

GUSTAVO CORRÊA UMEZAKI

**PROJETO E IMPLEMENTAÇÃO DE ROBÔ MÓVEL  
OMNIDIRECIONAL APLICADO A ENXAME DE  
ROBÔS**



UNIVERSIDADE FEDERAL DE UBERLÂNDIA

CURSO DE ENGENHARIA MECATRÔNICA

UBERLÂNDIA

2017

GUSTAVO CORRÊA UMEZAKI

**PROJETO E IMPLEMENTAÇÃO DE ROBÔ MÓVEL  
OMNIDIRECIONAL APLICADO A ENXAME DE ROBÔS**

**Monografia de Conclusão de Curso** apresentada no programa de graduação em Engenharia Mecatrônica da Universidade Federal de Uberlândia, como parte dos requisitos para obtenção do título de **BACHAREL EM ENGENHARIA MECATRÔNICA**.

Área de Concentração: Robótica.

Orientador: Prof. Dr. José Jean-Paul Zanlucchi de Souza Tavares.

UBERLÂNDIA MG

2017

UMEZAKI, G. C. **Projeto e Implementação de Robô Móvel Omnidirecional Aplicado a Enxame de Robôs**. 2017. Monografia de Conclusão de Curso (Bacharel) – Curso de Engenharia Mecatrônica, Universidade Federal de Uberlândia, Uberlândia. Páginas 73.

### Resumo

A evolução da tecnologia aliada à robótica tem como consequência o uso de robôs para inúmeras tarefas em diversas áreas, podendo hoje alcançar inclusive áreas de entretenimento, médica, atividade militar, pesquisa, orientação e domiciliar, ao contrário dos seus primórdios, quando tinha foco na indústria. Esse trabalho tem como objetivo projetar, produzir e testar uma estrutura de robô móvel com três rodas omnidirecionais, com o intuito de servir de modelo e incentivo em pesquisas e desenvolvimento da robótica cooperativa ou de enxame. Para tanto, buscou se aliar metodologias de projeto com fabricação assistida por computador, desde a concepção do desenho, passando pela integração com o processo de fabricação e maquinário e finalizando na programação de um microcontrolador de prototipagem rápida (Arduino) para testar o seu funcionamento. Obteve-se como resultado um projeto multidisciplinar desde sua concepção, execução até a entrega de um modelo de robô móvel omnidirecional (com o intuito de incentivar a pesquisa e desenvolvimento desse tipo de locomoção) e de custo reduzido (requisito importante do projeto já que visa sua replicação).

---

Palavras Chave: robótica, enxame, fabricação assistida por computador, Arduino, robô móvel, omnidirecional.

UMEZAKI, G. C. **Design and Implementation of Omnidirectional Mobile Robot Applied to Robot Swarm.** Course Completion Monograph (Bacharel) - Mechatronics Engineering Course, Federal University of Uberlândia, Uberlândia. Pages 73.

### **Abstract**

The evolution of technology combined with robotics results in the use of robots for numerous tasks in several areas, and it can now reach areas of entertainment, medical, military activity, research, guidance and domicile, as opposed to its beginnings, when it focused on industry. This work aims to design, produce and test a mobile robot structure with three Omnidirectional wheels, in order to become a model and incentive in the research and development of cooperative or swarming robotics. In this direction, it sought to combine design methodologies with computer-aided manufacturing and finalizing the programming of a rapid prototyping microcontroller (Arduino) robot to test its operation. As a result, this work presents a multidisciplinary project since its conception, execution to the delivery of an omnidirectional mobile robot model (with the intention of encouraging research and development of this type of locomotion) and a reduced cost (an important requirement of the project since it aims its replication).

---

Keywords: robotics, swarm, computer-aided manufacturing, Arduino, mobile robot, omnidirectional.

## Lista de Figuras

Figura 2.1 – Robôs modernos existentes no mercado atual (a) Robô cirurgião Da Vinci; (b) Robô Asimo da Honda; (c) Braço robótico UR3 da empresa <i>Universal Robots</i> . ....	6
Figura 2.2 - Esquema geral do sistema de controle de um robô móvel. (Adaptado de Secchi 2008). ....	9
Figura 2.3 – Diferentes ambientes de trabalho para robôs (a) Competição de robôs (ambiente interno); (b) Adaptação de frame da apresentação do robô Spot (ambiente externo). ....	11
Figura 2.4 - Ambiente estruturado (biblioteca) e não estruturado (depósito de linha de produção). ....	11
Figura 2.5 - Mecanismos de locomoção utilizados por sistemas biológicos. ....	12
Figura 2.6 - Tipos de rodas: padrão (a), castor (b), sueca (c) e esférica (d). ....	13
Figura 2.7 - Exemplo de disposição das rodas suecas sobre uma estrutura mecânica. ....	14
Figura 2.8 - Exemplos de roda Sueca. ....	14
Figura 2.9 - Organograma da estrutura RepRap. ....	18
Figura 3.1 - Fluxograma da metodologia de desenvolvimento do projeto. ....	21
Figura 4.1 - Esboços das plataformas. ....	26
Figura 4.2 - Impressora Prusa Mendel no seu local de trabalho. ....	28
Figura 4.3 - Imagem ilustrativa do motor utilizado. ....	30
Figura 4.4 – <i>Driver</i> ponte H dupla para motores DC baseado no <i>chip</i> L298N (a) Módulo dupla ponte H L298N; (b) Entradas e saídas do módulo. ....	30
Figura 4.5 - Exemplo de configuração de circuito ponte H. ....	32
Figura 4.6 - PWM. ....	33
Figura 4.7 - PWM no Arduino. ....	34
Figura 4.8 - Arduino Mega e Genuino Mega. ....	35
Figura 4.9 - Captura de tela da vista isométrica com linhas ocultas visíveis do primeiro protótipo. ....	36
Figura 4.10 - Captura de tela da vista frontal do primeiro protótipo. ....	36
Figura 4.11 - Captura de tela da vista lateral do primeiro protótipo. ....	36
Figura 4.12 - Kit robô móvel de roda omnidirecional. ....	41
Figura 4.13 - Captura de tela do software Fritzing com ilustração da montagem no Arduino. ....	42
Figura 4.14 - Movimentos a serem seguidos pelo robô na etapa de demonstração. ....	44
Figura 4.15 - Foto do protótipo montado. ....	51

Figura 6.1 - de tela do software Fritzing com a montagem esquemática da parte eletrônica do robô.....55

## Lista de Tabelas

Tabela 2.1 - Tabela comparativa dos materiais.	19
Tabela 4.1 - Requisitos do projeto.	24
Tabela 4.2 - Restrições do projeto.	25
Tabela 4.3 - Características do Módulo ponte H L298N.	31
Tabela 4.4 - Esquema de funcionamento dos pinos IN do Módulo L298N.	32
Tabela 4.5 - Características do microcontrolador Arduino Mega 2560.	35
Tabela 4.6 - Tempo de fabricação das peças.	39
Tabela 4.7 - Tabela de custos do protótipo para teste.	40

## Lista de Abreviaturas e Siglas

- ABS – Acrilomitrila Butadieno Estireno
- AC – *Alternating Current*
- AGV's – Veículos Guiados Automaticamente
- CAD – *Computer Aided Design*
- CAM – *Computer-Aided Manufacturing*
- CC – Corrente Contínua
- CIM – *Computer Integrated Manufacturing*
- CL – Controle Local do Sistema de Tração e Direção
- CtB – Capacidade Teórica da Bateria
- DC – *Direct Current*
- Et – Energia Total Armazenada
- GGT – Gerador Global de Trajetória
- GLT – Gerador Local de Trajetória
- IDE – *Integrated Development Environment*
- MAPL – *Manufacturing Automated Planning Laboratory*
- NC – *Numeric Control*
- NI-MH – Níquel-hidreto Metálico
- PETG – Politereftalato de Etileno Glicol
- PLA – Ácido Polilático
- PWM – *Pulse Width Modulation*
- RepRap – *Replicating Rapid-Prototyper*
- RP – *Rapid Prototypes*
- SLA – *Stereolithography Apparatus*
- TCE – Tempo de Consumo Estimado
- TPU – Poliuretano Termoplástico
- USB – *Universal Serial Bus*



## Sumário

CAPÍTULO I - INTRODUÇÃO .....	1
1.1 OBJETIVOS .....	2
1.2 JUSTIFICATIVA.....	2
CAPÍTULO II - FUNDAMENTAÇÃO TEÓRICA.....	3
2.1 PROJETO NA ENGENHARIA.....	3
2.2 ROBÓTICA MÓVEL .....	5
2.2.1 HISTÓRICO .....	6
2.2.2 CLASSIFICAÇÃO DOS ROBÔS.....	7
2.2.3 CONTROLE E TRAJETÓRIA .....	8
2.2.4 MORFOLOGIA.....	10
2.3 ENXAMES DE ROBÔS.....	14
2.4 FABRICAÇÃO ASSISTIDA POR COMPUTADOR.....	15
2.4.1 MANUFATURA INTEGRADA POR COMPUTADOR.....	15
2.4.2 DESENHO ASSISTIDO POR COMPUTADOR .....	16
2.4.3 MANUFATURA AUXILIADA POR COMPUTADOR.....	16
2.5 IMPRESSORA 3D.....	17
CAPÍTULO III - METODOLOGIA .....	20
3.1 NECESSIDADE .....	20
3.2 PROJETO CONCEITUAL .....	20
3.3 ANTI PROJETO .....	21
3.4 DETALHAMENTO.....	22
3.5 RESULTADOS DISCUSSÕES E TESTES .....	22
3.6 ENTREGA DO PROJETO .....	23
CAPÍTULO IV - ANÁLISE DOS RESULTADOS.....	24

4.1	NECESSIDADE .....	24
4.2	PROJETO CONCEITUAL .....	26
4.3	ANTI PROJETO .....	27
4.4	DETALHAMENTO.....	36
4.5	RESULTADOS DISCUSSÕES E TESTES .....	37
4.5.1	DA FABRICAÇÃO.....	37
4.5.2	DOS CUSTOS.....	40
4.5.3	DEMONSTRAÇÃO DO MOVIMENTO DO ROBÔ .....	42
	CAPÍTULO V - CONCLUSÕES .....	52
	BIBLIOGRAFIA .....	54
	APÊNDICE A... ..	55
	APÊNDICE B.....	56
	APÊNDICE C.....	61

# CAPÍTULO I

## INTRODUÇÃO

Atualmente a maioria dos equipamentos utilizados na indústria, na área médica, em pesquisas e até mesmo em domicílios é constituída de alta tecnologia. São exemplos as *smart* TVs com acesso à internet, equipamentos de ressonância magnética que permitem imagens detalhadas de alta resolução de batimentos cardíacos e robôs industriais com alta precisão.

Devido ao avanço dessa tecnologia, os robôs, que antes predominavam na indústria (robôs manipuladores), começaram a evoluir e fazer parte de toda a vida humana, se expandindo para a área da saúde (biônica), entretenimento, agricultura, aplicação militar, exploração espacial e de oceanos e como sistemas de orientação. Essa evolução concentrou esforços na construção de robôs moveis e autônomos.

Com a miniaturização de *hardware*, evolução dos circuitos integrados, o desenvolvimento de *softwares* cada vez mais específicos e confiáveis e a evolução da comunicação entre sistemas, os robôs que antes eram utilizados individualmente para determinadas tarefas, ganharam uma nova vertente de estudo e desenvolvimento, a robótica cooperativa ou de enxame, cuja multiplicação de membros para uma única tarefa pode conferir maior rapidez na sua solução, ou redundância (aumentando sua segurança), ou ampliar o alcance e dinamismo da robotização seja qual for sua área de aplicação.

Segundo Bekey (2005), espera-se uma crescente ênfase em aumentar o número de robôs cooperativos, para atuar em atividades como a construção de uma estação de energia solar na órbita terrestre, habitação da Lua e de Marte, ou criar instalações submarinas exigindo centenas ou milhares de robôs. Portanto, tem se como desafio o desenvolvimento de robôs autônomos, colônias autossuficientes, capazes de gerar sua própria energia.

Segundo Romero (2014), no Brasil, essa área vem se desenvolvendo muito nos últimos anos. Ele aponta que existem vários grupos de pesquisa espalhados nas principais universidades brasileiras justificando cada vez mais o estudo e produção de conteúdo intelectual sobre o assunto.

Destaca-se assim a necessidade de se contribuir para o estudo e desenvolvimento na robótica móvel e cooperativa.

## **1.1 OBJETIVOS**

O objetivo desse trabalho é projetar e implementar um protótipo de robô móvel omnidirecional modelo para criação de enxame de robôs.

Para atingir esse objetivo é necessário definir objetivos específicos, tais como:

- Especificar os requisitos do robô omnidirecional para uso em enxames;
- Projetar a estrutura do robô;
- Construir um protótipo de custo reduzido viabilizando sua reprodução para formação de enxame;
- Realizar testes (Demonstração da movimentação do protótipo).

## **1.2 JUSTIFICATIVA**

Atualmente os enxames de robôs são utilizados na substituição do homem em tarefas arriscadas, maçantes ou até mesmo impossíveis de serem executadas devido às limitações fisiológicas próprias da espécie humana (por exemplo, na exploração do fundo dos oceanos). Portanto, a construção de um robô omnidirecional modelo para criação de um enxame é sustentada pela sua aplicabilidade no estudo e desenvolvimento na área, trazendo aprendizado na concepção de um projeto multidisciplinar e complexo desde a sua idealização até sua execução.

# CAPÍTULO II

## FUNDAMENTAÇÃO TEÓRICA

Esse capítulo apresenta a fundamentação teórica pertinente a esse trabalho do qual se destacam os temas: projeto em engenharia, robótica móvel; rodas omnidirecionais e impressora 3D.

### 2.1 PROJETO NA ENGENHARIA

Visado alcançar a solução de problemas técnicos, engenheiros devem utilizar de toda sua capacidade, ideias, conhecimentos e talentos a fim de conceber projetos que tenham técnica, economia que apresentem preocupação com a ecologia.

Para tanto é necessário gerenciar recursos materiais, tecnológicos e financeiros para atender os objetivos de cada tipo de projeto.

Segundo Pahl, *et al.* (2005), desenvolver e projetar são interesses da engenharia que:

- Abrangem quase todos os campos da atividade humana;
- Aplicam leis e conhecimentos das ciências naturais;
- Adicionalmente se apoiam no conhecimento prático especializado;
- São em grande parte exercidas sob responsabilidade pessoal;
- Criam os pressupostos para concretização de ideias de solução.

Uma importante ferramenta para a atividade do projetista é a informática. Ferramentas metodológicas como CAD (*computer aided design*) transformam a estrutura de trabalho, facilitam a criatividade e o processo mental. Além disso, disponibilizam informações sobre os

dados do projeto, soluções consolidadas, e podem conseqüentemente ser utilizadas para otimização de novos desenvolvimentos.

Com relação ao aspecto organizacional, a produção integrada ao computador (*computer integrated manufacturing* – CIM), facilita o planejamento e o controle do processo do projeto.

Aliado a toda essa utilização da tecnologia como auxiliar para a técnica de trabalho, o conhecimento e desenvolvimento de materiais são de suma importância. Isso inclui o auxílio da microeletrônica e software, possibilitando inúmeras soluções. Assim, segundo Pahl, *et al.* (2005), cada vez mais as futuras soluções serão buscadas no contexto da mecatrônica.

A aprendizagem continuada é necessária para atender a grande quantidade de requisitos impostos ao projetista. Ela deve equilibrar a formação fundamental de ciências naturais e engenharia, com a modernização representada pela tecnologia de medição e controle, mecatrônica, tecnologia de produção, conhecimentos aplicados a produtos e projetos, bem como metodologias de projeto.

Em geral a metodologia de um projeto de engenharia (PAHL, *et al.*, 2005) se inicia com o entendimento da necessidade do contratante ou cliente, essa etapa é de suma importância porque acrescenta ao conhecimento do projetista a visão do cliente e o conhecimento de causa por esse adquirido. É importante detalhar por meio de lista os requisitos do projeto, restrições e os fatores ergonômicos e econômicos desejados. Ao fim dessa etapa o problema a ser resolvido deve ser tão bem definido quanto possível.

A próxima etapa consiste no uso da criatividade, conhecimento técnico e experiência na área, para apresentação de ideias de solução de todos os níveis possíveis, essa etapa também chamada de projeto conceitual visa caracterizar função e forma do produto sem se preocupar com a viabilidade. A ideia é esboçar o produto tendo em conta as restrições, requisitos impostos pelo contratante e a coerência com o contexto de quem utilizará essa solução (PAHL, *et al.*, 2005).

Seguindo as etapas dessa metodologia, a próxima etapa também chamada de anti projeto tem por finalidade levantar os parâmetros para orientação e condução da implementação da melhor alternativa apresentada no processo anterior. Ela leva em conta a análise econômico-financeira, análise de dimensões, análise de engenharia (que abrange a

fabricação e a caracterização do processo produtivo) e inclusive o mercado (PAHL, *et al*, 2005).

Depois de concluídas essas etapas, o detalhamento do projeto é de suma importância e é a próxima etapa a ser realizada. Nela o projetista deve representar partes e conjuntos tecnicamente em desenhos, além de especificar dimensões e processos de produção de forma que todos esses dados possam ser armazenados e posteriormente analisados e utilizados para otimização ou reprodução da solução. Pode se ainda incluir nessa etapa a implementação, teste e análise da solução, de forma que, se determinado algum problema ou melhoria possível, essa etapa ou qualquer outra seja refeita para garantir o sucesso do projeto (PAHL, *et al*, 2005).

A metodologia de projeto específica para esse trabalho será apresentada nos capítulos que seguem e todas as características aqui apresentadas de cada etapa podem ser observadas.

## **2.2 ROBÓTICA MÓVEL**

O século XXI chega com grandes avanços na robótica que fazem com que ela se apresente como algo cotidiano nas vidas dessa geração (SECCHI, 2008). Atualmente os robôs são utilizados em inúmeras situações que contemplam desde a indústria até a área médica. Sua utilização se dá pela segurança e qualidade oferecidos por essa tecnologia. A Figura 2.1 ilustra alguns desses robôs e suas áreas de atuação.

Com relação ao setor industrial, segundo Secchi (2008), a robótica sempre ofereceu um excelente compromisso entre produtividade e flexibilidade, uma qualidade uniforme dos produtos, uma sistematização dos processos e critérios. Dessa maneira, destacam-se o aumento da produtividade, alta flexibilidade, qualidade e melhoria na segurança como consequência da robotização.

