

**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA
SOUZA**

ETEC SYLVIO DE MATTOS CARVALHO

Curso Técnico em Mecatrônica M-tec

Geane Aparecida Correia dos Santos

Ana Clara Camilotti

Ana Heloisa do Santos

Kauê Vinicius Galo Leão

ROBÔ-SUMÔ GAJAK

Matão, SP

2024

SUMÁRIO

1	INTRODUÇÃO	3
2	OBJETIVOS (GERAL E ESPECÍFICOS)	4
2.1	Objetivo Geral	4
2.2	Objetivos Específicos	4
3	DESCRIÇÃO DO PROJETO	5
3.1	Apresentação Geral do Projeto	5
3.2	Materiais Utilizados e suas Especificações Técnicas	5
4	CRONOGRAMA DE ATIVIDADES	9
4.1	Atividades e Metas 1º Semestre de 2024	9
4.2	Atividades e Metas 2º Semestre de 2024	10
5	DESENVOLVIMENTO DO PROJETO	11
5.1	Descrição do Funcionamento	11
5.2	Montagem Física do Projeto	12
5.3	Desenvolvimento do programa do robô	14
6	CONSIDERAÇÕES FINAIS	17
	REFERÊNCIAS BIBLIOGRÁFICAS	18
	APÊNDICE	19

1 INTRODUÇÃO

Este documento apresenta o Projeto Técnico Robô sumo GAJAK referente ao projeto de Trabalho de Conclusão de Curso elaborado pela equipe.

Esse projeto foi escolhido por proporcionar aos estudantes uma oportunidade de desenvolver habilidades e competências relativas ao curso de Mecatrônica, contribuindo para sua formação e preparando-os para novas jornadas.

Isto posto, este documento encontra-se assim dividido:

No Capítulo 2, apresenta os objetivos geral e específicos desse documento.

No Capítulo 3, é feita uma apresentação geral do projeto e a lista de materiais utilizadas para a sua montagem física

O Capítulo 4 apresenta os cronogramas de atividades planejados e cumpridos no 1º e no 2º semestres de 2024 para a concretização do projeto.

O Capítulo 5 faz a descrição detalhada do projeto físico, apresentando o seu funcionamento, a sua montagem física e o programa aplicativo desenvolvido.

Na sequência são apresentadas as considerações finais sobre o projeto, apresentando as dificuldades encontradas e contornadas, e o resultado final obtido.

E, finalmente nas Referências Bibliográficas, são apresentadas toda a bibliografia utilizada para o desenvolvimento desse projeto.

2 OBJETIVOS (GERAL E ESPECÍFICOS)

2.1 Objetivo Geral

O objetivo geral do projeto é:

- Construir um robô mini sumô.

2.2 Objetivos Específicos

Os objetivos específicos do projeto são:

- Definir os componentes ideais para o robô;
- Elaborar o projeto estrutural do robô;
- Montar o robô, efetuando as conexões dos seus componentes;
- Desenvolver a programação do robô.

3 DESCRIÇÃO DO PROJETO

3.1 Apresentação Geral do Projeto

O robô tem o formato de um veículo, contendo duas rodas de borracha em uma das extremidades do chassi acionadas cada uma através de um motor CC, e uma roda fixa no centro da outra extremidade.

Sobre o chassi foi instalado uma placa de Arduino modelo Nano, o qual possui o microcontrolador Atmega 328P programado para que o robô se movimente através de comandos via celular, que está conectado ao Arduino através de um módulo bluetooth.

O robô é alimentado por uma bateria recarregável de 9V, instalado também no chassi.

A Figura 1 apresenta uma foto do nosso projeto final montado.

