



## **ESCOLA TÉCNICA ESTADUAL DEP. ARY DE CAMARGO PEDROSO**

### **TÉCNICO EM AUTOMÇÃO INDUSTRIAL**

Matheus Paskevics Romão

Rafael Felipe Brainich

José Carlos Vinhotti Júnior

Paulo Eduardo Alvarenga

#### **Projeto de Robô Sumô**

#### **Guiado por Arduino via controle (celular)**

Trabalho de Conclusão de Curso da Etec Deputado Ary de Camargo Pedroso, orientado pelo Prof. Cláudio Volcov apresentado como requisito parcial para obtenção do título de Técnico em eletrônica.

**Piracicaba 2024**

## **TERMO DE CONSENTIMENTO**

### **TRABALHO DE CONCLUSÃO DE CURSO – TCC**

Nós, alunos abaixo assinados, regularmente matriculados no **Curso Técnico de Automação Industrial na ETEC Deputado Ary de Camargo Pedroso do Município de**

**Piracicaba/SP** declaramos ter pleno conhecimento dos Regulamentos para realização do Trabalho de Conclusão de Curso do Centro Paula Souza. Declaramos, ainda que, o trabalho apresentado é resultado do nosso esforço e autorizamos a divulgação do mesmo pela instituição.

Piracicaba, 25 de junho de 2024

## **AGRADECIMENTOS**

Gostaríamos de agradecer aos coordenadores e professores da Etec Dep. Ary de Camargo Pedroso, pela dedicação e empenho em nos proporcionar conhecimento e auxílio necessário, para que venhamos estar aptos a ingressar no mercado de trabalho.

Não poderíamos deixar de agradecer a todos os envolvidos que colaboraram para o desenvolvimento do TCC.

## **EPIGRAFE**

“Não creio que haja uma emoção, mais intensa para um inventor do que ver suas criações funcionando. Essa emoção faz você esquecer de comer, de dormir, de tudo.”

**-Nikola Tesla**

## RESUMO

Nosso projeto se trata de Robôs Sumos.

O projeto foi criado com a intenção de trazer entretenimento e diversão ao público, visando uma ideia de competição entre seus usuários, trazendo uma experiência única e emocionante.

**Palavras-Chave:** Robô controlado, Robô controlado a base de Arduino.

## ABSTRACT

Our project is about Robots Sumos.

The project was created with the intention of bringing entertainment and fun to the public, aiming at an idea of competition between its users, bringing a unique and exciting experience.

**Key-Words:** Controlled robot, Arduino-based controlled robot.

# SUMÁRIO

1. Introdução.....	7
1.1. Justificativa.....	7
1.2. Objetivo.....	7
2. Robô Sumô.....	8
Figura 1: Robô Sumô Arduino para batalhas.....	8
2.1. Como funciona um robô sumô.....	9
2.2. Qual a origem da competição de robô sumô.....	10
2.3. Como funciona a competição de robô sumô.....	10
Figura 2: competição robô sumo.....	11
3. Processo de montagem.....	12
3.1. Carrinho com Trilhos.....	12
Figura 1: Base do carrinho com esteira com dois motores auxiliares.....	12
Figura 2: Parte superior do carrinho com esteira.....	13
Figura 3: Corpo do carrinho com esteira.....	14
Figura 4: Defesa/Ataque, parte frontal do carrinho com esteira.....	15
Figura 5: Driver motor ponte H dupla L298n, controladora driver L298n.....	16
3.2. Carrinho com Rodas.....	17
Figura 6: Base do carrinho com roda com dois motores auxiliares.....	18
Figura 7: BlackBoard Uno R3 (Arduino).....	19
Figura 8: Módulo bluetooth HC-05.....	21
Figura 9: Driver motor ponte H dupla L298n, controladora driver L298n.....	22
Figura 10: Protótipo de Teste Carrinho com Rodas.....	23
4. Materiais Utilizados.....	24
5. Linha de Programação dos Carrinhos.....	25
5.1. Linha de Programação do Carrinho com Rodas.....	25

5.2. Linha de Programação do Carrinho com Esteiras.....	27
<b>6. Trabalho Concluído.....</b>	<b>32</b>
Figura 1: Trabalho Finalizado.....	32

## **1. INTRODUÇÃO**

Temos em mente com esse projeto, trazer um momento descontraído para os alunos, ao mesmo tempo mostrar nossas ideias e incentivar os alunos a criarem seus próprios projetos de diferentes tipos de robôs para fins de aprendizado e entretenimento. Para este projeto tivemos a ideia de fazer algo mais compacto, mas que ao mesmo tempo seja extremamente funcional. Nesse documento vamos estar comentando sobre dificuldades e experiências que encontramos para executar nosso projeto e também alterações que foram feitas para concluí-lo da melhor maneira possível. Utilizamos nossos conhecimentos e de outros profissionais para o auxílio, durante o processo, e assim mesclando tecnologia e técnicas complementares para a finalização.

### **1.1. Justificativa**

Inicialmente pensamos em fazer uma lixeira inteligente onde cada tipo de material era separado para depois ser descartado da maneira correta, assim criando uma maneira sustentável de descarte de lixo. Porém depois de muito debate, decidimos optar por algo mais complexo e desafiador de fazer, uma coisa que fosse ao mesmo tempo divertida e compacta. Mudamos de ideia por conta da falta de tempo em relação a complexidade que seria realizar o projeto anterior.

