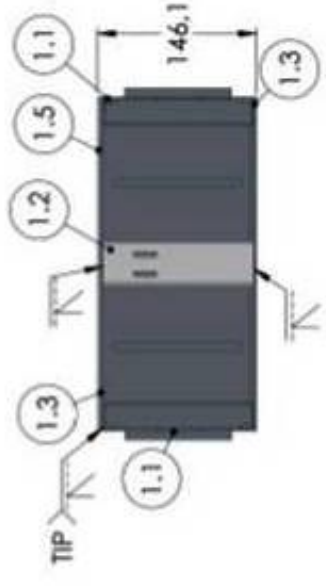
Quantity: 01

Page: 1 of 13

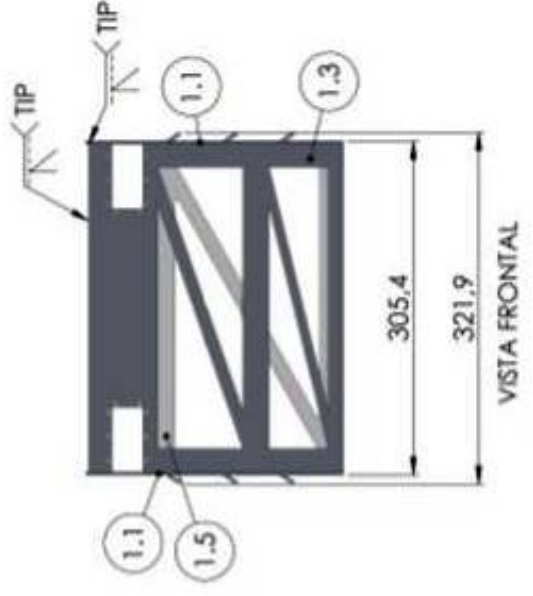
PI-Ro 2.1
VISTA FRONTAL



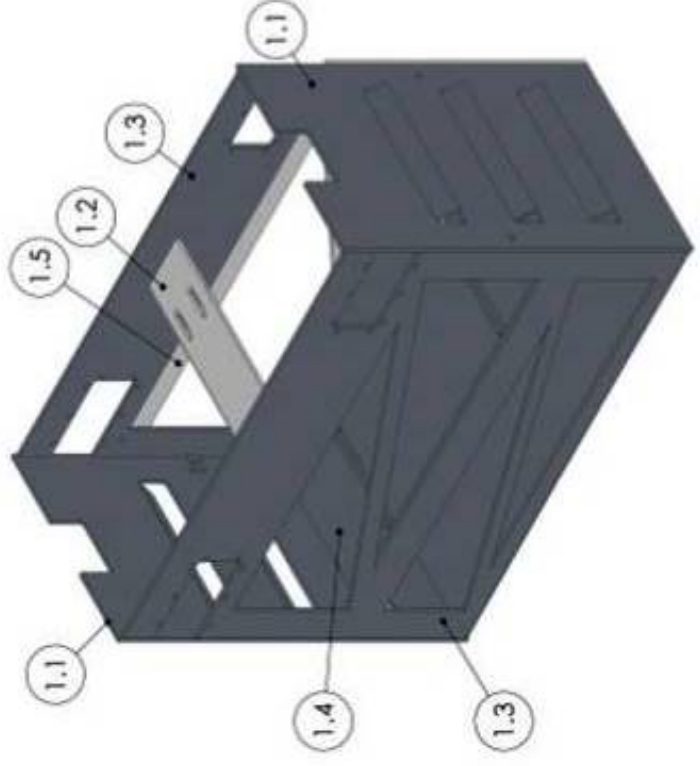
VISTA TRASEIRA



VISTA SUPERIOR



VISTA FRONTAL



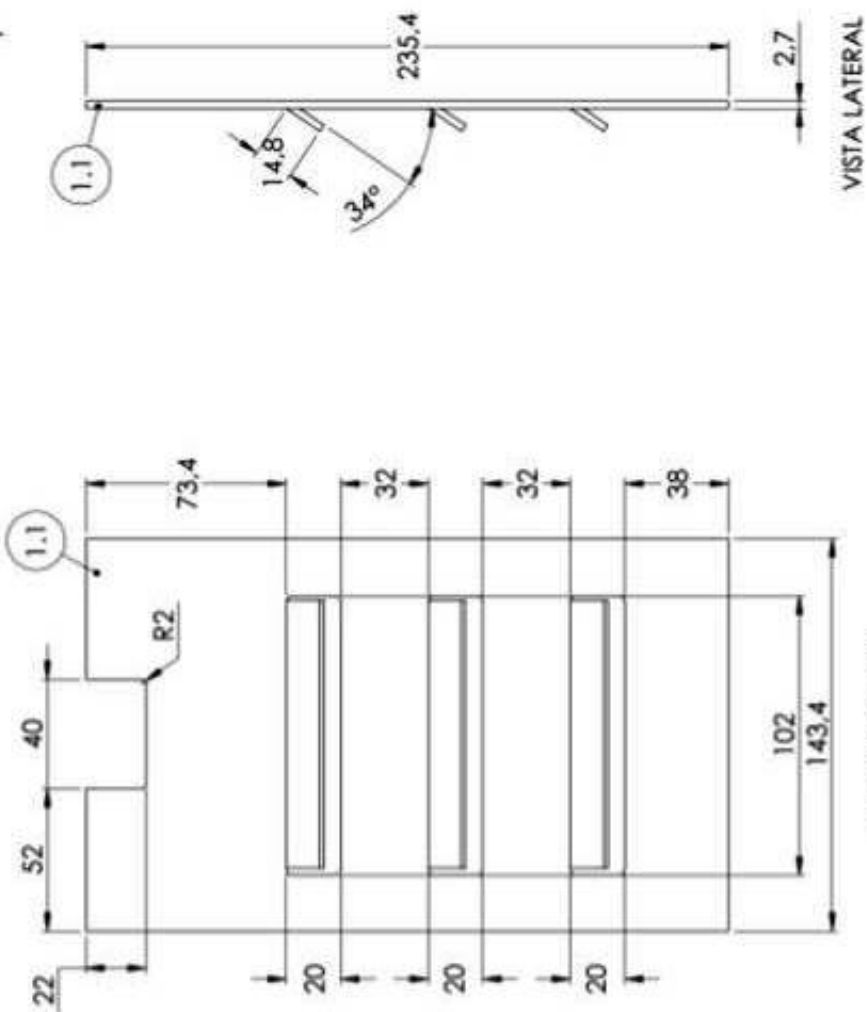
VISTA ISOMÉTRICA

ITEM	DESCRIÇÃO	QTD
1.5	CHAPA 05: 305,4 x 175 x 2,7mm	01
1.4	CHAPA 04: 148 x 254 x 2,7mm	01
1.3	CHAPA 03: 608,8 x 300 x 2,7mm	01
1.2	CHAPA 01: 138 x 47 x 2,7 mm	01
1.1	CHAPA 01: 143,4 x 235,4 x 2,7 mm	02

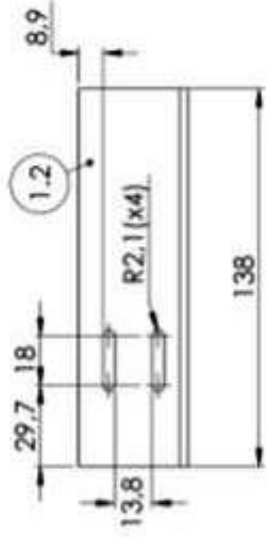
		BASE - ITEM 01	
Project: PI-Ro 2.1		Material: Alumínio	
Draw: Juliana Sartori		Thermal: N/A	
Agree: Marco Reis		Page: 2/13	
Date: 07/09/2017		Scale: 1:2	
Revisão: 0		Sheet: X	
Notes: 1. All measurements in mm 2. Unless otherwise specified 3. Default Tolerancing		Notes: 1. All measurements in mm 2. Unless otherwise specified 3. Default Tolerancing	

REV	DESCRIPTION	DRAW	DATE
0	Drawing elaboration	Juliana Sartori	07/09/2017

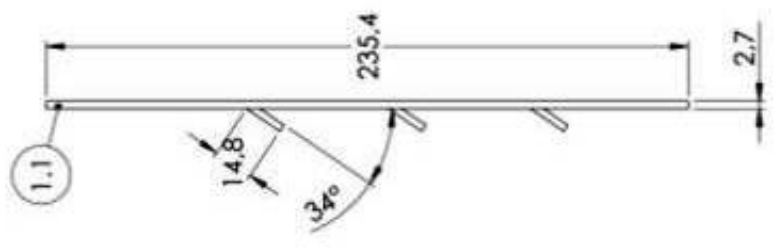
REV	DESCRIPTION	DRAW	DATE
0	Drawing elaboration	Juliana Sartori	07/09/2017



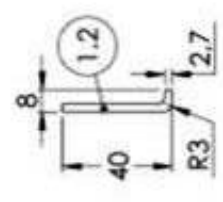
VISTA FRONTAL



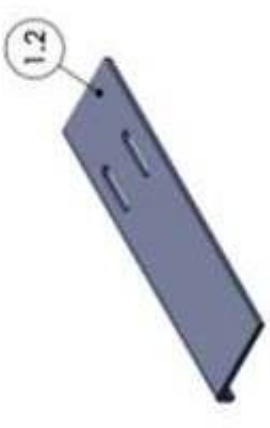
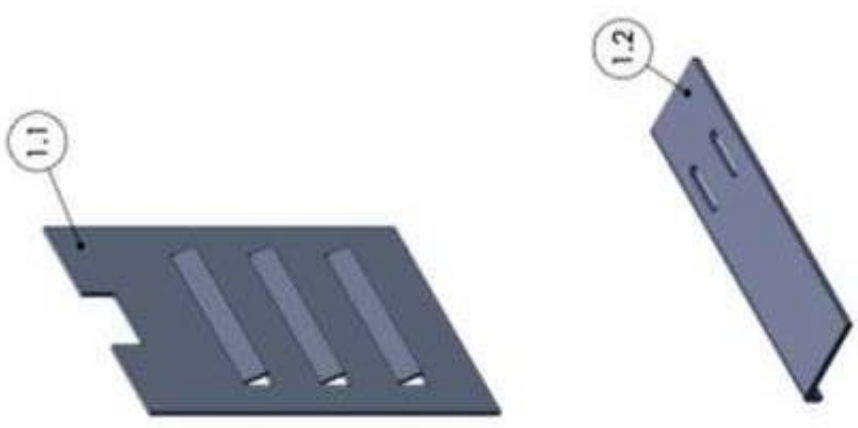
VISTA SUPERIOR