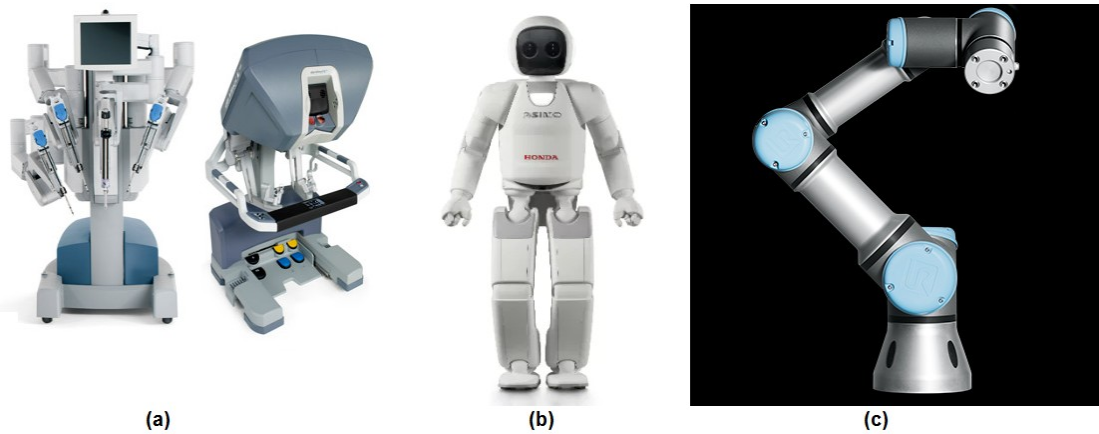


Figura 2.1 – Robôs modernos existentes no mercado atual (a) Robô cirurgião Da Vinci; (b) Robô Asimo da Honda; (c) Braço robótico UR3 da empresa *Universal Robots*.

Fonte: (a) (<http://www.mauriciorubinstein.com.br/sobre-a-cirurgia-robotica.asp>); (b) (<http://newatlas.com/new-honda-asimo-robot/32977/#gallery>); (c) (<https://www.universal-robots.com/pt/produtos/ur3/>). Acesso em 5 de Julho de 2017.

As últimas décadas testemunham um avanço significativo na área de robótica, muito em função dos novos recursos de *hardware* e *software* desenvolvidos. Em termos de *hardware* os computadores e dispositivos embarcados vêm sendo miniaturizados, tendo seus custos reduzidos e sua capacidade de processamento aumentada. Além disso, eles têm se tornado: mais robustos e precisos, consumindo menos energia e alcançando maior autonomia (ROMERO, *et al*, 2014). O desenvolvimento do *software* se reflete no desenvolvimento de programas de controle capazes de auxiliar na tomada de decisão, processamento de imagens, reconhecimento de voz e muitas outras funções que aliadas à evolução dos atuadores, possibilitam a sofisticação e capacidades dos robôs.

### 2.2.1 HISTÓRICO

No princípio dos anos 70 do século XX, introduziram-se na indústria os robôs manipuladores (SECCHI, 2008). Essa inserção, somada às inúmeras possibilidades de alteração desse novo elemento do processo produtivo incentivou pesquisadores a conseguir manipuladores mais rápidos, precisos e fáceis de programar. O avanço trazido originou uma nova etapa na automação industrial, que tornou a produção flexível com o surgimento da célula de fabricação robotizada.



Na maioria das vezes esses robôs eram utilizados em tarefas repetitivas, como na alimentação de máquinas presentes na célula de fabricação. Para que o mecanismo funcionasse, essas máquinas deveriam estar dispostas de tal maneira que estivessem acessíveis ao robô, numa área que compreendia a máxima extensão de suas articulações.

Mesmo com todo o sucesso desses robôs, eles sofriam de uma desvantagem fundamental: a falta de mobilidade (SIEGWART e NOURBAKHS, 2004).

De acordo com Secchi (2008), nos anos 80 do século XX, esse problema foi resolvido com o desenvolvimento dos Veículos Guiados Automaticamente (AGV's). Sua concepção inicial se baseava em um veículo móvel sobre trilhos capaz de transportar de forma eficaz materiais entre as diferentes zonas das cadeias de produção.

Esses veículos foram caracterizados pela baixa capacidade sensorial e de raciocínio, o que limitou a sua utilização em ambientes estruturados e com tarefas específicas e ordenadas em sequência. Isso quer dizer que, qualquer mudança capaz de afetar o desenvolvimento normal da navegação impossibilitava o veículo a retornar a sua atividade.

Segundo Siegart e Nourbakhsh (2004), em contraste a situação anterior, um robô móvel seria capaz de viajar por toda a fábrica, aplicando de forma flexível seus talentos onde ele é mais eficaz.

Por isso, nos anos 90 do século XX, surgiram os robôs móveis, veículos dotados de maior grau de inteligência e percepção, capacitando-os a desenvolver tarefa de propósito geral em qualquer tipo de ambiente.

### **2.2.2 CLASSIFICAÇÃO DOS ROBÔS**

Segundo Secchi (2008) existem três classes de robôs:

- Industriais;
- Médicos; e
- Móveis.

Os robôs industriais se caracterizam por ter estrutura mecânica articulada, que se move modificando as configurações das articulações segundo um controle de um equipamento

microprocessado. Eles podem mover elevadas cargas com ou sem velocidade elevada e com grande precisão nos movimentos.

Entre os robôs médicos existem os equipamentos desenvolvidos para tarefas cirúrgicas como o Da Vinci (Figura 2.1 (a)), capazes de realizar movimentos de grande precisão e complexidade; e os equipamentos de cooperação ou de reabilitação, tais como próteses inteligentes, que substituem membros e são capazes de realizar algumas funções mediante sinais nervosos ou musculares.

Os robôs móveis, segundo Secchi (2008), são dispositivos de transporte automático, ou seja, são plataformas mecânicas dotadas de um sistema de locomoção capazes de navegar através de um determinado ambiente de trabalho, dotados de certo nível de autonomia para sua locomoção, portando cargas.

A autonomia se refere a sistemas capazes de operar no ambiente do mundo real sem nenhuma forma de controle externo por longos períodos de tempo (BEKEY, 2005).

### **2.2.3 CONTROLE E TRAJETÓRIA**

Diferente dos robôs manipuladores, os robôs móveis trabalham em ambientes não estruturados e por isso tem que possuir a capacidade de “raciocinar” aquilo que foi alterado no ambiente para seguir em frente nas suas tarefas. Por isso, esses robôs tem um foco diferenciado, que segundo Secchi (2008), estão firmemente orientadas nas áreas de sensoriamento e “raciocínio”.

Dessa forma, o robô móvel deve gerar trajetórias para se guiar de maneira que não cometa erros. Essa trajetória só pode ser construída a partir de toda informação colhida dos sensores externos (ultrassom, laser, visão, entre outros) e deve ser suficiente para que o robô tenha maior autonomia possível.

Quando aplicados a sistemas físicos como robôs móveis, o planejamento e a reação são altamente complementares, sendo cada um crítico para o sucesso do outro (SIEGWART e NOURBAKHSI, 2004).

Por isso, essa percepção deve garantir a capacidade de determinar todas as variantes importantes do ambiente por meio de toda informação colhida pelos sensores com o objetivo de gerar os mapas específicos ou integrais do seu ambiente de trabalho, para que seu controle haja como se espera.

Além disso, o seu “raciocínio” deve ser capaz de tomar decisões a cada momento que este muda seu estado e sua posição em relação ao ambiente para alcançar seu objetivo. Essa capacidade se torna ainda mais evidente à medida que alterações no ambiente são dinâmicas e a máquina deve ser capaz de planejar, novas rotas seguras, ou modificar as anteriores para permitir que o mesmo alcance seu objetivo.

O esquema da Figura 2.2 apresenta a estrutura de controle de um robô móvel e as partes que compõem a arquitetura geral de controle.

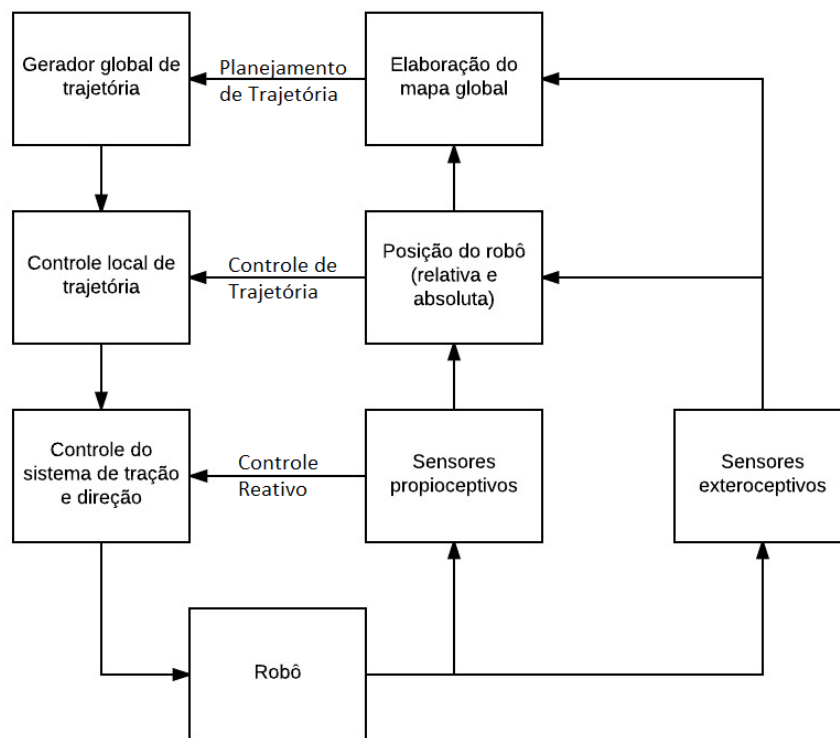


Figura 2.2 - Esquema geral do sistema de controle de um robô móvel. (Adaptado de Secchi 2008).

Segundo Siegwart e Nourbakhsh (2004), a classificação de sensores usam duas partes funcionais que se ligam: proprioceptivos/exteroceptivos e passivos/ativos.

Proprioceptivos medem valores internos ao robô, como velocidade dos motores, ângulos das juntas e a tensão da bateria. Enquanto que exteroceptivos adquirem informações do ambiente como medida de distâncias, intensidade da luz e amplitude do som. Isso quer

dizer que esses sensores captam medidas que devem ser interpretadas para que o sistema extraia características significativas do ambiente.

Com a captação de toda a informação gerada pelos seus sensores, o robô é capaz de gerar, de acordo com Secchi (2008), suas trajetórias globais e locais.

Gerador Global de Trajetória (GGT) seria o nível hierárquico superior, responsável por definir o caminho a ser seguido para realização de determinada tarefa e no caso de mudanças ao longo do caminho, redefinir essa trajetória.

Já o Gerador Local de Trajetória (GLT), como um nível intermediário, evita obstáculos e realiza correções necessárias para manobras. Ele mantém o robô atualizado e gera informação para ser armazenada pelo GGT. Pode ser desenvolvido com algoritmos clássicos ou até mesmo com inteligência artificial.

O Controle Local do Sistema de Tração e Direção (CL) é o nível inferior e interpretam as informações do GLT para gerar o controle dos motores de tração e direção, seus controles geralmente são desenvolvidos na teoria de controle clássico e buscam movimentação suave e que não causem problemas para o transporte da carga.

As estratégias de controle não são foco desse trabalho e, portanto, não serão abordadas.

#### **2.2.4 MORFOLOGIA**

Ao se tratar de morfologia, deve-se ter em mente que a tarefa que o robô realizará é aquilo que determina sua estrutura e características de tração e direção.

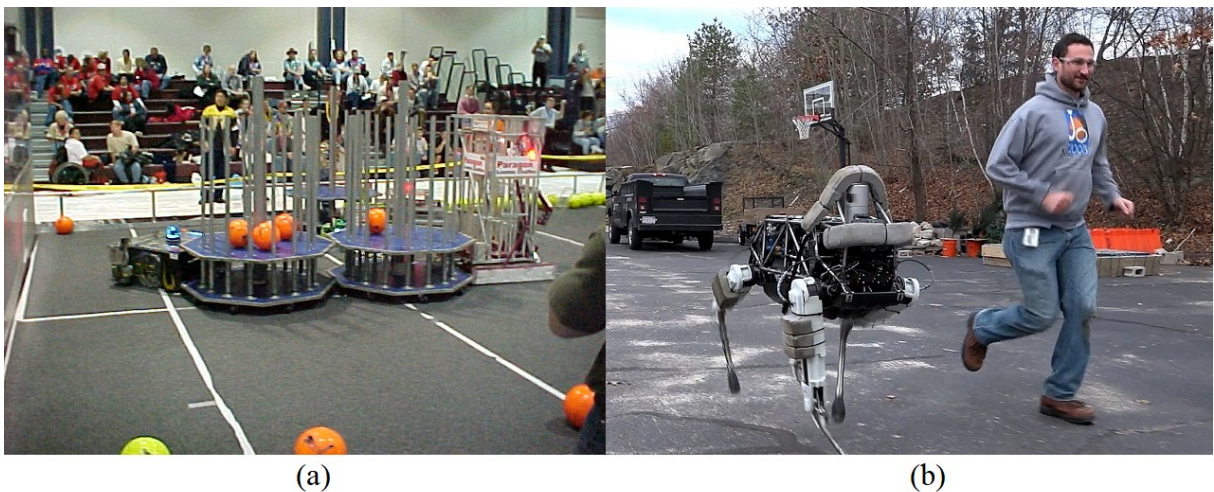
Sendo assim ao se tratar da morfologia dos robôs móveis, deve-se estabelecer: o ambiente em que o robô trabalha; seu sistema de locomoção; os tipos de rodas; a disposição das rodas; e sua tração e direção.

#### **AMBIENTE**

Segundo Secchi (2008), o ambiente pode ser interno ou externo. Quando o espaço de trabalho é delimitado por paredes e teto, caracteriza-se como interno, o qual na maioria das

vezes possui iluminação artificial. Quando o espaço não é delimitado e tem iluminação predominantemente natural, ele é chamado externo. A Figura 2.3 ilustra os dois tipos de ambientes citados.

Além disso, pode se ter, segundo a disposição de objeto, um ambiente estruturado ou não estruturado. No primeiro, objetos presentes são imóveis, e é possível distingui-los, pois possuem formas geométricas conhecidas. Já no segundo tipo de ambiente, bem mais dinâmico, podem ocorrer mudanças imprevisíveis, o que não permite determinação e caracterização dos objetos. A Figura 2.4 apresenta dois ambientes, um estruturado e outro não.



**Figura 2.3 – Diferentes ambientes de trabalho para robôs (a) Competição de robôs (ambiente interno); (b) Adaptação de frame da apresentação do robô Spot (ambiente externo).**

Fonte: (a) (<https://pt.wikipedia.org/wiki/Rob%C3%B4#/media/File:Zonezeal.jpg>); (b) (<https://www.youtube.com/watch?v=M8YjvHYbZ9w>). Acesso em 5 de Julho de 2017.



**Figura 2.4 - Ambiente estruturado (biblioteca) e não estruturado (depósito de linha de produção).**

Fonte: (SECCHI, 2008).

## LOCOMOÇÃO

Um robô móvel necessita de mecanismos de locomoção que garantam seu movimento dentro do seu ambiente de trabalho. Dadas às inúmeras possibilidades de formas de se locomover, esse é um parâmetro muito importante no estudo do design de um robô. Existem estudos inspirados na reprodução do que existe na biologia que possibilitam robôs a andar, pular, correr, nadar ou voar (Figura 2.5). Apesar disso, pode se dizer que rodas motorizadas são os mecanismos de melhor eficiência em solo plano (SIEGWART e NOURBAKHS, 2004).

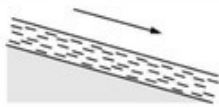
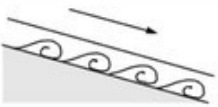

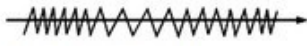

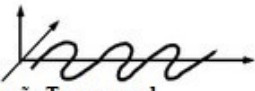






Tipo de movimento	Resistência ao movimento	Cinemática básica do movimento
Fluxo em um canal 	Forças hidrodinâmicas	Turbilhões 
Rastejar 	Forças de fricção	Vibração longitudinal 
Deslizar 	Forças de fricção	Vibração Transversal 
Correr 	Perda de energia cinética	Movimento oscilatório de um pêndulo duplo 
Pular 	Perda de energia cinética	Movimento oscilatório de um pêndulo duplo 
Andar 	Força gravitacional	Rolagem de um polígono 

Figura 2.5 - Mecanismos de locomoção utilizados por sistemas biológicos.

Fonte: Adaptado de (SIEGWART, *et al.*, 2004).

Sendo assim, considera-se que o robô pode ser terrestre: com patas, rodas ou esteiras. Também pode ser aquático (flutuante ou submarino) ou aéreo.

## TIPOS DE RODAS

Por se tratar de modelos de mecânica de implementação simplificada e boa efetividade os robôs com rodas motorizadas são os mais populares existentes. Além disso, por possuírem design que garante contato com o solo, os sistemas dotados de rodas tem equilíbrio estável, o que possibilita o projetista a se preocupar com outros problemas como tração, manobrabilidade e controle.

Para o design das rodas observam-se quatro grandes classes apresentadas na Figura 2.6.

