

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO
AMAZONAS
CAMPUS MAUÉS

KARTDUINO - CONTROLE DE UM CARRO ROBÔ
SIMULADO NO TINKERCAD

MAUÉS-AM

2020

LARISSA ZÓZIMO ANTUNES

TARSIS GERMANO MEDEIROS SAUNIER

KARTDUINO - CONTROLE DE UM CARRO ROBÔ

SIMULADO NO TINKERCAD

Projeto de Conclusão de Curso Técnico apresentado como requisito parcial para obtenção do título de Técnico de Informática do Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, Campus Maués.

Orientador: Professor Msc. Edcarlos da Silva Santana.

MAUÉS-AM

2020

SUMÁRIO

1 INTRODUÇÃO	1
2 OBJETIVOS	2
2.1 OBJETIVOS GERAL	2
2.2 OBJETIVOS ESPECÍFICOS.....	2
2.3 JUSTIFICATIVA.....	2
3 FUNDAMENTAÇÃO TEÓRICA.....	3
3.1 Robótica Educacional	3
3.2 Robôs.....	4
3.3 Simuladores	5
3.4 Simulador Tinkercad	6
3.5 Arduino	7
3.6 Kartduino	8
3.6.1 PONTE H L293D	8
3.6.2 MOTOR CC	9
3.6.3 SENSOR DE DISTÂNCIA ULTRASSÔNICO	9
4 METODOLOGIA	10
5 DESENVOLVIMENTO DO PROJETO	11
5.1 Simulação e Modelagem	11
5.2 Visão Geral do Kartduino	12
5.3 Simulação no Tinkercad	13
5.4 Código Fonte na programação do Kartduino	14
6 CONCLUSÃO E CONSIDERAÇÕES FINAIS	16
7 REFERÊNCIAS.....	17
APÊNDICE A.....	21

LISTA DE FIGURAS

Figura 01 - Classificação dos Robôs.....	4
Figura 02 - Interface Gráfica (Circuito).....	6
Figura 03 - Arduino Uno.....	7
Figura 04 - Ponte H L293D.....	8
Figura 05 - MOTOR CC.	9
Figura 06 - Sensor Ultrassônico.	9
Figura 07 - Modelagem do circuito Kartduino.....	11
Figura 08 - Modelo Geral do Kartduino.	12
Figura 09 - Modelagem (somente motores e ponte h).	13
Figura 10 - Kartduino Finalizado.	14
Figura 11 - Tela do Tinkercad-Arduino.....	14
Figura 12 - Exemplificação do código.	15

1 INTRODUÇÃO

Há uma grande diferença entre robôs e máquinas comuns. Para ser considerado um robô é necessário ser composto por sensores, *softwares* e manipuladores LOUSADA (2020). As áreas de aplicação da robótica vão muito além das industriais, contribuindo na medicina, no âmbito espacial, humanoide e com destaque na educação (Universidade Federal De São João Del-Rei, 2010).

MODELIX (2015) esclarece que a robótica educacional é uma metodologia de ensino que visa incentivar o aluno à investigação e à materialização dos conceitos assimilados no conteúdo curricular, indo muito além da construção de projetos e programação de robôs. A robótica proporciona um aprendizado prático que incrementa a capacidade de pensar e encontrar soluções para as problematizações propostas.

O simulador *Tinkercad* vem com a capacidade de produzir objetos 3D e simular circuitos com *Arduino*, que mostrou ter ferramentas tanto básicas como complexas, desde de *protoboards* até sensores. A programação do mesmo é completamente adaptável para ser utilizada no próprio *software* do *Arduino*. A compilação da simulação exibe na tela o funcionamento que condiz literalmente com a realidade, sendo possível modificar valores dos sensores na própria ferramenta para os testes.

Este projeto tem como objetivo de implementar o Kartduino, um carro robô simulado em uma ferramenta baseada na plataforma *Arduino*, a fim de impulsionar mais projetos, contribuindo com o avanço da robótica educacional no Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, Campus Maués (IFAM CMA).

2 OBJETIVOS

2.1 OBJETIVOS GERAL

Simular um protótipo de carro robótico utilizando a plataforma de *hardware* livre *Arduino*, a fim de impulsionar mais projetos de robótica no IFAM Campus-Maués.

2.2 OBJETIVOS ESPECÍFICOS

- Utilizar a plataforma *Arduino* para desenvolver o carro robótico;
- Encontrar uma ferramenta para a simulação de projetos com *Arduino*;
- Simular cada parte do Kartduino através da ferramenta.

2.3 JUSTIFICATIVA

Noemi (2019) aponta a robótica educacional como uma forma de aproximar os discentes da ciência e da tecnologia. Em função disso, a programação de robôs proporciona possibilidades que auxiliam na melhoria da qualidade de ensino. A execução de maneira prática e o treinamento do conhecimento adquirido podem estimular o interesse dos alunos e oferecer mais motivação para os estudos.

Os simuladores possibilitam que sejam feitas simulações de situações experimentais. Um exemplo disso é o *Tinkercad* utilizado neste projeto a fim de auxiliar no desenvolvimento da robótica educacional. A utilização de simuladores apresenta algumas vantagens, como a interatividade e participação legítima dos discentes, baixo custo, otimizador de tempo e os alunos podem testar diversas vezes sem medo de errar. Ademais essas ferramentas podem ser utilizadas no ensino a distância, que estão cada vez mais presentes na conjuntura atual, com a pandemia do COVID-19 (MENDES, 2019).

Portanto, com o Kartduino, procurou-se impulsionar os discentes à utilização dos simuladores, com destaque ao *Tinkercad*, que se mostrou uma ferramenta extremamente eficaz em simulações do *Arduino*. O mesmo possibilita o auxílio em diversos âmbitos do ensino, contribuindo com a robótica educacional, seja estimulando o raciocínio lógico ou fortalecendo o trabalho em equipe.

3 FUNDAMENTAÇÃO TEÓRICA

3.1 Robótica Educacional

A Robótica Educacional destaca-se entre os principais métodos tecnológicos empregados na educação. Do ponto de vista de Ouchana (2015), a robótica educacional está além de um otimizador de tempo, serve para estimular a criatividade e emancipação dos discentes de qualquer instituição pública ou privada, do ensino fundamental ao nível superior, trazendo a inovação que a educação necessita para promover novas formas de aprendizagem. Nesses novos tempos, é imperativo desenvolver novos métodos educacionais e tecnológicos, para competir em igualdade com os avanços industriais, sociais entre outros (ARGENTO, 2019).