VISTA LATERAL



VISTA LATERAL

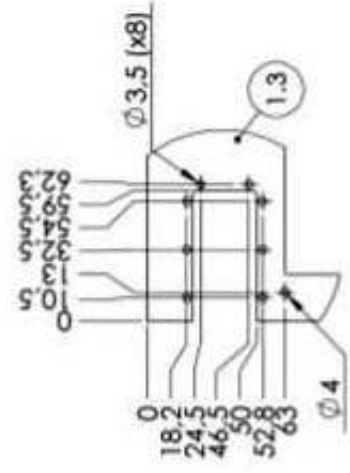
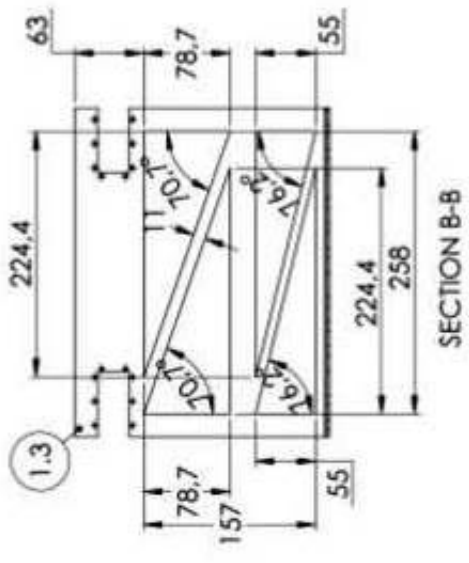
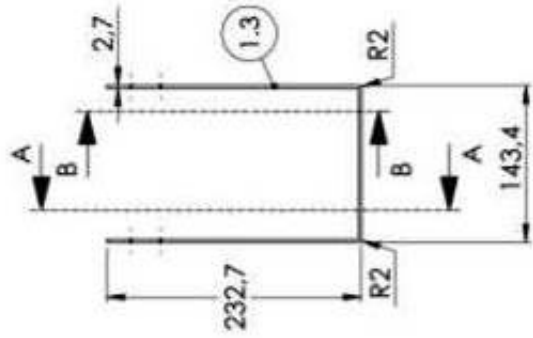
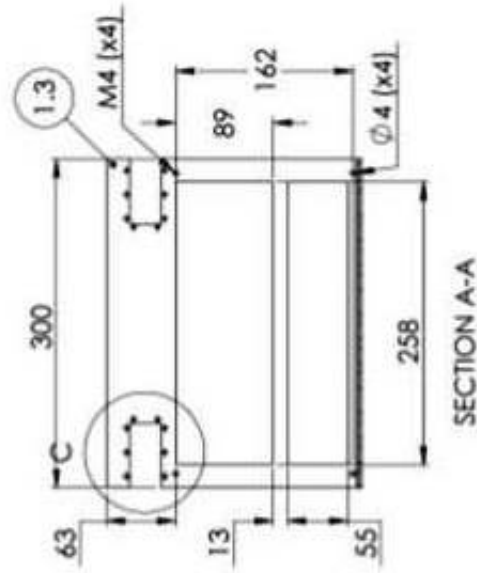


ITEM	DESCRIÇÃO	MATERIAL
1.2	CHAPA 02 : 138 x 47 x 2,7 mm	Alumínio
1.1	CHAPA 01 : 143,4 x 235,4 x 2,7 mm	Alumínio

		Drawn By:	BASE - ITENS 1.1 E 1.2 E ITEM 1.2	
		Project:	Pi-Ro 2.1	
		Drawn:	Juliana Sartori	
		Material:	Alumínio	
		Approved:	Marco Reis	
		Thermal:	N/A	
		Scale:	3/13	
		Date:	07/09/2017	
		Notes:	THE INFORMATION IN THIS DOCUMENT CAN NOT BE USED FOR ANY OTHER PURPOSE WITHOUT THE WRITTEN PERMISSION OF SENAI FIEB.	
		Revision:	0	

NOTA 01: CONSIDERAR QUE FILETES NÃO ESPECIFICADOS TÊM RAIO IGUAL A 2,0mm.

REV	DESCRIPTION	DRAW	DATE
0	Drawing elaboration	Juliana Sartori	07/09/2017



DETALHE C
ESCALA 2:5
FURAÇÃO TÍPICA PARA ENCAIXE
DO MOTOR MX-106T

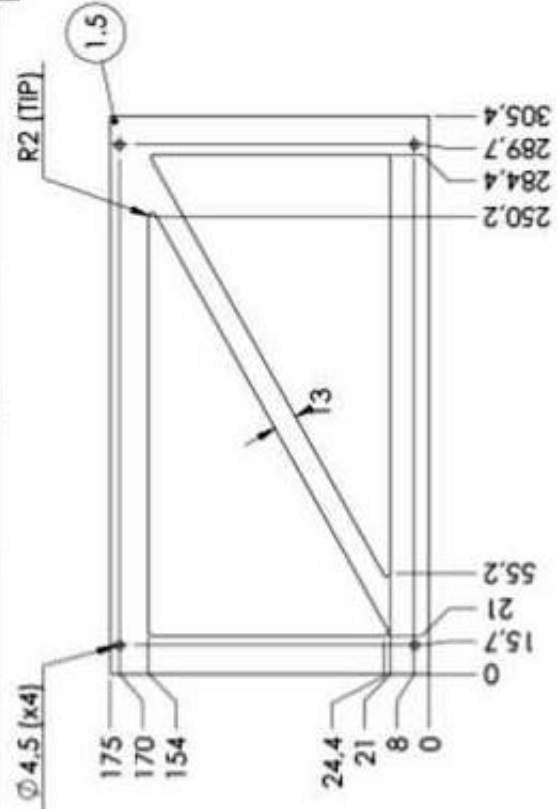
ITEM	Chapa 608.8 x 300 x 2.7mm	Alumínio
DESCRIÇÃO	BASE - ITEM 1.3	
MATERIAL	Alumínio	
Quantidade	01	
Rev	A3	
Projeto	FIR 2.1	
Desenho	Juliana Sartori	
Assinatura	Marco Reis	
Material	Alumínio	
Processo	N/A	
Page	4/13	
Date	07/09/2017	

1. All measurements in mm
 2. All dimensions are specified
 3. All dimensions are specified
 4. All dimensions are specified
 5. Detail drawing

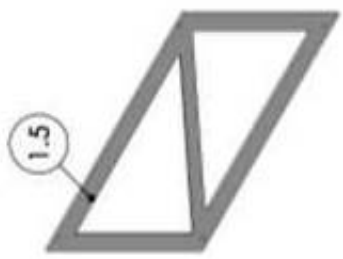
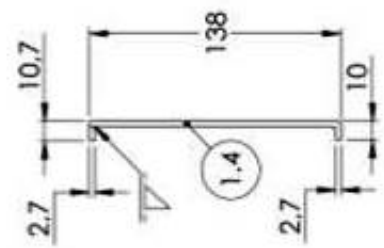
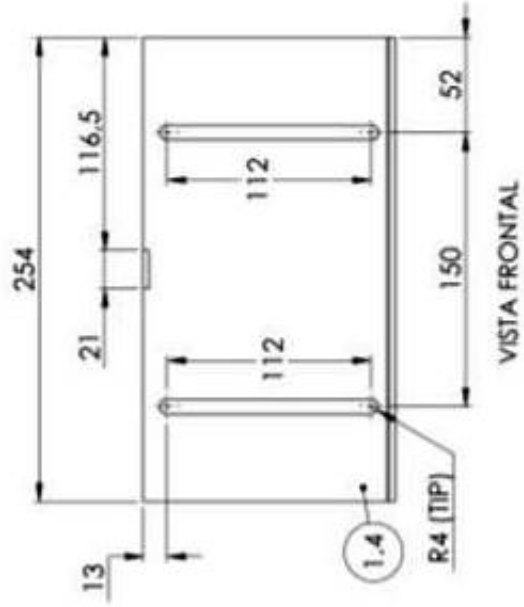
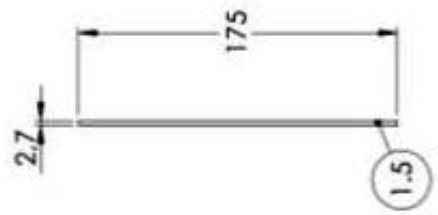
0

THIS DOCUMENT IS THE PROPERTY OF SENAI FIEB. IT IS TO BE USED ONLY FOR THE PROJECT FOR WHICH IT WAS CREATED. IT IS NOT TO BE REPRODUCED OR TRANSMITTED IN ANY FORM OR BY ANY MEANS, ELECTRONIC OR MECHANICAL, WITHOUT PERMISSION IN WRITING FROM SENAI FIEB.