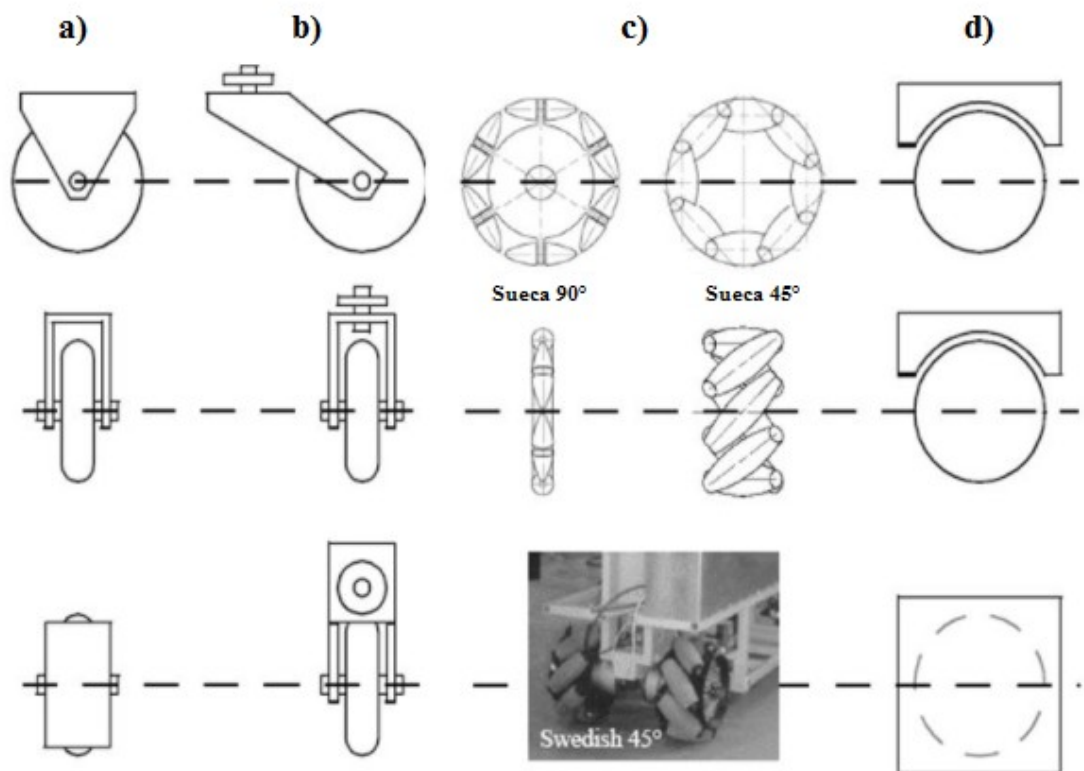


Figura 2.6 - Tipos de rodas: padrão (a), castor (b), sueca (c) e esférica (d).

Fonte: Adaptado do SIEGWART e NOURBAKSH, 2004.

A classe *a* representa uma roda padrão, a classe *b* uma roda castor, a classe *c* rodas suecas e a classe *d* rodas esféricas. A primeira e a segunda possuem dois graus de liberdade e precisam ser manobradas em relação ao eixo vertical para que girem em outra direção.

A roda sueca possui três graus de liberdade: rotação em torno do eixo motorizado, em torno dos seus roletes e em torno do seu ponto de contato. Sua principal vantagem é o movimento com baixa fricção por inúmeras trajetórias.

Dessa maneira, um arranjo triangular de três rodas Suecas (como mostra a Figura 2.7), torna possível o movimento omnidirecional, o que quer dizer que um robô com essa configuração de rodas pode se mover em qualquer direção sem que seja necessária a reorientação do mesmo.

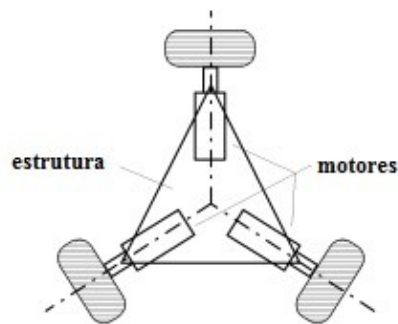


Figura 2.7 - Exemplo de disposição das rodas suecas sobre uma estrutura mecânica.

Fonte: Adaptado de (SECCHI, 2008).

A Figura 2.8 representa vários designs diferentes da roda sueca.



Figura 2.8 - Exemplos de roda Sueca.

Fonte: [http://www.societyofrobots.com/robot\\_omni\\_wheel.shtml#3vs4](http://www.societyofrobots.com/robot_omni_wheel.shtml#3vs4). Acesso em 5 de Julho de 2017.

## 2.3 ENXAMES DE ROBÔS

Existem inúmeras tarefas difíceis ou impossíveis para um único robô executar. Por isso, um crescente interesse em estudar e desenvolver grupos de robôs autônomos aparece na comunidade acadêmica. Segundo Bekey (2005), são exemplos desse tipo de tarefa:



exploração de ambientes perigosos ou inóspitos (o qual a falha de um robô não compromete a tarefa à medida que outro toma sua posição), carregamento de cargas complexas ou de grandes dimensões que inviabiliza o uso de um único robô ou em tarefas que possam ser realizadas de maneira mais rápida quando um grupo trabalha ao invés de um só membro (divisão de tarefas de sensoriamento e localização ou vários membros executando uma mesma atividade).

Esse grupo de robôs (também conhecido como enxame) pode ser constituído de robôs mais simples, de baixa capacidade individual que quando reunidos são capazes de executar as mais diversas tarefas (ROMERO, *et al*, 2014).

O desafio dessa área de estudo é desenvolver mecanismos que sejam escaláveis e controlados de forma eficiente.

Por se tratar de grandes grupos de robôs, algumas características são fundamentais para que um enxame seja constituído. São exemplos: a constante comunicação para possíveis mudanças entre os indivíduos em cooperação, a intercambiabilidade de membros sem a perda de informação (trabalha-se com redundância), a miniaturização e baixos custos por membro (ROMERO, *et al*, 2014).

## **2.4 FABRICAÇÃO ASSISTIDA POR COMPUTADOR**

Com o advento do computador, ferramentas complexas foram concebidas para serem utilizadas com o intuito de auxiliar processos de produção da indústria. Esses sistemas computacionais executam desde a digitalização do desenho da peça original até o controle das máquinas-ferramenta utilizadas na manufatura de diversos tipos de peças de diferentes materiais.

### **2.4.1 MANUFATURA INTEGRADA POR COMPUTADOR**

Manufatura integrada por computador (*Computer Integrated Manufacturing – CIM*), de acordo com Duimering *et al.* (1993), é um sistema que utiliza mecanismos de informação

computacional, com o objetivo de integrar ilhas de informação, bem como tecnologias de produção flexível e avançada, através do sistema organizacional de manufatura.

Essa tecnologia pode integrar várias áreas, entre elas: gerenciamento, planejamento, projeto, manufatura e distribuição. Nesse trabalho integram-se os sistemas de projeto e manufatura (CAD/CAM).

#### **2.4.2 DESENHO ASSISTIDO POR COMPUTADOR**

O desenho assistido por computador (CAD – *computer-aided design*) se destina a engenheiros, projetistas e qualquer outro profissional criativo. Ele oferece ferramentas para construção de modelos 3D, 2D, montagens de componentes, simulações, estimativa de custos, renderização (processo de obtenção do produto final de um processamento digital) e animação, oferecendo velocidade e facilidade de se trabalhar seja em projetos mecânicos ou elétricos (SOLID WORKS CORPORATION, 2003).

Além disso, ele possibilita que o arquivo criado possa ser traduzido e salvo numa grande variedade de aplicações de *software* e formatos de arquivo, sejam esses formatos nativos ao CAD, formatos CAD neutros, formatos para impressão 3D, imagem e documento, aplicativos externos entre outros padrões suportados.

#### **2.4.3 MANUFATURA AUXILIADA POR COMPUTADOR**

Entende-se por CAM (abreviação de *Computer-Aided Manufacturing*), ou manufatura auxiliada por computador, como sendo parte do processo de produção no qual o desenho CAD e todas as informações necessárias para a fabricação do produto são traduzidos em um programa, de comando específico para a máquina. É por meio do sistema CAM que se podem transferir todas as coordenadas e especificar a trajetória para que máquina-ferramenta efetue a fabricação da peça.

Vale dizer que sistemas CAM também são utilizados para cálculo de tempos de fabricação e simulação de operações de usinagem (tornos e fresas), corte a laser, corte a jato d'água, entre outros.

## 2.5 IMPRESSORA 3D

O início da impressão 3D ocorreu na década de 1980 (3DPI – 3D PRINTING INDUSTRY, 2017). Conhecidas por *Rapid Prototypes* (RP), essa tecnologia se diferenciava pela rápida prototipagem de forma mais econômica quando comparada a outros métodos de fabricação da indústria da época.

Tem-se registro que a origem oficial da impressão 3D ocorreu em 1983, com a patente do primeiro aparato de estereolitografia (*stereolithography apparatus - SLA*), tecnologia desenvolvida para prototipagem rápida, criada para produção e desenvolvimento de componentes 3D digitais de um projeto CAD, criada por Charles W. Hull, cofundador da *3D System Corporation*, uma das maiores organizações no setor (3DPI – 3D PRINTING INDUSTRY, 2017).

Após muita evolução, criação de ferramentas específicas e a queda de patente, em 2004 originou-se um projeto *open source*, ou seja, um projeto de conteúdo aberto, criado, publicado e desenvolvido de forma que qualquer pessoa possa construir sua própria impressora ou desenvolver novos equipamentos a partir dela. Inventada por Adrian Bowyer, essa impressora traria para o mundo, uma maneira de se ter impressoras 3D básicas a um custo muito menor do que as existentes no mercado, cerca de cem vezes mais baratas.

Apresentada pela comunidade RepRap (abreviação de *Replicatin Rapid-prototyper*), ela é primeira geração de máquinas de manufatura de auto-replicação de propósito geral. Essa máquina tem a forma de uma impressora 3D a ser usada numa superfície como uma mesa de trabalho comum. Ela é constituída de partes plásticas e componentes já existentes no mercado, como motores de passo, roscas sem fim, parafusos entre outros. Possibilita a impressão de peças reais, robustas e mecânicas, sendo assim, ela inclusive como o próprio conceito diz, pode imprimir peças a serem usadas para montar outra impressora semelhante a original. A estrutura geral de uma impressora RepRap é apresentada na Figura 2.9 (BOWYER, 2004).

Essa estrutura apresenta todo o esquema de *software* e *hardware* de uma impressora 3D baseada no modelo RepRap.

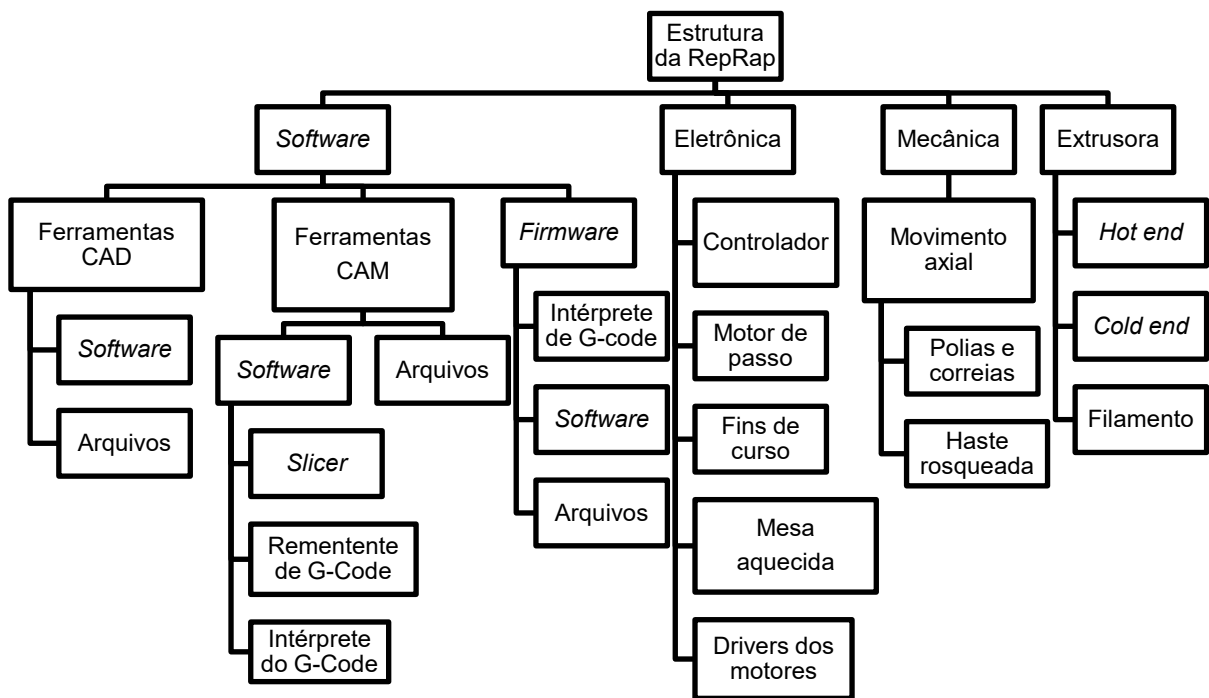


Figura 2.9 - Organograma da estrutura RepRap.

Fonte: Adaptado de imagem do site [http://reprap.org/wiki/File:RepRap\\_Component\\_Structure.svg](http://reprap.org/wiki/File:RepRap_Component_Structure.svg). Acesso em 18 de Julho de 2017.

O início do processo de fabricação se dá na criação e digitalização de um modelo da peça desejada, e depois a transferência desta geometria para um sistema CAM, com a finalidade de se obter um programa de Controle Numérico (NC, *Numeric Control*) para que a manufatura seja realizada.

Como explicado na seção 2.2, o sistema CAD é então utilizado para geração e detalhamento da peça e a ferramenta CAM é a responsável por calcular os parâmetros ótimos da fabricação e gerar o programa que será fornecido para a máquina ferramenta.

Uma das maneiras utilizadas de se conceber o programa de comandos específicos para a máquina é por meio de *Slicers*, rotina que fatia o modelo CAD em camadas e tem como saída o código G (G-code) de cada camada. G-code é a denominação dada para uma linguagem específica NC, que nada mais é do que as operações que a máquina-ferramenta deve executar em formato de programa codificado em um meio de armazenamento.

Sendo assim, de forma resumida a descrição completa do método de fabricação utilizado por esse tipo de impressora tem a seguinte rotina: faz-se um desenho em CAD que é transferido para um sistema CAM que gera o G-code. Depois disso, a máquina ferramenta

interpreta esse código e realiza as operações necessárias para a extrusão de material camada por camada para a produção da peça.

O material de uma impressora desse tipo varia, podendo ser de Ácido Polilático (PLA), ou o melhorado desse mesmo tipo (PLA+), ou Poliuretano Termoplástico (TPU), ou Politereftalato de Etileno Glicol (PETG), ou Acrilonitrila Butadieno Estireno (ABS), todos, materiais termoplásticos. O primeiro derivado de fontes renováveis como amido de milho e os dois últimos derivados de petróleo. Cada material tem suas características que o diferenciam e essas características são apresentadas na Tabela 2.1 (**IMPRESSÃO 3D FÁCIL, 2015**), a última linha da tabela foi adaptada para se tornar coerente com os valores atuais de mercado.

**Tabela 2.1 - Tabela comparativa dos materiais.**

<b>Propriedade/Desempenho</b>	<b>Menor</b>	<b>Médio</b>	<b>Maior</b>
<b>Ecológico</b>	ABS	PETG	PLA
<b>Brilho</b>	ABS	PLA	PETG
<b>Transparência</b>	ABS	PLA	PETG
<b>Rigidez/Dureza</b>	PETG	ABS	PLA
<b>Resistência a impactos</b>	PLA	ABS	PETG
<b>Flexibilidade</b>	PLA	ABS	PETG
<b>Contração</b>	PLA	PETG	ABS
<b>Precisão em detalhes e cantos</b>	ABS	PETG	PLA
<b>Qualidade de superfície da peça</b>	ABS	PETG	PLA
<b>Resistência a atritos</b>	PLA	ABS	PETG
<b>Resistência Química</b>	ABS	PLA	PETG
<b>Resistência altas temperaturas</b>	PLA	PETG	ABS
<b>Facilidade de pós-processamento</b>	PLA	PETG	ABS
<b>Usinabilidade</b>	PLA	PETG	ABS
<b>Densidade</b>	ABS	PLA	PETG
<b>Preço por cm<sup>3</sup>/grama/hora</b>	ABS	PLA	PETG

## **CAPÍTULO III**

### **METODOLOGIA**

A metodologia de desenvolvimento do projeto foi dividida nas etapas representadas no fluxograma da Figura 3.1. A especificação de cada atividade do fluxograma é feita a seguir.

Esse projeto foi idealizado, concebido e executado com o auxílio do laboratório de pesquisa Planejamento Automático da Manufatura (*Manufacturing Automated Planning Laboratory* - MAPL) e seus membros.

#### **3.1 NECESSIDADE**

O problema apresentado foi o interesse no estudo em robótica móvel no âmbito dos exames de robôs. Para iniciar essa linha de pesquisa foi proposto esse projeto que objetiva a modelagem e construção da estrutura modelo para um exame de robôs.

Os requisitos foram apresentados e fez-se uma lista de forma a definir a morfologia do robô que atende a sua finalidade.

#### **3.2 PROJETO CONCEITUAL**

Com a morfologia definida, por meio de *brainstorm*, esboços de desenhos e listas (requisitos, características e limitações), algumas opções de forma foram apresentadas e suas funções definidas.

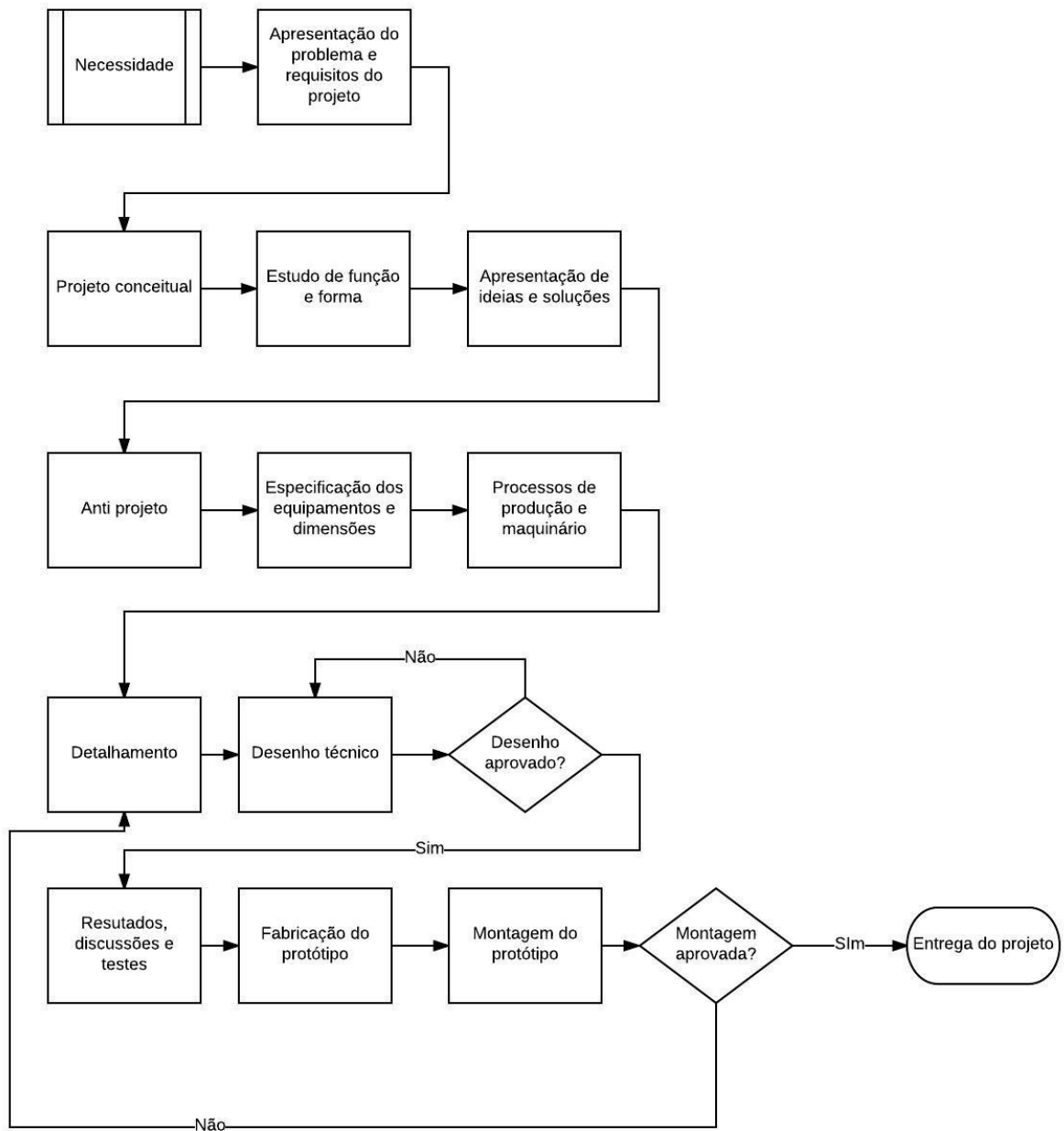


Figura 3.1 - Fluxograma da metodologia de desenvolvimento do projeto.