A Robótica Educacional é:

Uma metodologia de ensino cujo objetivo é instigar o aluno a construir os próprios conhecimentos a partir da realização de uma ação concreta, ou seja, ela proporciona a prática dos conhecimentos teóricos, a princípio, nas matérias de matemática e física, mas nada impede que ela seja trabalhada de forma multidisciplinar (ESCOLA EDUCAÇÃO, 2018).

Nesse caso, compreende-se a Robótica Educacional como um recurso que possibilita resolver a necessidade da prática científica na Educação básica, por se tratar de um dispositivo tecnológico que permite explorar e expressar as ideias dos educandos na resolução de problemas cotidianos. Como resultado, a utilização da ferramenta leva o aluno a ir além da simples observação das formas de solução e modelagem, oferecendo uma aprendizagem significativa e científica (BIDIN, 2019).

Em síntese, o Escolas Disruptivas (2020) explica que as aulas de robótica quando aplicadas dentro da metodologia *STEAM* (Ciências, Tecnologias, Engenharia, Arte e Matemática), traz um ponto de vista diferenciado para a física. Com a ajuda da robótica, o discente verá problemas do cotidiano serem resolvidos de modo objetivo e interessante, desta forma sua criatividade e curiosidade serão estimuladas.

3.2 Robôs

Os robôs são agentes físicos que executam tarefas manipulando o mundo físico. Esses robôs surgiram a partir da necessidade de trocar o trabalho humanizado pelo das máquinas por serem mais rápidos e apresentarem maior eficácia. Esse pensamento surgiu por volta da segunda metade do século XX e vem ganhando visibilidade até os tempos atuais (RUSSELL, 2013).

A classificação dos robôs utiliza diversos critérios, tais como: autonomia do sistema de controle, mobilidade da base, estrutura cinemática, forma de acionamento, graus de liberdade, geometria do espaço de trabalho, entre outros (Universidade Federal do ABC, 2010).

Figura 01 - Classificação dos Robôs.

Sistema de controle	Mobilidade da base	Estrutura Cinemática	Espaço de trabalho
Equipamentos teleoperados	{ Veículo teleoperado Manipulador teleop.		
Robôs	Móveis	{ Aquáticos Aéreos Terrestres	{ Submarinos Marinos
			{ Pernas Rodas
	Fixos	{ Paralelos Série	{ 3 - 6 GdL Cartesiano Cilíndrico Esférico Articulado SCARA

Fonte: Universidade Federal do ABC, 2010.

Para Gamero (2018), robôs móveis na indústria concedem inúmeros benefícios, como segurança, pois podem desviar de obstáculos ao interromper sua trajetória suavemente; facilidade na instalação, pois é possível adaptar a qualquer meio sem a necessidade de trilhos ou fitas magnéticas; versatilidade nas funções podendo ser controlado até por dispositivos móveis e configurados, custo e benefício adaptando as necessidades do usuário para determinada tarefa.

Em suma, um exemplo prático da utilização de robôs é o *Arduino*, para a automação residencial ou comercial, conectados a sensores de temperatura, umidade do ar, luminosidade, entre outros. Além disso, o *Arduino* tem ganhado espaço no campo da impressão 3D, e vem sendo adotado por usuários que se arriscam a projetar e produzir a sua impressora 3D do zero (DEAL TECHNOLOGIES, 2020).

3.3 Simuladores

O simulador é uma ferramenta que lembra o comportamento de um sistema diante de uma determinada situação, o que permite a execução de forma lógica antes da aplicação prática (CONCEITO.DE, 2014).

Segundo Calomeno (2017), com o avanço tecnológico, os simuladores permitiram ambientes de aprendizagem interativos, voltados para o entretenimento e, principalmente, para o âmbito educacional. Isso pode ser feito por meio de plataformas *on-line*, *softwares*, conteúdos multimídia, entre outros.

A utilização dos simuladores tem se tornado extremamente abrangente e essencial, em diversas áreas de conhecimento. Neste projeto, por exemplo, a simulação do *hardware* livre *Arduino* possibilitou a redução significativa de tempo e de custos econômicos, por se tratar de uma ferramenta gratuita, além de evitar possíveis acidentes relacionados aos circuitos eletrônicos.

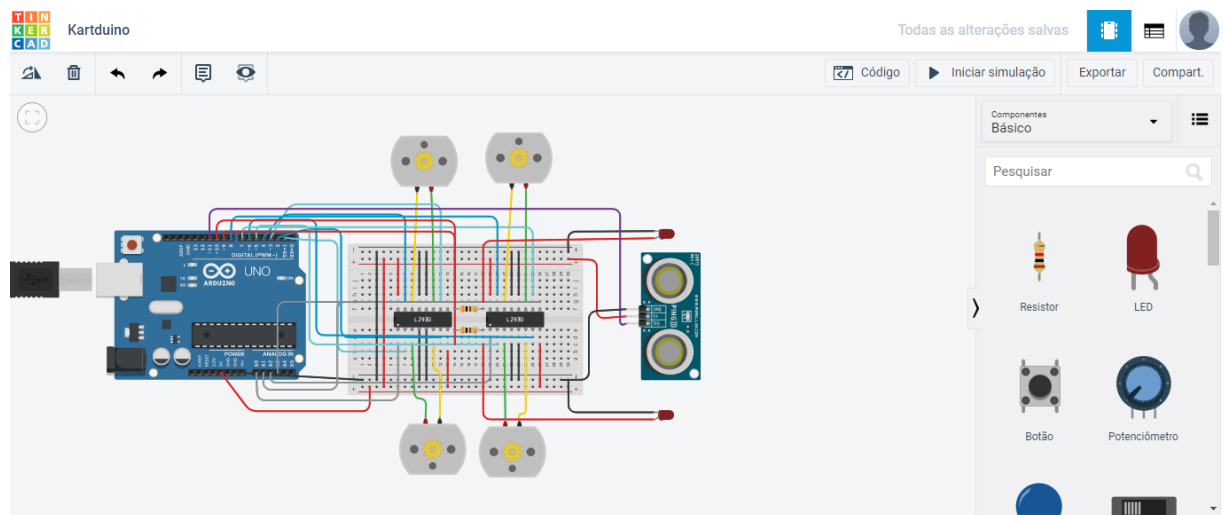
Um exemplo de simulador educacional é o *Younder*, um simulador de direção, onde o usuário tem uma experiência de aprendizagem imersiva por meio da simulação. Com telas ou realidade virtual, o cliente pratica, demonstra, aprende e recicla conhecimentos em um ambiente controlado, anulando riscos de acidentes na fase prática (YOUNDER, 2020).

