

ROBÔ SEGUIDOR DE LINHA: ARDUINO SEGUIDOR DE LINHA POR MEIO DE MÓDULOS

Alecio Galeno¹ Daniel Major¹ Erick Gasparino¹ Euller Vernizzi¹ Eduardo dos Santos¹ Flavio Machado¹ Genesio dos Reis¹ Josimar Silva Ribeiro¹ Jorlan da Cruz Araujo¹ Quelvin Tiery¹ Vinicius Frati¹

Universidade Ibirapuera
Av. Interlagos, 1329 – São Paulo – SP
josimar.ribeiro1@gmail.com

Resumo

Este trabalho tem como principal objetivo o estudo de técnicas e ferramentas utilizadas no desenvolvimento de trajetórias para robôs móveis, utilizando para isso a plataforma Arduino, um sistema de prototipação eletrônica que conta com hardware livre, dois motores e sensores funcionando paralelamente. Procura-se não somente desenvolver a parte física de um robô, como também o algoritmo para o problema proposto. Esta arquitetura é também conhecida por sua relativa facilidade de trabalho aliado a um baixo custo. Deseja-se que o robô consiga percorrer uma trajetória programada previamente.

Palavras-chave: Arduino, Robô, Trajetoria, Microcontrolador.

Abstract

This work has as main objective the study of techniques and tools used in the development of trajectories for mobile robots, using the Arduino platform, an electronic prototyping system that has free hardware, two motors and sensors running in parallel. We seek not only to develop the physical part of a robot, but also the algorithm for the proposed problem. This architecture is also known for its relative ease of work combined with low cost. The robot is intended to be able to travel a previously programmed path.

Keywords: Arduino, Robot, Trajectory, Microcontroller.

1. Introdução

Este trabalho apresenta os passos para construir um robô capaz de seguir uma linha no chão. A monografia começa com uma visão geral dos objetivos do trabalho e em seguida apresenta detalhes do planejamento, da montagem do hardware e da implementação do software que serão usados no robô. Na locomoção serão utilizados motores de corrente contínua alimentados por uma bateria, além de caixas de redução para aumentar o torque dos motores e circuitos que permitem inverter o sentido ou variar velocidade de rotação dos motores. Para detectar a presença da linha preta o robô estará equipado com emissores e receptores de luz na faixa infravermelho baseados na reflexão da luz pela cor branca e absorção pela luz preta. Para controlar tudo será utilizado o micro controlador Arduino.

2. Objetivos

Abaixo são listados os objetivos gerais e específicos para a execução deste trabalho.

2.2. Objetivo geral

O objetivo geral é utilizar o micro controlador Arduino para desenvolver um robô móvel terrestre, capaz de se deslocar em uma trajetória pré-determinada, desenvolver as ações para o robô seguir a rota estipulada até alcançar o seu destino, contendo sensores e dois motores funcionando paralelamente atendendo assim ao objetivo principal da matéria de Sistemas Paralelos e Distribuídos do curso de Ciência da Computação da Universidade Ibirapuera – Campus Chácara Flora.

2.3. Objetivos específicos

Abaixo, lista de objetivos:

- Entender a teoria por trás dos processos paralelos e distribuídos utilizados em nosso projeto;
- Realizar um estudo sobre dimensionamento de um motor para determinada tarefa ou, em nosso caso,

são dois motores de corrente contínua adequado para mover o robô entendendo o funcionamento deste;

- Aprender a teoria, a montar e utilizar uma ponte H, que em nosso projeto será utilizada para inverter o sentido de rotação do motor CC;
- Aprender a utilizar o micro controlador Arduino e a linguagem de programação utilizada nele;
- Estudar sobre PWM (modulação por largura de pulso) e como gerar este pulso utilizando o Arduino. Este estudo será utilizado para controlar a velocidade dos motores e assim conseguir realizar uma curva;
- Aprender a montar e utilizar sensores de infravermelho.

3.0. Construção

Atualmente, a plataforma com rodas é a mais comum. Comparado com outros, possui várias vantagens, como o custo relativamente baixo, a abundância de opções, além do design e construção simples. É altamente recomendado para iniciantes em robótica.