### 3.3 ANTI PROJETO

Restrições de custo econômico, estético e a escolha de materiais, bem como métodos de fabricação foram definidas de forma que a estrutura do laboratório fosse bem utilizada e colaborasse para a economia do projeto. Como a finalidade do projeto é a replicação do robô, essa escolha facilita a sua execução e valoriza o aparelhamento do laboratório.

Todos os equipamentos e componentes do projeto foram utilizados no próprio laboratório. Sendo assim, fez-se uma lista daquilo que estava disponível com relação à necessidade já apresentada.

As dimensões do robô foram definidas depois de analisados e mensurados os componentes que seriam inseridos nele, junto à necessidade estudada e sua finalidade.

Definiu-se então a proposta de robô a ser fabricado, o método de fabricação seus processos, máquinas e ferramentas.

### **3.4 DETALHAMENTO**

Definida a proposta a ser executada. O desenho das peças e do conjunto a ser construído foi feito. Como foi explicado na metodologia de um projeto de engenharia, utilizou-se de *software* para conceber a ideia. Dessa forma, fica fácil visualizar o produto acabado e modificá-lo caso haja necessidade.

A avaliação do desenho leva em conta o respeito às dimensões fixadas pelos requisitos, bem como a forma da estrutura, que deve ser a mais reduzida possível e resistente suficiente para suportar os componentes que serão fixados nas suas plataformas.

Por fim detalha-se a fabricação, seus processos e maquinário.

### **3.5 RESULTADOS DISCUSSÕES E TESTES**

Nessa etapa, além da fabricação do protótipo, foram registradas as quantidades de material utilizado e o tempo de trabalho da impressora para confecção de cada peça registrando medidas para se calcular os gastos relacionados à construção.

Além disso, registraram-se todos os erros relacionados ao processo de fabricação bem como suas soluções. Nessa etapa, o conhecimento do funcionamento da impressora, sua programação e pelo menos o conhecimento superficial do seu *hardware* são necessários.



A montagem do protótipo conta com a etapa de trabalhar no acabamento das peças e a montagem em si. Analisou-se a conformidade das peças e do conjunto para que caso houvesse algum problema o processo pudesse ser reavaliado e corrigido.

Para que a montagem fosse adequada, tanto as medidas pré-estabelecidas e suas tolerâncias deveriam ser respeitadas, como o conjunto deveria se comportar como o esperado, isso quer dizer que, os roletes das rodas deveriam deslizar com o menor atrito possível e a montagem das peças deveria acontecer de forma que todos os encaixes fossem feitos sem provocar tensão indesejada e sua função de fixação feita.

Por fim, trabalhou se na demonstração da movimentação do robô, testes com a montagem já realizada (inclusive com os dispositivos necessários para os testes de movimento), programação do microcontrolador, análise dos custos do projeto e do processo de fabricação, comparação de custos com o que existe no mercado, cálculo de autonomia das baterias e os problemas enfrentados durante toda a pesquisa e suas soluções encontradas.

### **3.6 ENTREGA DO PROJETO**

Ao término de todas as discussões e testes realizados com o robô móvel, bem como os desenhos das peças e montagem da sua estrutura, o robô será de posse da Universidade Federal de Uberlândia e poderá ser usado principalmente pelo MAPL para produção de outros robôs e para estudo de enxame de robôs, objetivo desse trabalho.

## CAPÍTULO IV

### ANÁLISE DOS RESULTADOS

#### 4.1 NECESSIDADE

A fundação do escopo do projeto é baseada em requisitos mostrados na Tabela 4.1.

Tabela 4.1 - Requisitos do projeto.

	REQUISITOS	INTENÇÃO do requisito	Requisito Explícito?	Prioridade
1	Robô omnidirecional para enxame	Estudo projeto e construção desse tipo de robô	Sim	Alta
2	Para uso interno	Prática e desenvolvimento em ambiente controlado	Sim	Alta
3	Custo Reduzido	Construção de número suficiente de robôs para estudo de enxame	Sim	Alta
4	Materiais	Projeto com uso de material existente para minimizar custo	Não	Alta
5	<i>Small size (World Cup)</i>	Requisito de robô em específico	Sim	Alta
6	Teste prático do processo de produção	Estudo do processo de fabricação e material utilizado	Não	Média

A escolha do sistema de locomoção composto por três rodas omnidirecionais se dá pela novidade em se pesquisar e desenvolver com esse tipo de sistema dentro do laboratório em que o projeto foi realizado.

O requisito custo reduzido é fundamental, pois o projeto tem como objetivo a construção de um modelo que será replicado, portanto, essa reprodução de várias outras unidades deve ser economicamente viável. Como foi dito, está inserida nesse custo reduzido, a utilização de material já adquirido pelo laboratório, utilização de equipamentos e fabricação no próprio laboratório e pesquisa de robôs semelhantes para comparação de preço.

Existem ainda limitações dadas por restrições internas ou externas ao projeto. Elas são apresentadas na Tabela 4.2.

**Tabela 4.2 - Restrições do projeto.**

RESTRICÇÕES	
1	Motores DC
2	Alimentação dos motores por bateria
3	Fabricação do chassi e peças em impressora 3D
4	Material da impressão ABS

A restrição de motores DC (*Direct Current* ou Corrente Contínua) e alimentação por baterias se explica por serem materiais já adquiridos pelo MAPL, além de serem materiais de baixo custo relativo e de dimensões que se adéquam ao requisito de construir um robô de dimensões restritas (*small size*).

A escolha por construir um robô com as dimensões da categoria *small size* se justifica porque o próprio desenvolvimento de enxame pode ser aplicado a competições, à medida que o mesmo seja planejado para se qualificar noutros aspectos necessários.

Partiu-se da premissa de que ampliações no projeto devem ser consideradas e, portanto, devem ser criadas plataformas para instalar placas com *drivers* dos motores ou placas de aquisição de dados (sensoriamentos) ou carregamento de materiais, em níveis (camadas acima do chassi – plataforma dos motores e rodas).

## 4.2 PROJETO CONCEITUAL

No processo de *brainstorm*, a ideia de como o robô seria começaram a ser definidas. Mais de um formato de chassi foi criado de forma a comportar o sistema de tração com 120° graus de angulação entre os eixos motores conforme deve ser para o funcionamento das três rodas omnidirecionais. A Figura 4.1 são capturas de tela dos esboços das formas que as plataformas poderiam ter.

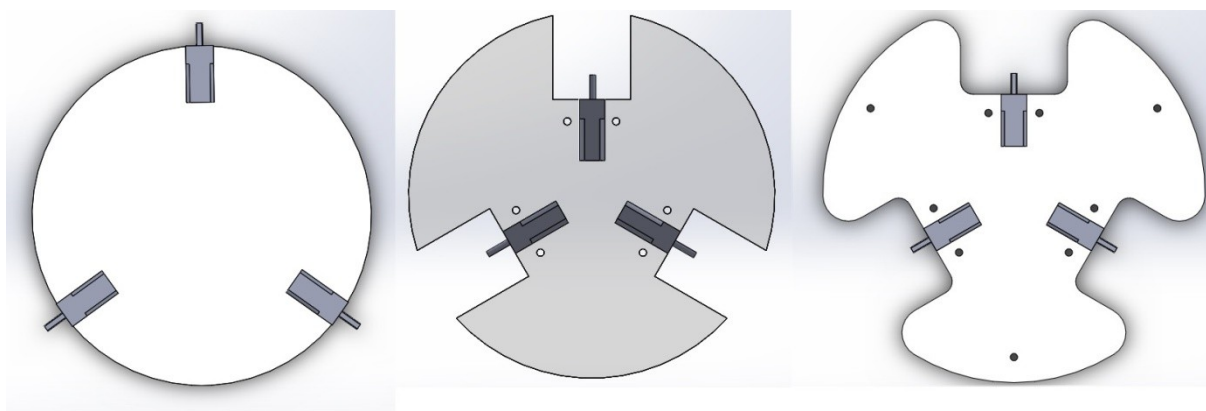


Figura 4.1 - Esboços das plataformas.

Fonte: Próprio autor.

A primeira opção se tratava de um disco maciço que foi trabalhado de forma a se retirar material desnecessário e comportar melhor as rodas, diminuindo a área ocupada pelo robô, o desperdício de material e gasto que seria gerado por esse desperdício.

As plataformas idealizadas foram definidas como discos maciços com furos para a colocação de dispositivos, componentes e placas, essas não foram alteradas porque a necessidade maior de espaço para os componentes justificava seu formato.

As rodas foram criadas por meio de pesquisa e comparação com o que já existe e durante a pesquisa não se encontrou um método específico de modelagem, portanto, inspirados pelas formas existentes, criou-se uma roda de forma que sua estrutura e rigidez não fossem prejudicadas, já que se desejam dimensões reduzidas.

Idealizaram-se compartimentos para colocação de baterias e suportes para fixação de componentes como os motores. Os desenhos finais e parciais serão apresentados na parte de detalhamento do projeto.

Criou se também hastes que suportariam as plataformas dispostas em camadas.

Os elementos de fixação dessas peças ficaram definidos como sendo parafusos e porcas de pequenas dimensões que não alterassem muito o peso do conjunto e que oferecessem fixação suficiente para que o conjunto não ficasse frouxo, possibilitando grandes variações com o movimento do robô.

Os eixos dos roletes das rodas omnidirecionais poderiam ser ou de material já existente no mercado, como eixos de aço maciço de pequeno diâmetro, ou fabricados no mesmo material das outras peças.

Para fabricação destinou-se dois métodos: a terceirização e busca pela confecção em acrílico para as plataformas ou a fabricação própria no laboratório MAPL em ABS com a impressora 3D disponível no mesmo.

Por fim determinou-se que seriam realizados testes com o robô com a finalidade de demonstrar sua movimentação. Para isso, se utilizaria de componentes de *hardware* Arduino, uma plataforma de prototipagem rápida e de baixo custo de aquisição. Essa ferramenta já possui várias placas (*Shields*) com inúmeras funções que auxiliam e facilitam o trabalho do projetista, tanto na parte de *hardware* como em *software*.

### 4.3 ANTI PROJETO

De posse de todas as opções, listas e estudo de viabilidade das possíveis soluções descritas na seção anterior, decidiu-se que a economia no projeto era prioridade para que a sua reprodução e estudo de enxame fosse facilitado.

De acordo com o item 4.1, determinou-se que o robô deve caber em um cilindro de 180 mm de diâmetro e ter altura máxima de 150 mm (**SMALL SIZE LEAGUE TECHNICAL COMMITTEE**, 2016).

Dessa forma, a utilização de processos de fabricação e máquinas do laboratório foi firmada. Para a fabricação decidiu se pela impressora 3D, o material escolhido foi o ABS.

A impressão do chassi e peças em impressora 3D se explica por ser um método amplamente aplicado na atualidade por facilitar a prototipagem de projetos, com material

(ABS) adequado à função das peças a serem impressas. Conforme comprovado na seção 2.4 (Tabela 2.1), o ABS apresenta vantajoso equilíbrio entre:

- Rigidez/dureza, resistência a impacto, flexibilidade e resistência ao atrito (sendo colocado nessas características como o material para impressão de qualidade média);
- Resistência à alta temperatura, facilidade de pós-processamento e usinabilidade (colocado como de alta qualidade);
- Preço (avaliação média).

Ainda que esse material possua baixa avaliação nos quesitos ecológicos, de brilho, transparência, precisão nos detalhes de cantos e qualidade da superfície das peças, essas características não foram consideradas como prioridade e não impactam de forma negativa a ponto de que a solução final fosse alterada.

A impressora 3D disponível é do modelo Prusa Mendel, ela possui *firmware* de nome Marlin que se comunica com um dos computadores do laboratório MAPL que possui o remetente de G-code chamado Repetier e inclui o *software* CAM *Slicer Curaengine*.

A Figura 4.2 é uma foto tirada da impressora no seu local de trabalho.

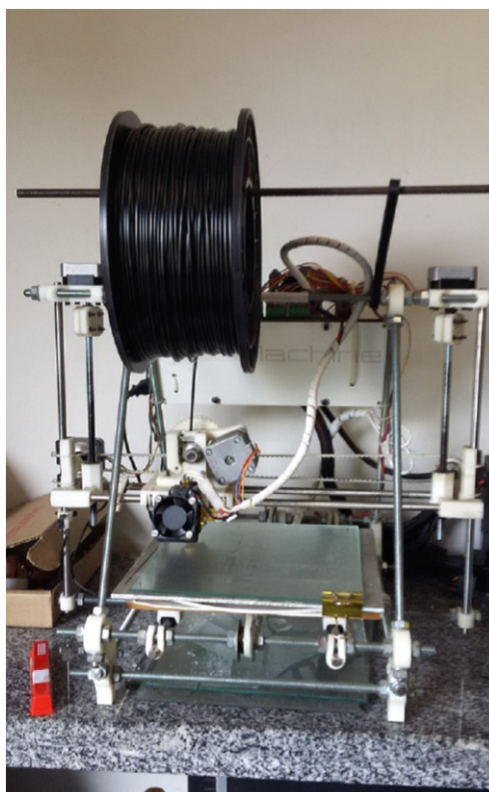


Figura 4.2 - Impressora Prusa Mendel no seu local de trabalho.

O *software* de CAD escolhido é o SolidWorks®. Ele oferece inúmeras ferramentas de forma simples e direta com botões para cada tipo de funcionalidade.

O desenho do protótipo foi feito em escala real, dada em mm no *software* SolidWorks®, que além de ser uma ferramenta gráfica amigável, proporciona a opção de ser gravado no arquivo de extensão .STL, formato de extensão utilizado para a impressão nos *softwares* utilizados na impressoras 3D.

O formato decidido para o chassi seria aquele que conforma as rodas e reduz sua área de ocupação.

As rodas foram decididas em comparação de detalhes e robustez de forma que o processo de fabricação escolhido oferecesse detalhamento e precisão para uma construção bem feita na impressora 3D. Essa escolha poderia sofrer alterações depois da análise da montagem.

Dessa forma, rodas, roletes, hastes de fixação das plataformas, as próprias plataformas, compartimentos para acoplamento de motores e eixos dos roletes seriam fabricados em ABS na impressora 3D (e terão seus desenhos parciais e totais mostrados na seção DETALHAMENTO).

Os motores utilizados (Figura 4.3) seriam motores de tipo DC, de 12 V de tensão, 100 rpm de velocidade, com construção do tipo excitação shunt, à prova de gotejamento da marca OEM, de engrenagens, com 16 Kgf.cm de torque, comutação por escova e possui redução na transmissão. Sua escolha se explica pelo fato de ser material já obtido pelo laboratório de baixo custo e dimensões que se adequam à necessidade.

De acordo com Fitzgerald (2006), uma máquina de corrente contínua (CC) como o motor DC tem enrolamentos de armadura (também chamado de rotor, que é a parte rotativa do sistema) e enrolamentos de campo (conhecido como estator). Seu princípio de funcionamento se baseia no fato de que o efeito da corrente contínua no enrolamento de campo cria um fluxo magnético em relação ao estator. De modo similar o efeito do comutador é tal que, quando uma corrente contínua flui através das escovas, a armadura também cria uma distribuição do fluxo magnético. A interação dessas duas distribuições de fluxo cria um conjugado que atua no sentido da rotação.

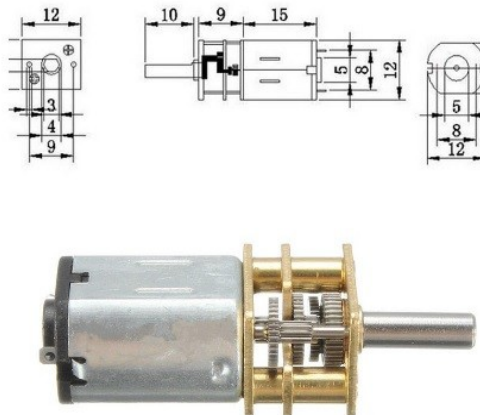


Figura 4.3 - Imagem ilustrativa do motor utilizado.

Fonte: <https://www.aliexpress.com/item-img/New-DC-12V-100RPM-Mini-Metal-Gear-Motor-with-Gearwheel-N20-3mm-Shaft-Diameter/32767629574.html>. Acesso em 6 de Julho de 2017.

A alimentação desse motor não pode ser feita diretamente do microcontrolador Arduino Mega, já que normalmente sua corrente necessária supera a das portas digitais desse componente. Sendo assim, foi utilizado um *driver* baseado no circuito integrado L298N (Figura 4.4) que possui ponte H dupla.