### **1.2 Objetivo**

Nosso objetivo com esse projeto é trazer entretenimento, diversão e aprendizado com o uso da automação e a tecnologia que a mesma nos proporciona. Incentivar os alunos ou técnicos a criarem seus próprios projetos, alavancando a popularidade dos robôs sumôs.

## 2. Robô Sumô

Robô sumo é um projeto desenvolvido com Arduino feito especialmente para batalhas. As competições com esses tipos de robôs estão cada vez mais famosas entre makers e estudantes também, principalmente pelo aprendizado proposto pelo desafio a cada montagem de robô.

Além de trazer um aperfeiçoamento no conhecimento sobre programação, esse robô estimula o trabalho em equipe pela necessidade de desenvolvimento de um projeto que possua competência técnica para vencer suas lutas.

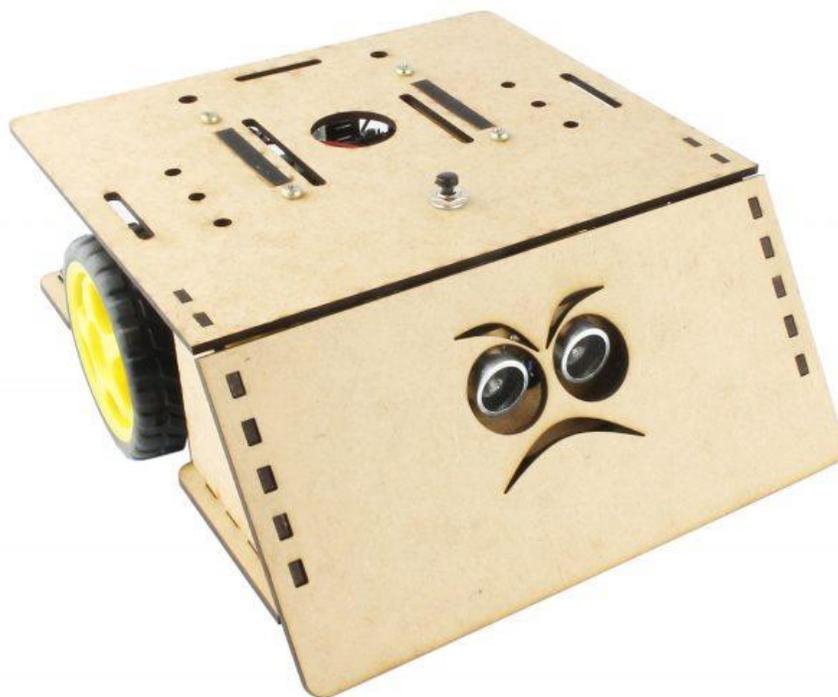


Fig 1: Robô sumo Arduino para batalhas

## 2.1. Como funciona um robô sumô?

Aqui está uma visão geral de como funciona um robô sumô:

**Estrutura do robô:** O robô sumô é construído com materiais leves e duráveis, geralmente plástico ou metal, para resistir a impactos e choques. Ele é projetado para ser ágil e capaz de se mover rapidamente no ringue.

**Sensores e eletrônicos:** O robô é equipado com uma variedade de sensores, como sensores infravermelhos, ultrassônicos ou de detecção de bordas, que ajudam a detectar a presença e a localização do oponente, bem como os limites do ringue.

**Atuadores e motores:** O robô é impulsionado por motores elétricos ou servos que controlam suas rodas ou esteiras. Estes motores são controlados por um microcontrolador ou um computador embarcado que processa os dados dos sensores e determina as ações do robô.

**Estratégia e programação:** Os robôs sumô são programados com estratégias de combate que determinam como eles devem se mover e reagir às ações do oponente. Isso pode envolver algoritmos de controle simples, como manter uma distância segura do oponente, ou técnicas mais avançadas, como reconhecimento de padrões e aprendizado de máquina.

Em resumo, um robô sumô é um dispositivo eletrônico complexo que combina hardware, software e estratégia para competir em um esporte de combate único e emocionante.

## **2.2. Qual a origem da competição de robô sumô?**

A competição de robô sumô é baseada nas famosas lutas de sumo japonesas, que tem como objetivo colocar dois oponentes frente a frente onde aquele que tirar o seu adversário primeiro da arena ou tocar o solo com qualquer parte do corpo do adversário, vence.

A versão dessa luta com robôs foi incorporada no ano de 2000 no Japão, onde admiradores do esporte colocavam pequenas máquinas controladas por rádio controles em uma batalha similar, com algumas regras atualizadas para essa nova modalidade. Normalmente esses tipos de robô são divididos em diferentes categorias de acordo com o peso que possuem.

## **2.3. Como funciona a competição de robô sumô?**

A competição de robô sumô funciona com controle manual ou autônomo, no primeiro caso (controle manual) os competidores controlam suas máquinas através de um joystick ou qualquer outro dispositivo, por exemplo um celular, controlando todos os movimentos do robô, no segundo caso (controle autônomo), esse controle é feito por uma programação prévia onde a máquina irá prever todas as etapas essenciais da luta, fazendo com que ele faça todos os movimentos e ataques por conta própria.

Na categoria autônomo (sem controle remoto) o microcontrolador o Arduino se faz presente, atuando como um cérebro eletrônico capaz de verificar todas as condições atuais do robô, através de sensores e módulos, fazendo ele tomar as decisões propostas através de uma certa programação.