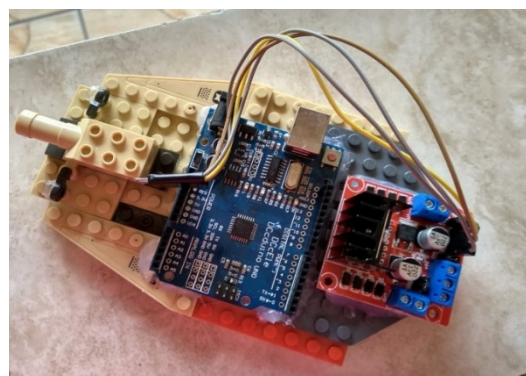


Figura 1 - Carro com placa Arduino

3.1. Componentes

Para esse robô, utilizamos os seguintes itens:

- **Base lego:** são peças modulares para a alocação dos hardwares adicionais;
- **Arduino Uno R1:** é uma placa baseada no microcontrolador ATmega328P. Suas especificações técnicas são:

- Tensão de operação 5V USB;
- Tensão de entrada Recomendada 7 - 12V;
- Tensão de entrada Máxima 6-20V;
- Pinos I/O Digitais 14 (onde 6 são saídas PWM);
- Pinos Digitais I/O PWM 6;
- Pinos entrada Analógicos 6;
- Corrente DC por Pino I/O 20 mA;
- Corrente DC para Pino 3.3V 50 mA;
- Flash Memory 32 KB (ATmega328P);
- SRAM 2 KB (ATmega328P);
- EEPROM 1 KB (ATmega328P);
- Clock Speed 16 MHz;
- LED_BUILTIN 13;
- Altura 68.6 mm;
- Comprimento 53.4 mm;
- Peso 25 g.

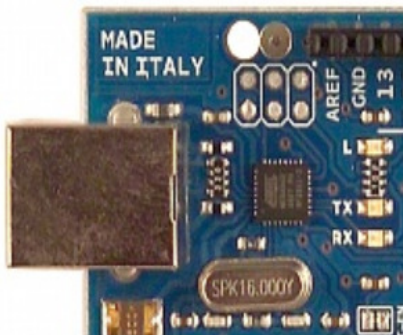


Figura 2 - Arduino Uno

- **IR TCRT5000:** é um componente eletrônico aplicado com frequência em projetos que envolva detectar obstáculos. Um sensor infravermelho pode ajudar a detectar a distância entre ele e um objeto e enviar essas informações ao microcontrolador. Suas especificações:
 - Tensão de operação: 5VDC;
 - Corrente máxima: 60 mA;
 - Comprimento de onda: 950nm;
 - Distância de detecção(máxima): 25mm.

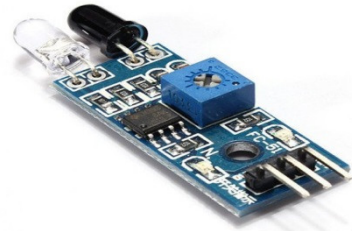


Figura 3 - Sensor Infravermelho

• **Módulo ESP8266:** é um chip que possui WIFI possibilitando a conexão de diversos dispositivos a internet como sensores, atuadores, etc.. Este módulo é usado para conectar-se ao dispositivo Android e se comunicar com o Arduino. O programa para o ESP8266 depende do plano de programação. Usamos a arquitetura do cliente-servidor para implementar este sistema. O dispositivo Android é o servidor deste sistema, já o ESP8266 e o Arduino, são clientes. Uma funcionalidade interessante é o fato de poder ser controlado a partir de um navegador de internet comum, usando uma interface projetada em HTML. Um smartphone Android pode ser usado para transmitir os comandos para a interface de controle. O uso do código abaixo faz com que o Arduino redefina o módulo, entre em uma rede, aguarde um pouco pela conexão, mostre seu endereço IP e inicie um servidor da web. Depois disso, o loop principal será iniciado e o microcontrolador aguardará os comandos. No código abaixo, o robô começa a se mover quando recebe um determinado comando ("cm1" a "cm4") e mantém esse estado até que um comando diferente seja recebido ("cm1" a "cm5"). Às vezes, o ESP8266 perde algumas mensagens e isso pode causar alguns problemas. Se, por exemplo, um comando "cm5" for perdido, o robô começará a mover-se mesmo após o lançamento de qualquer tecla de seta.