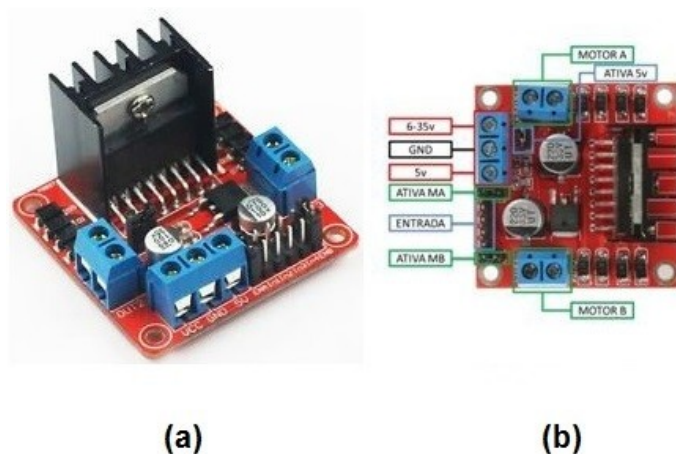


Figura 4.4 – *Driver* ponte H dupla para motores DC baseado no *chip* L298N (a) Módulo dupla ponte H L298N; (b) Entradas e saídas do módulo.

Fonte: <http://blog.vidadesilicio.com.br/arduino/modulo-ponte-h-l298n-arduino/>. Acesso 6 de Julho de 2017.

As características desse *driver* são apresentadas na Tabela 4.3.

É importante dizer que esse *driver* L298N pode ser alimentado de 4 V até um limite de 35 V (Tabela 4.3), sendo assim, pode ser usado para testes alimentado por baterias, por microcontroladores ou fontes externas.



Tabela 4.3 - Características do Módulo ponte H L298N.

<i>Driver Motor</i>	<b>Ponte H L298N</b>
<b>Tensão de operação</b>	4~35V
<b>Chip</b>	ST L298N
<b>Controle de 2 motores DC ou 1 de passo</b>	
<b>Corrente de operação máxima</b>	2A por canal ou 4A max
<b>Tensão lógica</b>	5V
<b>Corrente lógica</b>	0~36mA
<b>Limites de temperatura</b>	de -20 a 135°C
<b>Potência máxima</b>	25W
<b>Dimensões</b>	43X43X27mm
<b>Peso</b>	30g
<b>Potência dissipada</b>	25W

A ponte H é uma importante ferramenta utilizada para controlar o sentido de motores DC com possibilidade de se controlar também sua velocidade. É um circuito construído com transistores que chaveiam o sinal de entrada (E1 e E2) de forma alternada controlando o sentido do motor (alterando-se o sentido de corrente, altera-se o sentido de giro do motor). Além disso, como um motor DC consome corrente maior que a fornecida pela dos microcontroladores utilizados, necessitamos que o sinal de controle proveniente da saída digital da ponte seja capaz de suprir o consumo do motor. Para isso, o transistor também é utilizado como amplificador. A Figura 4.5 ilustra a configuração de uma ponte H.

Pode se concluir que polarizando  $Q_1$  e  $Q_4$  a corrente possui um sentido e polarizando  $Q_2$  e  $Q_3$  a corrente circula no sentido oposto pelo motor, dessa maneira, a partir de estímulos externos podemos controlar qual o sentido de rotação dos motores.

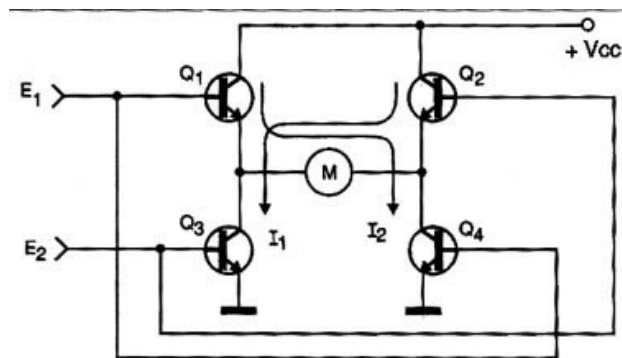


Figura 4.5 - Exemplo de configuração de circuito ponte H.

Fonte: <http://www.newtoncbraga.com.br/index.php/artigosmec/1213-ponte-h-com-pwm.html>. Acesso 6 de Julho de 2017.

Dado esse funcionamento da ponte H, deve se ter em mente que o módulo L298N possui duas pontes H e, portanto, pode controlar dois motores por placa. Para entender melhor o funcionamento desse componente, será apresentada a relação de seus pinos e suas funções.

Na Figura 4.4 (a) temos a identificação de pinos:

- Motor A e Motor B: conectores para os dois motores;
- 6- 35 V: porta de alimentação da placa com seus limites de funcionamento;
- Ativa 5 V: quando jumpeado, um regulador de tensão de 5 V da placa é utilizado e por isso não há a necessidade de alimentação da porta de 5 V e essa pode ser usada como fonte de alimentação de 5V;
- 5 V: em caso de não haver fonte de alimentação de mais de 6 V pode se alimentar a placa por essa entrada com esse valor de tensão;
- Ativa MA ou MB: quando alimentado aciona o motor com velocidade máxima, podendo ser utilizadas para se controlar essa velocidade tirando o jumper e alimentando o pino de 0 – 5 V onde 0 é o estado parado e 5 V a velocidade máxima;
- IN 1, 2, 3 e 4: cada um desses pinos serve para se controlar o sentido do seu respectivo motor (IN 1 e 2 -> Motor A; IN 3 e 4 -> Motor B) e seu esquema e mostrado na Tabela 4.4.

Tabela 4.4 - Esquema de funcionamento dos pinos IN do Módulo L298N.

IN1	IN2	ESTADO
0 V	0 V	Desligado
0 V	5 V	Sentido 1
5 V	0 V	Sentido 2
5 V	5 V	Freio

Outra vantagem do L298N é a resposta a sinais PWM (*Pulse Width Modulation* ou Modulação de Largura de Pulso). Se ao invés de sinais lógicos, utilizarmos sinais PWM é possível regularmos a tensão de saída e dessa forma, a velocidade do motor.

O PWM consiste em se enviar pulsos digitais em determinada frequência fixa (500 Hz no arduino) no nível de tensão total do sistema (no Arduino 5 V), ou seja, o sinal de saída do sistema é uma onda quadrada que alterna o pino em ON/OFF (ligado/desligado) rapidamente. A quantidade de tempo que essa onda dura é chamada de largura de pulso (*pulse width*). O que se faz é controlar a porcentagem de tempo em que a onda permanece em nível lógico alto, conhecida como taxa de largura de pulso ou ciclo de trabalho (*Duty Cycle*), dada pela razão  $T_{on}/T_{período}$  (BRÄUNL, 2006). Dessa maneira o ciclo de trabalho provoca uma mudança no valor médio da onda (Figura 4.6). Para o Arduino, esse variação pode ser de 0 V (0% de ciclo de trabalho) a 5 V (100% de ciclo de trabalho). Esse valor é traduzido na programação do Arduino em valores de 0 a 255 (dado que o registrador do timer dessa plataforma tem apenas 8 bits) (ARDUINO, 2017).

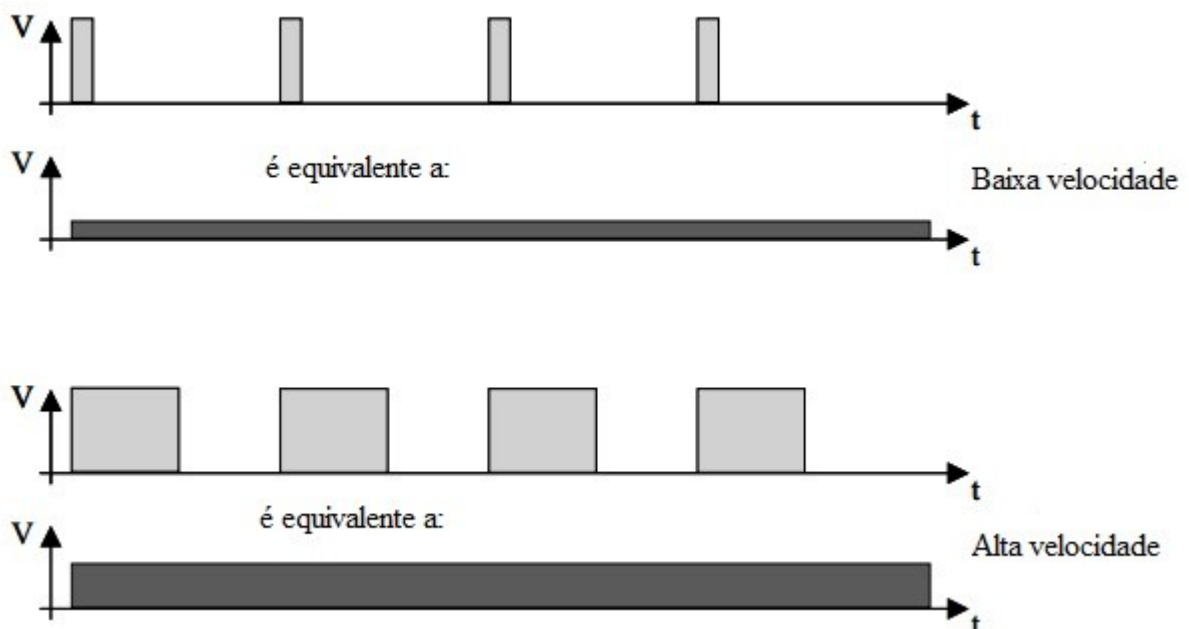


Figura 4.6 - PWM.

Fonte: Adaptado de (BRÄUNL, 2006)

A Figura 4.7 mostra o gráfico do ciclo de trabalho dos pinos PWM da plataforma Arduino.

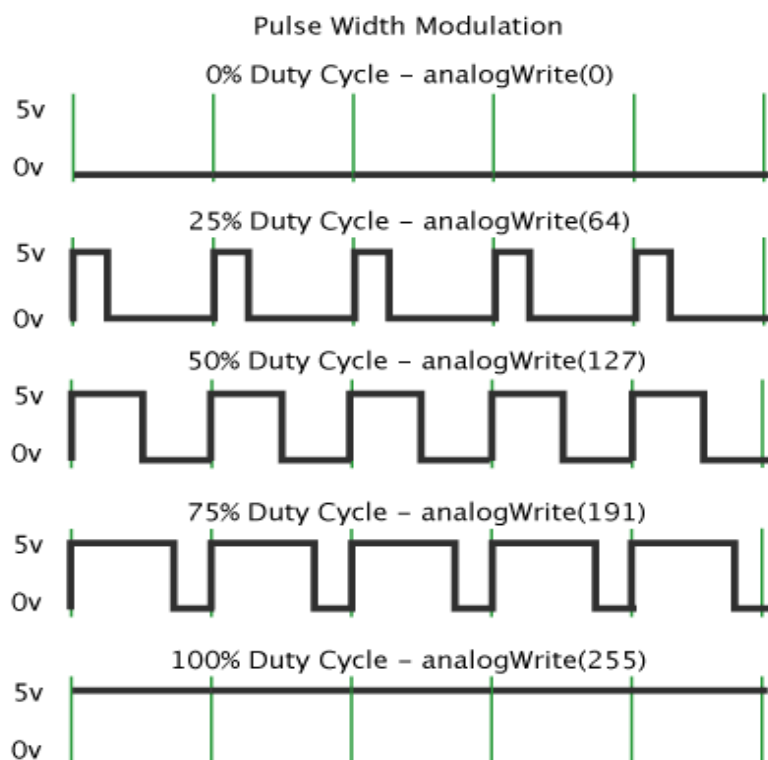


Figura 4.7 - PWM no Arduino.

Fonte: [www.arduino.cc](http://www.arduino.cc). Acesso em 1 de Agosto de 2017.

Para a programação e controle do módulo L298N, utilizou-se o microprocessador Arduino, de características dadas na Tabela 4.5. A Figura 4.8 ilustra esse componente.

Para se utilizar esse microcontrolador, primeiro se conecta a placa a um computador via IDE (*Integrated Development Environment*, ou Ambiente de Desenvolvimento Integrado) e depois programa-se com uma linguagem baseada em C/C++, transferindo esse código via cabo USB. Como o Arduino tem *software open source*, tanto a IDE como códigos básicos e instrutivos podem ser visualizados no site <https://arduino.cc>, onde também se encontram fóruns de programadores que buscam desenvolver essa plataforma, com inúmeras informações sobre *Shields* que podem ser usados junto ao Arduino.

Depois de programada, pode-se utilizar esse microcontrolador por fonte externa (AC-DC ou baterias) ou com o próprio cabo USB ligado a um computador como alimentação (Respeitando os valores de tensão de operação da Tabela 4.5).



Figura 4.8 - Arduino Mega e Genuino Mega.

Fonte: <https://www.arduino.cc/en/Main/arduinoBoardMega2560>. Acesso em 5 de Julho de 2017.

A fonte de alimentação de energia escolhida foi baterias recarregáveis de Lítio de modelo BRC18650 5000 mAh da marca UltraFire. Essa escolha se justifica pelo seu peso e dimensões, que se conformam bem ao tamanho e forma do robô, ao contrário de outras baterias mais potentes que seriam pesadas e ocupariam muito espaço.

Tabela 4.5 - Características do microcontrolador Arduino Mega 2560.

Microcontrolador	ATmega2560
Tensão de operação	5V
Tensão de entrada (recomendada)	7-12V
Tensão de entrada (limite)	6-20V
Pinos I/O Digitais	54 (of which 15 provide PWM output)
Pinos de entrada analógica	16
Corrente DC por pino I/O	20 mA
Corrente DC para pino de 3.3V	50 mA
Memória Flash	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz
Led imbutido	13
Comprimento	101.52 mm
Largura	53.3 mm
Peso	37 g

#### 4.4 DETALHAMENTO

Para que se pudesse trabalhar com a programação e ter melhores ideias de como fixar todos os componentes do robô um protótipo foi projetado e confeccionado. A montagem do conjunto de peças desse protótipo é mostrada a seguir (Figura 4.9, Figura 4.10 e Figura 4.11).

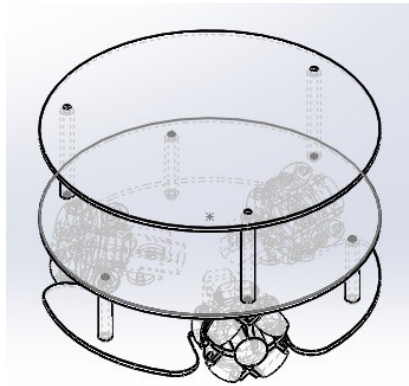


Figura 4.9 - Captura de tela da vista isométrica com linhas ocultas visíveis do primeiro protótipo.

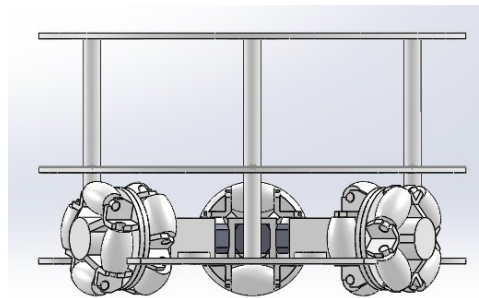


Figura 4.10 - Captura de tela da vista frontal do primeiro protótipo.

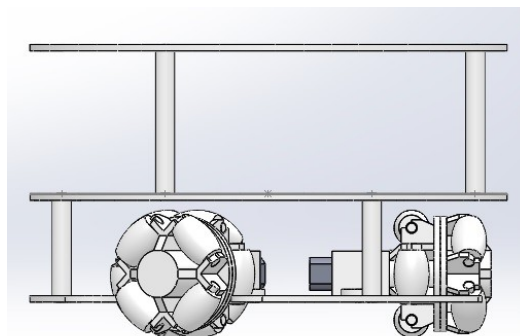


Figura 4.11 - Captura de tela da vista lateral do primeiro protótipo.

O detalhamento de peças é apresentado por completo no APÊNDICE B.

Quanto ao processo de fabricação, o chassi, as plataformas, os fixadores dos motores e as hastes de elevação entre plataformas foram todos produzidos na impressora 3D por extrusão e depósito de material ABS que depois de esfriados eram lixados com lima ou folha

de lixa para massas, em toda sua área da superfície incluindo bordas e cantos, para se retirar ranhuras e uniformizar o acabamento. Depois disso, cada peça que necessitava de furos era fixada numa morsa e tinha os furos retrabalhados por micro retífica. Os roletes do protótipo tiveram os diâmetros de eixo alargados para 2 mm com a micro retífica para se adequar aos pregos utilizados, isso foi feito porque o desenho original tinha dimensão de 1,6 mm e por falha da impressão esse diâmetro caiu para 1,35 mm.

No protótipo a maioria dos furos do chassi e das plataformas foi feita por micro retífica ou furadeira com brocas maiores porque o desenho inicial não contemplava a fixação de todas as placas.

A fixação das plataformas e componentes eletrônicos era feita por parafusos e porcas e a ligação entre os circuitos feitos por *jumpers* coloridos.

## **4.5 RESULTADOS DISCUSSÕES E TESTES**

### **4.5.1 DA FABRICAÇÃO**

No processo de fabricação as hastes ou eixos dos roletes das rodas fabricadas não funcionaram bem, porque tinham muito atrito com os roletes (de mesmo material) e mesmo com trabalho de acabamento não foram utilizadas. A compra de eixos maciços de pequeno calibre foi descartada pelo alto custo de peças tão específicas. Sendo assim a adequação encontrada foi de se utilizar pregos, cortados no tamanho necessário, esses não resistem ao rolamento dos roletes e, portanto, permitiam o funcionamento esperado das rodas.

Houve a necessidade de se fazer o acabamento das peças que por consequência do método de fabricação tinham superfície rugosa o suficiente para atrapalhar fixação de peças ou até mesmo o deslizamento (no caso dos roletes da roda omnidirecional). Sendo assim, essas peças foram lixadas com um lima de modo a alcançar o acabamento desejado. Além disso, outra adaptação feita nas rodas foi de colocar fita isolante ao redor dos roletes (de preferência na região central onde há o ponto de contato desses com o solo) para se aumentar a aderência desses ao solo.

Foi calculado o tempo gasto para a fabricação de cada peça a fim de se calcular posteriormente o custo de fabricação do protótipo, esse dimensionamento foi dado em horas e minutos aproximados que o próprio *software* do equipamento oferece.

A Tabela 4.6 relaciona tanto o tempo de fabricação quanto a quantidade de material utilizado.

Analisando todo o processo de fabricação (desenho e impressão), fica claro que modificações são necessárias durante todo o processo e recursos têm de ser administrados para cobrir esses gastos. Também fica claro que o conhecimento prévio sobre funcionamento e manutenção da impressora 3D é de suma importância para que a quantidade de retrabalho seja minimizada.