3.4 Simulador Tinkercad

Tinkercad é um *software* gratuito *on-line* de modelagem 3D e simulador de circuitos *Arduino*, muito conhecido por sua simplicidade e facilidade de utilização. O *Tinkercad* foi criado em 2010 na união Europeia pelo ex-engenheiro do *Google* Kai Backman e pelo cofundador Mikko Mononem, com o intuito de fazer modelagem 3D (WIKI, 2020).

De acordo com Neves (2017), a plataforma *Tinkercad* é extremamente simples, possui uma interface flexível e elegante. Além de ser gratuita a ferramenta, permite criar circuitos da plataforma de *hardware* livre *Arduino* e modificá-los rapidamente sem que haja o perigo de queimar componentes eletrônicos.

Figura 02 - Interface Gráfica (Circuito).



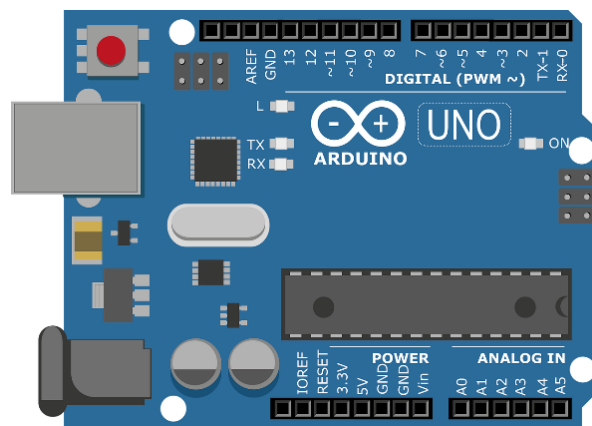
Fonte: Próprios Autores, 2020.

A figura 02 apresenta a interface da opção circuitos do *Tinkercad*, onde podemos arrastar os itens localizados na aba de componentes e formar circuitos apenas com alguns cliques. A interface mostra simplicidade e eficácia para se reproduzir protótipos como os da vida real, porém com benefício da utilização ilimitada dos componentes.

3.5 Arduino

Thomsen (2017) define *Arduino* como uma placa composta por um microcontrolador *Atmel*, circuitos de entrada/saída e que pode ser facilmente conectada a um computador e programada via IDE (*Integrated Development Environment*, ou Ambiente de Desenvolvimento Integrado) utilizando uma linguagem baseada em C/C++, sem a necessidade de equipamentos extras além de um cabo *USB*. Em termos práticos, as placas possuem funcionamento semelhante ao de um pequeno computador, no qual pode-se programar a maneira como suas entradas e saídas devem se comportar em meio aos diversos componentes externos que podem ser conectados às mesmas (MOTA, 2017).

Figura 03 - Arduino Uno.



Fonte: Seven au/Pixabay, 2017.

Outra vantagem que podemos citar sobre ele é que utiliza o conceito de *hardware* livre, o que significa que qualquer um pode montar, modificar, melhorar e personalizar o *Arduino*, partindo do mesmo *hardware* básico (THOMSEN, 2017). O objetivo seria elaborar um dispositivo que fosse ao mesmo tempo barato, funcional e fácil de programar, sendo dessa forma acessível a estudantes e projetistas amadores.

3.6 Kartduino

Para a idealização deste projeto utilizou-se como referência Almeida *et al.* (2015), o qual demonstra a construção de um carro robô controlado por dispositivos móveis utilizando o sistema operacional *Android* e uma placa *Arduino*.

A princípio almejava-se a prototipação deste projeto, porém, devido à conjuntura atual, a pandemia causada pela COVID-19, foi necessária a adaptação a este novo cenário. Como solução, desenvolveu-se a construção do Kartduino através do simulador *Tinkercad*, após a reestruturação dos objetivos deste projeto.

Os robôs móveis tornaram-se mais comuns em ambientes industriais e comerciais. Várias áreas utilizam os robôs móveis para a locomoção de matérias, como por exemplo, hospitais, armazéns, ambientes militares e de segurança (HISOUR, 2020).

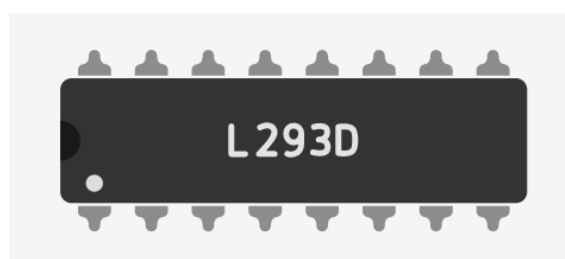
Kartduino é um carro robô móvel automático guiado por um sensor ultrassônico, que tem como objetivo apoiar novos projetos, para futuras implementações relacionadas ao tema, como carros robôs controlados via *bluetooth*, ou guiados por sensores.

3.6.1 PONTE H L293D

Para o controle dos motores com *Arduino* é utilizado o componente chamado Ponte H, um sistema similar ao de várias chaves ligadas ao terminal do motor, através das correntes altas. O *Arduino* atua abrindo e fechando as chaves (LIMA, 2017).

Na simulação, esse componente tem como objetivo controlar os dois motores CC do Kartduino, de forma independente.

Figura 04 - Ponte H L293D.



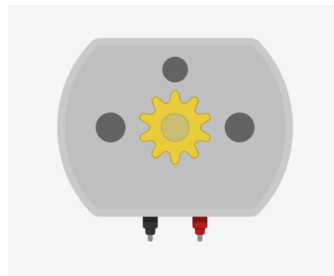
Fonte: Próprios Autores, 2020.

3.6.2 MOTOR CC

Trata-se de um motor abastecido por corrente contínua (CC), sendo este alimentado por uma bateria ou qualquer outra fonte alimentação CC. O motor é controlado apenas variando a sua tensão, diferente do motor de corrente alternada (CA) (SILVEIRA, 2020).

Na simulação, o motor é utilizado para a locomoção do Kartduino. Foram utilizado quatro motores para maior mobilidade. Por se tratar de um simulador, não se faz necessário especificar o modelo utilizado.

Figura 05 - MOTOR CC.



