

## ROBÔ MÓVEL AUTÔNOMO “SEGUE FAIXA”

<sup>1</sup>Reinaldo Gomes da Rocha  
<sup>2</sup>Eder Firmino de Moraes  
<sup>3</sup>João Alves Bento

### Resumo:

Neste trabalho foi desenvolvido um robô móvel autônomo, movido por energia elétrica através de duas baterias de 9 V acopladas em um chassi de acrílico, capaz de transitar de forma independente da intervenção humana, sendo capaz de tomar decisões assim que a situação exigir. Para realização da automação foi utilizada a plataforma de prototipagem Arduino, modelo UNO R3, como controlador. O Arduino é uma plataforma baseada em microcontroladores que possibilitam que esse dispositivo possa interagir com o ambiente e ter controle sobre atuadores, além de uma unidade central de processamento, periféricos como memórias (tanto de dados quanto de programa), comunicação serial, conversores analógico-digitais, dispositivos PWN, comparadores, entre outros. Com o desenvolvimento desse projeto, foi verificada a possibilidade de um robô móvel percorrer vários trajetos sem a intervenção humana, de forma totalmente independente e autônoma, apenas seguindo uma faixa de cor branca ou preta pintada em uma superfície, capaz de tomar decisões, tais como, virar para a esquerda ou direita.

**Palavras-Chave:** Arduino. Automação. Veículo Autônomo. Sensores. Motor DC. Segue Faixa.

## AUTOMOTIVE MOBILE ROBOT “FOLLOW BAND”

### Abstract:

In this work an autonomous mobile robot was developed, powered by two 9V batteries coupled in an acrylic chassis, able to travel independently of human intervention, being able to make decisions as the situation demands. To perform the automation, the Arduino prototype platform, model UNO R3, was used as controller. The Arduino is a platform based on microcontrollers that enable this device to interact with the environment and have control over actuators, as well as a central processing unit, peripherals such as memories (both data and program), serial communication, analog-digital, PWN devices, comparators, among others. With the development of this project, it was verified the possibility of a mobile robot to cross several routes without human intervention, in a totally independent and autonomous way, only following a strip of white or black color painted on a surface, able to make decisions, such as turning left or right.

**Keywords:** Arduino. Automation. Autonomous Vehicle. Sensors. DC motor. Follow Band.

### 1. Introdução

Há uma constante busca no desenvolvimento de novas tecnologias cada vez mais avançadas, capazes de facilitar ou até mesmo substituir o trabalho humano, tanto nas mais simples

---

<sup>1</sup> Acadêmico de Engenharia Mecânica da UniEVANGÉLICA Centro Universitário de Anápolis-Go, Brasil. Bacharel (Ciência da Computação Anhanguera de Anápolis). reinaldo.rocha.g@gmail.com

<sup>2</sup> Acadêmico de Engenharia Mecânica da UniEVANGÉLICA Centro Universitário de Anápolis-Go, Brasil. eder2012mrs@gmail.com

<sup>3</sup> Mestre (Ciências Moleculares Universidade Estadual de Goiás, Brasil). joaoalvesbento@gmail.com

das tarefas, como trocar de canal de um televisor sem a intervenção direta, através do controle remoto, até nas mais complexas, como o monitoramento cardíaco de um paciente que informa seu médico, de forma automática, caso haja alguma alteração que precisaria de maior atenção.

A robótica também vem avançando bastante nos últimos anos, já temos robôs para quase tudo: robôs que apagam incêndios, robôs que transportam objetos, até mesmo para fins de espionagens. Esse tema vem atraindo muito o interesse da sociedade, bastante representado em ficção científica, onde muitos aspectos já são realidades e outros ainda estão em desenvolvimento [3].

O Arduino como plataforma de prototipagem nos permite desenvolver vários tipos de dispositivos para diversas finalidades, desde pisca-pisca, controle de acesso, portão eletrônico, rastreamento por GPS, até robôs para fins diversos, as possibilidades de sua utilização são inúmeras, sendo limitado apenas por sua imaginação. O baixo custo do Arduino e seus componentes é um fator atrativo para o desenvolvimento de pequenos projetos e/ou projetos conceito, tornando o investimento inicial de desenvolvimento menor. Uma das vantagens da utilização do Arduino é a possibilidade de desenvolvimento de diversos projetos, como os citados acima, com pouco ou quase nenhum conhecimento de eletrônica [5].