No aspecto citado, pode se descrever que as seguintes características foram responsáveis por retrabalho:

- Reestruturação do projeto para adequação às necessidades não identificadas desde o início;
- Desvio de furos e medidas acima da tolerância;
- Acabamento inferior ao esperado.

No que diz respeito à impressora 3D as seguintes medidas foram tomadas e devem ser observadas antes mesmo que a fabricação comece:

- Potência fornecida para cada motor dos eixos da impressora (facilmente modificada por meio de parafuso existente no *driver* controlador do motor);
- Limpeza do bico de extrusão para evitar entupimento e falha na quantidade de material que será extrudado;
- Calibração dos motores dos eixos e do alimentador do bico de extrusão a fim de que os erros sejam minimizados;
- Uso de matéria prima de qualidade (evitar usar ABS muito antigo, conservado ao tempo, pois pode ressecar e causar bolhas na impressão).



Tabela 4.6 - Tempo de fabricação das peças.

Peças	Quantidade	Tempo de fabricação (cada) [hs]	Tempo de fabricação total [hs]	Metragem filamento ABS [m]
<b>PLATAFORMAS</b>				
Chassi	1	02:20:00	02:20:00	8,245
Pisos superiores	2	02:53:00	05:46:00	21,912
<b>HASTES DE CONEXÃO ENTRE PLATAFORMAS</b>				
Menor	3	00:22:00	01:06:00	0,987
Maior	3	00:25:00	01:15:00	1,464
<b>PARTES DA RODA</b>				
Parte fixa	3	01:27:00	04:21:00	6,633
Parte eixo motor	3	01:30:00	04:30:00	6,843
Roletes	24	01:23:00	08:18:00	11,538
Fixação dos motores	3	00:58:00	02:54:00	2,868
		<b>Subtotal</b>	30:30:00	60,49
<b>RETRABALHO</b>				
Fixação dos motores	3	00:58:00	02:54:00	2,868
Parte fixa	3	01:27:00	04:21:00	6,633
Parte eixo motor	3	01:30:00	04:30:00	6,843
Roletes	24	01:23:00	08:18:00	11,538
		<b>Subtotal</b>	20:03:00	27,882
		<b>Total</b>	50:33:00	88,372

Dessa maneira, pode se dizer que alguns custos seriam minimizados à medida que a manutenção do equipamento tivesse sido feita e o projeto conceitual tivesse sido trabalho de forma mais extensa, abrangendo todas as possibilidades de adaptações a serem feitas antes da fabricação do primeiro protótipo.

## 4.5.2 DOS CUSTOS

Para a análise de custos foram considerados os seguintes parâmetros:

Na fabricação:

- Honorários do projetista;
- Custo gasto de energia para alimentação da impressora 3D;
- Custo do material de impressão (ABS).

Nos equipamentos acoplados ao protótipo de teste:

- Motores;
- Elementos de fixação (parafusos);
- Elementos usados como eixo dos roletes (pregos);
- Elementos de fixação (porcas);
- *Shield de driver* dos motores (L298N);
- Baterias;
- *Mini protoboard*;
- *Jumpers*;
- Fita isolante.

A Tabela 4.7 relaciona o custo dos itens relacionados anteriormente.

Tabela 4.7 - Tabela de custos do protótipo para teste.

Despesas	Quantidade	Valor
<b>Honorários do projetista</b>	-	-
<b>Energia</b>	20,86[kWh]	R\$ 17,50
<b>Filamento ABS</b>	88,372 [m]	R\$ 68,00
<b>Motores</b>	3[un.]	R\$ 60,00
<b>Parafusos</b>	30[un.]	R\$ 1,50
<b>Pregos</b>	30[un.]	R\$ 1,20
<b>Porcas</b>	30[un.]	R\$ 2,00
<b>Motor shield LN298</b>	2[un.]	R\$ 44,00
<b>Baterias com carregador</b>	4[un.](baterias) 1[un.] (carregador)	R\$ 31,37
<b>Mini protoboard</b>	1[un.]	R\$ 1,50
<b>Jumpers</b>	120 [un.]	R\$ 9,00
<b>Fita isolante</b>	1[un.]	R\$ 6,00
	Total	R\$ 242,07

É necessário dizer que apesar de não serem considerados no custo final os honorários do projetista, foi considerado o retrabalho de peças, que aumentam o custo com fabricação tanto no material quanto nos seus processos e é parte fundamental desse projeto, já que adequações e melhorias foram estudadas durante e depois o processo de fabricação do primeiro protótipo.

A fim de comparar produtos existentes no mercado, por meio de pesquisa online identificou-se um robô que se assemelha à necessidade do projeto de pesquisa e encontrou-se o representado na Figura 4.12.



**Figura 4.12 - Kit robô móvel de roda omnidirecional.**

**Fonte:** <http://www.robotshop.com/en/3wd-48mm-omni-wheel-mobile-robot-kit.html>. Acesso em 2 de junho de 2017.

O robô encontrado inclui:

- 3 rodas omnidirecionais de 48 mm;
- 3 motores coreless Namiki de 12 V com encoder;
- 1 controlador Arduino 328;
- 1 expansão Arduino IO;
- 3 sensores ultrassônicos.

Apesar do protótipo construído nesse trabalho não possuir todas as ferramentas que o kit pesquisado (que possui valor de venda de US\$ 298.00), entende-se que a fabricação de um robô original além de economizar recursos financeiros, viabiliza desenvolvimento e pesquisa tanto nessa fabricação (foco desse trabalho) como na multiplicação e estudo de enxame desse

robô, uma vez que o investimento em múltiplos robôs como esse sugerido, teria um valor exorbitante (quase três vezes) quando comparado ao do projeto.

### 4.5.3 DEMONSTRAÇÃO DO MOVIMENTO DO ROBÔ

A demonstração do movimento do robô se baseou em duas etapas, na primeira etapa montou-se a parte eletrônica do robô, estimou-se a autonomia das baterias e programou-se o Arduino para realizar a movimentação, e na segunda etapa, fez-se um vídeo dessa movimentação e registrou-se quanto tempo funcionaram na prática as baterias.

#### PRIMEIRA ETAPA

Para ilustrar a montagem da parte eletrônica do robô utilizou-se do *software* Fritzing (*open source* com aplicativo para montagem e simulação de kits da Arduino), a Figura 4.13 ilustra a montagem com as peças do Arduino enquanto o ANEXO A, a montagem esquemática de componentes.

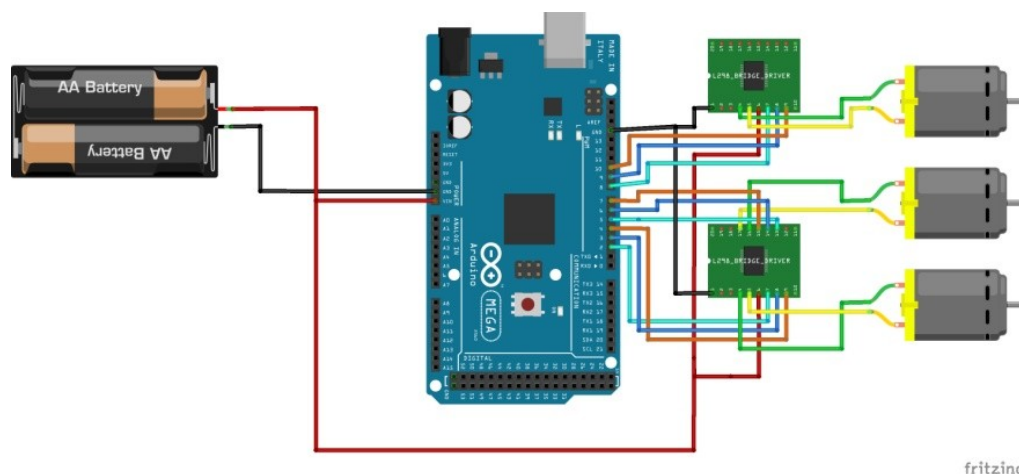


Figura 4.13 - Captura de tela do software Fritzing com ilustração da montagem no Arduino.

A estimativa da autonomia das baterias se baseia em artigos práticos ([www.dourolight.com](http://www.dourolight.com) acesso em 7 de junho de 2017) e segue os seguintes cálculos:

#### CAPACIDADE DA BATERIA

A capacidade da bateria, registrada pelo fabricante, dada em ampere-hora [Ah], tem sua utilização por motivo de segurança (no caso de baterias de chumbo-ácido) delimitada a 75% da capacidade nominal, podendo inutilizar ou reduzir o seu tempo de vida quando

utilizada ao máximo. Dessa maneira, mesmo que de material diferente a estimativa foi feita seguindo esse padrão.

Sendo assim, a capacidade teórica da bateria ( $CtB$ ) é:

$$CtB = 0,75 * \text{capacidade nominal [Ah]}$$

As baterias utilizadas nesse trabalho tem 5 Ah de capacidade então:

$$CtB (5 Ah) = 0,75 * 5 = 3,75 Ah$$

#### ENERGIA TOTAL ARMAZENADA

Outro parâmetro importante para o cálculo da autonomia é a energia total armazenada ( $Et$  [Wh]), dada pela multiplicação da tensão nominal da bateria pela sua capacidade, sendo assim:

Considerando que são duas baterias de 5 Ah,

$$Et (3,75 Ah e 3,7 V) = 13,875 * 2 = 27,75 Wh$$

#### CONSUMO ESTIMADO

Dessa forma, o tempo de consumo estimado ( $TCE$  [h]), dado pela razão entre a energia total e a potência dissipada é:

$$TCE = \frac{\text{Energia Total [Wh]}}{\text{Potência Dissipada [W]}} = \frac{27,75}{53,06} = 0,523 h = 32 \text{ minutos}$$

A potência dissipada considera os gastos de energia para os pinos utilizados do Arduino, os *drivers* de motor e os motores em si. O motor em questão trabalhando com 6 V, alcança no máximo 50 rpm e consome 120 mA de corrente.

Com relação à programação, foi feito um algoritmo no qual o robô deveria seguir a ordem de movimentos apresentados na Figura 4.14. As setas verdes (linhas tracejadas) representam a direção em que o carro se moverá, enquanto as setas vermelhas (linhas contínuas) mostram o sentido de movimento da roda em questão. As rodas que não tem sentido marcado se encontram em estado parado como será mostrado na programação.

Os motores nomeados e marcados servem como referência para a programação.

Vale lembrar que feito o programa, basta conectar o Arduino ao computador via cabo USB e transferir seu conteúdo por uma função própria dessa IDE.

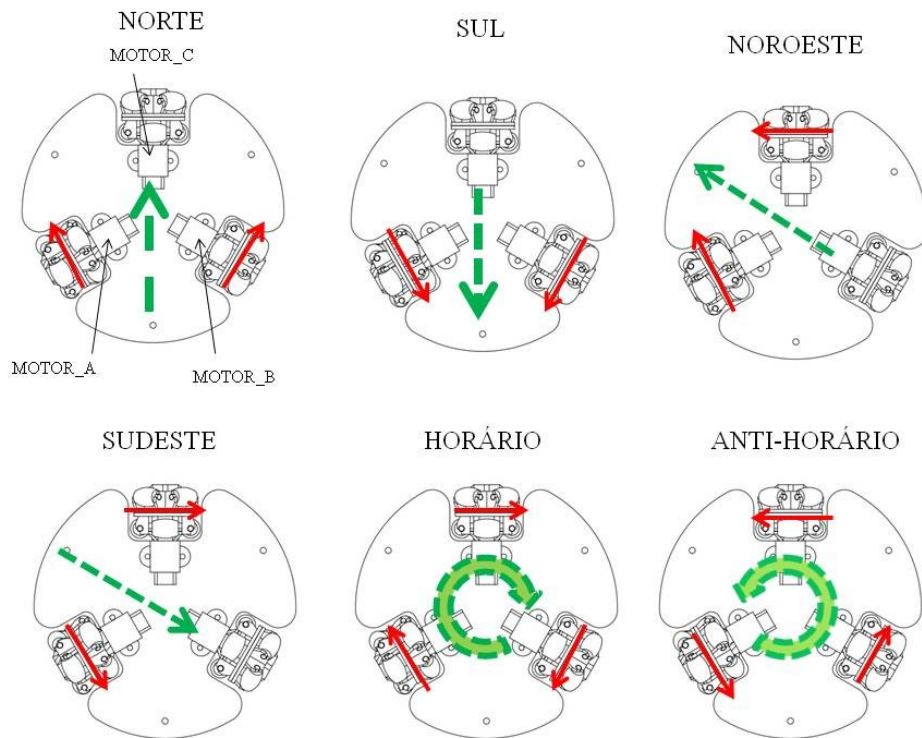


Figura 4.14 - Movimentos a serem seguidos pelo robô na etapa de demonstração.

Como já foi dito a linguagem utilizada para programar o Arduino é similar à linguagem C de programação com sintaxe particular e funções que facilitam o desenvolvimento de projetos com bibliotecas já existentes na própria comunidade Arduino, *open source*.

A IDE para programação tem ambiente programado em Java e baseado em outros *softwares open source*.

Os programas no Arduino são divididos em três partes principais:

#### ESTRUTURA

A sua estrutura compreende de duas funções definidas como:

*Setup* ( ) – chamada sempre que se inicia uma *sketch* ( ou esboço, como são chamadas as páginas abertas na IDE para se escrever os programas). Ela é usada para iniciar variáveis,

modos dos pinos, iniciar bibliotecas. Essa função roda apenas uma vez após cada vez que se liga o Arduino ou se reseta a placa.

*Loop* ( ) – Depois de feito o *setup* (função anteriormente explicada), essa função é repetida consecutivamente até que o Arduino seja desligado ou o programa faça alguma chamada de saída. É ela quem permite mudanças e respostas no programa. É utilizada para controlar a placa do Arduino.

### VARIÁVEIS (variáveis e constantes)

Variáveis presentes nessa linguagem podem ser constantes (*HIGH*, *LOW*, *INPUT*, *OUTPUT*), tipos de dados (*void*, *boolean*, *char*, *unsigned char*), de conversão (*char*, *byte*, *int*, *word*).

### FUNÇÕES

Existem várias funções e bibliotecas criadas para esse microcontrolador e se destacam, por exemplo, as de controle de pinos, aquelas que vão realizar comandos nos pinos. Exemplos: *pinMODE* ( ), *digitalWrite* ( ), *digitalRead* ( ), *analogWrite* ( ), entre outras.

Tudo que foi brevemente explicado aqui será explicado de forma mais completa na apresentação do código do programa de movimentação do robô.

Apresentadas as partes principais de um programa de Arduino, o algoritmo criado para a demonstração do movimento do robô será apresentado e explicado em partes e apresentado por completo no ANEXO C.

### DECLARAÇÃO DOS PINOS

```
//-----DECLARATION OF PINS-----

//-----MOTOR A-----
int IN1_1 = 2;
int IN2_1 = 4 ;
int vel_A = 3;

//-----MOTOR B-----
int IN3_1 = 5;
int IN4_1 = 7;
int vel_B = 6;

//-----MOTOR C-----
int IN1_2 = 8;
int IN2_2 = 10;
int vel_C = 9;
```

```
#define MOTOR_A 1
#define MOTOR_B 2
#define MOTOR_C 3
#define CLOCKWISE true
#define COUNTERCLOCK false
```

A fim de facilitar a programação e se utilizando da maneira mais correta comercialmente (língua em inglês), essa parte do programa se destina à definição de pinos a serem utilizados e os nomes que serão empregados no programa, foi feita inclusive antes da função *setup* ( ). Os números indicados à frente de cada linha de código (ex.: `int IN1_1 = 2`) mostram a posição física do pino a ser utilizado, para esse exemplo o pino indicado na placa pelo numeral 2 será utilizado.

### FUNÇÃO *SETUP*

```
void setup() {
  //----PINS INITIALIZATION----
  pinMode(IN1_1, OUTPUT);
  pinMode(IN2_1, OUTPUT);
  pinMode(IN3_1, OUTPUT);
  pinMode(IN4_1, OUTPUT);
  pinMode(IN1_2, OUTPUT);
  pinMode(IN2_2, OUTPUT);
  pinMode(vel_A, OUTPUT);
  pinMode(vel_B, OUTPUT);
  pinMode(vel_C, OUTPUT);
}
```

Como foi explicada anteriormente essa função tem como finalidade, inicializar os pinos que serão utilizados do Arduino. Nesse caso, todos foram definidos como “*OUTPUT*”, ou seja, são designados com baixa impedância, podendo fornecer corrente de até 40 mA para outros circuitos ou dispositivos. A função *pinMode* ( ) que vem da própria biblioteca do Arduino funciona com dois argumentos, o primeiro indicando qual o pino que será configurado e o segundo o modo que ele está ativado (*input*, *output* ou *input\_pullup*).

### PROTÓTIPOS DE FUNÇÕES

```
//----PROTOTYPES----
void forward_move (void);
void backward_move (void);
void north_west (void);
void south_east (void);
void clock_move (void);
void counter_clock_move (void);
void quiet (void);
```



Como boas práticas de programação, foi feita a lista de protótipos de funções que são declarações que omitem o corpo da função, mas especificam: nome, tipos de argumentos e retorno, essas informações permitem ao programador, discriminar de forma mais simples as funções que serão utilizadas e apresentá-las antes de serem chamadas de fato.

Como podem ser observadas, essas funções não tem entrada nem saída, funcionam quando são chamadas e toda sua descrição será detalhada na apresentação de cada função posteriormente.

### FUNÇÃO PRINCIPAL (MAIN)

```
//-----MAIN-----
void loop() {
  forward_move();
  backward_move();
  north_west();
  south_west();
  clock_move();
  counter_clock_move();
  quiet();
}
```

A função *loop* foi programada de modo que cada chamada tem as configurações de movimento do robô em um sentido, até que após completo o ciclo ele as recomece.

Dessa maneira, ele fará um movimento à frente, um movimento para trás, depois na diagonal em sentido noroeste, diagonal contrária (sudeste), depois da um giro completo no sentido horário, giro completo no sentido anti-horário (como mostrado na Figura 4.14) e termina a série com uma parada breve, reiniciando assim a série até que a bateria acabe ou o Arduino seja desconectado da fonte de energia (baterias).

As funções mostradas não possuem parâmetros de entrada e, portanto, ao serem chamadas, são executadas e passam para a próxima instrução.

