



SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL: SENAI
VALINHOS CURSO: DESENVOLVIMENTO DE SISTEMAS

1TDS2

DESENVOLVIMENTO TÉCNICO

MIGUEL SARTI
NICOLE CAU
PABLO DELGADO
VINICIUS PEREIRA
VINICIUS ROCHA
VITOR ARGERI

VALINHOS
2024



SUMÁRIO

1	Lista de Componentes de Hardware.....	3
2	Explicação do Sistema Operacional do Arduino.....	4
3	Lógica de Controle do Carrinho.....	5
4	Fluxograma do Código.....	6
5	Configuração da Tecnologia Sem Fio.....	7
6	Diagrama de Montagem do Carrinho.....	8
7	Código Fonte Implementado.....	9



1 LISTA DE COMPONENTES DE HARDWARE

- Arduino Uno ou microcontrolador similar
- Módulo Bluetooth (HC-05 ou HC-06)
- Driver do motor (L298N ou similar)
- 2 motores de rotação
- 2 motores DC com caixa de w48:1
- 4 rodas adicionais
- Bateria AA
- Compartimento de pilhas AA e um conjunto de pilhas alcalinas AA
- Cabo USB para conectar a placa ao IDE
- Chassi com rodas e espaço para montagem de eletrônicos



2 SISTEMA OPERACIONAL DO ARDUÍNO

O Arduino UNO não possui um sistema operacional tradicional. Ele opera sem um OS, usando firmware que você programa diretamente. Na prática, você escreve programas (chamados sketches) na linguagem C/C++ usando a Arduino IDE, uma ferramenta simples e intuitiva. Esses programas são então carregados e executados diretamente no microcontrolador ATmega328P da placa.

O processo de programação envolve duas funções principais: `setup()`, que é executada uma vez quando a placa é ligada ou reiniciada, e `loop()`, que é executada repetidamente, permitindo que o Arduino responda a eventos e execute tarefas continuamente.

Para projetos mais complexos que exigem multitarefa e gerenciamento de tempo real, é possível utilizar um Sistema Operacional em Tempo Real (RTOS) como o FreeRTOS. No entanto, isso não é comum para a maioria dos usos do Arduino UNO, que normalmente se beneficia da simplicidade e do controle direto oferecido pelo microcontrolador sem a necessidade de um sistema operacional intermediário.



3 LÓGICA DE CONTROLE DO CARRINHO

Definições de Pinos

Os pinos do Arduino são definidos para controlar os motores do carrinho. Cada motor é controlado por dois pinos de sinal e **um** pino de velocidade (PWM):

- **IN1** e **IN2** controlam o Motor 1.
- **IN3** e **IN4** controlam o Motor 2.
- **ENA** controla a velocidade do Motor 1.
- **ENB** controla a velocidade do Motor 2.

Configuração Inicial (**setup**)

A função **setup** configura todos esses pinos como saídas, o que permite que o Arduino controle os motores:

- O pino IN1 é configurado como saída.
- O pino IN2 é configurado como saída.
- O pino IN3 é configurado como saída.
- O pino IN4 é configurado como saída.
- O pino ENA é configurado como saída.
- O pino ENB é configurado como saída.

Funções de Movimento

Mover para Frente (**moveForward**)

Para mover o carrinho para frente, o código ativa os pinos de controle para que ambos os motores girem para frente. Isso é feito configurando IN1 e IN3 como HIGH (ligados) e IN2 e IN4 como LOW (desligados). Além disso, a velocidade dos motores é definida no máximo (255) usando os pinos ENA e ENB.

Mover para Trás (**moveBackward**)

Para mover o carrinho para trás, o código inverte os sinais dos pinos de controle. IN1 e IN3 são configurados como LOW, e IN2 e IN4 como HIGH. Isso faz com que ambos os motores girem para trás. A velocidade também é definida no máximo.

Virar à Esquerda (**turnLeft**)

Para virar o carrinho à esquerda, o motor esquerdo é parado (IN1 e IN2 são configurados como LOW), enquanto o motor direito continua girando para frente (IN3 é HIGH e IN4 é LOW). Isso faz com que o carrinho gire para a esquerda. A velocidade do motor direito é definida no máximo.