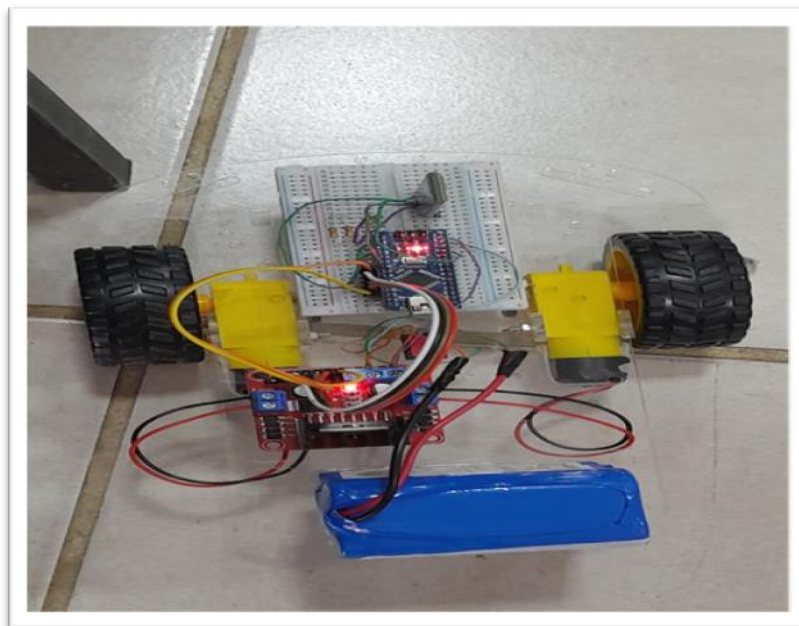


Figura 1 – Foto do projeto montado
FONTE: Elaborado pelos autores (2024).

3.2 Materiais Utilizados e suas Especificações Técnicas

Os materiais utilizados em nosso projeto estão apresentados na Tabela 1.

Tabela 1: Lista de materiais utilizados no projeto

Material	Quantidade	Preço Unitário	Preço Total
Arduino Nano Rev3	1	45,00	45,00
Mini Protoboard	1	15,00	15,00
Motor Shield L298N	1	13,00	13,00
Motor DC	2	15,00	30,00
Roda de Borracha	2	9,00	18,00
Roda Boba	1	28,77	28,77
Chassi	1	20,00	20,00
Modulo Bluetooth HC-06	1	48,90	48,90
Cabo USB	1	22,11	22,11
Bateria 9V Recarregável	1	38,00	38,00
Kit jumper Macho-Macho (20)	1	25,50	25,50
Kit jumper Macho-Fêmea (20)	1	28,00	28,00
Clip para bateria	1	24,00	24,00
Kit parafusos	1	60,00	60,00
Kit Abraçadeiras	1	35,00	35,00
Kit Resistor 220R (10)	1	1,14	1,14
Pushbotton	1	38,60	38,60
Kit resistor de 10Ω (10)	1	0,40	0,40
Fita dupla fácil	1	33,00	33,00
Total			524,42

FONTE: Elaborado pelos autores (2024).

Os materiais listados na Tabela 1 foram custeados pelo Professor Rogério Varavallo.

A seguir temos uma breve descrição dos principais materiais utilizados.

Arduino Nano

O Arduino Nano (figura 2) é uma pequena placa eletrônica que funciona como o cérebro de projetos de tecnologia. Ele é fácil de programar e pode controlar uma variedade de dispositivos, como motores e luzes. Você escreve códigos em uma linguagem especial para dizer ao Arduino o que fazer. É muito usado para criar robôs, sistemas automatizados e projetos interativos. É uma ferramenta popular para quem quer explorar a eletrônica e a programação.



Figura 2: Arduino Nano
FONTE: Mercado Livre

Módulo Shield L298N

O Módulo Shield L298N é um módulo eletrônico usado para controlar motores em projetos de robótica e automação. Ele pode controlar até dois motores de corrente contínua (DC) ou um motor de passo. O shield possui circuitos integrados que permitem o controle da direção e da velocidade dos motores. Além disso, ele pode fornecer a corrente necessária para o funcionamento dos motores, geralmente de até 2 amperes por canal. É comumente utilizado com placas como o Arduino para projetos que exigem movimento controlado.

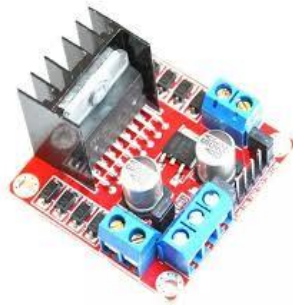


Figura 3: Módulo L298N
FONTE: Mercado Livre

Motor DC

Um motor DC (corrente contínua) é um tipo de motor elétrico que funciona com energia de corrente contínua. Ele converte energia elétrica em movimento rotacional. O controle da velocidade e da direção do motor é feito ajustando a voltagem e a polaridade da corrente elétrica que o alimenta. Esses motores são amplamente usados em diversas aplicações, como brinquedos, ventiladores e veículos elétricos, devido à sua simplicidade e eficiência.