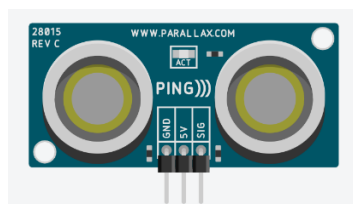
Fonte: Próprios Autores, 2020.

3.6.3 SENSOR DE DISTÂNCIA ULTRASSÔNICO

Sensor ultrassônico é geralmente utilizado em projetos onde é necessário medir distância ou evitar colisões. A distância é calculada a partir da emissão de um pulso sonoro, sendo a partir dele calculada a distância.

Na aplicação, utilizou-se o sensor ultrassônico unido à ponte H para o controle dos motores CC com objetivo de movimentar o Kartduino.

Figura 06 - Sensor Ultrassônico.



Fonte: Próprios Autores, 2020.

O funcionamento do sensor ultrassônico é:

Semelhante ao sonar dos morcegos, o sistema do sensor ultrassônico funciona com o princípio de funcionamento a emissão de uma onda sonora de alta frequência. O objeto a ser detectado (que é capaz de refletir essa onda) resulta em um eco, que é convertido em sinais elétricos. A detecção desse eco depende da intensidade e da distância entre o objeto e o sensor. É a partir disso que se sabe se o objeto está dentro dos parâmetros estabelecidos, ou mesmo se está no local para o sensor identificá-lo (BALLUFF, 2018).

4 METODOLOGIA

Como caracteriza Silva (2017), metodologia é uma união de procedimentos para garantir a legitimidade do conhecimento adquirido. Desta forma, podemos partir do princípio de que este projeto irá abranger o caráter de pesquisa qualitativa, analisando a complexidade de determinado problema e contribuindo no processo da mudança, o que facilita a mais variada compressão dos indivíduos (DIEHL, 2004).

Empregou-se o estudo de caso na qual se busca a aplicação prática de conhecimentos para soluções de problemas sociais, encontrados no IFAM Campus-Maués (BOAAVENTURA, 2004).

O projeto foi realizado no Instituto Federal de Educação, Ciência e Tecnologia, na cidade de Maués-AM, localizada a 258 KM de Manaus.

A metodologia aplicada neste projeto, pautada nas citações acima, segue os seguintes procedimentos:

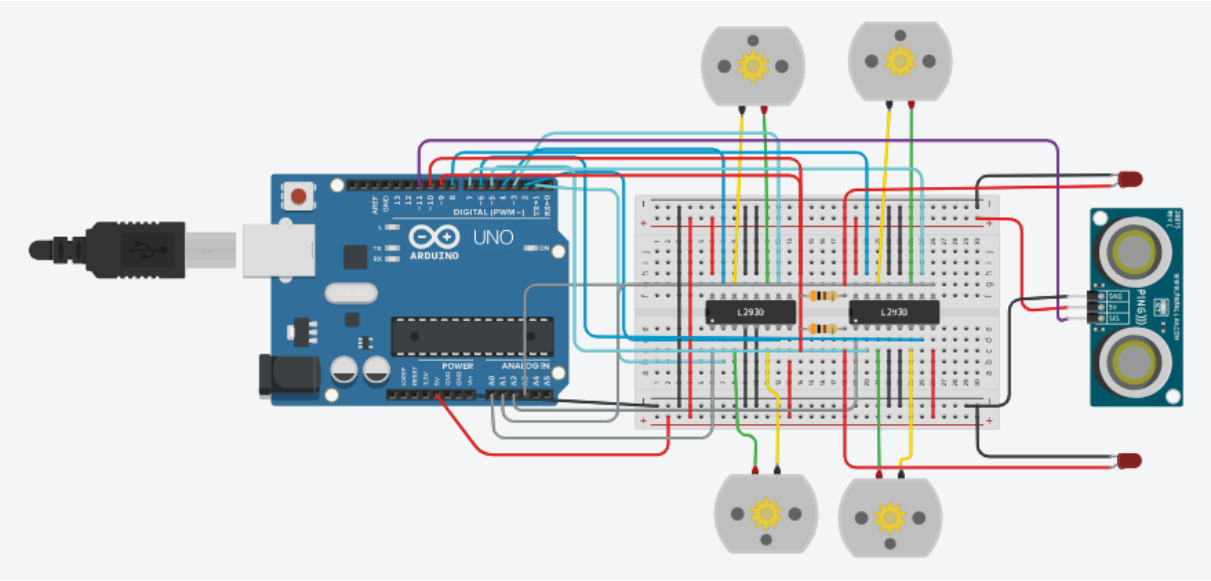
- a) **Planejamento:** revisão de literatura, análise em artigos encontrados em sites, revistas eletrônicas e bibliográficas.
- b) **Reconhecimento dos componentes:** seleção dos componentes que considerassem a eficiência e preço acessível para desenvolver o protótipo *Kartduino*.
- c) **Simulação:** representação da parte física do projeto utilizando a plataforma *Arduino* através da ferramenta *Tinkercad*.
- d) **Desenvolvimento do software:** produção do código fonte para o controle do *Kartduino*.
- e) **Testes e Resultados:** realização de testes funcionais com a finalidade de analisar e expor os resultados obtidos.

5 DESENVOLVIMENTO DO PROJETO

5.1 Simulação e Modelagem

A modelagem do Kartduino pode ser vista na figura 07 logo abaixo:

Figura 07 - Modelagem do circuito Kartduino.



Fonte: Próprios autores, 2020.

Algo a se destacar do simulador *Tinkercad* é que ao concluir-se o projeto, a própria construção do modelo pode ser utilizada de base para outros projetos *Arduino*, bastando organizar os *jumpers* e componentes.