### FUNÇÃO DE TRAVAMENTO INDIVIDUAL DOS MOTORES

```
//-----FUNCTION TO STOP INDIVIDUALLY THE MOTORS-----
void stop_motor(int motor)
{
  if (motor == MOTOR_A)
  {
    digitalWrite(2, HIGH);
    digitalWrite(4, HIGH);
    analogWrite(3, 255);
  }
  else if (motor == MOTOR_B)
```

```

{
  digitalWrite(5, HIGH);
  digitalWrite(7, HIGH);
  analogWrite(6, 255);
}
else if (motor == MOTOR_C)
{
  digitalWrite(8, HIGH);
  digitalWrite(10, HIGH);
  analogWrite(9, 255);
}
}
}

```

Essa função tem como parâmetro de entrada o motor que se quer deixar em modo parado. Como foi mostrado, o comando para acionamento não tem link direto entre o Arduino e o motor, ou seja, passa pelo *driver* de motor L298N.

Vale lembrar que todos os pinos utilizados nesse programa já foram configurados como de saída de sinal (*OUTPUT*), essa configuração aliada à da função *digitalWrite*, faz com que esses pinos (2, 4, 5, 7, 8 e 10) forneçam na saída 5 V, sendo assim, o primeiro argumento da função dita o pino destinado e o segundo argumento a configuração do sinal. A terceira função de cada condicional, designada aos pinos 3, 6 e 9, configura a porta como uma saída analógica com *duty cycle* de 255 (sempre ligado) por se tratar de uma parada forçada, ou seja, ao invés de se deixar o motor sem sinal (e suscetível a erros de impedância ou variação da placa), aplica-se tensão máxima nos dois polos, de modo que ele ficará travado, garantindo assim que esse motor ficará parado. Dessa maneira, entende-se que essa função *analogWrite*, utilizada em pinos que possuem PWM, configuram no seu primeiro argumento o pino a ser acionado e no segundo o *duty cycle* desejado.

## FUNÇÃO PARA CONTROLAR OS MOTORES

```

//----FUNCTION TO CONTROL THE ENGINES----
void power_motor(int motor, boolean way, int velocity)
{
  if (motor == MOTOR_A)
  {
    digitalWrite(2, way);
    digitalWrite(4, !way);
    analogWrite(3, velocity);
  }
  else if (motor == MOTOR_B)
  {
    digitalWrite(5, way);
    digitalWrite(7, !way);
    analogWrite(6, velocity);
  }
  else if (motor == MOTOR_C)
  {

```

```

    digitalWrite(8, way);
    digitalWrite(10, !way);
    analogWrite(9, velocity);
  }
}

```

Essa função tem como objetivo controlar o movimento das rodas. Ela tem 3 argumentos de entrada, o primeiro indicando qual motor, o segundo indicando sentido de rotação via variável booleana definida no início do programa e a terceira o *duty cycle* desejado para o PWM.

A diferença que caracteriza essa função é que diferente da anterior, o sentido do giro do motor é expresso pelo sentido da corrente aplicada a ela. Essa foi expressa justamente pela variável booleana, em caso de sentido *CLOCKWISE* (horário) ou *COUNTERCLOCK* (anti-horário).

## FUNÇÕES DE MOVIMENTAÇÃO

```

//----FORWARD----
void forward_move (void)
{
  power_motor(MOTOR_A, COUNTERCLOCK, 255);
  power_motor(MOTOR_B, CLOCKWISE, 255);
  stop_motor(MOTOR_C);
  delay(2500);
}
//----BACKWARD----
void backward_move (void)
{
  power_motor(MOTOR_A, CLOCKWISE, 255);
  power_motor(MOTOR_B, COUNTERCLOCK, 255);
  stop_motor(MOTOR_C);
  delay(2500);
}
//----RIGHT DIAGONAL----
void south_east (void)
{
  power_motor(MOTOR_A, CLOCKWISE, 255);
  stop_motor(MOTOR_B);
  power_motor(MOTOR_C, COUNTERCLOCK, 255);
  delay(2500);
}
//----LEFT DIAGONAL----
void north_west (void)
{
  power_motor(MOTOR_A, COUNTERCLOCK, 255);
  stop_motor(MOTOR_B);
  power_motor(MOTOR_C, CLOCKWISE, 255);
  delay(2500);
}
//----CLOCKWISE----
void clock_move (void)
{

```

```

power_motor(MOTOR_A, COUNTERCLOCK, 255);
power_motor(MOTOR_B, COUNTERCLOCK, 255);
power_motor(MOTOR_C, COUNTERCLOCK, 255);
delay(2500);
}
//----COUNTERCLOCKWISE
void counter_clock_move (void)
{
    power_motor(MOTOR_A, CLOCKWISE, 255);
    power_motor(MOTOR_B, CLOCKWISE, 255);
    power_motor(MOTOR_C, CLOCKWISE, 255);
    delay(2500);
}
//----QUIET----
void quiet (void)
{
    stop_motor(MOTOR_A);
    stop_motor(MOTOR_B);
    stop_motor(MOTOR_C);
    delay(2500);
}

```

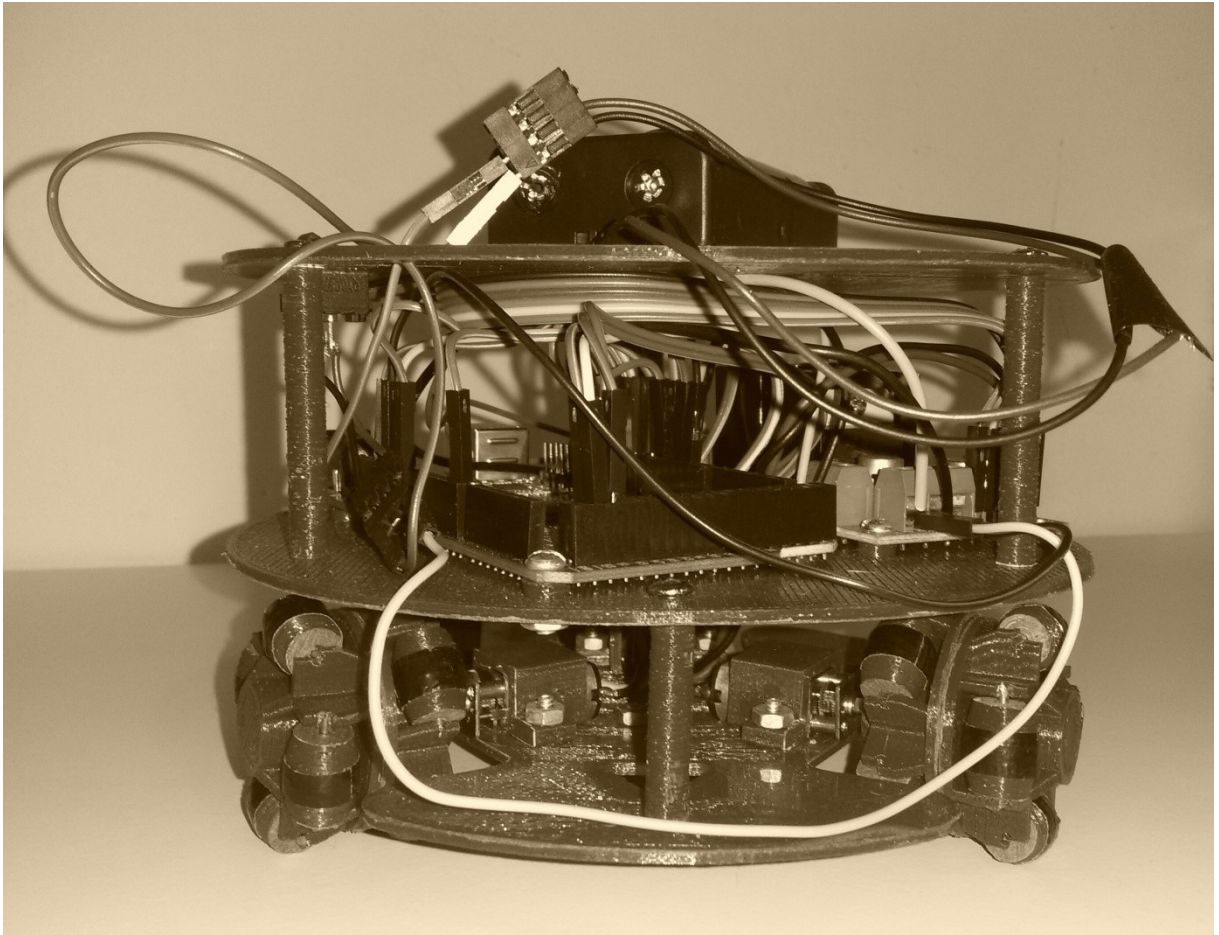
Finalmente, as funções que regem o movimento do robô são as apresentadas. Todas são autoexplicativas uma vez que suas antecessoras já foram descritas. A diferença é que, a cada término de movimento, a função *delay* foi aplicada, adicionando uma espera de 2500  $\mu$ s antes que a função seja terminada. Como pode ser visto todas as funções são chamadas sem argumento de entrada e saída e a última delas (*quiet*) simplesmente garante que todos os motores ficarão parados. Observe que a sequência apresentada segue exatamente a Figura 4.14 lida da esquerda para a direita de cima para baixo.

Vale lembrar que nenhum tipo de controle foi feito no código, o que quer dizer que as rodas foram apenas acionadas de forma discreta. O funcionamento foi um sucesso e credita-se a isso o fato dos motores serem novos e, portanto, não apresentarem diferença de impedância ou qualquer outro tipo de problema que poderia desvia-los da movimentação programada.

## SEGUNDA ETAPA

O protótipo final é apresentado na Figura 4.15.

A autonomia das baterias na prática alcançou um número muito próximo do cálculo teórico (de 29 minutos de funcionamento do robô), o que mostra que essa avaliação pode ser usada para prever o gasto do robô nessa e em novas configurações, desde que se respeite a identificação de todo o gasto de potência que ele despenderá.



**Figura 4.15 - Foto do protótipo montado.**

## **CAPÍTULO V**

### **CONCLUSÕES**

O projeto de um robô móvel é complexo. O levantamento de requisitos e restrições limita financeira e estrategicamente como se deve trabalhar. O maquinário disponível e o conhecimento de como se utilizar dele, mostrou-se de suma importância para a evolução do projeto. O estudo do processo de fabricação permite adequações que visem economia de tempo e dinheiro.

Houve a possibilidade de compra de peças de diferentes materiais, porém, como o custo e as adequações para seu uso eram inviáveis, tornou-se mais interessante, a utilização de materiais mais baratos e de fácil acesso.

Obteve-se sucesso na construção da estrutura de um modelo de robô móvel. Com a metodologia de projeto aplicada, foi possível construir um robô de custo reduzido (investimento de R\$242,07 por robô, em comparação a um similar de custo U\$298,00), com material e maquinário disponível na universidade onde o projeto foi realizado, de forma que sua replicação para estudo de exame de robôs se caracteriza viável.

O levantamento de requisitos foi todo respeitado e, portanto o robô tem as características estruturais desejadas e com o detalhamento do projeto, sua replicação ou melhoria para estudo e desenvolvimento na área estará disponível.

A demonstração do movimento do robô apesar de simplificada mostra a principal característica que se desejava ao se construir um robô com três rodas suecas, a possibilidade de que ele realizasse movimento omnidirecional.

Sugere-se aos interessados nessa área de pesquisa, o projeto e confecção das placas de circuitos de sensoramento e controle, um sistema de recarga das baterias acoplado ao sistema

ou externo, de forma que não seja necessário o desligamento do robô e avaliar a viabilidade de outras formas de alimentação do robô (como por exemplo, energia solar).

É importante ressaltar que a utilização desse robô modelo com as adequações necessárias, pode fomentar pesquisa e desenvolvimento, por exemplo, de equipes de futebol de robôs ou estudos de enxames de robôs, contribuindo assim para a formação de engenheiros, graças ao seu caráter multidisciplinar, incluindo mecânica, eletrônica, microcontroladores, controle, lógica, programação, gestão de projetos, bem como o uso de ferramentas (*softwares* de CAD e CAM) e maquinário de fabricação (como por exemplo, a impressora 3D).

## BIBLIOGRAFIA

**3DPI - 3D Printing Industry.** The Authority on 3D Printing - The Free Beginner's Guide. Site da 3DPI. [Online] <https://3dprintingindustry.com/3d-printing-basics-free-beginners-guide/history/>. Acesso em: 20 de janeiro de 2017.

**ARDUINO.** Arduino. What is Arduino? [Online].[www.arduino.cc](http://www.arduino.cc). Acesso em: 12 de Fevereiro de 2017.

**BEKEY, George A.** 2005. *Autonomous Robot. From Biological Inspiration to Implementation and Control.* Cambridge : The MIT Press, 2005. p. 580.

**BRÄUNL, Thomas.** 2006. *Embedded Robotics. Mobile Design and Applications with Embedded Systems.* 2ª Edição. Germany : Springer, 2006. p. 458.

**BOWYER, Adrian.** 2004. RepRap. Bem vindo a RepRap.org. [Online] RepRap Ltd, 2004. <http://www.reprap.org/wiki/RepRap/pt>. Acesso em: 20 de Janeiro de 2017.

**FITZGERALD, A. E., KINGSLEY JR., Charles e UMANS, Stephen D.** 2006. MÁQUINAS ELÉTRICAS. 6ª Edição. [S.l.] : Bookman, 2006. p. 607.

Integrated Manufacturing: Redesign the Organization before Implementing Flexible Technology. **DUIMERING, P. Robert, SAFAYENI, Frank e PURDY, Lyn.** 1993. Cambridge : Mass, 1993, Sloan Management Review.

**JENKIN, Michael e DUDEK, Gregory.** 2000. *Computational Principles of Mobile Robots.* New York : Cambridge University Press, 2000. p. 280.

**PAHL, Gerhard; BEITZ, Wolfgang; FELDHUSEN, Jörg; GROTE, Karl-Heinrich.** 2005. Projeto na Engenharia. Fundamentos do Desenvolvimento Eficaz de Produtos. Métodos e Aplicações. [trad.] Nazem Nascimento e Hans Andreas Werner. [S.l.] : Edgard Blücher, 2005. p. 432.

**ROMERO, Roseli Aparecida F.; PRESTES, Edson; OSÓRIO, Fernando; WOLF, Denis.** 2014. *Robótica Móvel.* Rio de Janeiro : Grupo Editora Nacional, 2014. p. 303.

**SECCHI, Humberto A.** 2008. Uma Introdução aos Robôs Móveis. [trad.] Cynthia Netto de ALMEIDA e Felipe Nascimento MARTINS. [S.l.] : s.n., 2008. p. 81. Título original: Una Introducción a los Robots Móviles.

**SIEGWART, Roland e NOURBAKHS, Illah R.** 2004. *Introduction to Autonomous Mobile Robots.* Cambridge : The MIT Press, 2004. p. 322.

**SOLID WORKS CORPORATION.** 1997. SolidWorks Premium The power you need to drive innovation. Site da Dassault Systemes. [Online] Dassault Systemes, 1997. [http://www.solidworks.com/sw/docs/3DS\\_2017\\_SWK\\_Premium\\_Datasheet.pdf](http://www.solidworks.com/sw/docs/3DS_2017_SWK_Premium_Datasheet.pdf). Acesso em 23 de Janeiro de 2017.



### APÊNDICE A

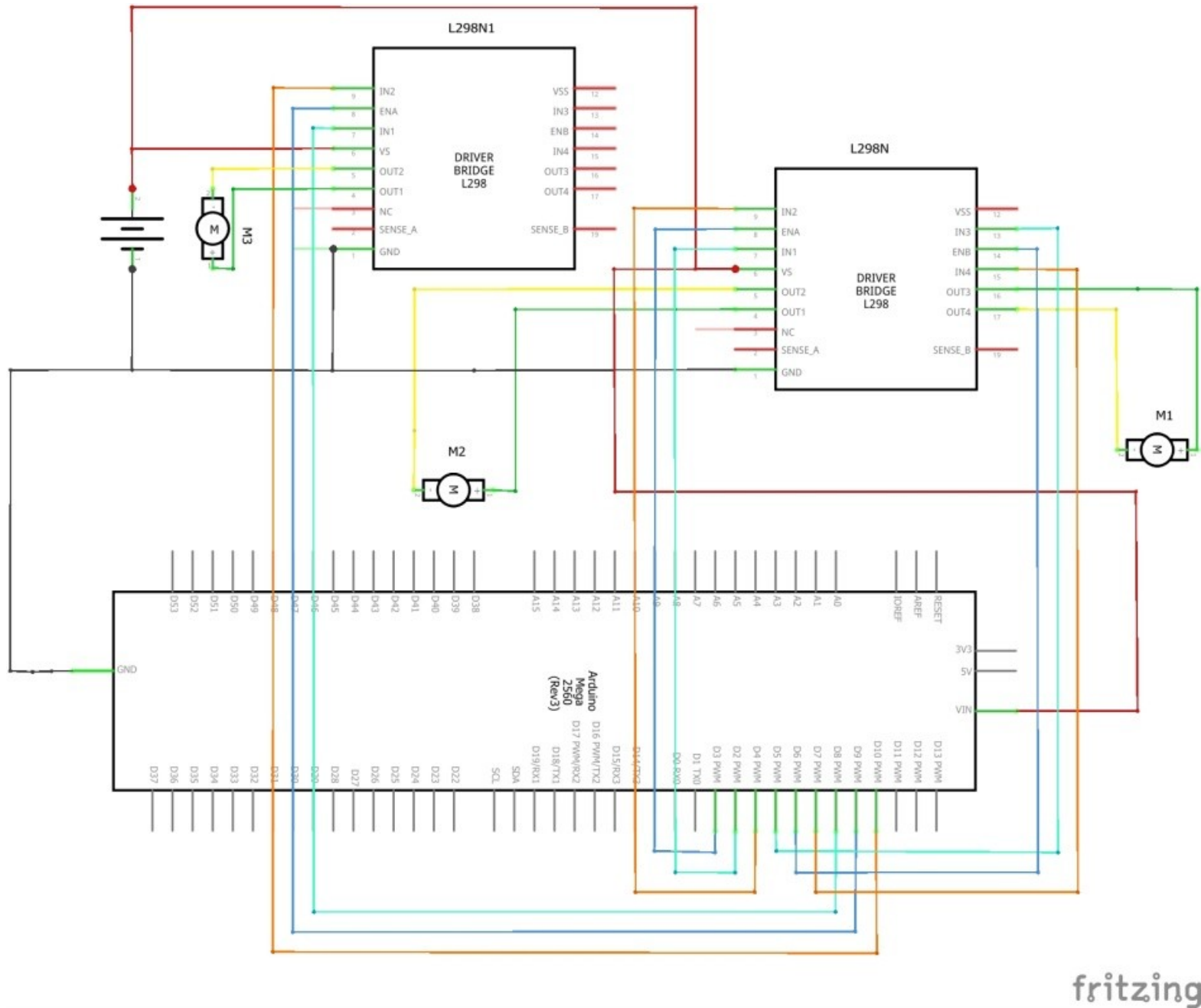
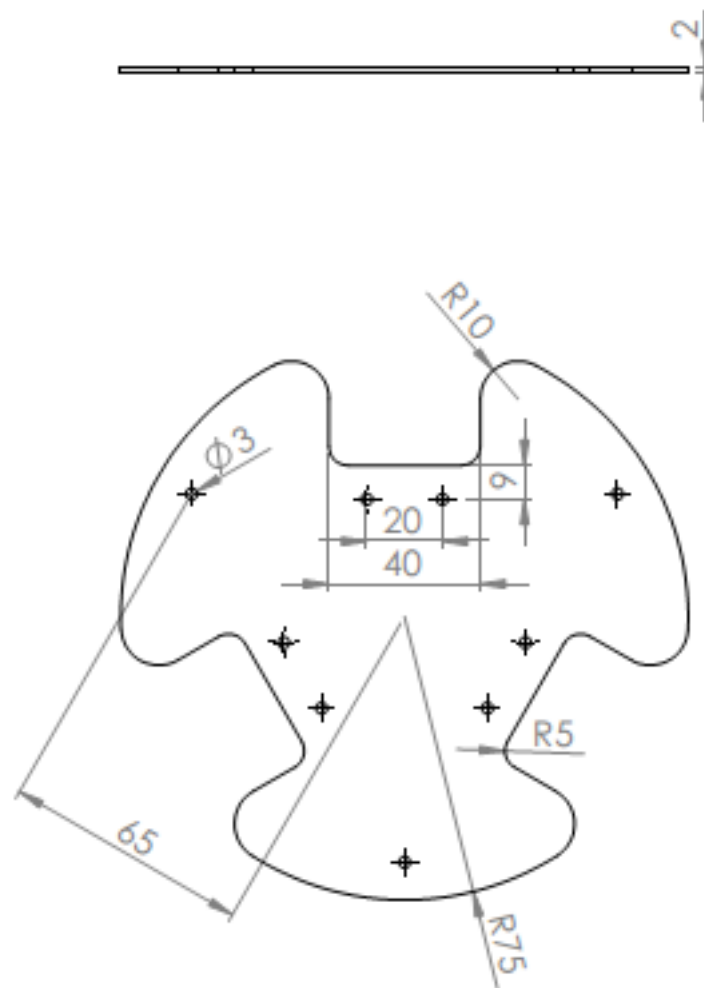