Figura 4 - Módulo Wi-fi Esp8266

- **Ponte H L298N:** Este driver é baseado no circuito tipo ponte H e é dedicado para controle de motor DC. O mesmo possui dois canais e permite controlar velocidade e sentido de rotação de até dois motores ao mesmo tempo.

Possui terminais parafusáveis furos nas extremidades da placa para fixação ao protótipo. Suas especificações:

- Tensão de Operação: 4~35v;
- Chip: ST L298N (Datasheet);
- Controle de 2 motores DC ou 1 motor de passo;
- Corrente de Operação máxima: 2A por canal ou 4A max;
- Tensão lógica: 5v;
- Corrente lógica: 0~36mA;
- Limites de Temperatura: -20 a +135°C;
- Potência Máxima: 25W;
- Dimensões: 43 x 43 x 27mm;
- Peso: 30g.

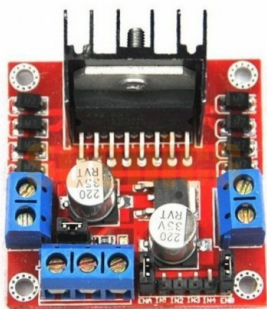


Figura 5 – Ponte H L298N

4. Etapas do projeto

Iniciamos com o planejamento do projeto. Utilizamos partes do PMI para conseguir gerenciá-lo. Após isso, discutimos mais a fundo as suas etapas e suas tarefas. Utilizamos a ferramenta Trello para conseguir acompanhar melhor o avanço do projeto. Com isso, conseguimos identificar as dificuldades dos membros e dividir melhor as tarefas.

Em resumo, os passos para desenvolvimento do projeto são:

- Planejamento do projeto;
- Distribuição das tarefas e acompanhamento do Trello;
- Definição do hardware a ser utilizado;
- Construção do chassi em Lego;
- Instalação do Moto DC;
- Instalação e configuração do módulo ponte H L298N;
- Ligação e programação do módulo ESP8266;
- Instalação e ligação dos módulos TCRT5000;
- Desenvolvimento do Site;
- Teste em pista;
- Ajustes de velocidade;
- Adaptação da pista;
- Documentação.

5. Dificuldades do projeto

O projeto foi um grande desafio. Tivemos que aprender a função dos módulos eletrônicos, suas conexões e como fazer com que eles trabalhem em conjunto para atingir o objetivo.

Na etapa de montagem do motor, encontramos vários obstáculos para a montagem e conexões com as portas PWM para controle de velocidade. Foi necessário um ajuste fino e inúmeros testes para chegarmos aos valores corretos.

O ESP possui diversos detalhes para ser codificado que exigiu muita atenção e troubleshooting.

Nos módulos TCRT5000, a distância entre a pista e o sensor causou sérias dificuldades. Esse fator foi um dos decisivos para mudarmos de uma base em Lego para outra em madeira e ferro. Com isso, obtivemos melhor estabilidade dos componentes.

6. Sistemas distribuídos e paralelos

Na construção do nosso projeto, conseguimos identificar vários pontos referenciados no livro *Sistemas Distribuídos e Paralelos*.

O primeiro deles foi identificar a arquitetura centralizada do nosso robô. Os módulos adicionais enviam as informações para o Arduino UNO. Para melhor assimilação segue um diagrama:

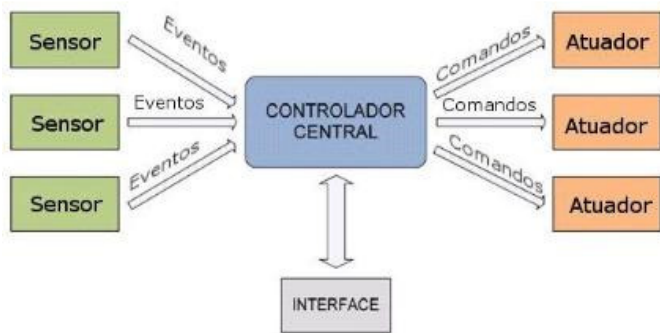


Figura 5 – Arquitetura Centralizada

Podemos assimilar da seguinte forma:

- Controlador central: Arduino Uno;
- Sensor: TCRT5000;
- Atuador: Ponte H L298N com o conjunto de motores;
- Interface: ESP8266 com uma interface web e/ou IDE Arduino.