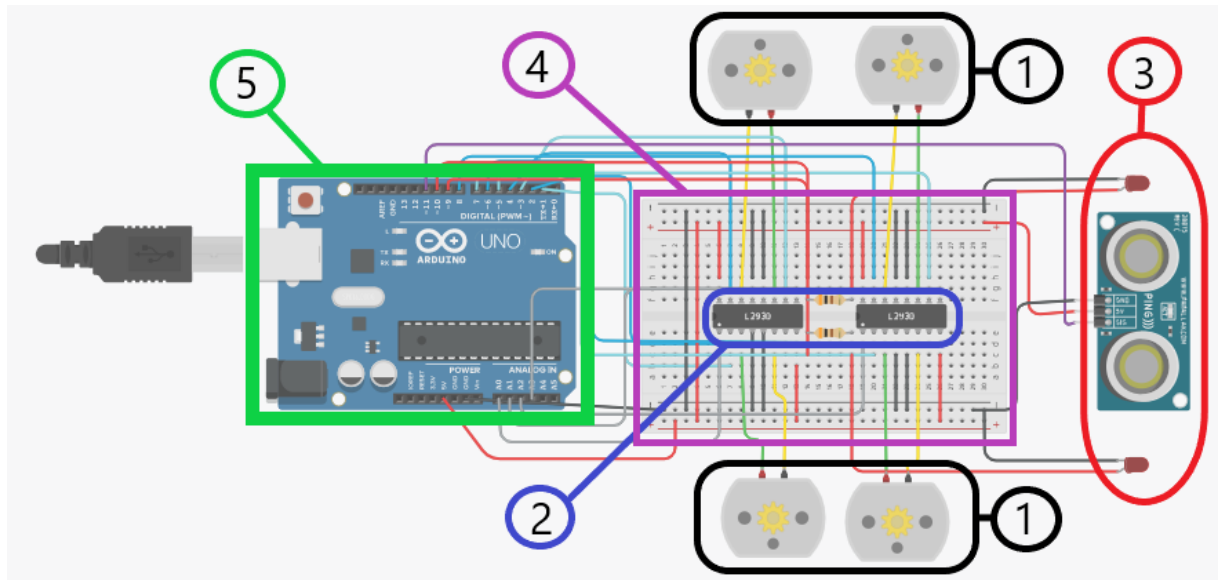
Neste projeto, utilizamos os seguintes materiais:

Itens utilizados	Quantidade
Arduino Uno R3	1
Acionador de motor de ponte H	2
Motor CC	4
LED vermelho	2
300 Ω Resistor	2
Sensor de distância ultrassônico	1

5.2 Visão Geral do Kartduino

A modelagem lógica do projeto foi feita no próprio site *Tinkercad*. A Figura 8 fornece a visão geral do modelo Kartduino.

Figura 08 - Modelo Geral do Kartduino.



Fonte: Próprios autores, 2020.

Os elementos utilizados no desenvolvimento do Kartduino são apresentados pelos números de 1 a 5 e estão retratados a seguir:

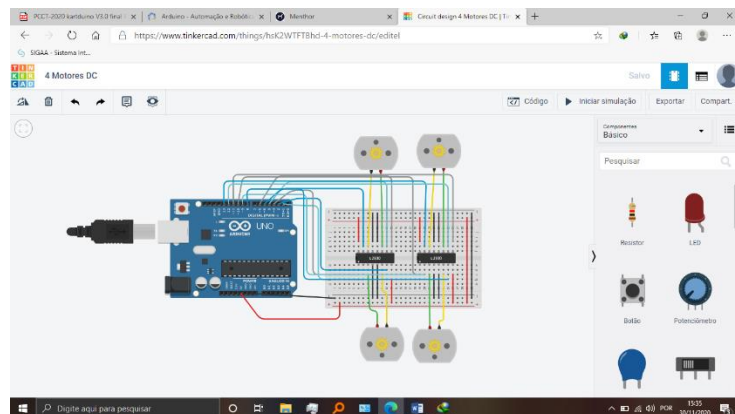
1. **Motores CC:** o motor de corrente contínua é controlado variando a sua tensão.
2. **Ponte H (L293D):** é utilizada a fim de controlar o sentido da corrente, ou seja, comandar os motores.
3. **Sensor ultrassônico e LEDs:** o sensor é utilizado para medir a distância junto à Ponte H, onde é possível a movimentação lógica dos motores. Os LEDs simulam os faróis do Kartduino.
4. **Protoboard:** é a base onde são interligados todos os componentes eletrônicos junto dos *jumpers*.
5. **Arduino uno R3:** é o principal elemento do Kartduino, tem a função de processar o código fonte e conduz os sinais para os componentes exercerem suas devidas funções.

5.3 Simulação no Tinkercad

Fundamentado na indicação do orientador, encontramos o curso *IoT* que é uma iniciativa da *SAMSUNG*, a partir de um dos módulos do curso disponibilizado pela plataforma, o qual divulgou a ferramenta *on-line Tinkercad*.

Na sequência, iniciou-se a montagem do Kartduino no *Tinkercad*. Primeiramente explorou-se o funcionamento da parte principal do Kartduino, que é a motora, onde encontram-se apenas os motores CC e os circuitos de ponte H. As conexões são feitas através de *jumper*, entre o circuito ponte H e os motores CC, com auxílio da *protoboard*. Logo após, foi utilizado um código simples, o qual contava apenas com comandos para controlar os motores.

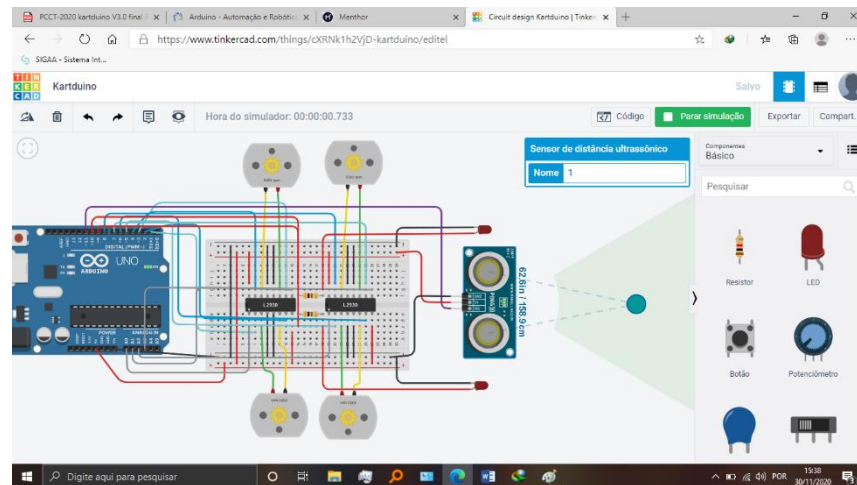
Figura 09 - Modelagem (somente motores e ponte h).



Fonte: Próprios autores, 2020.

Por fim, implementou-se os faróis e o sensor ultrassônico. Os faróis representados por LEDs, conectados por meio de *jumpers* e resistores e o sensor ultrassônico foi conectado ao *Arduino* para automatizar o sistema motor do Kartduino. O código foi amplamente melhorado, com métodos junto do sensor ultrassônico, a fim de evitar possíveis colisões e acelerar automaticamente.