REV	DESCRIPTION	DRAW	DATE
0	Drawing elaboration	Juliana Sartori	07/09/2017



VISTA LATERAL



ITEM	DESCRIÇÃO	MATERIAL
5	Chapa 305,4 x 145 x 2,7mm	Alumínio

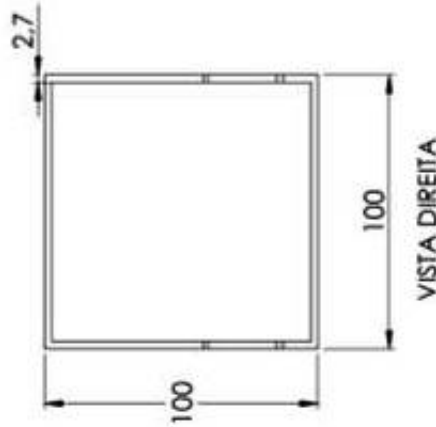
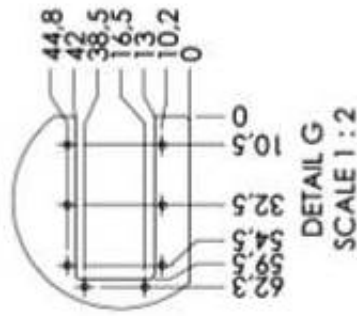
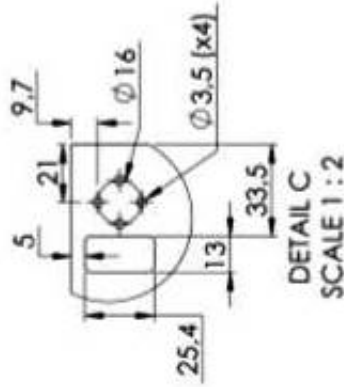
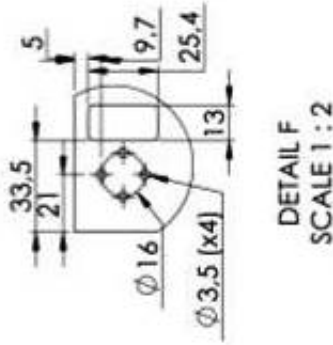
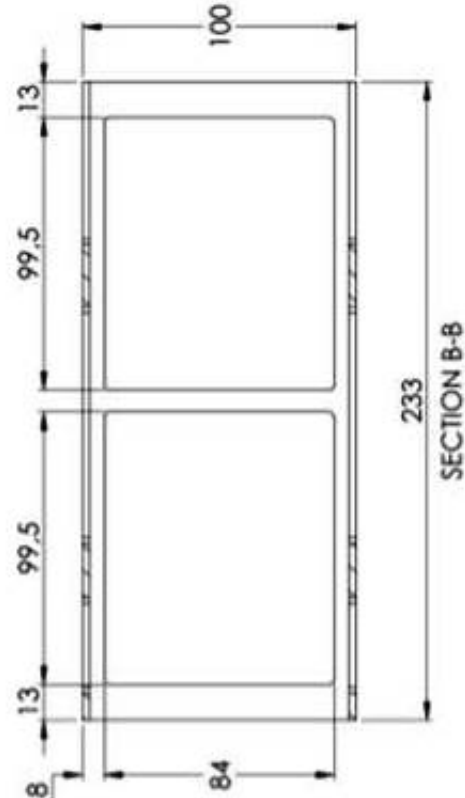
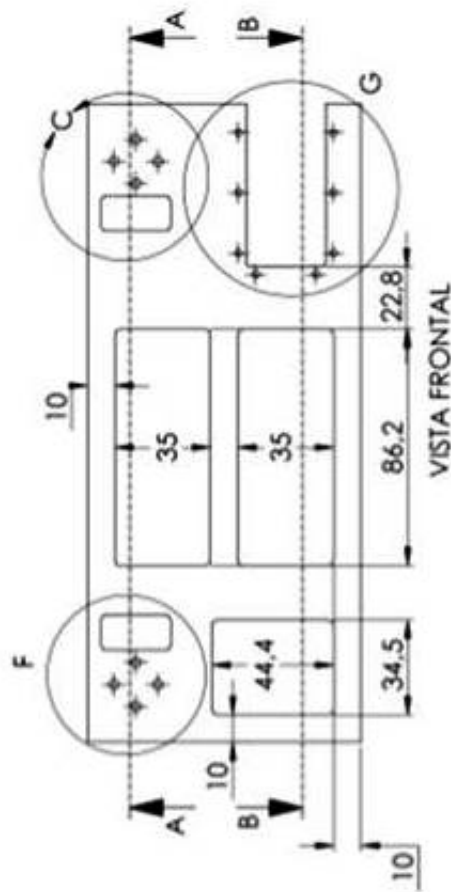
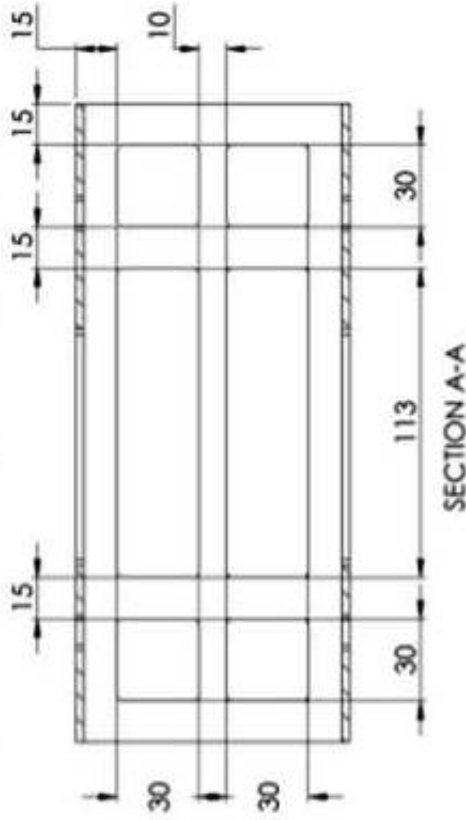
BASE - ITENS 1.4 E 1.5	
Quantidade	1.2
Unidade	---
Material	A3
Projeto	PRo 2.1
Desenho	Juliana Sartori
Material	Alumínio
Thermat	N/A
Assinatura	Marco Reis
Revisão	5/13
Assinatura	Marco Reis
Revisão	5/13



Notas:
1. Atualizar o nome do projeto para o nome do projeto.
2. Atualizar o nome do projeto.
3. Definir o nome do projeto.




































































































Revisão: 0

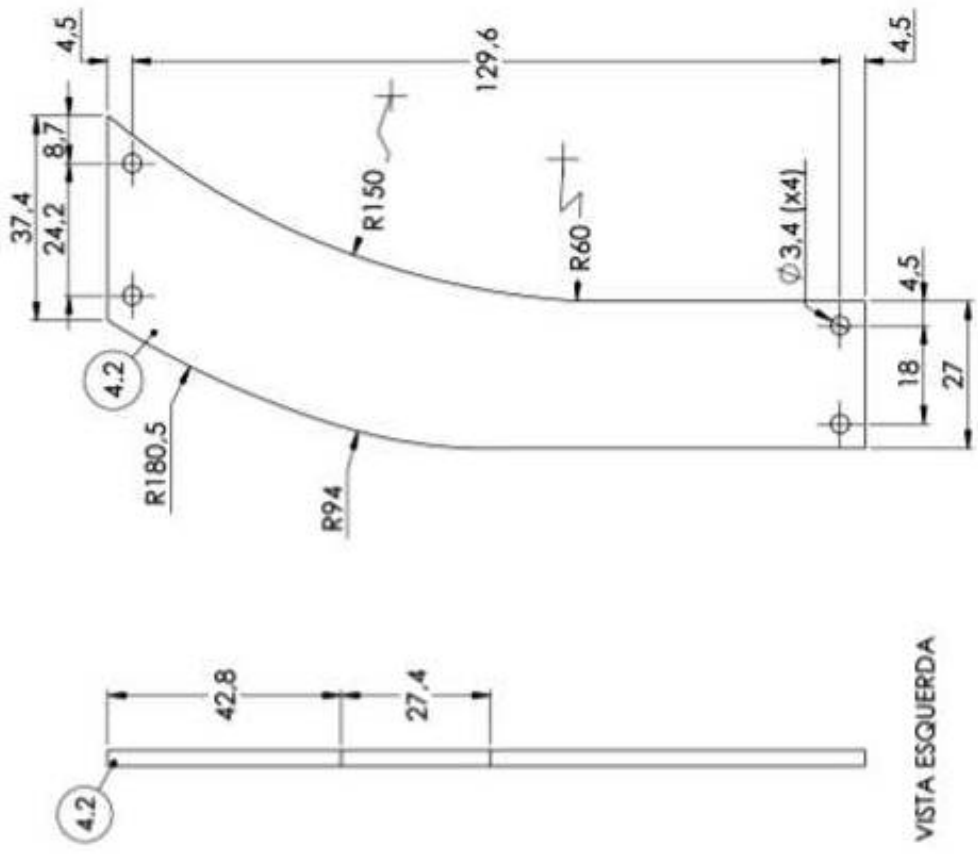
THE INFORMATION IN THIS DOCUMENT CAN NOT BE COPIED, REPRODUCED OR USED FOR ANY PURPOSES WITHOUT THE WRITTEN PERMISSION OF THE COMPANY. ALL RIGHTS ARE RESERVED.



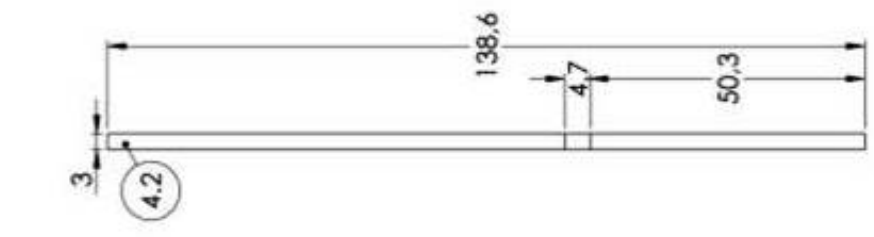
VISTA ISOMÉTRICA

NOTA 01: CONSIDERAR QUE FILETES NÃO ESPECIFICADOS TÊM RAIO IGUAL A 2mm.