Figura 4: Motor DC
FONTE: Mercado Livre

Módulo Bluetooth HC-06

O módulo Bluetooth HC-06 é um dispositivo que permite a comunicação sem fio entre um microcontrolador, como o Arduino, e outros dispositivos Bluetooth, como smartphones ou computadores. Ele é usado para enviar e receber dados via Bluetooth, facilitando a conexão e controle remoto de projetos eletrônicos. O HC-06 é

fácil de configurar e não requer muitos ajustes, sendo ideal para quem quer adicionar funcionalidades de comunicação Bluetooth a seus projetos.



Figura 5: Módulo Bluetooth HC-06
FONTE: Mercado Livre

4 CRONOGRAMA DE ATIVIDADES

4.1 Atividades e Metas 1º Semestre de 2024

Na Tabela 2 estão listadas as atividades e metas concluídas no 1º semestre.

Tabela 2: Cronograma de atividades e metas no 1º Semestre de 2024

Atividade	Fev.		Março		Abril		Maio		Junho	
	1ª Qu inz en a	2ª Qu inz en a	1ª Qu inz en a	2ª Qu inz en a	1ª Qu inz en a	2ª Qu inz en a	1ª Qu inz en a	2ª Qu inz en a	1ª Qu inz en a	2ª Qu inz en a
Montagem da equipe	x									
Definição do nome do robô		x								
Definição dos objetivos do projeto		x								
Definição da lista de materiais / orçamentos			x							
Construção do chassi				x						
Redação do relatório até o capítulo 3				x	x					
Redação do capítulo 3					x	x	x	x		
Montagem do robô					x	x	x	x		
Programação do robô							x	x		
Batalha de Robôs									x	x

FONTE: Elaborado pelos autores (2024).

4.2 Atividades e Metas 2º Semestre de 2024

Na Tabela 3 estão registradas as atividades e metas que estão sendo concluídas no 2º Semestre.

Tabela 3: Cronograma de atividades e metas no 2º Semestre de 2024

Atividade	Julho		Agosto		Setembro		Outubro		Novembro	
	1ª Qui n z e n a	2ª Qu inz e n a	1ª Qu inz e n a	2ª Qu inz e n a	1ª Qui n ze n a	2ª Qu inz e n a	1ª Qui n ze n a	2ª Qui n ze n a	1ª Qui n ze n a	2ª Qui n ze n a
Escolha do chassi	x	x	x							
Desenvolvimento circuito elétrico			x	x						
Análise de Necessidades		x	x	x	x	x	x	x		
Construção do Chassi			x	x						
Testes Funcionais					x	x	x	x		
Ajustes e Correções					x	x	x	x		
Revisão do Projeto						x	x	x	x	
Ajustes Finais								x	x	
Preparação da Apresentação								x	x	
Entrega do projeto										x
Entrega do Boneco projeto técnico										x
Correções do projeto										x

FONTE: Elaborado pelos autores (2024).

5 DESENVOLVIMENTO DO PROJETO

5.1 Descrição do Funcionamento

A Figura 6 apresenta o esquema elétrico de toda a parte eletrônica do projeto.