## **1.1. Objetivo**

Este trabalho consiste no desenvolvimento de um robô móvel autônomo que utiliza sensores ópticos reflexivos em conjunto com o Arduino. O funcionamento consiste em seguir uma faixa branca ou preta que será usada como o delimitador do trajeto, para esse projeto será usado uma faixa preta.

## **2. Referencial teórico**

### **2.1. Robótica**

De acordo com MEDEIROS:

Robô é uma máquina que se parece com um ser humano e executa várias ações complexas (caminhar, falar) de seres humanos ou um meio (ou máquina) que automaticamente executa tarefas complicadas e repetitivas ou ainda um mecanismo guiado por controle automático<sup>4</sup>.

Essa descrição nos mostra uma definição mais básica sobre robô, mas um robô não é apenas isso, pode ser muito mais complexo, por exemplo, um robô pode executar tarefas repetitivas,

---

<sup>4</sup> P.02

tele operadas, pré-programada ou de forma autônoma, tomando decisões de acordo com sua percepção da uma situação do ambiente, mas isso será conforme suas limitações de reação e antecipação dos problemas [3].

Um robô pode ser de base fixa, como os braços robóticos de uso industrial, ou de base móvel, podendo se locomover no ambiente. Os robôs são utilizados em várias situações, tais como, tarefas que coloquem os seres humanos em risco de vida ou prejudicial à saúde.

Uma das dificuldades em se desenvolver robôs são os parâmetros obtidos através de sensores e demais componentes que interagem com o ambiente, suas informações não são precisas o suficiente para se obter 100% de certeza na decisão a ser tomada, pois o próprio ambiente não está sempre estático, emitindo sempre os mesmos parâmetros, o ambiente muda constantemente. Uma solução abordada é a teoria das probabilidades, que vem tendo sucesso nessas incertezas do ambiente e no próprio robô [3].

Outra solução que está sendo pesquisada e muito adotada nos sistemas inteligentes autônomos são os sistemas de navegação. Com essa função um robô, sabe sua própria posição, pode deslocar de um ponto a outro, calculando a melhor trajetória. Mas saber sua própria posição é muito problemático. Uma possível solução para esse problema é a utilização de sistemas GPS<sup>5</sup>, que consiste na tecnologia de localização por satélite [3].

## **2.2. Planejamento e geração de trajetória**

Na robótica, trajetória pode ser definida através da posição atual dos atuadores no robô, que são calculadas em dois passos: escolha de um ponto de destino e o ponto atual do robô, com essas duas referências é possível calcular a trajetória a ser seguida.

A dificuldade em gerar as rotas está em descobrir a sequência de movimentos, a trajetória deve ser construída evitando os obstáculos entre o ponto atual do robô e o destino. Para isso é possível à utilização de vários pontos intermediários, pela trajetória em que o robô vai passar, possibilitando o desvio dos obstáculos [3].

---

<sup>5</sup> Sigla do inglês *Global Position System*, em português, Sistema de Posicionamento Global.