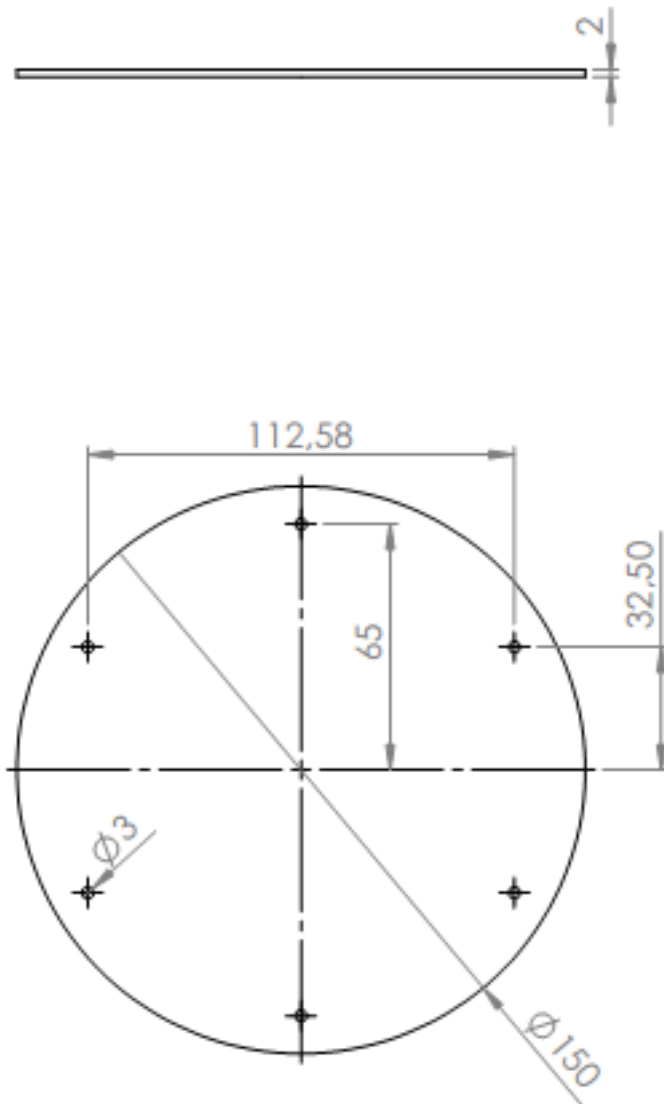


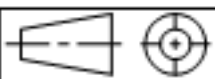
Figura 6.1 - de tela do software Fritzing com a montagem esquemática da parte eletrônica do robô.

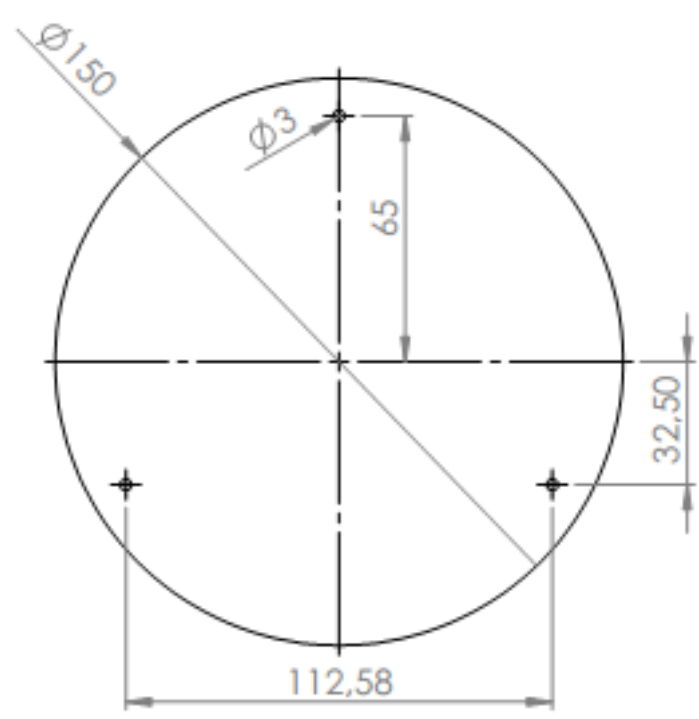
## APÊNDICE B




Escala: 1:2		UNIVERSIDADE FEDERAL DE UBERLÂNDIA	
A4		Título Chassi térreo	
Data 03/07/2017	Autor Gustavo Correa Umezaki		Folha 1

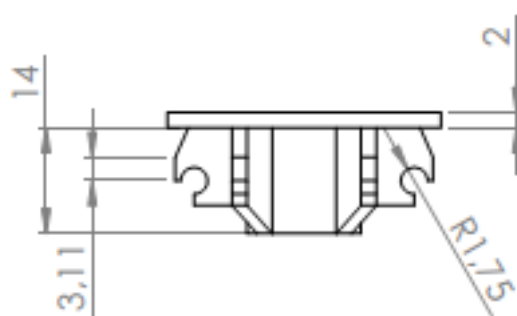
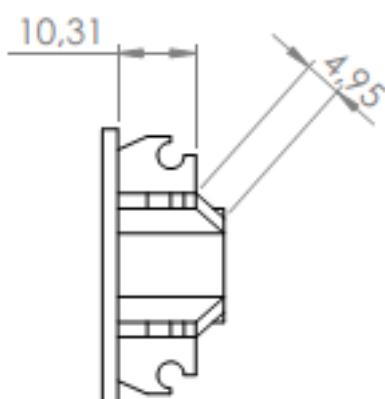
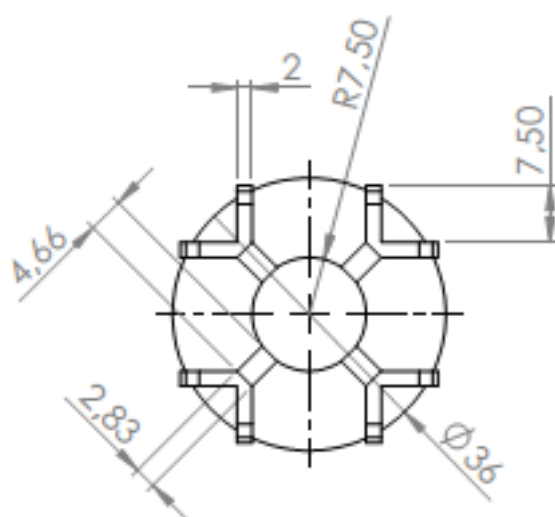
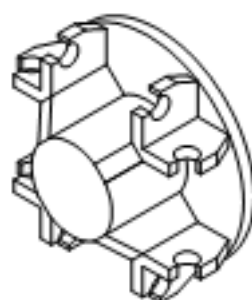
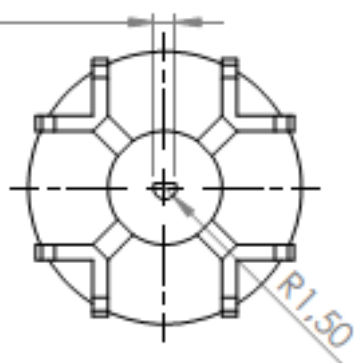


Escala: 1:2		UNIVERSIDADE FEDERAL DE UBERLÂNDIA	
A4		Título Chassi do 1º andar	
Data 03/07/2017		Autor Gustavo Correa Umezaki	
			Folha 2



Escala: 1:2		UNIVERSIDADE FEDERAL DE UBERLÂNDIA	
A4		Titulo Chassi do 2º andar	
Data 03/07/2017	Autor Gustavo Correa Umezaki		Folha 3

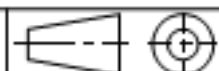
2,76



Escala: 1:1

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

A4



Título

Roda - parte motora e parte fixa

Data

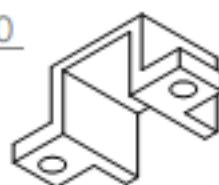
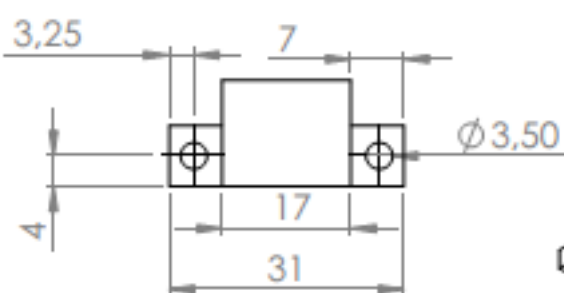
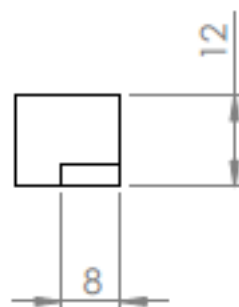
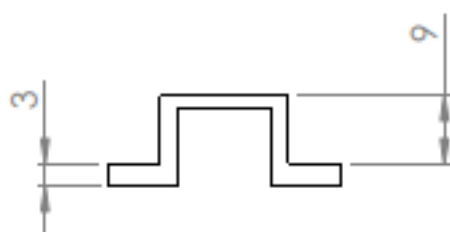
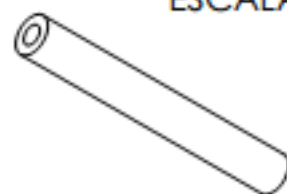
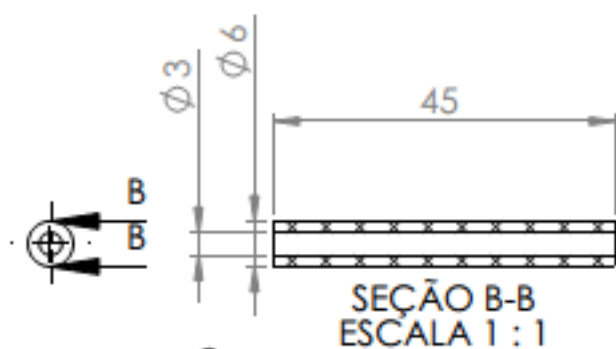
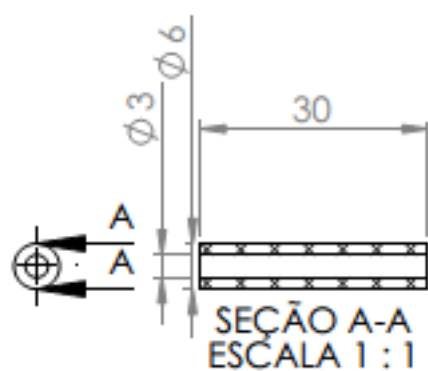
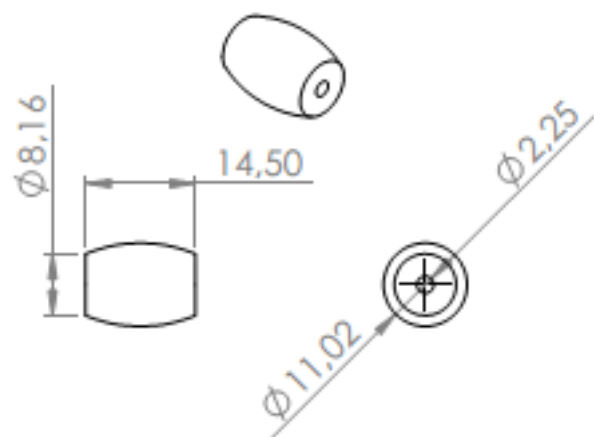
03/07/2017

Autor

Gustavo Correa Umezaki

Folha

4



Escala: 1:1		UNIVERSIDADE FEDERAL DE UBERLÂNDIA	
A4		Título Rolete, hastes e fixador de motor	
Data 03/07/2017		Autor Gustavo Correa Umezaki	Folha 5

## APÊNDICE C

```

//-----DECLARATION OF PINS-----

//-----MOTOR A-----
int IN1_1 = 2;
int IN2_1 = 4 ;
int vel_A = 3;

//-----MOTOR B-----
int IN3_1 = 5;
int IN4_1 = 7;
int vel_B = 6;

//-----MOTOR C-----
int IN1_2 = 8;
int IN2_2 = 10;
int vel_C = 9;

#define MOTOR_A 1
#define MOTOR_B 2
#define MOTOR_C 3
#define CLOCKWISE true
#define COUNTERCLOCK false

void setup() {
  //-----PINS INITIALIZATION-----
  pinMode(IN1_1, OUTPUT);
  pinMode(IN2_1, OUTPUT);
  pinMode(IN3_1, OUTPUT);
  pinMode(IN4_1, OUTPUT);
  pinMode(IN1_2, OUTPUT);
  pinMode(IN2_2, OUTPUT);
  pinMode(vel_A, OUTPUT);
  pinMode(vel_B, OUTPUT);
  pinMode(vel_C, OUTPUT);
}

//-----PROTOTYPES-----
void forward_move (void);
void backward_move (void);
void north_west (void);
void south_east (void);
void clock_move (void);
void counter_clock_move (void);
void quiet (void);

//-----MAIN-----
void loop() {
  forward_move ();
  backward_move ();
  north_west ();
  south_east ();
  clock_move ();
  counter_clock_move ();
  quiet ();
}

//-----FUNCTION TO STOP INDIVIDUALLY THE MOTORS-----
void stop_motor (int motor)
{

```

```

if (motor == MOTOR_A)
{
    digitalWrite(2, HIGH);
    digitalWrite(4, HIGH);
    analogWrite(3, 255);
}
else if (motor == MOTOR_B)
{
    digitalWrite(5, HIGH);
    digitalWrite(7, HIGH);
    analogWrite(6, 255);
}
else if (motor == MOTOR_C)
{
    digitalWrite(8, HIGH);
    digitalWrite(10, HIGH);
    analogWrite(9, 255);
}
}
//----FUNCTION TO CONTROL THE ENGINES----
void power_motor(int motor, boolean way, int velocity)
{
    if (motor == MOTOR_A)
    {
        digitalWrite(2, way);
        digitalWrite(4, !way);
        analogWrite(3, velocity);
    }
    else if (motor == MOTOR_B)
    {
        digitalWrite(5, way);
        digitalWrite(7, !way);
        analogWrite(6, velocity);
    }
    else if (motor == MOTOR_C)
    {
        digitalWrite(8, way);
        digitalWrite(10, !way);
        analogWrite(9, velocity);
    }
}
//----FORWARD----
void forward_move (void)
{
    power_motor(MOTOR_A, COUNTERCLOCK, 255);
    power_motor(MOTOR_B, CLOCKWISE, 255);
    stop_motor(MOTOR_C);
    delay(2500);
}
//----BACKWARD----
void backward_move (void)
{
    power_motor(MOTOR_A, CLOCKWISE, 255);
    power_motor(MOTOR_B, COUNTERCLOCK, 255);
    stop_motor(MOTOR_C);
    delay(2500);
}
//----RIGHT DIAGONAL----
void south_east (void)
{

```



```
power_motor(MOTOR_A, CLOCKWISE, 255);
stop_motor(MOTOR_B);
power_motor(MOTOR_C, COUNTERCLOCK, 255);
delay(2500);
}
//----LEFT DIAGONAL----
void north_west (void)
{
    power_motor(MOTOR_A, COUNTERCLOCK, 255);
    stop_motor(MOTOR_B);
    power_motor(MOTOR_C, CLOCKWISE, 255);
    delay(2500);
}
//----CLOCKWISE----
void clock_move (void)
{
    power_motor(MOTOR_A, COUNTERCLOCK, 255);
    power_motor(MOTOR_B, COUNTERCLOCK, 255);
    power_motor(MOTOR_C, COUNTERCLOCK, 255);
    delay(2500);
}
//----COUNTERCLOCKWISE
void counter_clock_move (void)
{
    power_motor(MOTOR_A, CLOCKWISE, 255);
    power_motor(MOTOR_B, CLOCKWISE, 255);
    power_motor(MOTOR_C, CLOCKWISE, 255);
    delay(2500);
}
//----QUIET----
void quiet (void)
{
    stop_motor(MOTOR_A);
    stop_motor(MOTOR_B);
    stop_motor(MOTOR_C);
    delay(2500);
}
```