Para iniciar a batalha dois robôs são colocados frente a frente numa espécie de arena, que nada mais é do que o local onde os robôs ficarão durante a disputa, geralmente sendo uma chapa de madeira de cor branca com a borda de cor preta da competição, dependendo da competição pode ser até uma madeira pintada de preto com as bordas brancas.

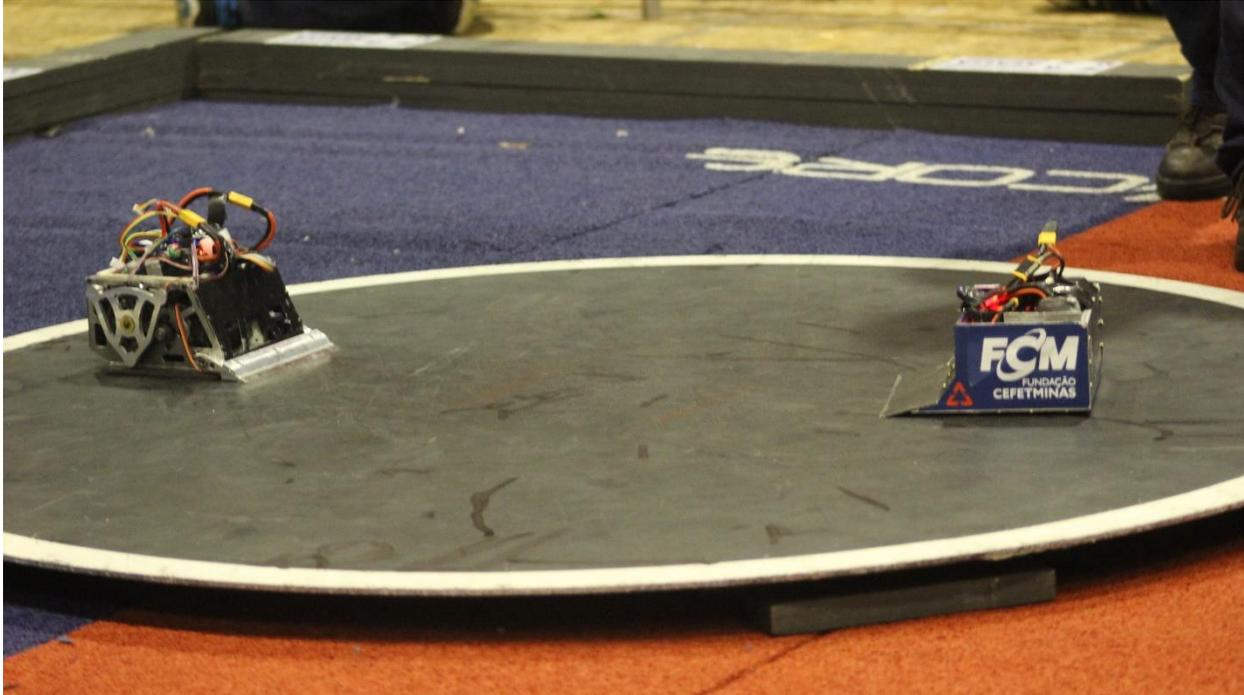


Fig 2 : competição robô sumo

O objetivo dessa competição é que os robôs (oponentes) se enfrentem por um tempo determinado até algum deles for expulso para fora do ringue pelo oponente. Caso o tempo da batalha acabe e nenhum dos robôs forem expulsos da arena, a decisão caberá os jurados que iram decidir o vencedor a partir das pontuações geradas pelos golpes durante a rodada. Um detalhe muito importante é que se o robô parar de funcionar ou não funcionar durante a batalha será aberta uma contagem de tempo para que o robô tenha uma chance de recuperar os seus movimentos, caso contrário, o mesmo será derrotado.

Esse tipo de luta contém três rounds, aquele que vencer dois deles, será considerado o campeão. Com um sistema de chaves, os oponentes vão se enfrentando até que restem apenas dois que irão se enfrentar na final, definindo assim o verdadeiro campeão.