### 2.3. Arduino

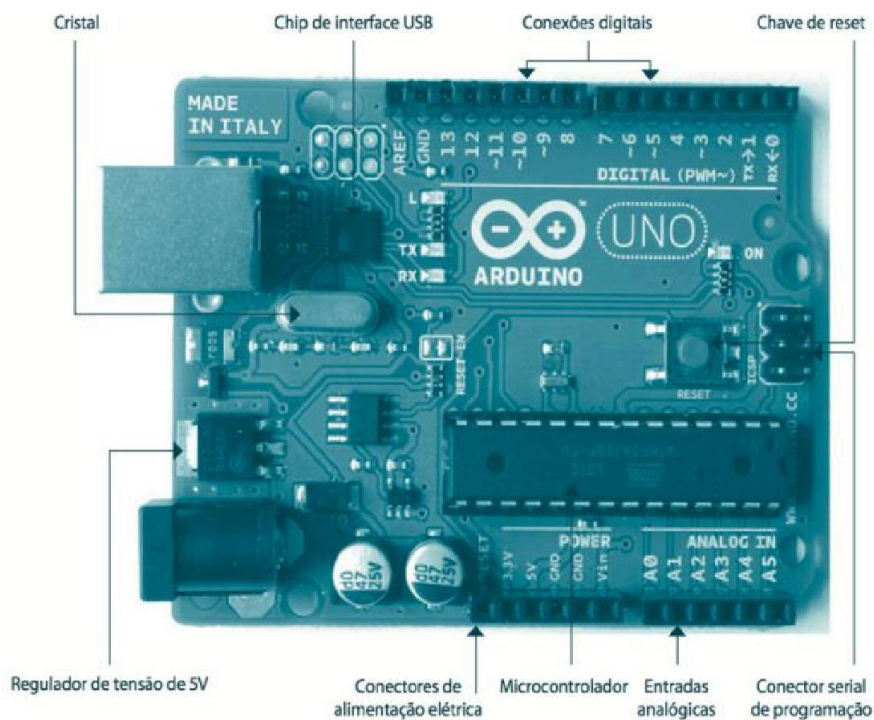
Para STEVAN JUNIOR:

Arduino é uma plataforma *open-source*, ou *software* livre<sup>6</sup>. [...]. Essa plataforma é baseada em microcontroladores, que são unidades de processamento com diversos periféricos internos ao mesmo circuito integrado, que possibilitam que esse dispositivo possa interagir com o ambiente e ter controle sobre atuadores no mesmo, por possuir internamente, além de uma unidade central de processamento, periféricos como memórias (tanto de dados quanto de programa), comunicação serial, conversores analógico-digitais, dispositivos *PWN*, comparadores, entre outros<sup>7</sup>.

Simplemente, é possível a conexão de circuitos eletrônicos aos seus terminais, possibilitando o controle de dispositivos, por exemplo, ligar ou desligar motores, controlar a temperatura e a luminosidade.

A Figura 1 mostra um Arduino Uno e seus principais componentes

**Figura 1:** Placa do Arduino Uno



Fonte: [4]<sup>8</sup>

Sua simplicidade de conexões e expansões dos circuitos eletrônicos ganhou mais adeptos, qualquer pessoa com um pouco de interesse pode criar protótipos mais simples e de forma mais rápida.

<sup>6</sup> “Software livre é o software que é distribuído acompanhado de seu código-fonte, e é liberado sob termos que garantem aos usuários a liberdade de estudar, adaptar/modificar e distribuir o software” [6].

<sup>7</sup> P.124

<sup>8</sup> P. 8

O principal componente do Arduino é o microcontrolador, a maioria restante dos componentes é para a comunicação da placa com o computador e fornecimento de energia elétrica.

O microcontrolador Arduino é um chip de 28 pinos que está encaixado em um soquete no centro da placa (vide Figura 1). São parte desse microcontrolador, a memória, o processador e toda a eletrônica necessária aos pinos de Entrada/Saída.

Um importante fator sobre a plataforma Arduino é a possibilidade de redução da variedade de escolhas a um único microcontrolador padrão. Isso significa que, ao mudar de projeto não será necessária toda uma nova análise das vantagens e desvantagens dos diversos tipos de microcontroladores que melhor se encaixaria nesse novo projeto.

Qualquer projeto utilizando Arduino se baseia na eletrônica e em diversos componentes, cada um com sua função específica. A seguir, serão apresentados os conceitos básicos dos três principais componentes utilizados no projeto deste trabalho.

#### 2.4. Programação para Arduino

Para programar o Arduino utilizamos a linguagem C, e para a importação do código para o arduino foi utilizado o *software* Arduino IDE na versão 1.8.7. A programação para Arduino é dividida, basicamente, nas funções *setup()* e *loop()*. A seguir estão algumas funções que podem ser usadas na programação de um Arduino.