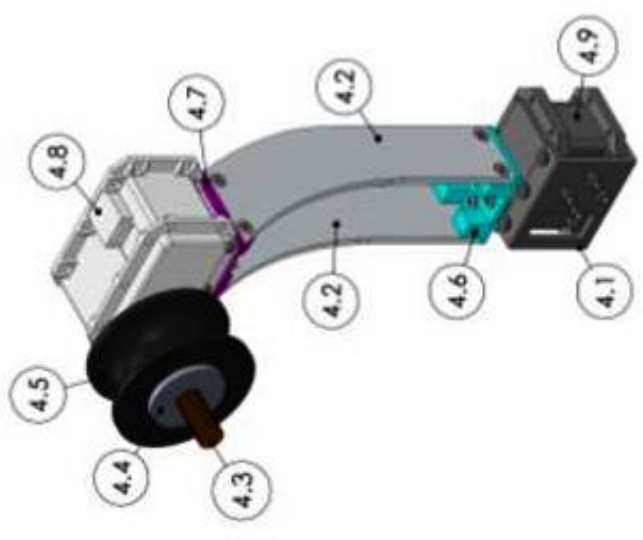
 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>		 <small>FUNDACÃO DAS INDÚSTRIAS DE CAXIAS DE PÓS</small>		 <small>BR Robotics</small>			
--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--	---	--	--	--



VISTA ESQUERDA



VISTA DIREITA



VISTA ISOMÉTRICA

ITEM	DESCRIÇÃO	QTD	MATERIAL
4.9	Motor MX-28	05	—
4.8	Motor MX-106T-31	05	—
4.7	Garra- Chapa IV	05	Alumínio
4.6	Garra- Chapa III	05	Alumínio
4.5	Roldana externa	05	Borracha
4.4	Roldana interna	05	Alumínio
4.3	Eixo Roldana	05	Alumínio
4.2	Garra- Chapa I	10	Alumínio
4.1	Suporte garra	05	Alumínio

Projeto: PH-2.1

Desenho: Juliana Sartori

Aprova: Marco Reis

Data: 07/09/2017

Revisão: 0

Observações:

1. Até 1000mm
2. Semelhante ao modelo
3. Default tracing

Material: Alumínio

Thermal: N/A

Page: 8/13

CONJUNTO ROLDANA - ITEM 04

ITEM 4.2

REV	DESCRIPTION	DRAW	DATE
0	Drawing elaboration	Juliana Sartori	07/09/2017

4

3

2

1

F

F

E

E

D

D

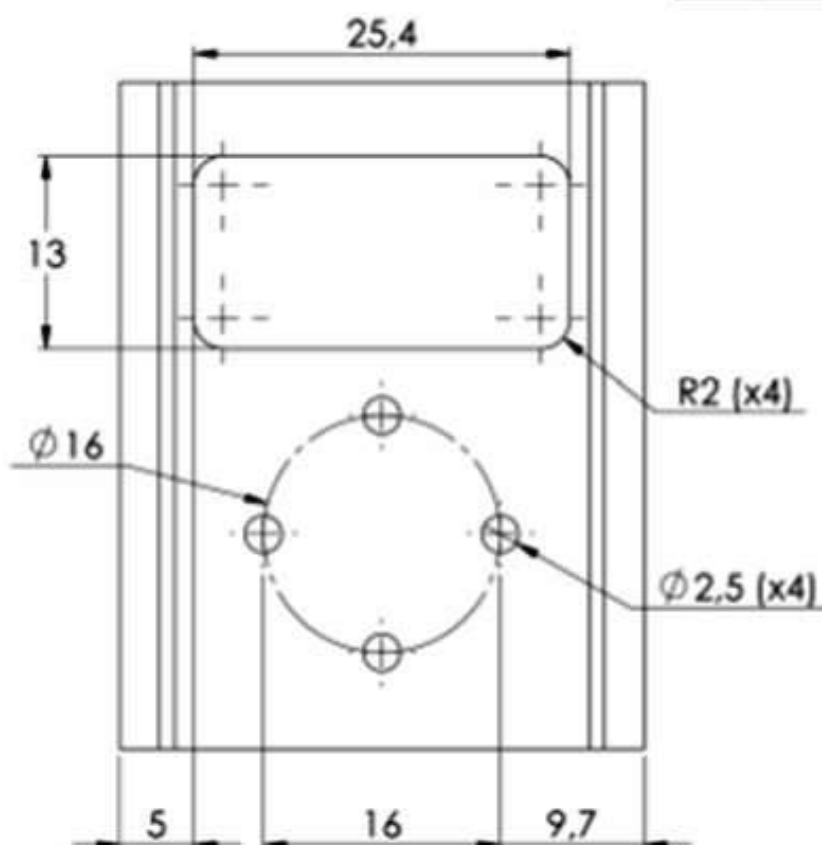
C

C

B

B

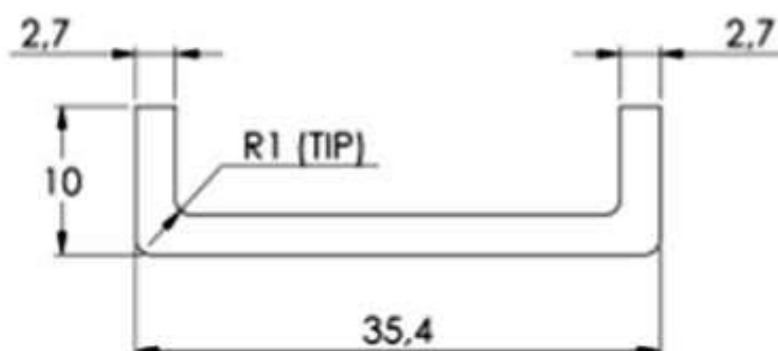
REV	DESCRIPTION	DRAW	DATE
0	Drawing elaboration	Juliana Sartori	07/09/2017



VISTA FRONTAL



VISTA DIREITA



VISTA INFERIOR



VISTA ISOMÉTRICA



Notes:
1. All measurements in mm
unless otherwise specified
2. Remove cutting edge
3. Default finishing

Tolerances:
mm ± 0,01
mm ± 0,01
mm ± 0,1
mm ± 0,2

Revisão:
0



Draw Title:

CONJUNTO ROLDANA ITEM 4.1

Project:

Pi-Ro 2.1

Draw:

Juliana Sartori

Agree:

Marco Reis

Date:

07/09/2017

Material:

Alumínio

Thermal T:

N/A

Scale:

2:1

Quantity:

05

Size:

A4

Finishing:

Page:

9/13

THE INFORMATION IN THIS DOCUMENT CANNOT BE COPIED, GIVEN AWAY OR USED FOR OTHERS MEANS EXCEPT THOSE CONTAINED IN THE CONTRACT TERMS.

4

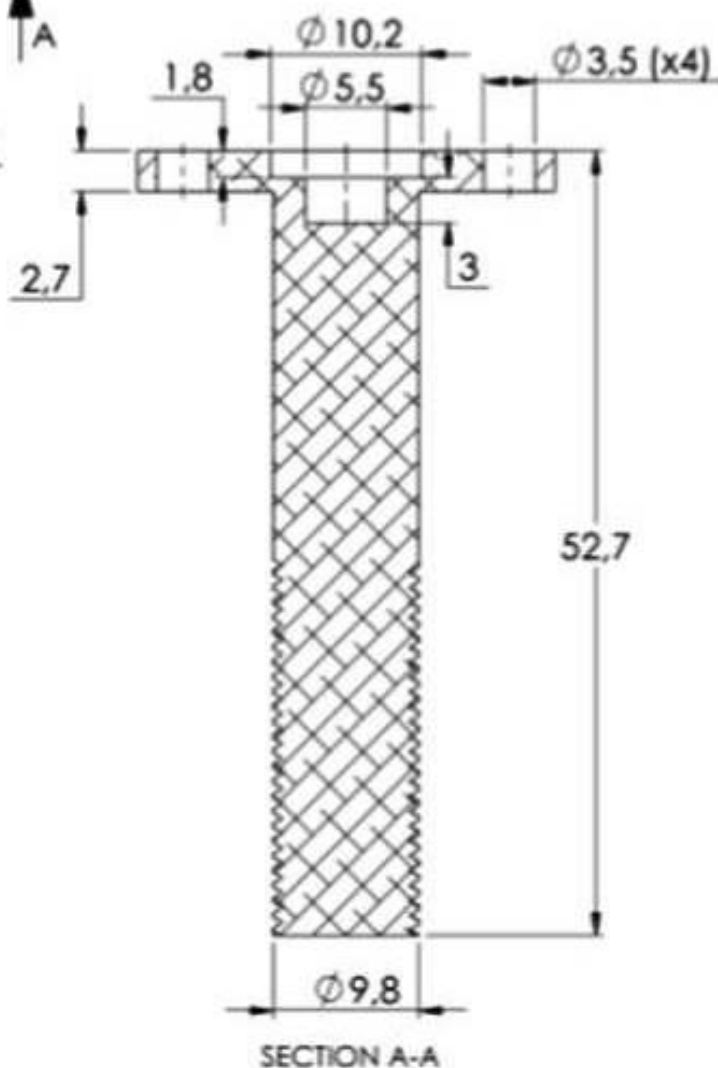
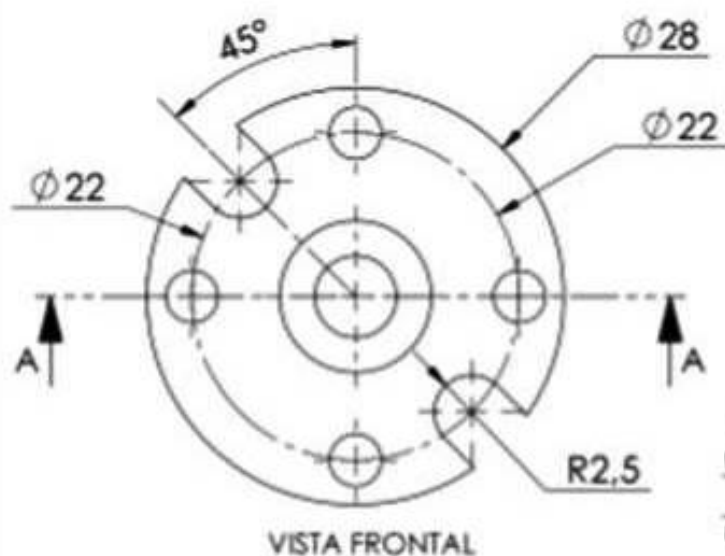
3