### 3. Processo de montagem

#### 3.1. Carrinho com Trilhos:

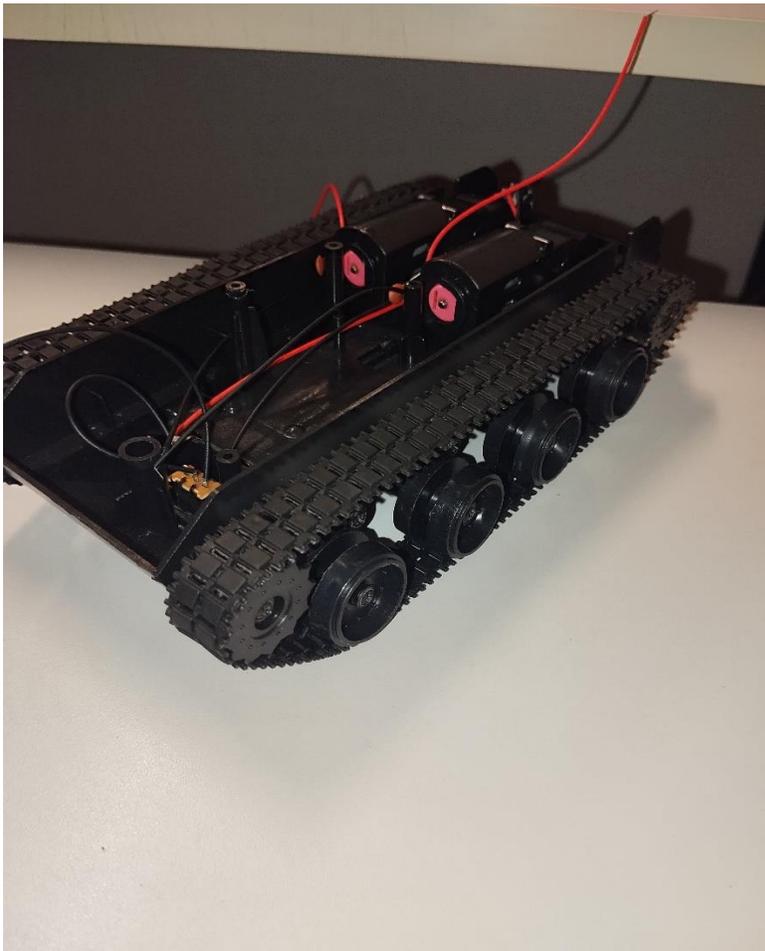


Fig 1- Base do carrinho com esteira com dois motores auxiliares



Fig 2- Parte superior do carrinho com esteira



Fig 3- Corpo do carrinho com esteira



Fig 4- Defesa/Ataque, parte frontal do carrinho com esteira