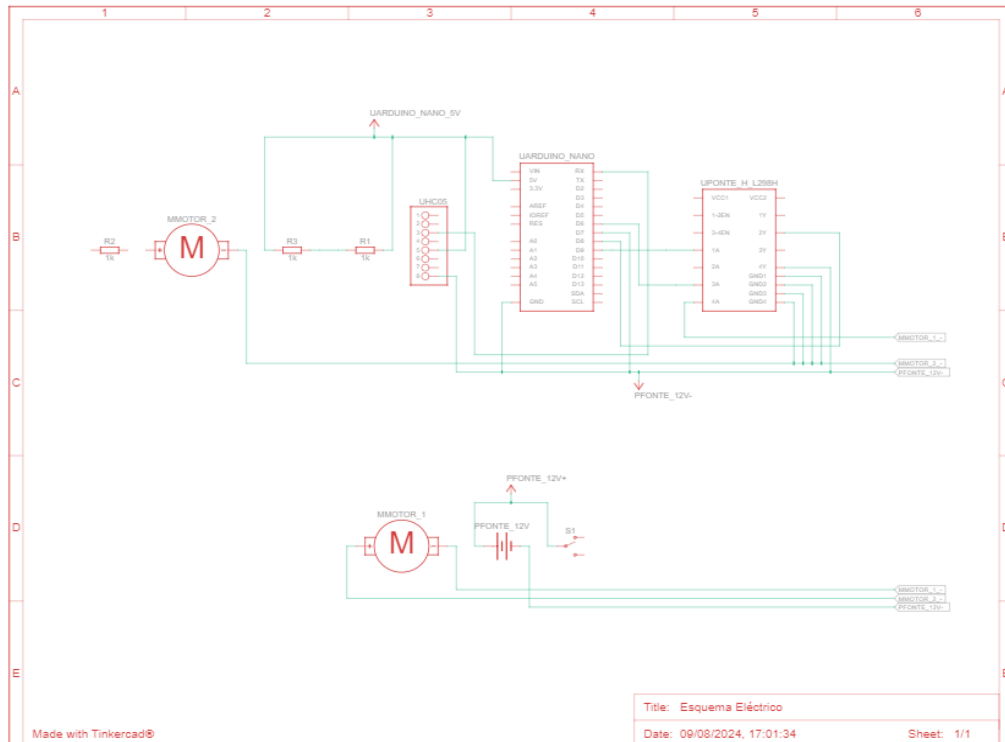


Figura 6 – Esquema elétrico do robô
FONTE: Elaborado pelos autores (2024).

No esquema elétrico temos o Arduino Nano ligado ao módulo 298N, que é uma ponte H dupla para acionamento dos dois motores do robô (conforme apresentado na página 6), e também a um módulo bluetooth que está conectado a um celular via aplicativo.

O funcionamento do robô é bastante simples: através de comandos via celular, o Arduino Nano aciona os motores via ponte H para movimentá-los conforme o comando recebido. Temos 8 tipos de comandos programados:

- Comando “I”: movimenta os motores para que o robô se movimente para a frente;
- comando for “G” aciona os motores para o robô ir para frente e para direita.
- Quando o comando for “B” aciona os motores para o robô ir para trás.
- Quando o comando for “H” aciona os motores para o robô ir para trás e esquerda.

- Quando o comando for “J” aciona os motores para o robô ir para trás e direita,
- Quando o comando for “L” aciona os motores para o robô ir para esquerda,
- Quando o comando for “R” aciona os motores para o robô ir para direita,
- Quando o comando for “S” desligará os motores, após isso os comandos estarão finalizados.

5.2 Montagem Física do Projeto

As figuras 7 a 10 apresentam a sequência da montagem física do nosso projeto.



Figura 7 – Peças utilizadas no robô
FONTE: Elaborado pelos autores (2024).

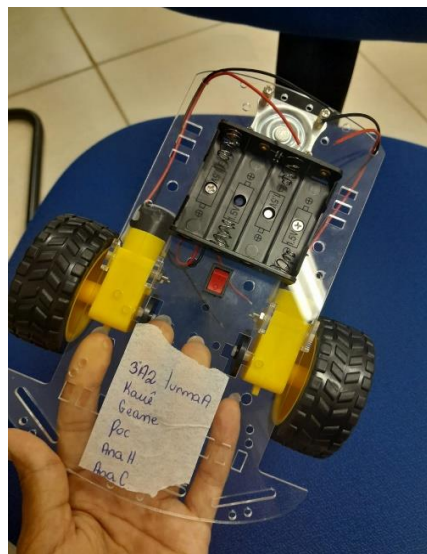


Figura 8 – Instalação das rodas e motores no chassi
FONTE: Elaborado pelos autores (2024).



Figura 9 – Instalação do Arduino e módulos no chassi
 FONTE: Elaborado pelos autores (2024).



Figura 10 – Instalação da bateria no chassi
 FONTE: Elaborado pelos autores (2024).

As Figuras 11 e 12 apresentam os testes finais efetuados em nosso projeto.

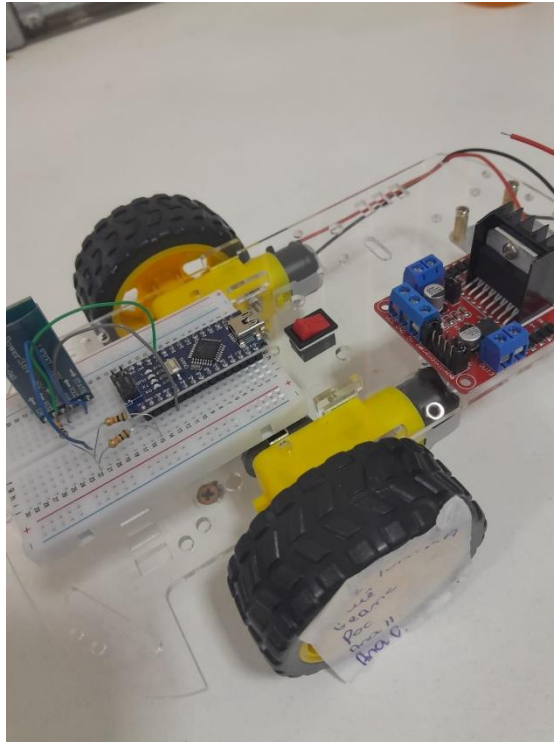


Figura 11 – Teste do hardware
FONTE: Elaborado pelos autores (2024).