Figura 10 - Kartduino Finalizado.

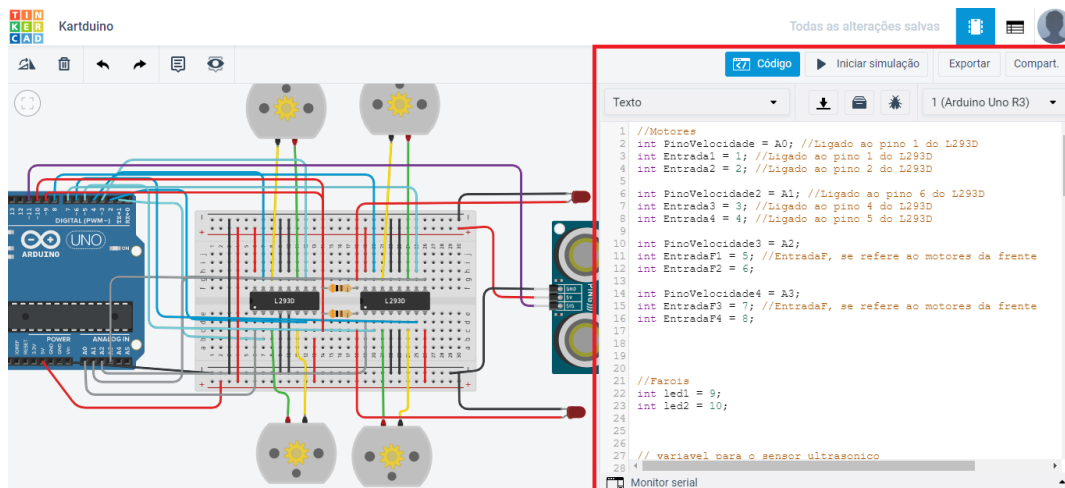


Fonte: Próprios autores, 2020.

5.4 Código Fonte na programação do Kartduino

Na Figura 11, é apresentada um trecho do código fonte utilizado no Kartduino, produzido na ferramenta *Tinkercad*.

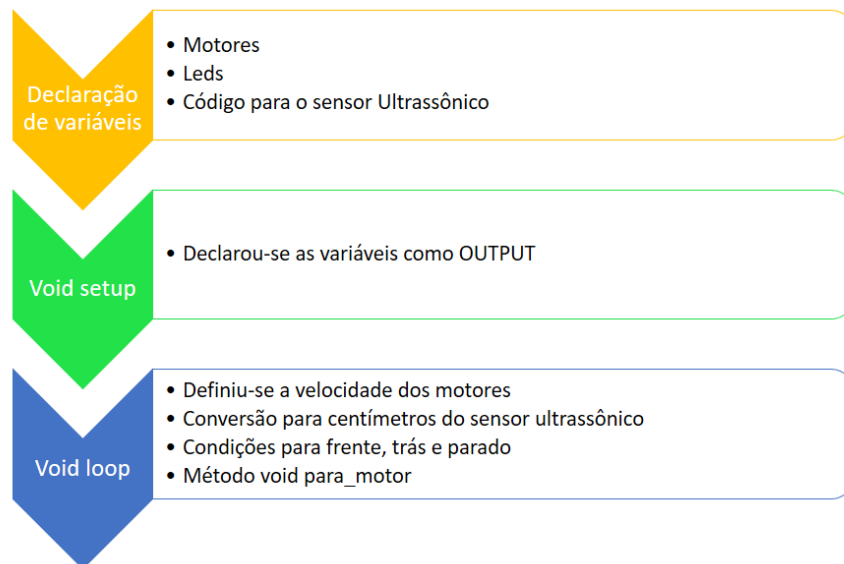
Figura 11 - Tela do *Tinkercad*-Arduino.



Fonte: Próprios autores, 2020.

A Figura 12 apresenta um breve resumo do código utilizado no Kartduino. Basicamente, o controle dos movimentos do Kartduino é composto pelo sensor ultrassônico, que envia os dados analisados para o circuito ponte H, que movimenta os motores CC de acordo com o sentido solicitado.

Figura 12 - Exemplificação do código.



Fonte: Próprios autores, 2020.

6 CONCLUSÃO E CONSIDERAÇÕES FINAIS

O presente trabalho expôs o Kartduino, um carro robô controlado pelo simulador *on-line Tinkercad*, que apresentou maneiras eficientes e novas possibilidades de simular e trabalhar com o *hardware* livre *Arduino*, atendendo todos os parâmetros dos objetivos do projeto. Diante da conjuntura atual, a pandemia causada pelo COVID-19, esta ferramenta foi de suma importância para a realização deste projeto, sendo um ótimo otimizador de tempo e custo-benefício.

Diante dos objetivos propostos, obteve-se resultados extremamente satisfatórios. Trabalhar com o *Arduino* no *Tinkercad* mostrou-se útil na simulação do Kartduino. No simulador, foram produzidos circuitos completos e totalmente funcionais que, se aplicados na forma prática, certamente há de se ter o funcionamento e a prototipação garantidos.

Como pontos a serem destacados, por meio deste projeto encontrou-se a oportunidade da aplicação dos conhecimentos práticos e teóricos obtidos ao longo do Curso Técnico Integrado em Informática. Dentre as vantagens de se utilizar um simulador para a prototipação na plataforma *Arduino*, destaca-se a produtividade em curto prazo de tempo, tendo em vista a demora na chegada de componentes do exterior, além de reduzir consideravelmente os custos, levando em consideração o preço dos componentes necessários para a prototipação. Os pontos a serem melhorados estão em prototipar o Kartduino para a comparação com a simulação do *Tinkercad*, com o objetivo de mostrar a eficiência do simulador.

Este projeto tem potencial de divulgar uma nova ferramenta para auxiliar no desenvolvimento da robótica educacional no IFAM-Campus Maués. Dessa forma, os discentes, ao tomarem conhecimento da robótica educacional, poderão utilizar este projeto para impulsionar a produção de novos trabalhos, aplicando os conhecimentos com os materiais exibidos neste projeto.

7 REFERÊNCIAS

ALMEIDA, T. *et al.* **CARDUINO-UM CARRO ROBÔ CONTROLADO POR DISPOSITIVOS MÓVEIS ANDROID.** Formiga, MG. Disponível em: www.researchgate.com. Acesso em: 30 jan. 2020.