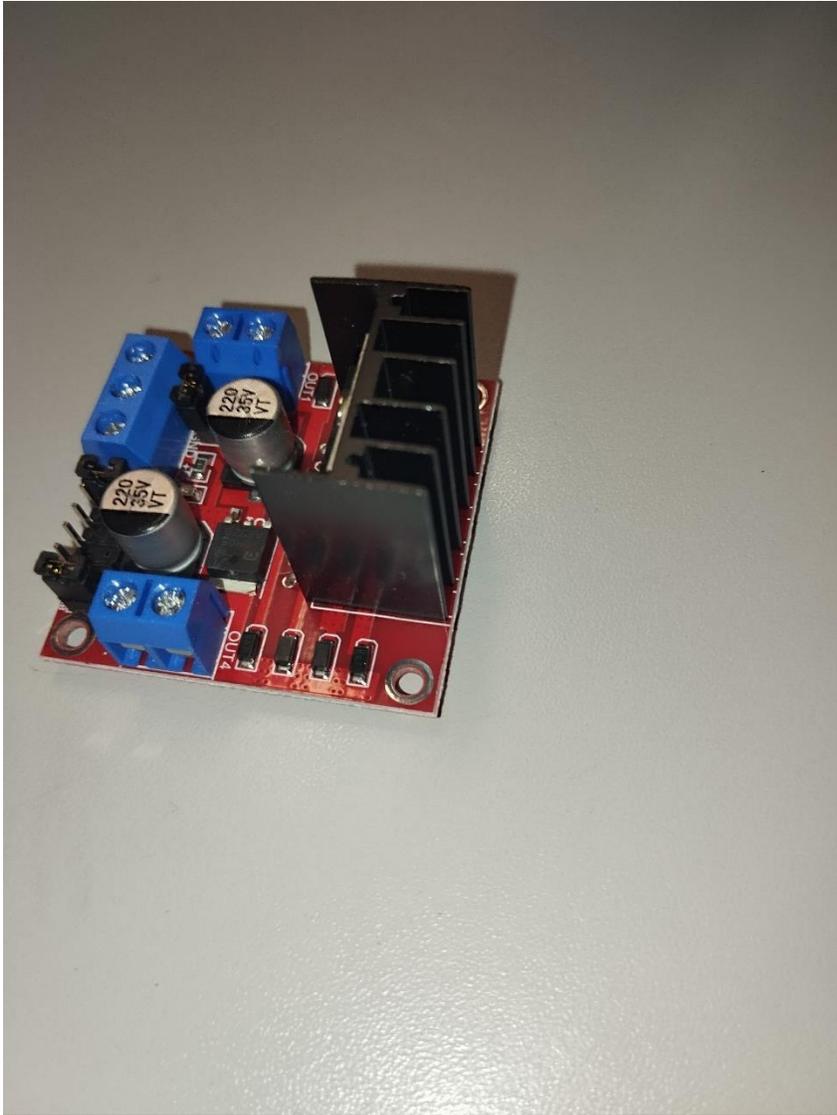


Fig 5- Driver motor ponte H dupla L298n, controladora driver L298n

### 3.2. Carrinho com Rodas:

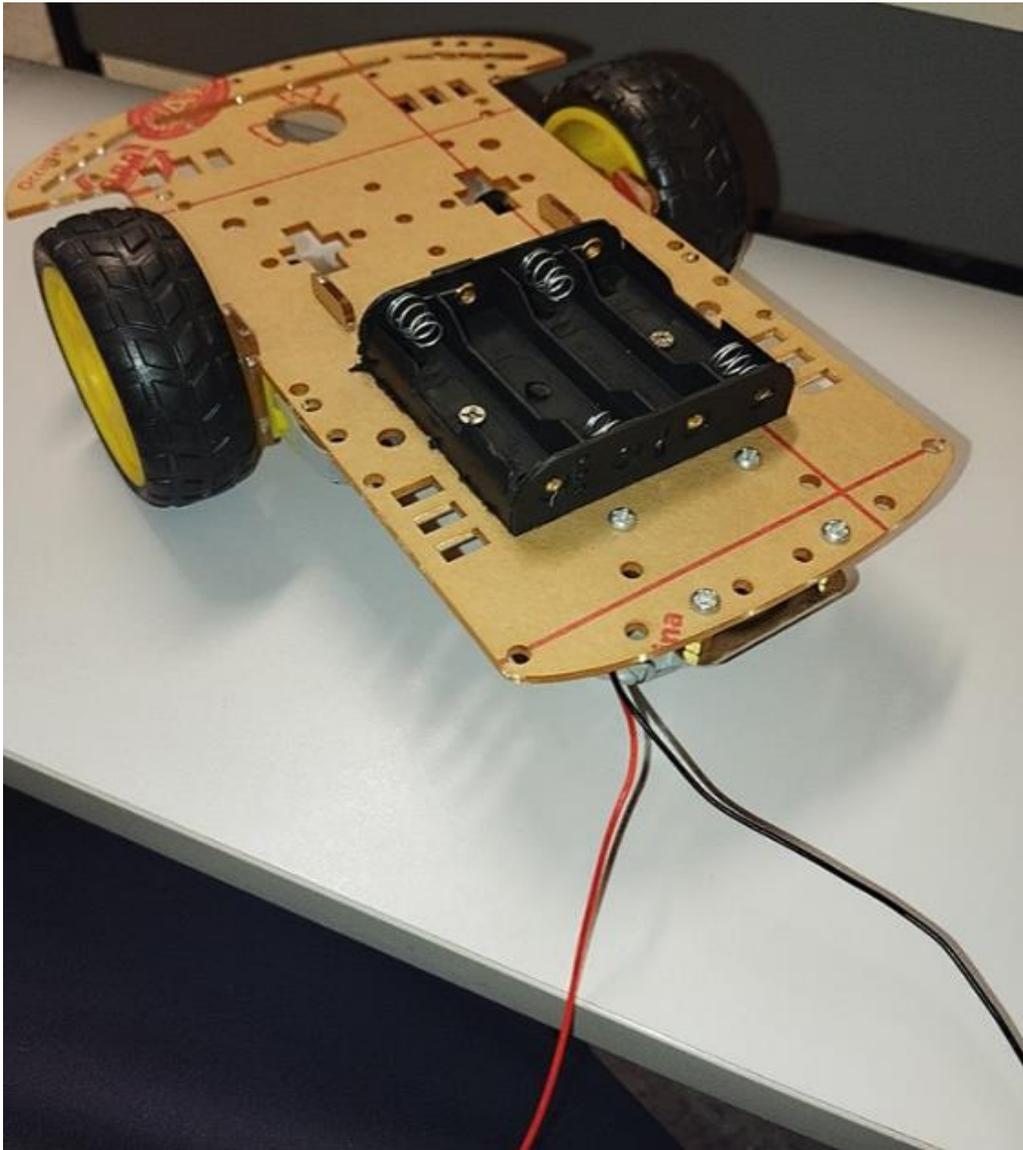


Fig 6- Base do carrinho com roda com dois motores auxiliares



Fig 7- BlackBoard Uno R3 (Arduino)