2

1

A

REV	DESCRIPTION	DRAW	DATE
0	Drawing elaboration	Juliana Sartori	07/09/2017



Notes:

1. All measurement in mm unless otherwise specified
2. Remove cutting edge
3. Default finishing

Revisão:

0

Tolerances:

±0.001
±0.01
±0.1
±0.2



Draw Title:

CONJUNTO ROLDANA ITEM 4.3

Project:

PI-Ro 2.1

Draw:

Juliana Sartori

Agree:

Marco Reis

Date:

07/09/2017

Material:

Alumínio

Thermal T:

N/A

Scale:

2:1

Quantity:

05

Size:

A4

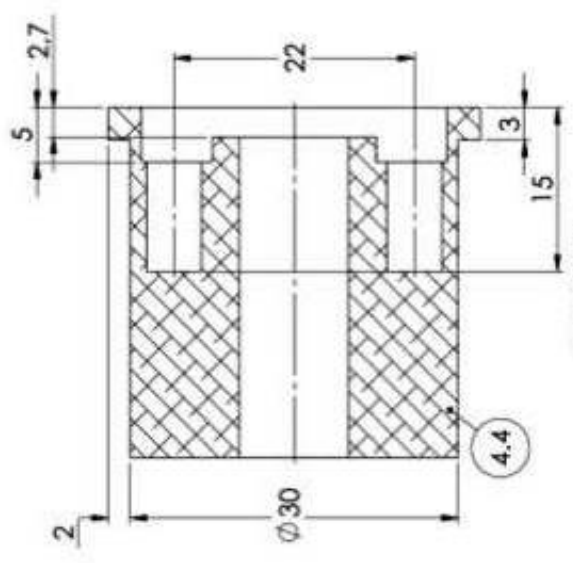
Finishing:

Page:

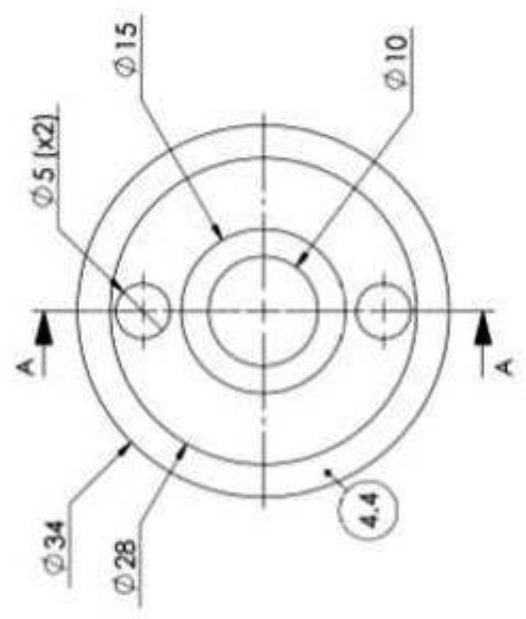
10/13

THIS INFORMATION IN THIS DOCUMENT CANNOT BE COPIED, GIVEN AWAY OR USED FOR OTHERS BEINGS EXCEPT THOSE CONTAINED IN THE CONTRACT TERMS.

REV	DESCRIPTION	DRAW	DATE
0	Drawing elaboration	Juliana Sartori	07/09/2017



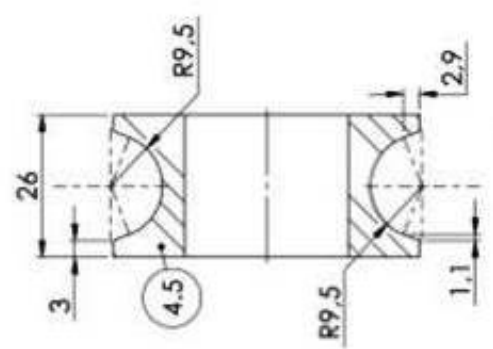
SECTION A-A



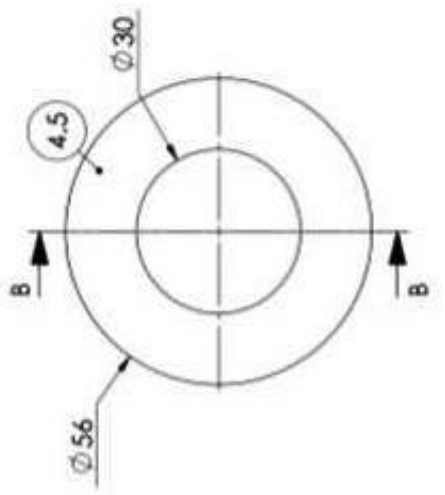
VISTA FRONTAL





VISTA ISOMÉTRICA



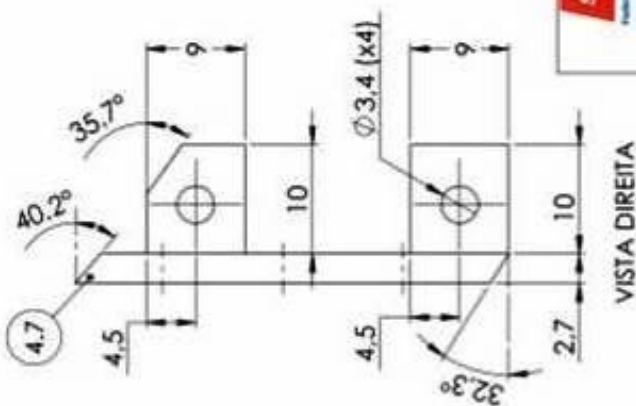
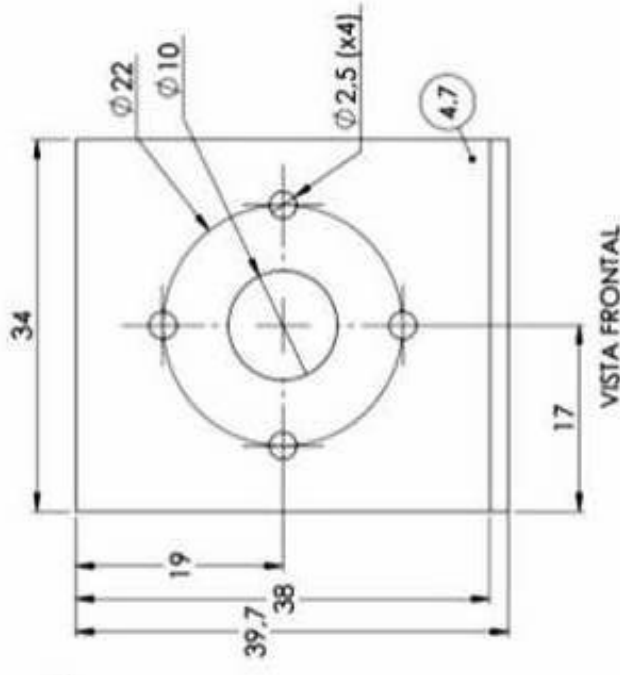
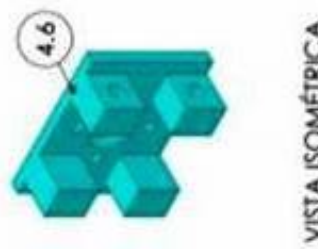
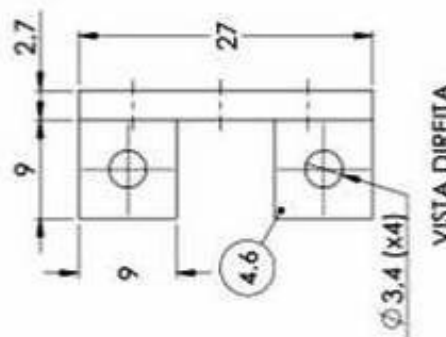
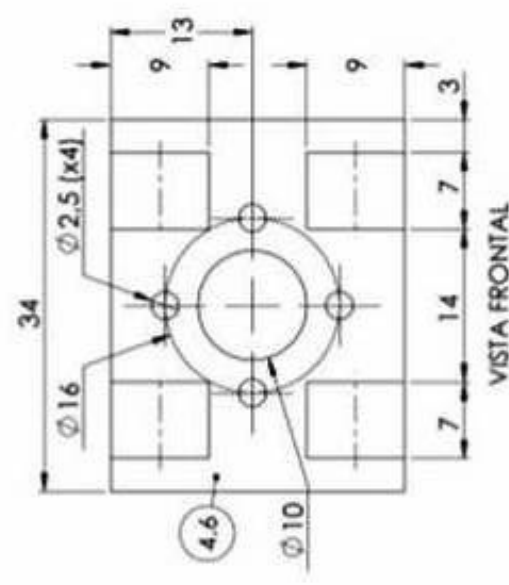
SECTION B-B
ESCALA 1:1



VISTA FRONTAL
ESCALA 1:1

		Draw Title	CONJUNTO ROLDANA ITEM 4.4 E 4.5
		Project	PRo 2.1
1. All representations of the 2. Revision (drawing stage) 3. Default hatching		Drawn	Juliana Sartori
1. All representations of the 2. Revision (drawing stage) 3. Default hatching		Approved	Marco Reis
1. All representations of the 2. Revision (drawing stage) 3. Default hatching		Material	Aluminio
1. All representations of the 2. Revision (drawing stage) 3. Default hatching		Thermal	N/A
1. All representations of the 2. Revision (drawing stage) 3. Default hatching		Scale	A3
1. All representations of the 2. Revision (drawing stage) 3. Default hatching		Quantity	05
1. All representations of the 2. Revision (drawing stage) 3. Default hatching		Revision	2:1
1. All representations of the 2. Revision (drawing stage) 3. Default hatching		Page	11/13
1. All representations of the 2. Revision (drawing stage) 3. Default hatching		THE INFORMATION IN THIS DOCUMENT IS UNCLASSIFIED AND IS NOT TO BE RELEASED TO THE PUBLIC WITHOUT THE AUTHORITY OF THE CONTRACTING AGENCY	

REV	DESCRIPTION	DRAW	DATE
0	Drawing elaboration	Juliana Sartori	07/11/2017
A	Modificação de dimensões	Juliana Sartori	18/11/2017



		Draw Title	CONJUNTO ROLDANA - ITENS 4.6 E 4.7		Scale	1:2
		Project	PI-Ro 2.1		Quantity	02
		Draw	Juliana Sartori		Rev	A3
		Material	Alumínio		Finishing	
		Thermal	N/A		Page	12/13
		Sign	Marco Reis			
		Date	07/09/2017			
		Revisão	0			

THE INFORMATION IN THIS DOCUMENT IS NOT BE COPIED, STORED OR REPRODUCED IN ANY FORM OR BY ANY MEANS WITHOUT THE WRITTEN PERMISSION OF THE COMPANY.