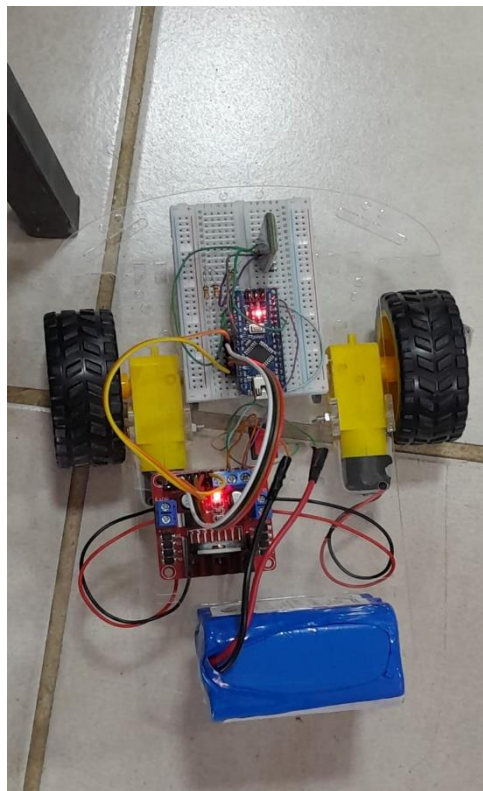
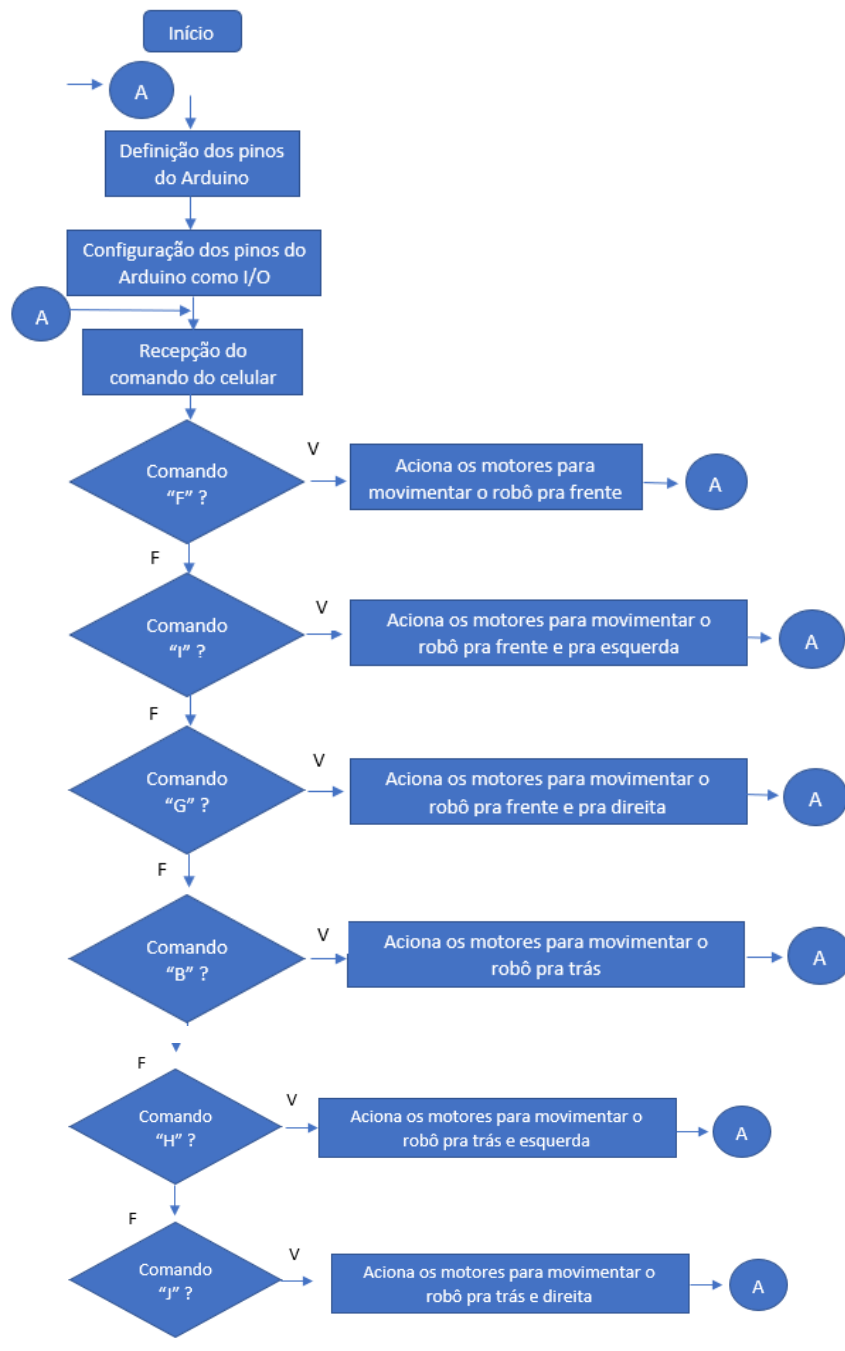