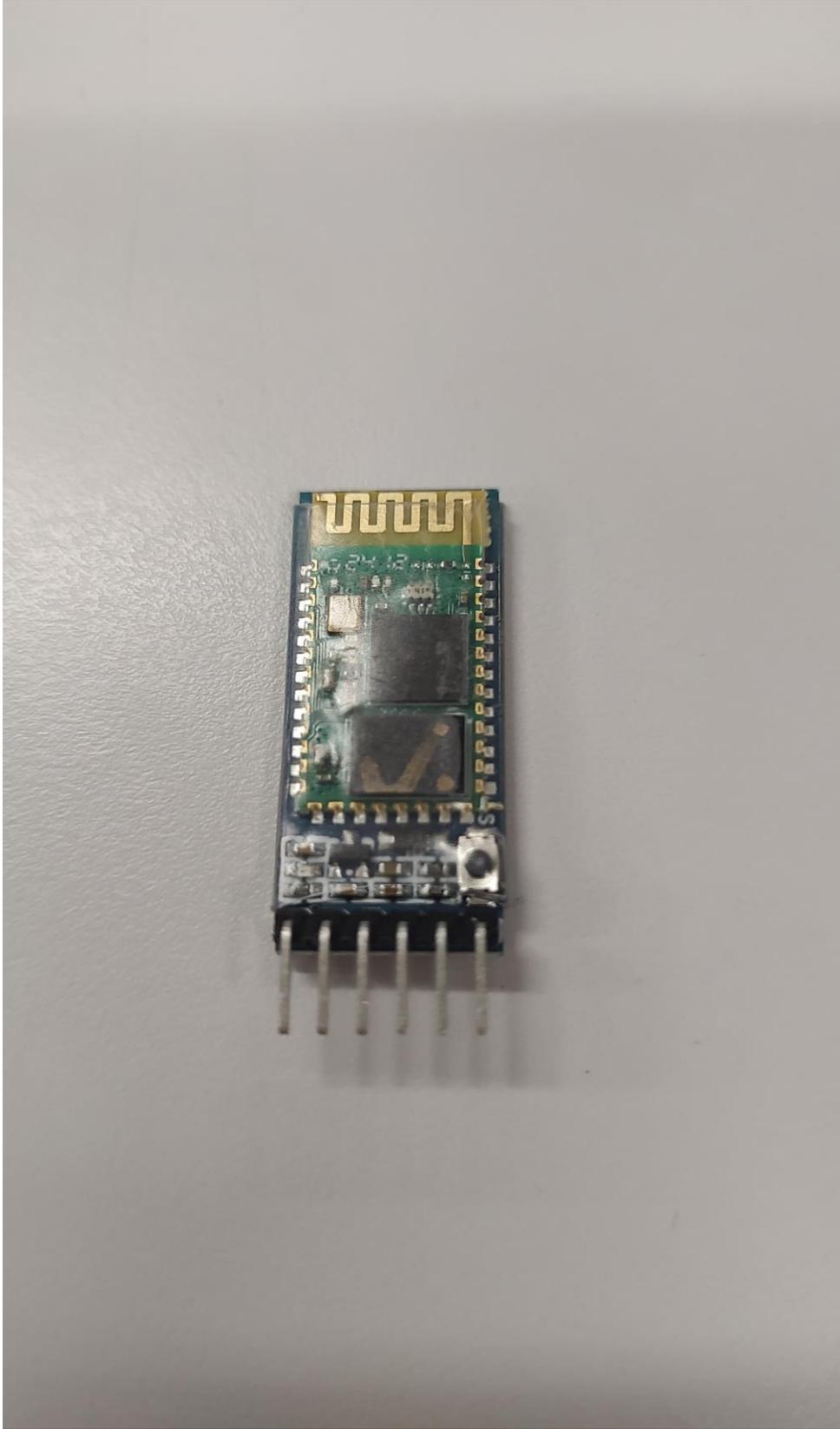


Fig 8 - Módulo bluetooth HC-05

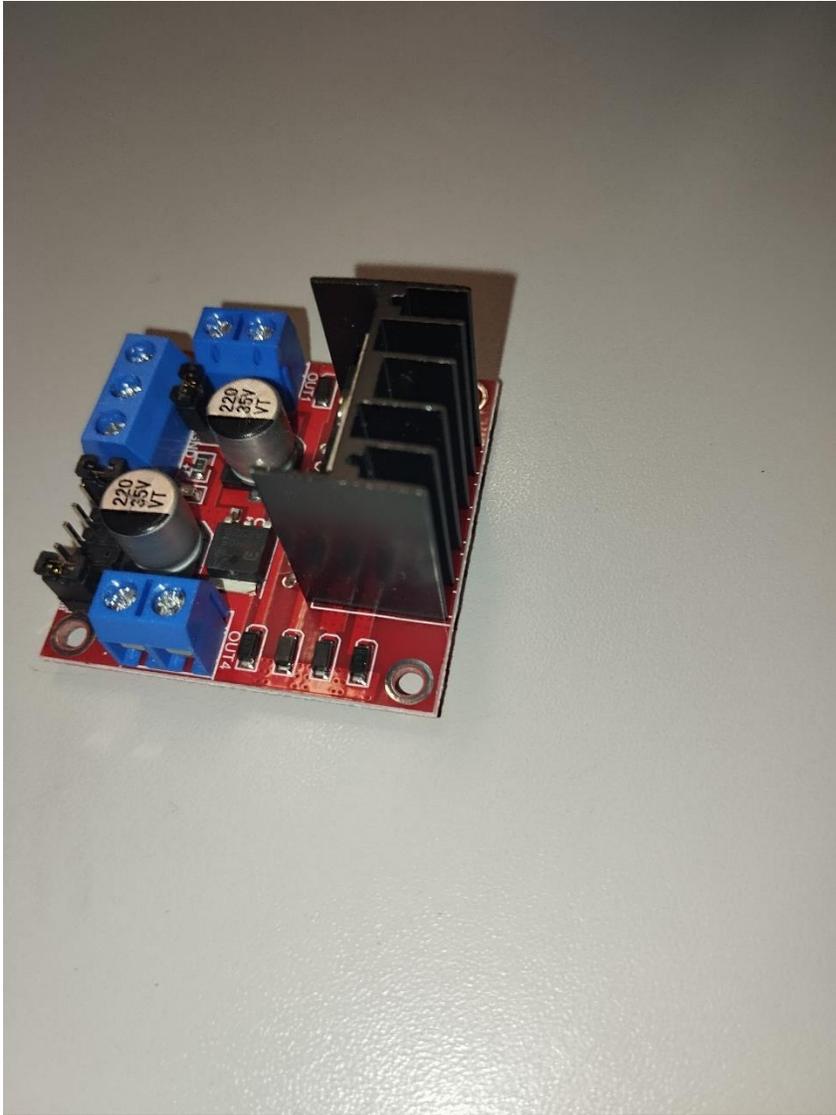


Fig 9 - Driver motor ponte H dupla L298n, controladora driver L298n

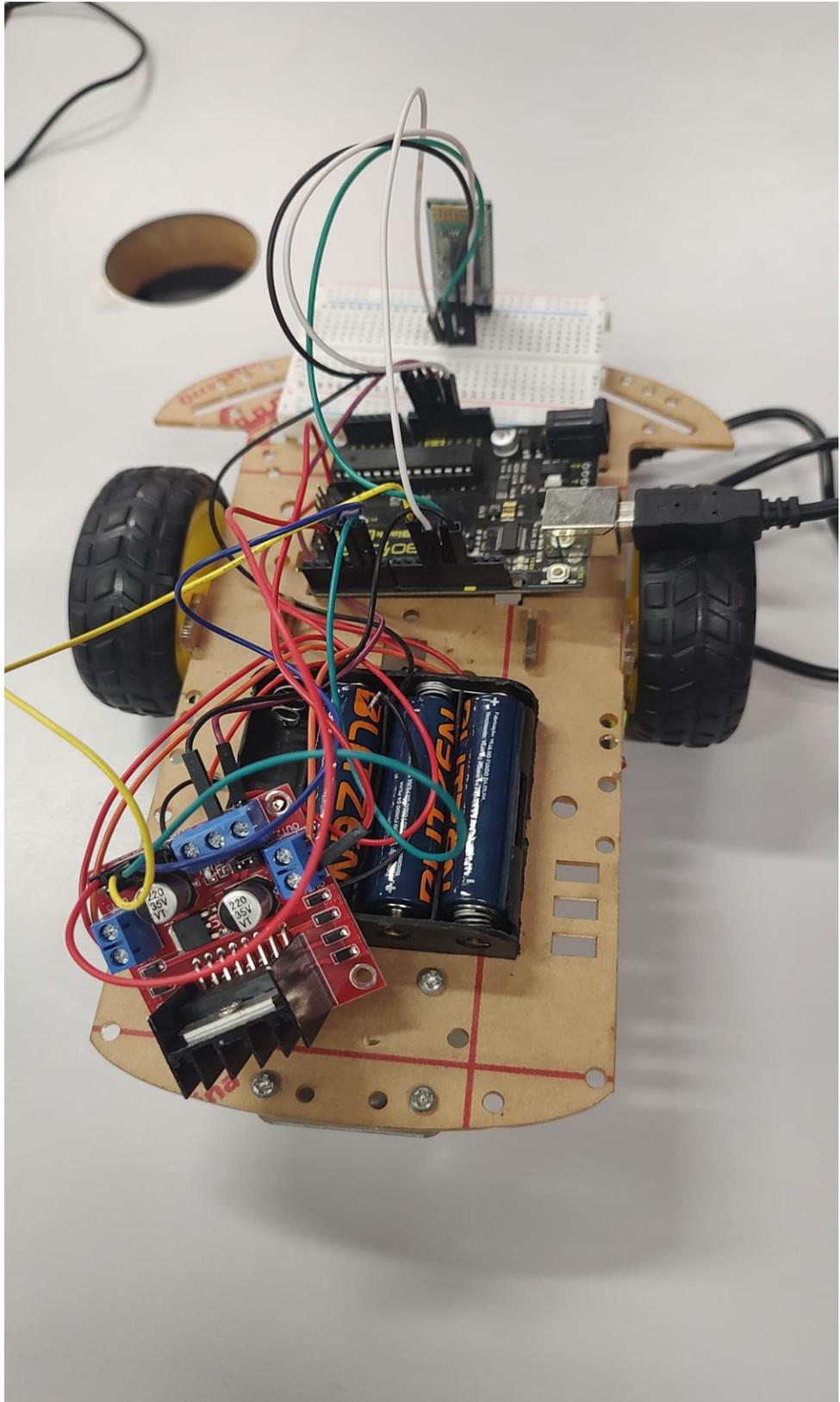


Fig 10: Protótipo de Teste Carrinho com Rodas

#### **4. Materiais e preços utilizados no TCC:**

Segue a lista dos componentes utilizados nesse projeto:

1. ARDUINO UNO R3 – 1, Preço: R\$94,90
2. PROTOBOARD – 2, Preço: R\$14,00 (unidade)
3. Fios Macho/Macho, Preço: R\$14,00 (40 peças)
4. Fios Macho/Fêmea, Preço: 15,00 (40 Peças)
5. MÓDULO HC-05 – 2, Preço: R\$42,90 (Cada)
6. CONTROLADORA DRIVER L298n – 2, Preço: R\$21,85 (unidade)
7. Kit Chassi 2 Rodas com Base de Madeira – 1, Preço: R\$45,60
8. Chassi Robótico Robô Tipo Tanque Esteira Lagarta – 1, Preço: R\$144,60
9. IMPRESSORA 3D;
10. PILHAS – 4, Preço: R\$19,00 (as quatro)

## 5. Linha de Programação dos Carrinhos

### 5.1. Linha de Programação do Carrinho com Rodas

```
#include <SoftwareSerial.h>
SoftwareSerial bluetooth(13, 12); //13 = TX 12 = RX
```

```
#define p1m1 6
#define p2m1 5
#define pvelm1 2
#define p1m2 3
#define p2m2 11
#define pvelm2 8
```

```
int vel;
```

```
class Motor {
public:
  int p1,p2,pv;
  Motor(int p1, int p2, int v){
    this->p1=p1;
