

**UNIVERSIDADE FEDERAL DE OURO PRETO**

**Mestrado em Ciência da Computação**

**Especificação Sistêmica de Carro Robô Seguidor de Linha  
com Sistema Avançado Utilizando Controle Proporcional**

Autor:

Rodolfo Labiapari Mansur Guimarães -  
rodolfolabiapari@decom.ufop.br

Professor:

Ricardo Augusto Rabelo Oliveira - rrabelo@gmail.com

Ouro Preto - MG  
6 de fevereiro de 2017

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Referencial Teórico</b>	<b>1</b>
2.1	Equação Diferencial Linear . . . . .	1
2.2	Controlador Proporcional . . . . .	1
2.2.1	Visão Geral . . . . .	1
2.3	Modelamento Teórico-Matemático . . . . .	3
2.3.1	Definições . . . . .	3
2.3.2	Teoria de Controle Proporcional (P) . . . . .	5
2.3.3	Teoria de Controle Proporcional Integral (PI) . . . . .	5
2.3.4	Teoria de Controle Proporcional Derivativo (PD) . . . . .	5
2.3.5	Teoria de Controle Proporcional Integral-Derivativo (PDI) . . . . .	6
<b>3</b>	<b>Elementos do Projeto</b>	<b>6</b>
3.1	Microcontrolador . . . . .	6
3.2	Energização . . . . .	7
3.3	Sensores . . . . .	8
3.4	Atuadores . . . . .	11
<b>4</b>	<b>Implementação</b>	<b>11</b>

# 1 Introdução

Atualmente é possível construir Carro Robô Seguidor de Linha utilizando poucos componentes, e com poucas linhas de programação.

É possível encontrar facilmente em sites de venda de eletrônicos, kits de sensores e atuadores de baixo custo prontos para serem acoplados à placa de prototipagem e assim, a construção de um carro seguidor.

O desenvolvimento do sistema controlador também segue o mesmo princípio de facilitação de uso. Utilizando uma plataforma de prototipagem tal como Arduino, é possível escrever o controle de um carro seguidor com poucas linhas de instruções. Isso pode ser levado como um desafio até para crianças e adolescentes com criatividade e entusiasmo para desenvolver um sistema completo e funcional como incentivo à robótica.

Entretanto, o que diferencia de grandes competidores de amadores é o algoritmo de controle do carro. Algoritmos com controles simples funcionarão e completarão a prova, mas terão grande dificuldade. O que diferencia carros simples à complexos e velozes seguidores de linha não são seus componentes, organização e a precisões desses, mas sim a adaptação de seu controle. Com isso, o objetivo deste trabalho é especificar um carro seguidor de linha onde ele deverá possuir todos os equipamentos necessários para a conclusão de seu objetivo além de uma adaptação na avaliação e atuação de seus sensores para que este possa obter melhores resultados usando cálculos matemáticos mais elaborados a seu favor como Controle Proporcional.

Na Seção 2 será feita uma introdução às definições matemáticas dos temas abordados no texto. Nesta Seção encontra-se todos os detalhes técnico-matemáticos para o entendimento do trabalho. Na Seção 3 é descrito brevemente sobre os equipamentos utilizados no projeto para seu funcionamento. Por fim, na Seção 4 é descrito um pseudocódigo do procedimento proposto.

## 2 Referencial Teórico

### 2.1 Equação Diferencial Linear

Equações Diferenciais Lineares são equações que suas soluções podem ser somadas a fim de produzir uma nova solução.

É considerada linear quando satisfaz as características de que cada coeficiente  $a_n$  e o termo de não-homogeneidade só dependem da variável independente, no caso  $x$ ; e a variável dependente, no caso  $y$ , e suas derivadas são de primeiro grau. Assim, ela deve representar a seguinte forma

$$f(x) = a_n(x) \frac{d^n y}{dx^n} + a_{n-1}(x) \frac{d^{n-1} y}{dx^{n-1}} + \cdots + a_1(x) \frac{dy}{dx} + a_0(x)y \quad (1)$$

### 2.2 Controlador Proporcional

#### 2.2.1 Visão Geral

A teoria de controladores proporcionais se baseiam em sistemas realimentados. Tais podem ser divididos em basicamente três partes sendo elas:

- Sistema a ser controlado;
- Controlador (também conhecido como compensador); e
- Realimentação.

O sistema a ser controlado é constituído por atuadores capazes de efetuar as ações necessárias. Os outros dois elementos têm como finalidade fazer com que o desempenho do sistema possua estabilidade e opere com certa precisão e agilidade, seguindo as especificações uma vez estabelecidas. Uma vez estabelecido como o sistema deverá ser desenvolvido, os controladores e sua realimentação serão item essencial para que o processo ocorra de forma estável. Tudo isso pode ser observado no diagrama exibido na Figura 1.