- *setup()*: Responsável por iniciar variáveis, definição dos pinos (*INPUT* ou *OUTPUT*) inicializar bibliotecas, etc. Será executada apenas uma vez, após a placa ser ligada ou acontecer o reset;
- *loop()*: É executada após a função *setup()*, a função *loop()* faz exatamente o que o nome sugere, será executada repetidamente enquanto a placa estiver ligada;
- *analogWrite()*: Adiciona uma onda PWM<sup>9</sup>, que pode ser usada para variar o brilho de um LED ou a velocidade de um motor. Sintaxe: *analogWrite(pino, valor)*;
- *pinMode()*: Configura um determinado pino para funcionar como uma entrada ou saída. Sintaxe: *pinMode(pino, modo)*;
- *digitalRead()*: Função responsável por ler o valor de um pino digital. Sintaxe: *digitalRead(pino)*.

---

<sup>9</sup> Sigla em inglês, *Pulse-Width Modulation*, ou Modulação por Largura de Pulso.

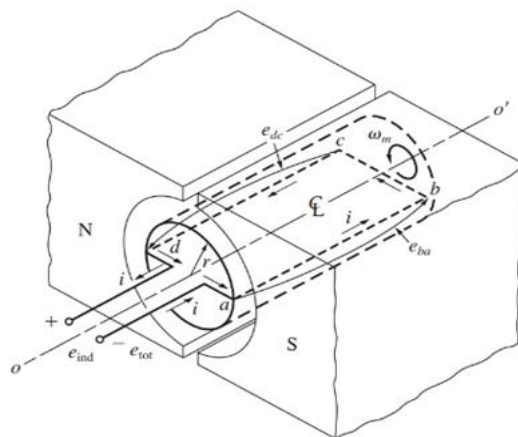
A alteração do comportamento elétrico dos pinos é realizada através da função *pinMode()*. Pinos configurados como entrada, *INPUT*, através da função *pinMode()*, estarão em um estado de alta-impedância, isso faz com que seja possível, por exemplo, ler informações de sensores. Já os pinos configurados como saída, *OUTPUT*, estarão em um estado de baixa-impedância, o que significa que podem fornecer corrente para outros circuitos.

## 2.5. Motor DC

Os motores DC (*Direct Current*) ou motores CC (Corrente Contínua) é uma máquina elétrica rotativa que transforma energia elétrica, corrente contínua, em energia mecânica [2]. Essas máquinas são motores que trabalham aproveitando as forças de atração e repulsão geradas por ímãs ou eletroímãs. A velocidade é controlada apenas variando a tensão.

Motores DC funcionam devido às propriedades dos ímãs e ao campo magnético criado quando circula uma corrente elétrica por um condutor.

**Figura 2:** Funcionamento do motor de corrente contínua



Fonte: [2]<sup>10</sup>

Quando uma corrente elétrica flui através de um condutor no meio de um campo magnético, uma força eletromotriz é gerada no condutor que faz com que ele se mova e, portanto, gira o eixo central. Uma vez atingida a posição de equilíbrio, a direção da corrente que flui através do condutor é comutada e o rotor gira novamente procurando o equilíbrio novamente.