    this->p2=p2;
    this->pv=v;
    pinMode(p1, OUTPUT);
    pinMode(p2, OUTPUT);
    pinMode(pv, OUTPUT);
    digitalWrite(p1,LOW);
    digitalWrite(p2,LOW);
    analogWrite(pv,0);
  }
}
```

```
void frente(int v) {
```

```
digitalWrite(p1,HIGH);  
digitalWrite(p2,LOW);  
analogWrite(pv,v);  
  
}
```

```
void tras(int v) {  
  
digitalWrite(p1,LOW);  
digitalWrite(p2,HIGH);  
analogWrite(pv,v);  
  
}
```

```
void para() {  
  
digitalWrite(p1,LOW);  
digitalWrite(p2,LOW);  
analogWrite(pv,0);  
  
}
```

```
void freia() {  
  
digitalWrite(p1,HIGH);  
digitalWrite(p2,HIGH);  
analogWrite(pv,100);  
  
}  
};
```

```
Motor *md=new Motor(p1m1,p2m1,pvelm1);  
Motor *me=new Motor(p1m2,p2m2,pvelm2);
```

```
void setup(){

Serial.begin(9600);
vel=100;
bluetooth.begin(9600);

}

void loop() {

char c = bluetooth.read();

if(c== 'L'){
  md->frente(vel);
  me->frente(vel);
  Serial.println("Frente");
}

if(c== 'R'){
  md->tras(vel);
  me->tras(vel);
  Serial.println("re");
}

if(c== 'F'){
  md->tras(vel);
  me->frente(vel);
  Serial.println("esquerda");
}

if(c== 'B'){
  md->frente(vel);
  me->tras(vel);
```

```
Serial.println("direita");  
}
```

```
if(c== 'S'){  
  md->para();  
  me->para();  
  Serial.println("parado");  
}  
}
```