4

3

2

1

F

E

D

C

B

F

E

D

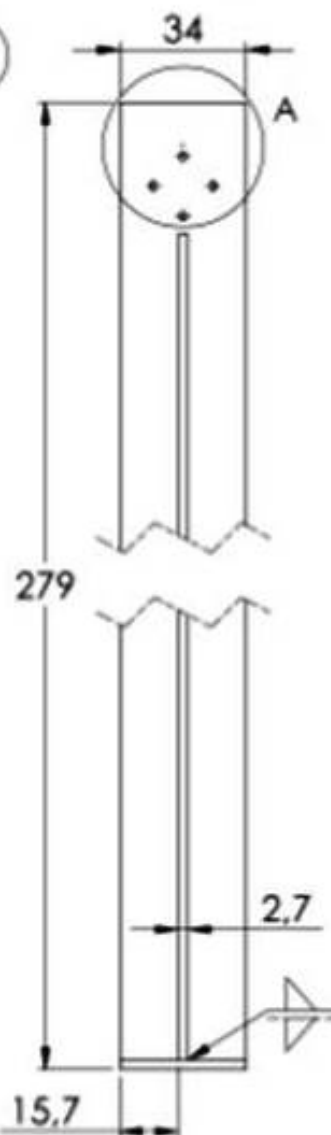
C

B

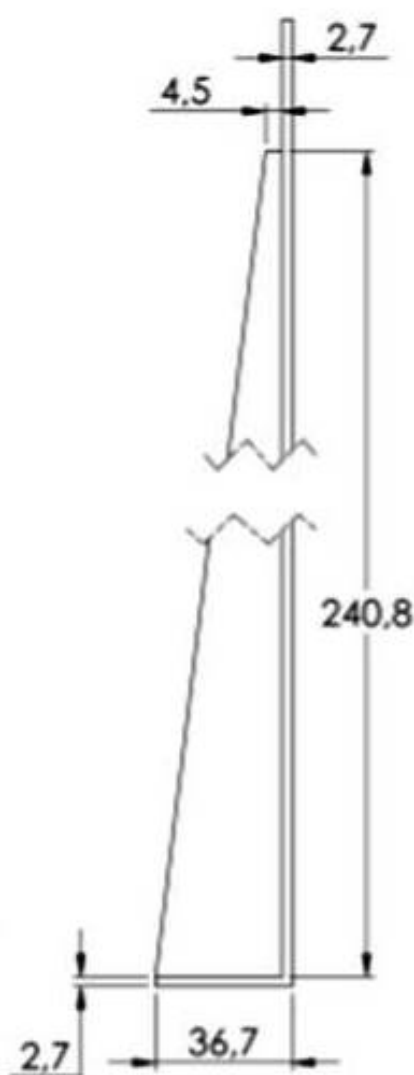
A

REV	DESCRIPTION	DRAW	DATE
0	Drawing elaboration	Juliana Sartori	07/09/2017

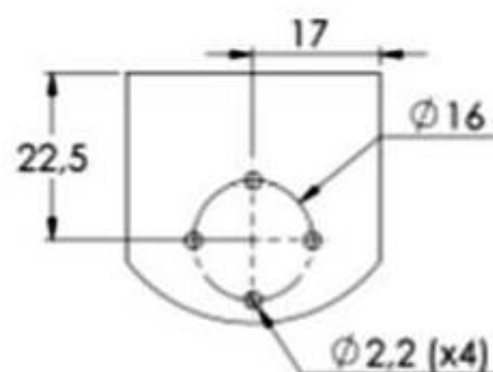
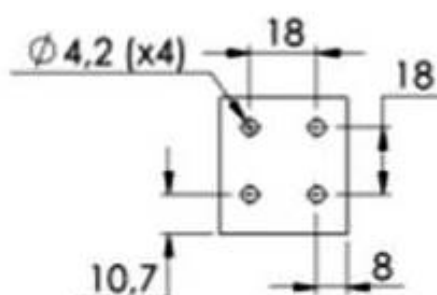
5



VISTA FRONTAL



VISTA DIREITA

DETAIL A
SCALE 1 : 1

VISTA INFERIOR

VISTA ISOMÉTRICA

		Draw Title:		Scale:
		HASTE CENTRAL - ITEM 5		1:2
		Project:		Quantity:
		PI-Ro 2.1		01
		Draw:		Size:
		Juliana Sartori		A4
		Material:		Finishing:
		Alumínio		
		Thermal T:		Page:
		N/A		13/13
Notes: 1. All measurement in mm unless otherwise specified 2. Remove cutting edge 3. Default finishing		Tolerances: max ±0.001 max ±0.01 max ±0.1 max ±0.2		Date: 07/09/2017
Revisado: 0				THE INFORMATION IN THIS DOCUMENT CANNOT BE COPIED, GIVEN AWAY OR USED FOR OTHERS BEINGS EXCEPT THOSE CONTAINED IN THE CONTRACT TERMS.

4

3

2

1

Diagramas eletro-eletrônicos

Logbook

CONFIGURAÇÃO DOS LIMITES DE GIRO DOS MOTORES

Objetivos

O teste teve como objetivo estabelecer os limites de giro dos motores em seus controladores, com base nos limites físicos da estrutura do robô.

Descrição do teste

É criado um “Controller manager” que conecta os motores e publica em um tópico as informações destes. As juntas do robô são posicionadas manualmente em suas posições máximas e mínimas, então o tópico “motor_states” é monitorado para verificar as posições dos motores.

DATA

8 AGOSTO 2018

LOCALIDADE

SENAI CIMATEC
SALVADOR - BAHIA

Mandruvah team

Cleber
Carlos
Ícaro
Davi

17:00

Foram coletados os limites de giro dos motores com id 11, 12, 13, 21, 22 e 23.

17:05

Ajustamos as posições iniciais dos controladores das juntas com base na posição “home” da simulação no *MovelIt!*. Nesse momento, verificamos que o valor que é publicado no controlador para mover a junta é a posição em radianos em relação à posição inicial que foi determinada no controlador.

TESTE DE MOVIMENTAÇÃO DOS SERVO-MOTORES

Objetivos

Verificar o comportamento do robô executando alguns movimentos em um dos braços.

DATA

10 AGOSTO 2018

Descrição do teste

Os motores são alimentados e seus controladores executados. A partir daí, valores de posição são publicados e o comportamento do robô verificado.

LOCALIDADE

SENAI CIMATEC
SALVADOR - BAHIA

Mandruvah team

Cleber
Carlos
Ícaro
Davi

15:20

O braço do robo foi levantado até a posição “home” com as duas juntas sendo movimentadas ao mesmo tempo. Antes de atingir a posição determinada, o motor com id 21 apresentava erro de overload.

16:10

Quando o robô começa o movimento já proximo da posição final, “home”, o braço consegue alcançar o objetivo. Depois de cerca de 5 minutos nessa posição, um erro de overheat é apresentado.

TESTE DE MOVIMENTAÇÃO DO BRAÇO

Objetivos

Verificar possíveis motivos para erro de "Overload" na junta 12-22 apresentado em teste anterior.

Descrição do teste

Os motores são alimentados e seus controladores executados. A partir daí, valores de posição são publicados e o comportamento do robô verificado.

DATA

13 AGOSTO 2018

LOCALIDADE

SENAI CIMATEC
SALVADOR - BAHIA

Mandruvah team

Cleber
Carlos
Ícaro
Davi

16:50

Foi verificado que os motores estão configurados para operar com 100% do torque, não sendo assim essa a causa do problema.

17:15

Percebemos também que o problema acontece com maior frequência quando as juntas do braço e da unidade de tração são acionadas ao mesmo tempo. Quando é acionada uma junta por vez, o "Overload" acontece menos vezes.

17:32

É levantada a suspeita de que a falta do cabo de sincronização nos motores da junta pode ser a causa da falha. Com o cabo conectado, o problema não aconteceu.

TESTE DE CONVERSOR DA PLACA DE POWER MANAGEMENT

Objetivos

Verificar possíveis problemas do conversor da placa de power management e sua resposta de saída .