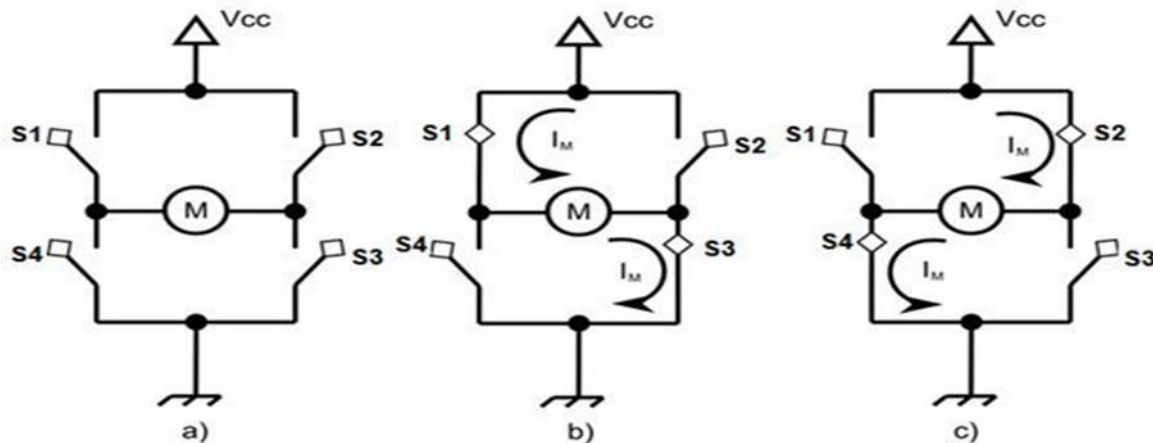
## 2.6. Driver Motor Ponte H L298N

Este *driver* ponte H é baseado no chip L298N, construído para controlar cargas indutivas, solenoides, motores DC e motores de passo. Com este componente é possível controlar de forma independente a velocidade e rotação de dois motores DC ou 1 motor de passo.

<sup>10</sup> P. 406

Este dispositivo recebe esse nome devido sua configuração do circuito em forma de H. Utiliza quatro chaves (S1, S2, S3 e S4) que são acionadas de forma alternada. Conforme a configuração entre elas, os motores vão girar para um sentido ou para outro.

**Figura 3:** Circuito Ponte H



**Fonte:** Portal Vida de Silício<sup>11</sup>

a) quando nenhuma das chaves estiverem fechadas, o motor estará parado. b) quando as chaves S1 e S3 estiverem acionadas e as chaves S2 e S4 estiverem desligadas o motor irá girá para um sentido. c) quando as chaves S2 e S4 estiverem acionadas e as chaves S1 e S3 estiverem desligadas, o motor irá girá para outro sentido.

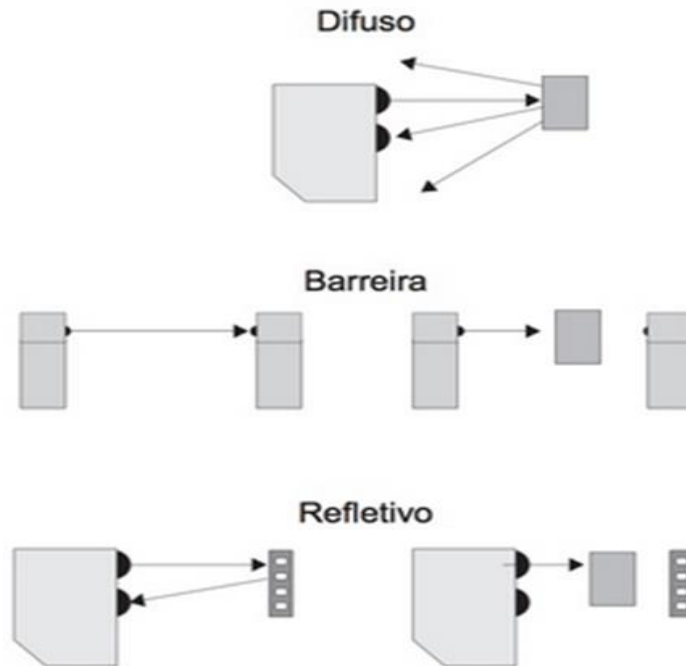
## 2.7. Sensores Fotoelétricos

Os sensores fotoelétricos podem ser utilizados para diversas finalidades, tais como, esteira de empacotamento, processo de embalagem, robótica, em muito mais. Seu formato também tem uma grande variedade, podendo ser escolhido de acordo com os fatores de utilização, podendo ficar expostos ao clima ou de acordo com o objeto que será detectado. Também há modelos que variam sua capacidade em relação à distância do objeto a ser detectado.

Seu funcionamento consiste basicamente na transmissão e recepção luminosa para detectar a presença de um determinado objeto a certa distância. Alguns tipos de sensores utilizam a interrupção do feixe de luz para detectar o objeto, já outros utilizam a reflexão da luz no próprio objeto, como podemos observar na figura abaixo.

<sup>11</sup> Disponível em: <https://portal.vidadesilicio.com.br/driver-motor-com-ponte-h-l298n/>. Acesso em nov. 2018.