## 5.2. Linha de Programação do Carrinho com Esteira

```
#include <SoftwareSerial.h>
SoftwareSerial bluetooth(13, 12); //13 = TX 12 = RX
```

```
#define p1m1 6
#define p2m1 5
#define pvelm1 2
#define p1m2 3
#define p2m2 11
#define pvelm2 8
```

```
int vel;
```

```
class Motor {
public:
  int p1,p2,pv;
  Motor(int p1, int p2, int v){
    this->p1=p1;
    this->p2=p2;
    this->pv=v;
    pinMode(p1, OUTPUT);
    pinMode(p2, OUTPUT);
    pinMode(pv, OUTPUT);
    digitalWrite(p1,LOW);
    digitalWrite(p2,LOW);
    analogWrite(pv,0);
  }
}
```

```
void frente(int v) {
```

```
  digitalWrite(p1,HIGH);
  digitalWrite(p2,LOW);
```

```

    analogWrite(pv,v);

}

void tras(int v) {

    digitalWrite(p1,LOW);
    digitalWrite(p2,HIGH);
    analogWrite(pv,v);

}

void para() {

    digitalWrite(p1,LOW);
    digitalWrite(p2,LOW);
    analogWrite(pv,0);

}

void freia() {

    digitalWrite(p1,HIGH);
    digitalWrite(p2,HIGH);
    analogWrite(pv,100);

}
};

Motor *md=new Motor(p1m1,p2m1,pvelm1);
Motor *me=new Motor(p1m2,p2m2,pvelm2);

void setup(){

```

```
Serial.begin(9600);
vel=100;
bluetooth.begin(9600);

}

void loop() {

char c = bluetooth.read();

if(c== 'L'){
  md->frente(vel);
  me->frente(vel);
  Serial.println("Frente");
}

if(c== 'R'){
  md->tras(vel);
  me->tras(vel);
  Serial.println("re");
}

if(c== 'F'){
  md->tras(vel);
  me->frente(vel);
  Serial.println("esquerda");
}

if(c== 'B'){
  md->frente(vel);
  me->tras(vel);
  Serial.println("direita");
}
```

```
if(c== 'S'){  
    md->para();  
    me->para();  
    Serial.println("parado");  
}  
}
```

## 6. Projeto Concluído

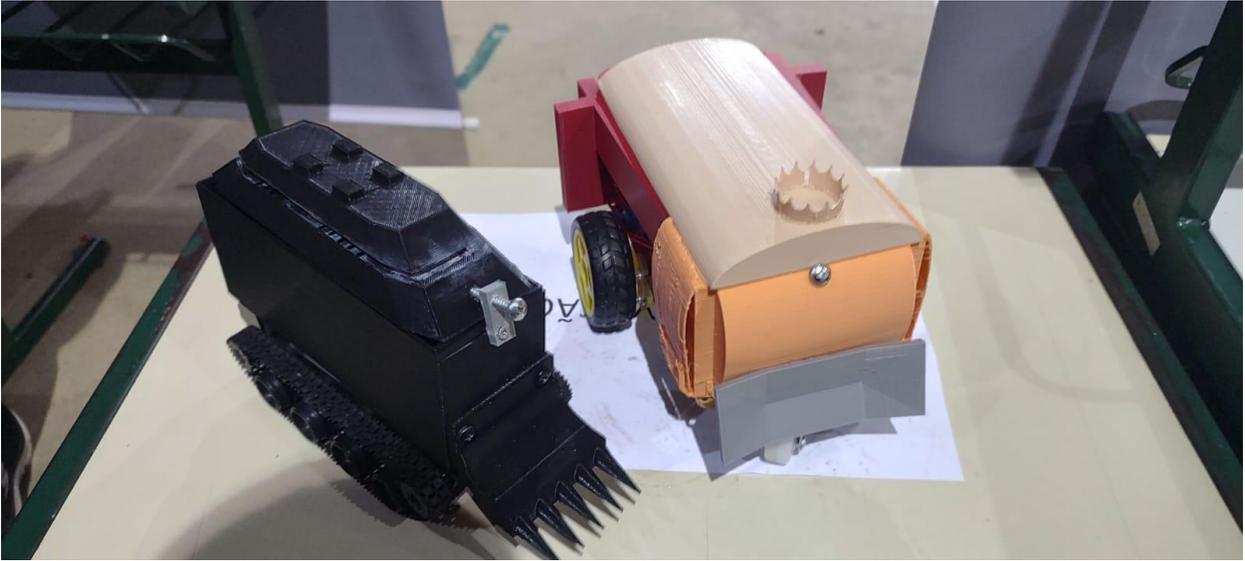


Fig 1. Trabalho finalizado.