DATA

25 SETEMBRO 2018

Descrição do teste

O conversor é retirado da placa e então testado com fonte de alimentação e sua saída observada com um multímetro. O conversor testado é do modelo UWE-12/10-Q12PB-C

LOCALIDADE

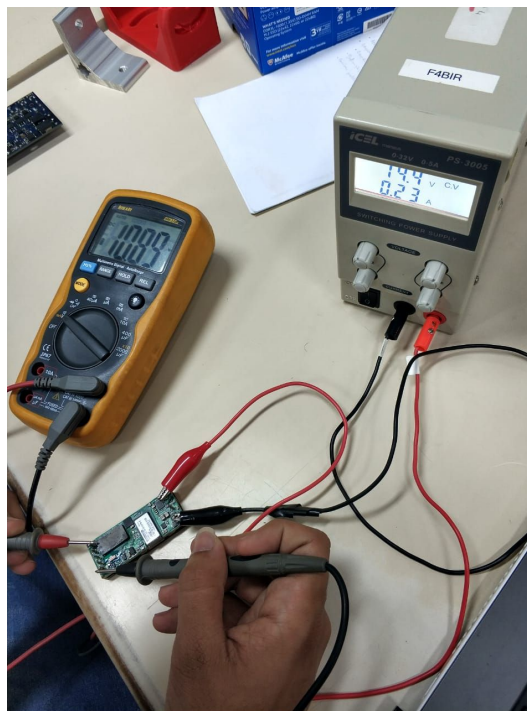
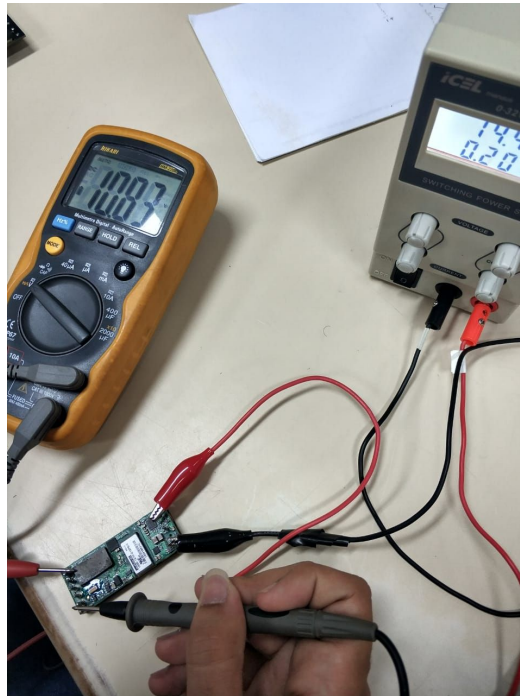
SENAI CIMATEC
SALVADOR - BAHIA

Mandruvah team

Cleber
Carlos
Ícaro
Davi

16:05

O conversor retirado da placa foi testado com alimentação de uma fonte de tensão com 14 Volts, e o mesmo consumiu um valor de cerca de 200 mA. Sua tensão de saída ficou em aproximadamente 10 Volts.

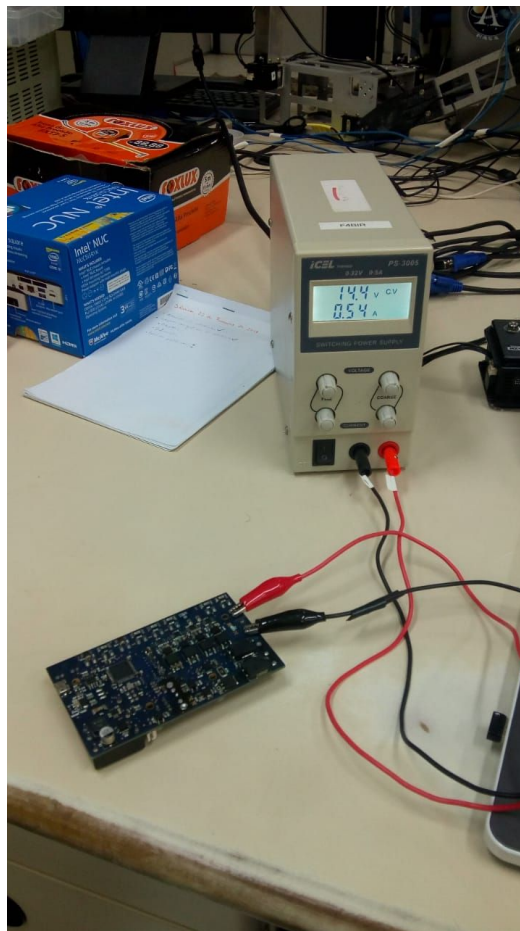


16:12

A corrente requisitada pelo conversor defeituoso oscila entre 200 mA e 250 mA. Sua temperatura, ao contrário do conversor que está funcionando corretamente, não aumenta e o conversor permanece frio.

16:28

O conversor que já está na Power Management permanece apresentando funcionamento correto. Sua temperatura aumenta quando permanece ligado.



TESTE DOS MOTORES/CONTROLLER_MANAGER

Objetivos

Identificar se há algum motor defeituoso que pode estar "sujando" a comunicação dos motores.

DATA

04 OUTUBRO 2018

Descrição do teste

Um motor é conectado e o arquivo controller_manager.launch é executado e verifica-se se o motor foi encontrado. Em seguida, são inseridos os demais motores, um a um, para que se perceba se a comunicação ainda acontece.

LOCALIDADE

SENAI CIMATEC
SALVADOR - BAHIA

Mandruvah team

Cleber
Carlos
Ícaro
Davi

13:40

Todos os 18 motores estavam conectados, utilizando apenas os componentes (cabos e hub) da ROBOTIS. Quando o controller_manager.launch foi executado, nenhum motor foi encontrado.

13:42

Com apenas um motor conectado, o controller_manager o encontrou.

13:55

O teste prosseguiu até que, quando o motor de ID 14 foi conectado, o controller_manager não encontrou mais motores.

14:10

O motor de ID 14 foi removido e o teste continuou. O mesmo erro aconteceu quando o motor de ID 3 foi adicionado. Esse motor também foi retirado.

14:15

O teste seguiu até o último motor, nenhum motor aparentemente defeituoso foi encontrado. A comunicação funcionou bem com os 16 motores restantes.

14:45

Dois motores novos foram conectados. O controlador foi executado por volta de 20 vezes, em todos os testes todos os motores foram encontrados.

TESTE DA PLACA DE POWER MANAGEMENT

Objetivos

Verificar possíveis problemas da montagem da placa de power management sem a presença de um dos conversores DC/DC

Descrição do teste

Os capacitores de acoplamento do regulador de tensão para o Atmega 32U4 são soldados e então a placa é alimentada com 14.4 Volts. A temperatura é monitorada com um multímetro com termopar, e os níveis de tensão com um multímetro comum.

DATA

05 OUTUBRO 2018

LOCALIDADE

SENAI CIMATEC
SALVADOR - BAHIA

Mandruvah team

Cleber
Carlos
Ícaro
Davi

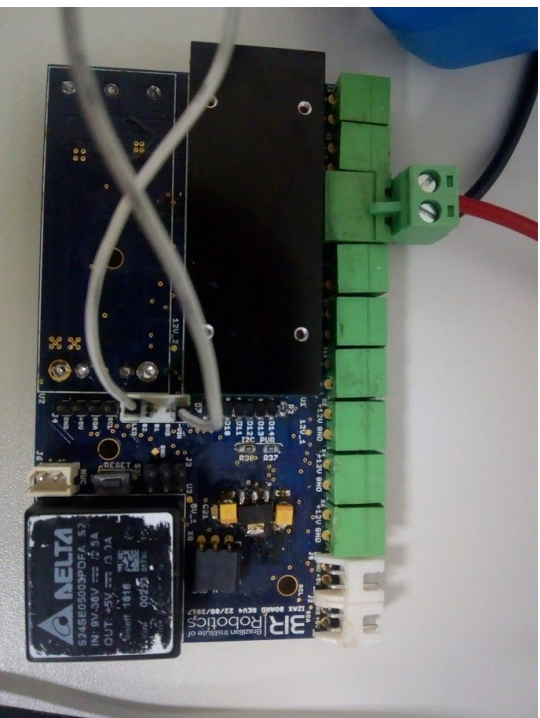
15:20

Capacitores de acoplamento foram soldados na placa, respeitando a polarização estabelecida no projeto de power management.

Capacitores de tântalo de 10uF de 16 Volts.

15:25

A placa foi alimentada com uma fonte de tensão à 14.4 Volts. Para que haja o funcionamento da placa é necessário realizar um curto entre os pinos -Vin e S2. Desta maneira inicia-se a operação da placa.



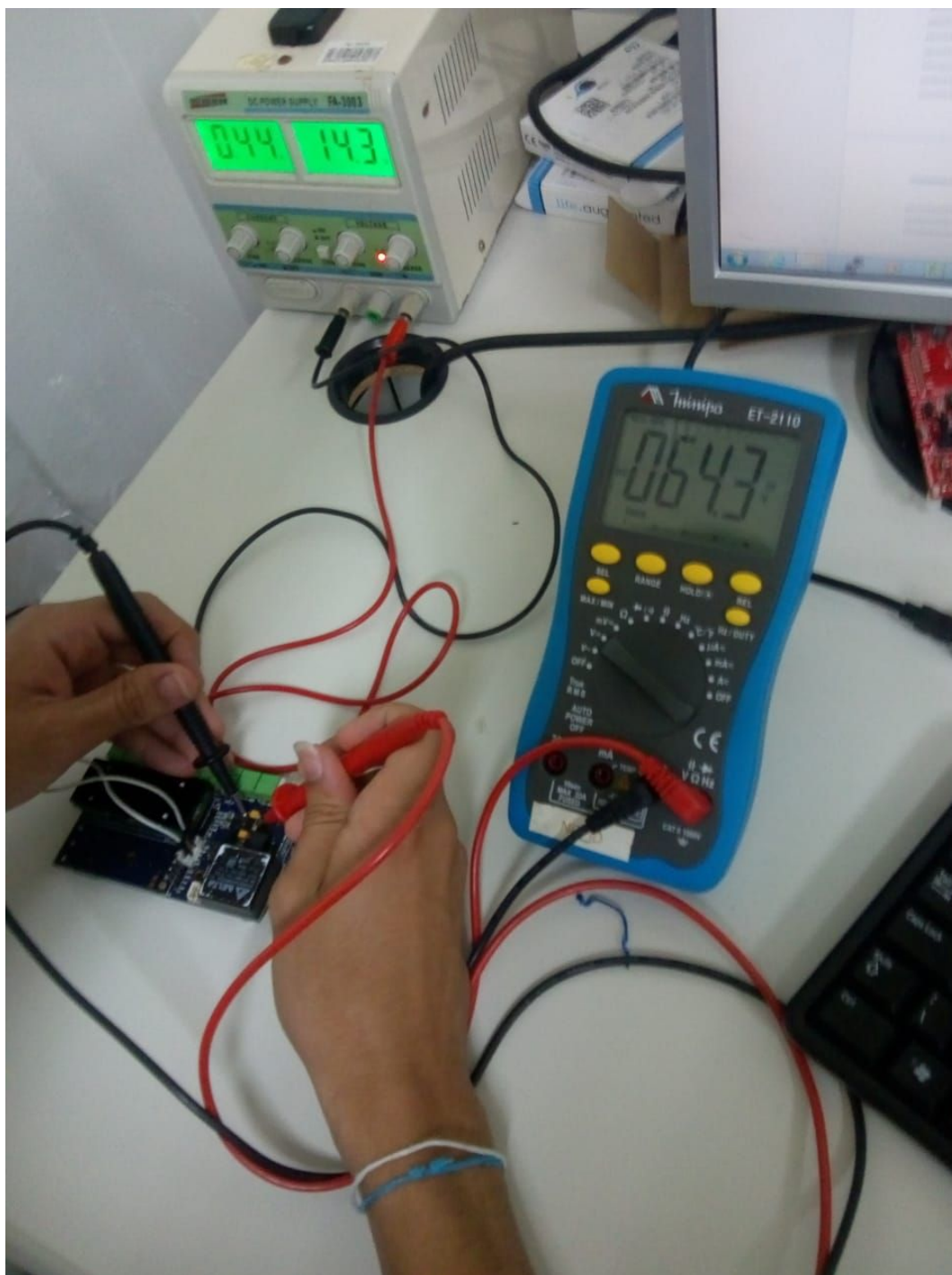
15:27

A placa apresentou um aumento da temperatura do regulador de tensão 5 Volts, chegando a alcançar temperaturas em cerca de 110°C.