Figura 12– Teste geral do robô
FONTE: Elaborado pelos autores (2024).

5.3 Desenvolvimento do programa do robô

A programação do robô foi desenvolvida no aplicativo Arduino IDE versão 1.8.1

A Figura 13 apresenta o fluxograma do programa



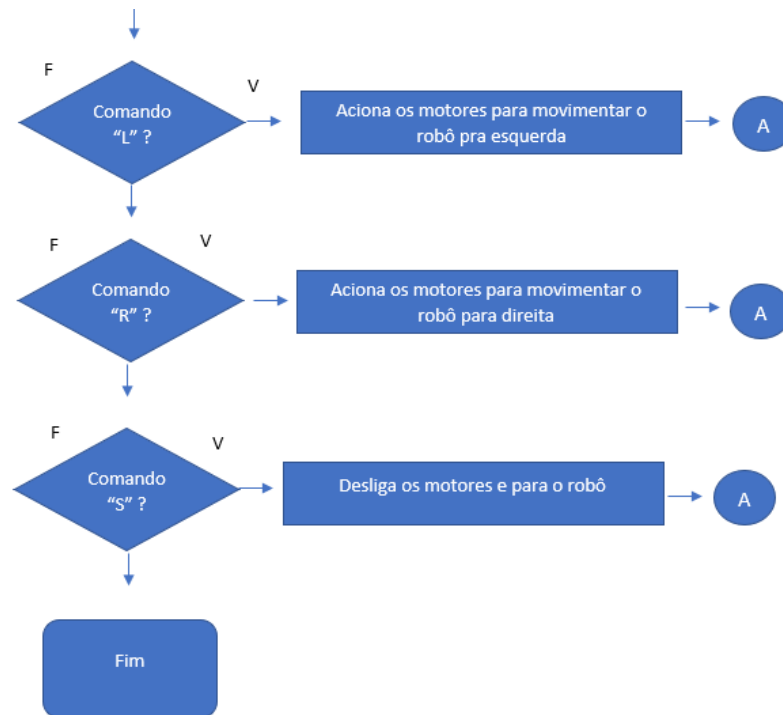


Figura 13 – Fluxograma do programa do robô
 FONTE: Elaborado pelos autores (2024).

Conforme apresentado no fluxograma, o funcionamento dele começa após fazer o upload do programa no Arduino, fazer as definições dos pinos do Arduino, depois fazer as configurações dos pinos no Arduino, fazer a recepção do comando no celular.

Quando o comando for “F” aciona os motores para o robô ir pra frente.

Quando o comando for “L” aciona os motores para o robô ir para frente e para esquerda.

Quando o comando for “R” aciona os motores para o robô ir para frente e para direita.

Quando o comando for “B” aciona os motores para o robô ir para trás.

Quando o comando for “H” aciona os motores para o robô ir para trás e esquerda.

Quando o comando for “J” aciona os motores para o robô ir para trás e direita.

Quando o comando for “L” aciona os motores para o robô ir para esquerda.

Quando o comando for “R” aciona os motores para o robô ir para direita,

Quando o comando for “S” desligará os motores, após isso os comandos estarão finalizados.