**Figura 4:** Principais tipos de sensores ópticos: difuso, barreira e refletivo.



Fonte: [1]<sup>12</sup>

### 3. Metodologia

#### 3.1. Materiais utilizados

Foi utilizado uma lista de componentes necessários para montar o robô móvel autônomo. Abaixo segue a lista dos componentes.

- Arduino UNO R3

Utilizado para controlar o robô.

- 01 *Kit* Robótica 2WD

O *kit 2WD* é destinado à montagem de robô móvel em um chassi feito de acrílico. Também é composto por dois motores DC com caixa de redução 1:48, em cada motor é conectado uma roda. Outro componente desse *kit* é uma roda boba para a sustentação do chassi.

O chassi tem dimensão 21,2 x 15,2 cm e as rodas, 7 x 7 x 2,6 cm cada.

- Ponte H

A Ponte H controla a rotação e a velocidade dos motores.

- 02 Baterias de 9 V

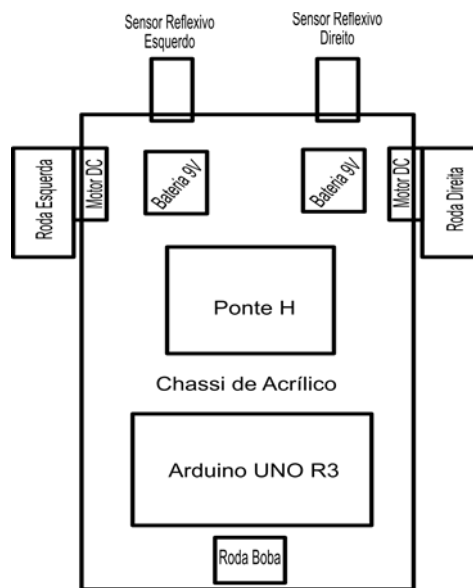


### 3.2. Montagem dos componentes

Utilizando a placa Arduino UNO para controlar os dois motores e os sensores refletivos para indicar a direção a ser tomada.

Para montar os componentes foi utilizado o chassi de acrílico. As disposições dos componentes foram organizados conforme esquema na figura 5.

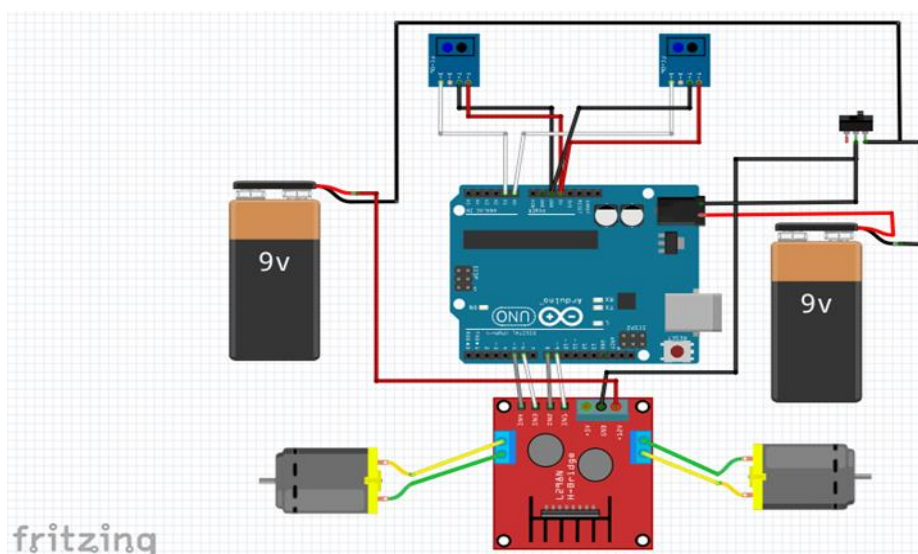
**Figura 5:** Esquema de montagem



Fonte: Autores

A montagem da parte elétrica foi esquematizado utilizando o *software Fritzing*, na sua versão 0.9.3, figura 6.