ARGENTO, H. **Importância de aprender robótica no Ensino Fundamental I.** Disponível em: <https://www.professordofuturo.com.br/importancia-de-aprender-robotica-no-ensino-fundamental>. Acesso em: 29 jan. 2020.

BALLUFF. **Sensor ultrassônico: como ele funciona e de que modo pode ajudar a sua indústria?** Disponível em: <https://balluffbrasil.com.br/sensor-ultrassonico-como-ele-funciona-e-de-que-modo-pode-ajudar-a-sua-industria/>. Acesso em: 23 nov. 2020.

BIDIN, S. Administradores. **A Robótica Educacional.** Disponível em: <https://administradores.com.br/artigos/a-rob%C3%B3tica-educacional>. Acesso em: 30 nov. 2020.

BOAAVENTURA, E.M. *Metodologia da pesquisa*. São Paulo: Atlas, 2004.

CALOMENO, C. SIMULADORES EDUCACIONAIS: DEFINIÇÕES E APROPRIAÇÕES COMO OBJETOS DE APRENDIZAGEM. **Educação**, Paraná, v. 21, n. 1, p. 257-269, abr./2017.

CONCEITO.DE. **Conceito de Simulador.** Disponível em: conceito.de/simulador. Acesso em: 30 nov. 2020.

DEAL TECHNOLOGIES. **O Arduino e o mundo da automação e robótica.** Disponível em: <https://www.deal.com.br/blog/o-arduino-e-o-mundo-da-automacao-e-robotica/>. Acesso em: 30 nov. 2020.

DIEHL, A. A; TATIM, D. C. **Pesquisa em ciências sociais aplicadas: métodos e técnicas.** São Paulo: Prentice-Hall, 2004.

ESCOLAS DISRUPTIVAS. **Ensino de física: entenda como a robótica pode auxiliar.**

Disponível em: <https://escolasdisruptivas.com.br/steam/ensino-de-fisica/>. Acesso em: 30 nov. 2020.

ESCOLA EDUCAÇÃO. **Robótica Educacional: Descubra o que é e quais os benefícios para os alunos.**

Disponível em: <https://escolaeducacao.com.br/robotica-educacional/>. Acesso em: 28 jan. 2020.

GAMERO, I. **Robôs Industriais: tudo o que você precisa saber!** Disponível em:

<https://www.pollux.com.br/blog/robos-industriais-tudo-o-que-voce-precisa-saber/>. Acesso em: 1 fev. 2020.

HISOUR. **ROBÔ MOVÉL.** Disponível em: <https://www.hisour.com/pt/mobile-robot-42899/>. Acesso em: 30 nov. 2020.

LIMA, I. **Aprenda a utilizar a ponte H L293D com Arduino.** Disponível em:

<https://autocorerobotica.blog.br/aprenda-utilizar-ponte-h-l293d-com-arduino/>. Acesso em: 6 nov. 2020.

LOUSADA, R. **O que é Robótica: conceito, história e evolução.** Disponível em:

<https://blog.eletrogate.com/o-que-e-robotica-conceito-historia-e-evolucao/>. Acesso em: 22 nov. 2020.

MENDES, M. **USO DE SIMULADORES NO ENSINO DE FÍSICA.** Disponível em:

<https://educador.brasilecola.uol.com.br/estrategias-ensino/uso-simuladores-no-ensino-fisica.htm#:~:text=O%20uso%20de%20simuladores%20apresenta,um%20experimento%20com%20muitas%20vari%C3%A1veis>. Acesso em: 22. nov. 2020.

MODELIX. **O que é Robótica Educacional.** Disponível em: <http://modelix.cc/o-que-e-robotica-educacional/>. Acesso em: 22 nov. 2020.

MOTA, A. **O QUE É ARDUINO E COMO FUNCIONA?** Disponível em:

<https://portal.vidadesilicio.com.br/o-que-e-arduino-e-como-funciona/>. Acesso em: 24 jan. 2020.

NEVES, M. S. das. **Apostila de introdução ao Tinkercad (antigo Circuits.io)**. Disponível em: <https://docplayer.com.br/65200375-Apostila-de-introducao-ao-tinkercad-antigo-circuits-io.html>. Acesso em: 11 nov. 2020.

NOEMI, D. **O que é robótica educacional e como implantar na escola**. Disponível em: <https://escolasdisruptivas.com.br/steam/o-que-e-robotica-educacional-e-como-implantar-na-escola/#:~:text=Conhe%C3%A7a%20as%20principais%20vantagens%20da,melhorar%20a%20qualidade%20de%20ensino..> Acesso em: 22 nov. 2020.

OUCHANA, D. Revista educação. **O que é a robótica educacional e quais são os ganhos para o aprendizado**. Disponível em: <https://revistaeducacao.com.br/2015/12/02/o-que-e-a-robotica-educacional-e-quais-sao-os-ganhos-para-o-aprendizado/>. Acesso em: 29 jan. 2020.

RUSSEL, S.; NORVIG, P. **Inteligência Artificial**. 2ª Edição, Rio de Janeiro, Campus, 2013.

SILVA, C. N. N. da. Metodologia científica descomplicada: pesquisa e pratica para iniciante. **EDITORA IFB**. Acesso em: 22 nov. 2020.

SILVEIRA, C. B. **Motor CC: Saiba como funciona e de que forma Especificar**. Disponível em: <https://www.citisystems.com.br/motor-cc/>. Acesso em: 6 nov. 2020.

THOMSEN, A. **O que é Arduino?** Disponível em: <https://www.filipeflop.com/blog/o-que-e-arduino/>. Acesso em: 24 jan. 2020.

Universidade Federal De São João Del-Rei **Aplicações da Robótica**. São João Del-Rei, Minas Gerais. Acesso em: 30 nov. 2020

Universidade Federal do ABC. **Fundamentos de Robótica**. Disponível em: <http://pgene.ufabc.edu.br/docentes/Riascos/ensino/disciplinas/Robotica/FundamentosRobotica.html>. Acesso em: 5 fev. 2020.

WIKI. **Tinkercad**. Disponível em: <https://en.wikipedia.org/wiki/Tinkercad>. Acesso em: 11 nov. 2020.

YOUNDER. **Simuladores de direção educacionais para treinamentos práticos**. Disponível em: <https://younder.com.br/solucoes/simuladores-de-direcao-educacionais/>. Acesso em: 30 nov. 2020.