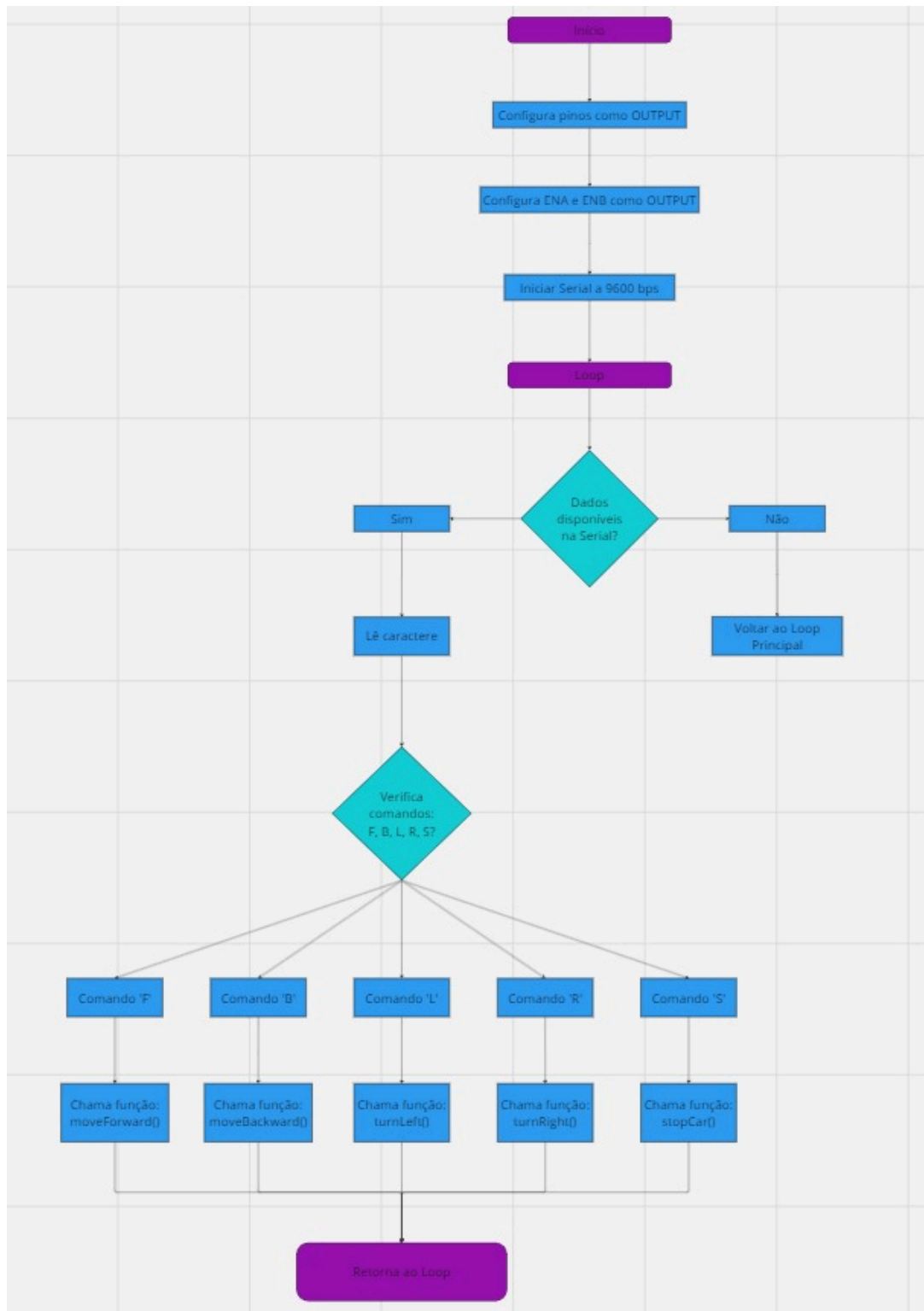
Virar à Direita (**turnRight**)

Para virar o carrinho à direita, o motor direito é parado (IN3 e IN4 são configurados como LOW), enquanto o motor esquerdo continua girando para frente (IN1 é HIGH e IN2 é LOW). Isso faz com que o carrinho gire para a direita. A velocidade do motor esquerdo é definida no máximo.