**Figura 6:** Esquema elétrico



Fonte: Autores

Conforme visto na figura 6, o motor DC Esquerdo foi ligado nas portas *OUT1* e *OUT2* da Ponte H e o motor direito, nas portas *OUT3* e *OUT4*. Os pinos *VCC* e *GND* dos sensores reflexivos, foram ligados nas portas *5V* e *GND* do Arduino, respectivamente, essas portas são responsáveis pela alimentação dos sensores, e as portas *A0* e *A1* do Arduino foram ligados nas portas *A0* do sensor direito e esquerdo, respectivamente. O pino *D0* é responsável pela saída da informação de leitura do sensor. Uma das baterias 9V foi usada para a alimentação da Ponte H, através das portas *+12V* e *GND*, a outra bateria foi utilizada para a alimentação do Arduino. Para controlar os motores foram ligadas as portas *IN1*, *IN2*, *IN3* e *IN4* da Ponte H nas portas *~10*, *~9*, *~6* e *~5*, respectivamente.

### 3.3. Algoritmo

O algoritmo utilizado necessita de dois sensores reflexivos para a obtenção dos parâmetros para as demandas de decisão de virar para a direita ou para a esquerda.

A lógica se dá da seguinte forma:

*Se (SensorDireito está em cima da faixa preta)*

*Vira para a direita*

*Se (SensorEsquerdo está em cima da faixa preta)*

*Vira para a esquerda*

Para que o robô consiga seguir a faixa preta, essa verificação ficará dentro na função *loop()*, pois será executada repetidamente enquanto a placa estiver ligada.

Na função *setup()*, serão configurada as portas *~5*, *~6*, *~9* e *~10*, do Arduino, como portas de saída. Estas configurações serão configuradas através da função *pinMode()*, onde serão passados, por parâmetro, o respectivo número da porta e a informação que a porta será usada como porta de saída.

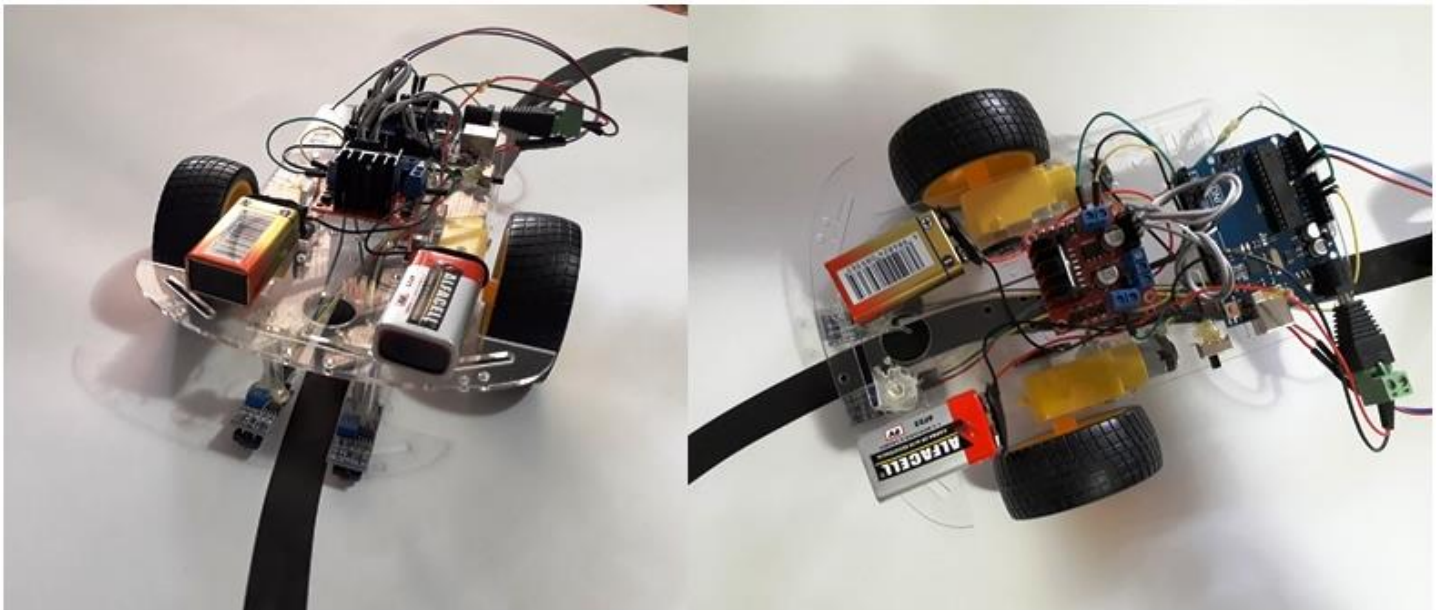
A velocidade e a direção, para frente ou para trás, do robô é feito através da função *analogWrite()*, sendo passado por parâmetro o número da porta do Arduino e a velocidade, que vai de 0 (zero) à 255, sendo 0 (zero) para parado e 255 para velocidade máxima de rotação do motor.