O código fonte desenvolvido no Arduino IDE é apresentado no Apêndice A.

6 CONSIDERAÇÕES FINAIS

O desenvolvimento deste projeto teve como objetivo a criação e implantação de uma solução inovadora para descrever o tema do projeto, e os resultados obtidos superaram nossas expectativas iniciais. Através de uma análise aprofundada, testes e refinamentos contínuos, conseguimos alcançar uma implantação com um nível de precisão e eficiência que atendeu plenamente aos objetivos propostos. Através de uma abordagem metódica, conseguimos criar uma solução eficiente, que atendeu aos objetivos propostos, com alta precisão e desempenho. A implantação trouxe impactos positivos, como melhorias ou resultados alcançados, comprovando a eficiência do trabalho. O sucesso do projeto é evidenciado pela qualidade dos resultados refletido o empenho e a dedicação ao longo de todas as fases de desenvolvimento.

REFERÊNCIAS BIBLIOGRÁFICAS

USINAINFO.*INSPIRAÇÕES DA ESTRUTURA DO ROBÔ SUMÔ*.2008.Acesso:19 ago.2024.

FERREIRA, A.; *COMO MONTAR E PROGRAMAR UM ROBÔ LUTADOR DE SUMÔ*.2022.Acesso em:19 ago.2024.

CARDOSO.; MARQUES B.; *REPERTÓRIO DO ROBÔ*.2021.Acesso:19 ago.2024.