APÊNDICE A

```
//Motores do Kartduino;

//Aqui declaramos as variáveis dos Pinos de velocidade nas entradas analógicas do Arduino,
para o controle da velocidade;

// Aqui declaramos as variáveis das EntradasX ao Arduino, para controlar para qual sentido
os motores irão girar.

int PinoVelocidade = A0; //Ligado ao pino 1 do L293D
int Entrada1 = 1; //Ligado ao pino 1 do L293D
int Entrada2 = 2; //Ligado ao pino 2 do L293D

int PinoVelocidade2 = A1; //Ligado ao pino 6 do L293D
int Entrada3 = 3; //Ligado ao pino 4 do L293D
int Entrada4 = 4; //Ligado ao pino 5 do L293D

int PinoVelocidade3 = A2;
int EntradaF1 = 5; //EntradaF, se refere ao motores da frente
int EntradaF2 = 6;

int PinoVelocidade4 = A3;
int EntradaF3 = 7; //EntradaF, se refere ao motores da frente
int EntradaF4 = 8;

//Farois (Leds) do Kartduino
//Duas variáveis para os leds
int led1 = 9;
int led2 = 10;
```

```
// variável para utilizarmos a conversão para centímetros
int cm = 0;

//Código fornecido pelo próprio Tinkercad para o funcionamento do sensor ultrassônico
long readUltrasonicDistance(int triggerPin, int echoPin)
{
    pinMode(triggerPin, OUTPUT); // Clear the trigger
    digitalWrite(triggerPin, LOW);
    delayMicroseconds(2);
    // Sets the trigger pin to HIGH state for 10 microseconds
    digitalWrite(triggerPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(triggerPin, LOW);
    pinMode(echoPin, INPUT);
    // Reads the echo pin, and returns the sound wave travel time in microseconds
    return pulseIn(echoPin, HIGH);
}

//Void para ligarmos as variáveis declaradas anteriormente ao Arduino.
void setup()
{
    //Utiliza-se o comando pinMode (variável, OUTPUT), para declararmos como a saída dos
    dados do Arduino, ou seja, os dados lidos pelo Arduino, irão fazer os motores e Leds realizar
    comandos de acordo com o programado.
    pinMode(PinoVelocidade, OUTPUT);
    pinMode(Entrada1, OUTPUT);
    pinMode(Entrada2, OUTPUT);
```

```
pinMode(PinoVelocidade2, OUTPUT);
pinMode(Entrada3, OUTPUT);
pinMode(Entrada4, OUTPUT);
```

```
pinMode(PinoVelocidade3, OUTPUT);
pinMode(EntradaF1, OUTPUT);
pinMode(EntradaF2, OUTPUT);
```

```
pinMode(PinoVelocidade4, OUTPUT);
pinMode(EntradaF3, OUTPUT);
pinMode(EntradaF4, OUTPUT);
```

```
pinMode(led1,OUTPUT);
pinMode(led2,OUTPUT);
}
```

//Void loop são as sequências de comandos que serão executados em looping (várias vezes), ou até definirmos um limite para a parada.

```
void loop()
```

```
{
```

//Define a velocidade de rotação dos motores do Kartduino, escolhemos 500 pois é melhor para a demonstração no Tinkercad.

//Utiliza-se o comando analogWrite(variável, variável) para converter os dados analógicos do Arduino para os motores, assim sendo possível o melhor controle da velocidade.

```
int velocidade = 500;
analogWrite(PinoVelocidade, velocidade);
analogWrite(PinoVelocidade2, velocidade);
analogWrite(PinoVelocidade3, velocidade);
analogWrite(PinoVelocidade4, velocidade);
```

```
// Leitura do Sensor ultrasonico;
```

// O sensor ultrassônico do Tinkercad vem com padrões americanos de distância, então é necessário a conversão desses valores por meio deste cálculo.

```
cm = 0.01723 * readUltrasonicDistance(11, 11);
```

//O if é utilizado para caso “se” o sensor ultrassônico detectar certa distância, execute uma sequência de comandos.

```
if (cm>80) {
```

//Se a distancia do Kartduino for maior que 80 centímetros, os motores executam os comandos para que o robô se movimente para a frente.

```
digitalWrite(Entrada1, HIGH);
```

```
digitalWrite(Entrada3, HIGH);
```

```
digitalWrite(Entrada2, LOW);
```

```
digitalWrite(Entrada4, LOW);
```

```
digitalWrite(EntradaF1, HIGH);
```

```
digitalWrite(EntradaF3, HIGH);
```

```
digitalWrite(EntradaF2, LOW);
```

```
digitalWrite(EntradaF4, LOW);
```

```
digitalWrite(led1,LOW);
```

```
digitalWrite(led2,LOW);
```

```
}
```

```
if (cm<=80 && cm>40) {
```

//Se a distancia do Kartduino for menor e igual 80 centímetros, e, maior que 40, os motores executam os comandos para que o robô se movimente para trás.

```
digitalWrite(Entrada1, LOW);
```

```
digitalWrite(Entrada3, LOW);
```

```
digitalWrite(Entrada2, HIGH);
```

```

digitalWrite(Entrada4, HIGH);
digitalWrite(EntradaF1, LOW);
digitalWrite(EntradaF3, LOW);
digitalWrite(EntradaF2, HIGH);
digitalWrite(EntradaF4, HIGH);
digitalWrite(led1,HIGH);
digitalWrite(led2,HIGH);
delay(200);
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
delay(200);
}

```

```

if (cm<=40) {

```

//Se a distancia do Kartduino for menor ou igual a 40 centímetros, os motores executam os comandos para que o robô permaneça no local, para evitar colisões.

```

digitalWrite(led1,HIGH);
digitalWrite(led2,HIGH);
para_motor();
}

```

```

}

```

//O método void para_motor() é utilizado para o Kartduino parar seus movimentos, ao final de cada if do Void loop, ele para por 1 segundo, depois retorna para a próxima leitura.

```

void para_motor()
{
digitalWrite(Entrada1, LOW);

```

```
digitalWrite(Entrada2, LOW);  
digitalWrite(Entrada3, LOW);  
digitalWrite(Entrada4, LOW);  
digitalWrite(EntradaF1, LOW);  
digitalWrite(EntradaF2, LOW);  
digitalWrite(EntradaF3, LOW);  
digitalWrite(EntradaF4, LOW);  
delay(1000);  
}
```