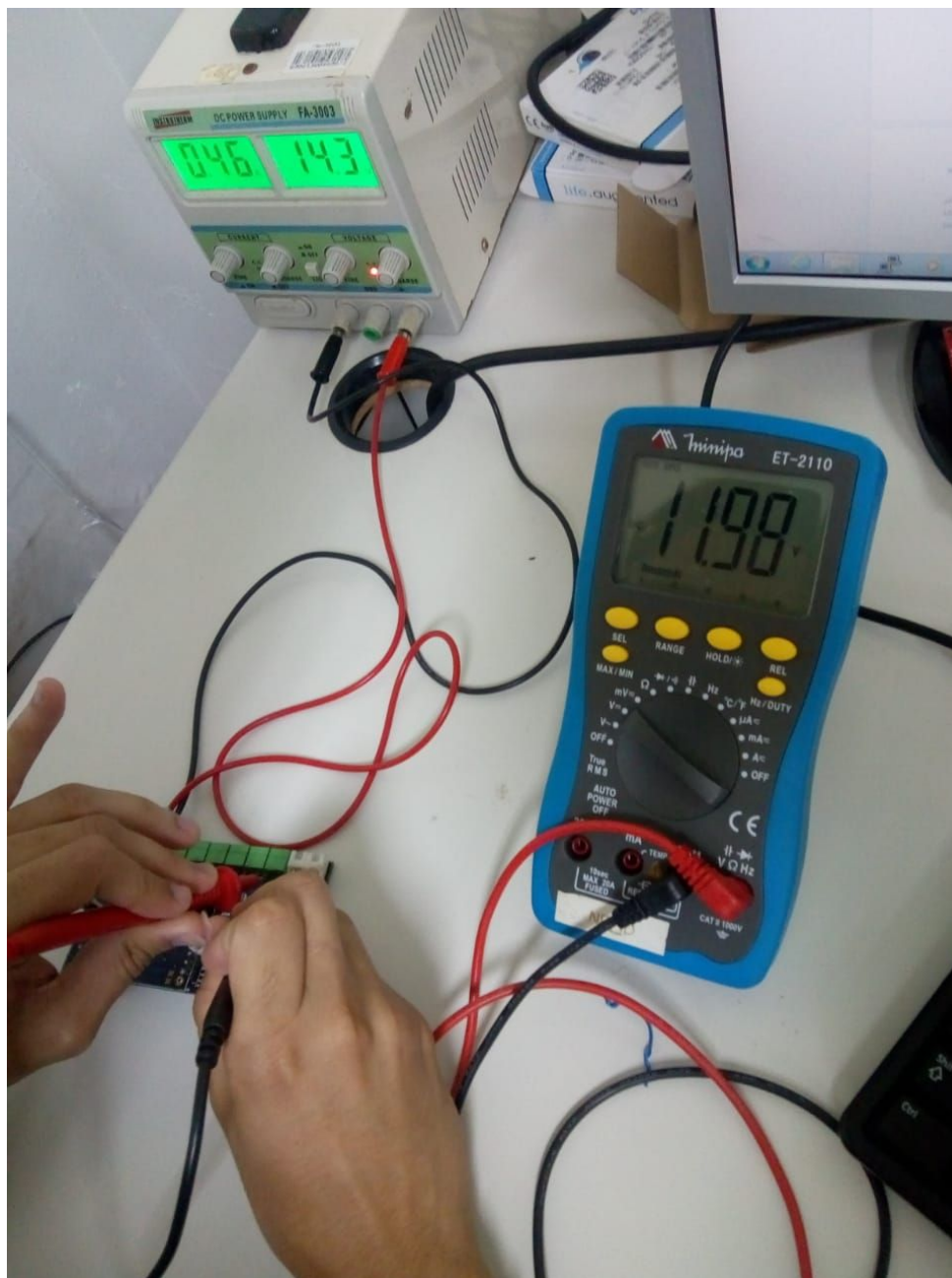
15:28

Os níveis de tensão gerados pelos reguladores também apresentaram níveis fora do esperado. O regulador de 5 volts apresentou tensões extremamente baixas, revelando um possível mal funcionamento, em torno de 70 milivolts



15:30

O conversor ainda presente na placa apresentou um funcionamento normal, gerando 12 Volts nas 3 portas de saída fixa, como esperado.



TESTE COM O ROBÔ NA LINHA

Objetivos

Montar setup do teste, acessar a NUC remotamente e movimentar o robô na linha

DATA

19 OUTUBRO 2018

Descrição do teste

O robô é posicionado na linha, é estabelecido um acesso remoto, são iniciados os controladores e movimenta-se o robô.

LOCALIDADE

SENAI CIMATEC
SALVADOR - BAHIA

Mandruvah team

Cleber
Davi
Ícaro
Carlos

10:40

O robô foi levado até o estacionamento do SENAI CIMATEC onde há uma linha de transmissão de testes. O setup foi montado com uma bateria automotiva, um inversor de frequência 12 - 110V para conectar a fonte da NUC e os motores ligados diretamente à bateria.

11:05

Com o robô na linha e a NUC ligada e conectada a uma rede local exclusiva do teste, conectamos um notebook com o robô via SSH e assim tivemos acesso remoto ao terminal do robô.

11:10

Ao tentar carregar o *controller_manager.launch* nenhum motor foi encontrado. Conferimos então todas as conexões dos hubs de comunicação, nenhum mal contato foi encontrado. Percebemos que um dos terminais do conector do conversor USB-RS485 estava solto da placa.

13:20

Corrigimos o terminal solto do conversor. Novamente carregamos o *controller_manager.launch*, aconteceram erros de “*checksum*” e de “*wrong packet prefix*” e apenas 4 dos 18 motores foram encontrados.

13:40

Após os erros de comunicação, verificamos que quando corrigimos o conversor, os fios D+ e D- foram trocados acidentalmente.

13:45

Ainda depois de corrigir a conexão invertida, não conseguimos encontrar todos os motores, foi levantada a hipótese de que a bateria automotiva era o problema. Medimos a tensão e estava em 12,15V, decidimos voltar ao laboratório e testar com a fonte de bancada.

14:20

Repetimos o teste alimentando o robô com a fonte de bancada e o teste correu bem, nenhum dos erros anteriores aconteceu.

Lista de componentes

ELIR project - BILL OF MATERIAL

							\$3.60		
id	component	description	brand	part number	power/current	connection	unit cost [R\$]	quantity	total cost [R\$]
01	interface board		Phidgets	1019_1B	500mA (max)	USB	R\$ 272.00	1	R\$ 272.00
02	proximity sensor		ETT CO. Ltd	E18-D80NK npn	<25mA	Digital Output	R\$ 29.00	5	R\$ 145.00
03	temperature sensor		Texas Instruments	LM35	10mA	Analog Output	R\$ 7.38	1	R\$ 7.38
04	gps		Swift Navigation	Piksi 2.3.1	5V, 500mW	USB		1	R\$ 0.00
05	imu		XSENS	Mti-1	44mW	USB		1	R\$ 0.00
06	ultrasonic sensor		Maxbotix	EZ-1	5V, 2mA	Analog Output	R\$ 107.82	1	R\$ 107.82
07	current sensor		Phidgets	1122_0	5V, 10mA	Analog Output	R\$ 106.02	3	R\$ 318.06
08	lwir camera		FLIR	Lepton 1.0	140mW	I2C		1	R\$ 0.00
09	bridge board I		STM	STM32F401 RE	160mA/0.64W	USB	R\$ 49.79	1	R\$ 49.79
10	bridge board II		STM	STM32L432	140mA/0.56W	USB	R\$ 39.56	1	R\$ 39.56
11	joint hub		Mandrurah	-	12V	RS485	R\$ 25.20	3	R\$ 75.60
12	servomotor I		Dynamixel	MX-28	1.4A/16.8W	RS485	R\$ 900.00	5	R\$ 4,500.00
13	servomotor II		Dynamixel	MX-106	5.2A/62.4W	RS485	R\$ 1,800.00	13	R\$ 23,400.00
14	adapter 485-usb							1	R\$ 0.00
15	battery		Inspired Energy		89Wh/6,2Ah		R\$ 879.98	2	R\$ 1,759.97
16	power management board		-			USB	R\$ 2,200.00	1	R\$ 2,200.00
17	multiplex board		Inspired Energy	EB325A	15mA/0.36W	I2C	R\$ 1,296.00	1	R\$ 1,296.00
18	central processing		Intel	NUC515RYK			R\$ 4,300.00	1	R\$ 4,300.00
19	stereo camera		Stereolabs	ZED camera	380mA	USB	R\$ 1,800.00	1	R\$ 1,800.00
20	cabo usb								