As informações capturadas pelos sensores reflexivos são lidos através da função *digitalRead()*. A função *digitalRead()* ficará dentro da função *setup()*, pois essas informações serão obtidos a cada interação, repetidamente, enquanto a placa estiver ligada.

#### 4. Resultados e Discussões

A proposta do projeto foi concluída parcialmente. A construção do robô móvel foi realizada com sucesso, vide figura 7, todos os componentes eletrônicos instalados funcionaram bem. O robô móvel realizou vários trajetos de forma satisfatória em superfícies claras e que não tenham muitas variações de cores. Os sensores reflexivos não se comportaram conforme o esperado em pisos com cerâmicas, principalmente as mais escuras. De um modo geral, consideramos que o projeto foi realizado com sucesso, pois conseguimos construir o robô móvel autônomo e desenvolver o algoritmo responsável por obter os dados dos sensores e tomar uma decisão.

**Figura 7:** Robô finalizado



**Fonte:** Autores

#### 5. Conclusões

Considerando a utilização de componentes profissionais, como sensores usados em linhas de produção de fábricas, a construção do robô móvel autônomo em um tamanho, que possibilite o transporte de objetos ou até mesmo de pessoas, se mostrou possível. O robô móvel desenvolvido neste projeto se torna ponto de partida para novos trabalhos, através da implantação de mais recursos, tais como, sensor de obstáculos, sensores de pontos de parada, *timers* para indicação na hora de saída ou chegada em um ponto de embarque e desembarque. É possível citar um exemplo prático da utilização profissional de robôs autônomos, como o *Smart Transport Robots*, que

transportam componentes através da logística na fábrica da BMW, na cidade de *Wachtersdorf* na Alemanha.

**Figura 8:** *Smart Transport Robots*



Fonte: *BimmerFile – Daily BMW News*<sup>13</sup>

## 6. Referências

- [1] CAMARGO, Valter Luís Arlindo de. **Elementos de automação**. São Paulo. Editora Érica, 2014.
- [2] CHAPMAN, Stephen J. **Fundamentos de máquinas elétricas [recurso eletrônico]**. Tradução: Anatólio Laschuk. – 5. ed. – Dados eletrônicos. – Porto Alegre: AMGH, 2013.
- [3] MEDEIROS, Adelardo Adelino Dantas de. ... [et al.]; organização Roseli Aparecida Francelin Romero... [et al.]. **Robótica Móvel**. 1. ed. - [Reimpr.]. - Rio de Janeiro: LTC, 2017. il.; 28 cm.
- [4] MONK, Simon. **Programação com Arduino [recurso eletrônico]: começando com Sketches**. Porto Alegre. Editora Bookman. 2013.

---

<sup>13</sup> Disponível em: <http://www.bimmerfile.com/2016/11/21/bmw-shows-off-logistics-concepts-from-data-goggles-to-electric-trucks>. Acesso em dez. 2018

[5] OLIVEIRA, Cláudio Luís Vieira. Cláudio Luís Vieira Oliveira; Humberto Augusto Piovesana Zanetti. **Arduino descomplicado: como elaborar projetos de eletrônica** – São Paulo : Érica, 2015.

[6] STEVAN JUNIOR, Sérgio Luiz. SILVA, Rodrigo Adamshuk. **Automação e instrumentação industrial com arduino: teoria e projetos**. São Paulo. Editora Érica. 2015.