APÊNDICE

// CARRO CONTROLADO POR BLUETOOTH

```
// === Pinos utilizados no driver L298 === //
#define IN1 9 //o pino IN1 do Driver será ligado ao pino 9 do arduino
#define IN2 8 //o pino IN2 do Driver será ligado ao pino 8 do arduino
#define IN3 7 //o pino IN3 do Driver será ligado ao pino 7 do arduino
#define IN4 6 //o pino IN4 do Driver será ligado ao pino 6 do arduino

// ===== Variáveis utilizadas ===== //
char Texto_Recebido; //variável do tipo caracter (char) - recebe os dados do bluetooth
byte velocidade = 100 //variável do tipo byte - usada no controle da velocidade

void setup() {
    // === Definição dos pinos de saída === //
    pinMode(IN1, OUTPUT); //configura o pino do IN1 como saída
    pinMode(IN2, OUTPUT); //configura o pino do IN2 como saída
    pinMode(IN3, OUTPUT); //configura o pino do IN3 como saída
    pinMode(IN4, OUTPUT); //configura o pino do IN4 como saída
    pinMode(13, OUTPUT); //configura o pino do 13 (ligado ao led L) como saída
    Serial.begin(9600); //Inicia a comunicação serial
}

void loop() {
    // === Dados recebidos na comunicação === //
    if (Serial.available() > 0) { // "SE" tiver dados disponíveis na comunicação, faça...
        Texto_Recebido = Serial.read(); //o valor da leitura serial é atribuído à variável
        Texto_Recebido
    }

    // === Movimentação do carro === //
    if (Texto_Recebido == 'F') { // "SE" a letra recebida for "F", faça...
        Serial.println("Para Frente"); //imprime o texto "Para Frente" no monitor serial
        digitalWrite(IN1, 1); //envia nível lógico alto para o pino IN1 - liga o motor 1
        digitalWrite(IN2, 0); //envia nível lógico baixo para o pino IN2 - para frente
        digitalWrite(IN3, 1); //envia nível lógico alto para o pino IN3 - liga o motor 2
        digitalWrite(IN4, 0); //envia nível lógico baixo para o pino IN4 - para frente
        digitalWrite(13, 1); //liga o led L ligado ao pino 13
    }

    else if (Texto_Recebido == 'I') { // "MAS SE" a letra recebida for "I", faça...
        Serial.println("Para Frente e Esquerda"); //imprime o texto "Para Frente e Esquerda" no
        monitor serial
        digitalWrite(IN1, 1); //envia nível lógico alto para o pino IN1 - liga o motor 1
        digitalWrite(IN2, 0); //envia nível lógico baixo para o pino IN2 - para frente
        digitalWrite(IN3, 1); //envia nível lógico alto para o pino IN3 - liga o motor 2
        digitalWrite(IN4, 0); //envia nível lógico baixo para o pino IN4 - para frente
    }

    else if (Texto_Recebido == 'G') { // "SE" a letra recebida for igual a 'G', o carro se
        movimenta para Frente Direita.
        Serial.println("Para Frente e Direita"); //imprime o texto "Para Frente e Direita" no monitor serial
        digitalWrite(IN1, 1); //envia nível lógico alto para o pino IN1 - liga o motor 1
        digitalWrite(IN2, 0); //envia nível lógico baixo para o pino IN2 - para frente
        digitalWrite(IN3, 1); //envia nível lógico alto para o pino IN3 - liga o motor 2
        digitalWrite(IN4, 0); //envia nível lógico baixo para o pino IN4 - para frente
    }
}
```

```

else if (Texto_Recebido == 'B') {      //"SE" a letra recebida for igual a 'B', o carro se
    movimenta para Trás.
    Serial.println("Para Trás"); //imprime o texto "Para Trás" no monitor serial
    digitalWrite(IN1, 0); //envia nível lógico baixo para o pino IN1 - liga o motor 1
    digitalWrite(IN2, 1); //envia nível lógico alto para o pino IN2 - para trás
    digitalWrite(IN3, 0); //envia nível lógico baixo para o pino IN3 -liga o motor 2
    digitalWrite(IN4, 1); //envia nível lógico alto para o pino IN4 - para trás
}

else if (Texto_Recebido == 'H') {      //"SE" a letra recebida for igual a 'H', o carro se
    movimenta para Trás e esquerda.
    Serial.println("Para Trás e Esquerda"); //imprime o texto "Para Trás e Esquerda" no monitor serial
    digitalWrite(IN1, 0); //envia nível lógico baixo para o pino IN1 - liga o motor 1
    digitalWrite(IN2, 1); //envia nível lógico alto para o pino IN2 - para trás
    digitalWrite(IN3, 0); //envia nível lógico baixo para o pino IN3 - liga o motor 2
    digitalWrite(IN4, 1); //envia nível lógico alto para o pino IN4 - para trás
}

else if (Texto_Recebido == 'J') {      //"SE" a letra recebida for igual a 'J', o carro se
    movimenta para Trás e direita.
    Serial.println("Para Trás e Direita"); //imprime o texto "Para Trás e Direita" no monitor serial
    digitalWrite(IN1, 0); //envia nível lógico baixo para o pino IN1 - liga o motor 1
    digitalWrite(IN2, 1); //envia nível lógico alto para o pino IN2 - para trás
    digitalWrite(IN3, 0); //envia nível lógico baixo para o pino IN3 - liga o motor 2
    digitalWrite(IN4, 1) //envia nível lógico alto para o pino IN4 - para trás
}

else if (Texto_Recebido == 'L') {      //"SE" a letra recebida for igual a 'L', o carro se
    movimenta para a esquerda.
    Serial.println("Para Esquerda"); //imprime o texto "Para Esquerda" no monitor serial
    digitalWrite(IN1, 0); //envia nível lógico baixo para o pino IN1 - liga o motor 1
    digitalWrite(IN2, 1); //envia nível lógico alto para o pino IN2 - para trás
    digitalWrite(IN3, 1); //envia nível lógico alto para o pino IN3 - liga o motor 2
    digitalWrite(IN4, 0); //envia nível lógico baixo para o pino IN4 - para frente
}

else if (Texto_Recebido == 'R') {      //"SE" a letra recebida for igual a 'R', o carro se
    movimenta para a direita.
    Serial.println("Para Direita"); //imprime o texto "Para Direita" no monitor serial
    digitalWrite(IN1, 1); //envia nível lógico alto para o pino IN1 - liga o motor 1
    digitalWrite(IN2, 0); //envia nível lógico baixo para o pino IN2 - para frente
    digitalWrite(IN3, 0); //envia nível lógico baixo para o pino IN3 - liga o motor 2
    digitalWrite(IN4, 1); //envia nível lógico alto para o pino IN4 - para trás
}

else if (Texto_Recebido == 'S') {      //"SE" a letra recebida for igual a 'S', o carro deve parar.
    Serial.println("Parado"); //imprime o texto "Parado" no monitor serial
    digitalWrite(IN1, 0); //envia nível lógico baixo para o pino IN1 - desliga
    digitalWrite(IN2, 0); //envia nível lógico baixo para o pino IN2 - o motor 1
    digitalWrite(IN3, 0); //envia nível lógico baixo para o pino IN3 - desliga
    digitalWrite(IN4, 0); //envia nível lógico baixo para o pino IN4 - o motor 2
    digitalWrite(13,0); //desliga o led L ligado ao pino 13
}

```