O Arduino terá o código com todas as regras e controles. Fará a leitura dos resultados obtidos pelos sensores. A partir desses resultados, enviará os comandos para o H L298N que passará ao conjunto de motores e em seguida chegará as rodas.

Podemos acompanhar através da IDE ou até melhor, através de um navegador web. O ESP8266 permitirá a comunicação de browser com o Arduino. Podemos apenas obter o resultado das leituras ou desenvolver um código que permita o envio de informações e até mesmo a programação através do navegador.

Pensando no site, utilizamos uma hospedagem externa. Essas hospedagens, geralmente, utilizam servidores paralelos para a execução dos seus serviços de hospedagem. Isso permite uma otimização do uso do hardware e a consequente redução nos custos.

No livro, temos um tópico de virtualização. Podemos virtualizar vários servidores em plataformas como o VirtualBox, VmWare e Hyper-V. Com isso podemos ter uma redundância do serviço. Se um servidor tiver algum problema, o serviço estará disponível no outro.

A gestão da distribuição de conexões pode ser realizadas por frontends e equipamentos de rede, como, por exemplo, o BIG-IP.

Esse trabalhará como um middleware capaz de identificar se o serviço está disponível ou não. Pode

também trabalhar para melhorar a performance do servidor web fazendo um balanceamento de carga.

Fazendo a analogia com o nosso Arduino, poderíamos montar dos robôs seguidores de linhas que permitam a conexão e gerenciamento remoto através do módulo ESP8266. Pensando em preservar seus componentes o maior tempo possível, o ideal é que eles sejam utilizados o mesmo tempo nesse contexto. Para fazer esse balanceamento, poderíamos ter um frontend que teria a função de um middleware para fazer a distribuição e contagem de execuções. Teríamos um balanceamento das cargas de serviços.

Poderíamos ter também um serviço paralelo, utilizando um middleware que gerenciasse 2 Arduínos, imaginemos que os robôs seguidores de linha terão a função de limpar a pista. O middleware enviará o comando para os 2 simultaneamente, com isso, cada um limpará metade da pista diminuindo pela metade o tempo da tarefa.

No livro, encontramos também capítulos que abordam a comunicação. A comunicação é essencial para a conversa entre os nós. O ESP8266 se comunica utilizando a rede a Wi-Fi. Ele receberá um endereço IP, e rodará o protocolo HTTP que utiliza o protocolo TCP para transporte.

Adicional a isso podemos adicionar um DNS para controlar o nome dos ESP conectados à rede e facilitar a identificação humana desses componentes.

7. Referências Bibliográficas

ARDUINO (Acessado em 2019). Site oficial Arduino. <https://www.arduino.cc>.

FLIPFLOP. (Acessado em 2019). Blog Controle de velocidade de motor DC com PWM no Arduino UNO. <https://www.filipeflop.com/blog/controle-motor-dc-pwm-arduino-uno/>.

ELETROGATE. (Acessado em 2019). Blog IoT com módulo WiFi Esp8266 – Básico. <https://blog.eletrogate.com/iot-com-modulo-wifi-esp8266-basico/>.

Tanenbaum, Andrew S. (2007). Sistemas Distribuídos, edição 2. Pearson Education.

EXPLORE EMBEDDED. (Acessado em 2019).

Blog Arduino Support for ESP8266 with simple test code.

https://www.exploreembedded.com/wiki/Arduino_Support_for_ESP8266_with_simple_test_code.

EMBARCADOS. (Acessado em 2019). Blog

Usando as saídas PWM do Arduino.

<https://www.embarcados.com.br/pwm-do-arduino/>.