Parar o Carrinho (**stopCar**)

Para parar o carrinho, todos os pinos de controle dos motores são configurados como LOW, interrompendo a alimentação dos motores. As saídas de velocidade ENA e ENB são definidas para 0, garantindo que os motores parem completamente

4 FLUXOGRAMA DO CÓDIGO



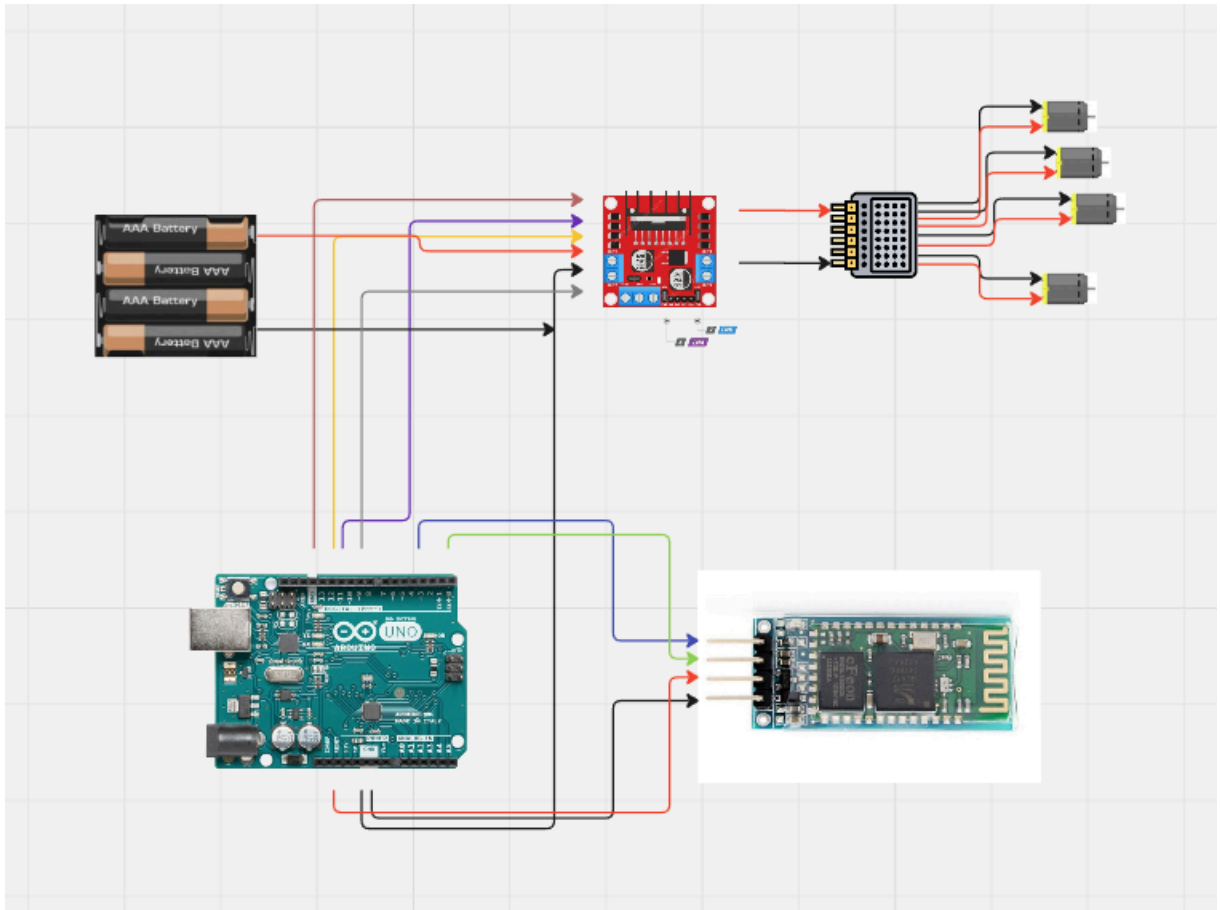


5 CONFIGURAÇÃO DA TECNOLOGIA SEM FIO

```
Serial.begin(9600); // Define a taxa de transmissão serial em 9600 bps
}

void loop() {
  if (Serial.available() > 0) {
    char command = Serial.read();
    switch(command) {
      case 'F':
        moveForward();
        break;
      case 'B': // Trás
        moveBackward();
        break;
      case 'L':
        turnLeft();
        break;
      case 'R':
        turnRight();
        break;
      case 'S': // Parar
        stopCar();
        break;
    }
  }
}
```

6 DIAGRAMA DE MONTAGEM DO CARRINHO





7 CÓDIGO FONTE IMPLEMENTADO

Link do repositório:

<https://github.com/Vinirocha388/logicacarro.git>