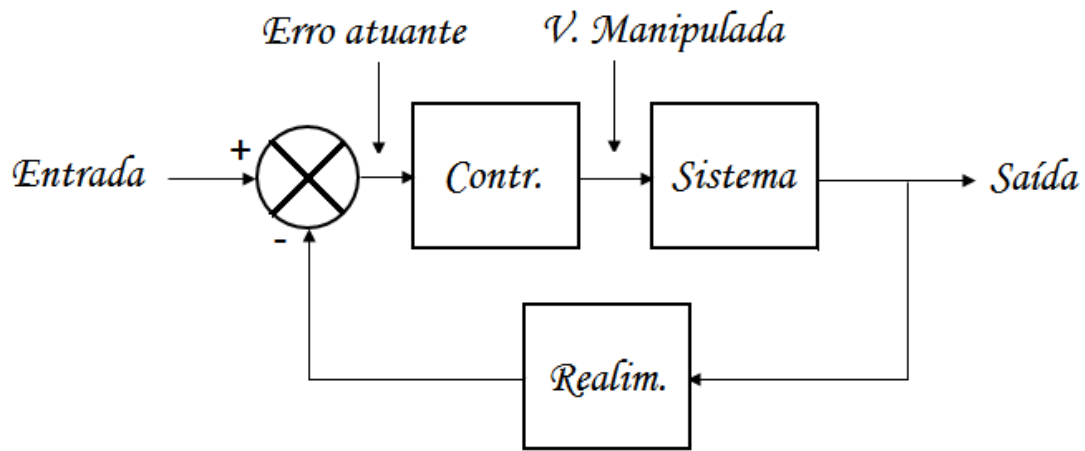


Figura 1: Estrutura de Sistema de Controle Geral.

A realimentação é o ponto chave para a diferenciação de sistemas controladores comuns. Um sistema de controle realimentado compara, instantaneamente, o valor de saída anterior, com o valor de referência existente na entrada do sistema como os sensores. O resultado desta comparação é o centro de toda adaptabilidade estudada e é denominado erro atuante. Este é levado ao controlador, que produz o chamado sinal de controle, cuja função resume-se em reduzir o desvio entre a saída e o sinal desejado. Como o nome sugere, em um controlador proporcional a saída do mesmo, também conhecido como sinal de controle (ou ação de controle), é diretamente proporcional ao sinal de erro, ou seja, ao erro atuante.

Sabendo-se em da proporcionalidade direta entre o sinal de controle e o sinal de erro, é possível afirmar que

$$a(t) \propto e(t) \quad (2)$$

onde o controle é diretamente proporcional à seu erro.

Entretanto, não é possível realizar operações matemáticas exatas com o sinal de proporção da fórmula. Par isso, deve-se admitir então uma constante de proporcionalidade entre as mesmas. Esta possui o nome de ganho proporcional e é

representado pela variável  $f$ . Sendo assim

$$a(t) = f * e(t) \quad (3)$$

Dessa forma, o sistema matemático mostrado na Equação 3 poderá ser representado pelo diagrama exibido na Figura 2.

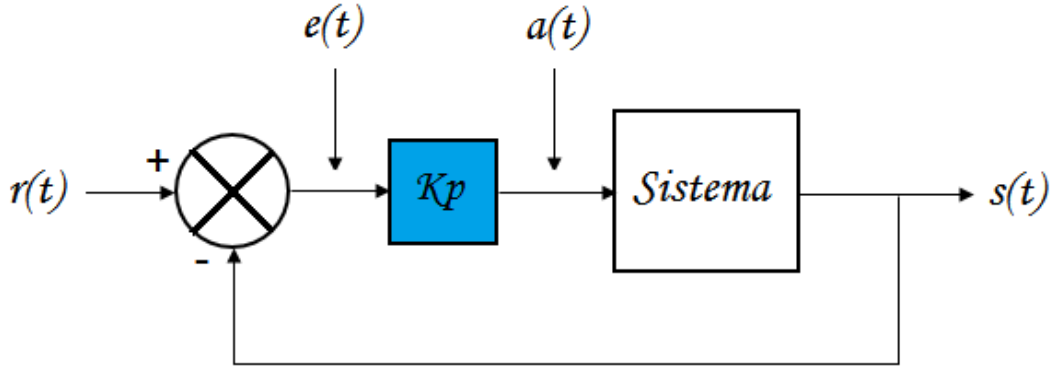


Figura 2: Sistema no Domínio do Tempo Geral.

Sendo assim, como fórmula final, temos

$$a(t) = f * (r(t) - s(t)) \quad (4)$$

onde  $a(t)$  é a atuação,  $e(t)$  representa o erro,  $r(t)$  é o valor de início de execução de controle e  $s(t)$  é saída do controle. Todos representando o valor no tempo  $t$ .

## 2.3 Modelamento Teórico-Matemático

### 2.3.1 Definições

Para iniciar a discussão sobre sistema de controle proporcional, primeiramente será definido os termos utilizados.

**Alvo:** É a posição que desejamos que o carro esteja em relação à faixa, ou seja, centralizado respeitando a devida linha.

**Erro:** Diferença entre a posição atual e o *Alvo*. Pode ser qualquer valor no conjunto dos  $\mathbb{R}$ .

**Proporção:** Fator que determinará quão distante o carro está da linha. Por exemplo são as proporções: ‘para a esquerda’, ‘para a direita’, ‘para a extrema esquerda’, ‘pouco para a direita’, etc. É representado pela constante de variação  $K_p$ .

**Integral:** Proporciona o erro acumulado sobre o tempo corrido. Isso significa que este irá informar se o carro vem estando na linha nos últimos momentos ou não. Representada pela constante de variação  $K_i$ .

**Derivada:** Taxa no qual exibe a oscilação do robô sobre a linha, representada pela constante de variação  $K_d$ .

Ambos *Integral* e *Derivada* utilizam o valor *Proporção*.

O procedimento de execução é baseado no seguinte pseudocódigo:

1. Realiza o cálculo inicial da posição atual.
2. Calcula o erro baseado na posição atual.
3. Ele então comandará os motores para fazer uma girada:
  - Brusca caso o erro for grande; ou
  - Leve caso o erro for pequeno.

Basicamente, a magnitude do giro tomado é proporcional ao *erro*.

4. Repete os passos até completar o objetivo.

Com esses passos, saímos de um controle simples para um Contro Proporcional mais eficaz.

Mesmo se depois desses passos o valor de *erro* não decrementar ou decrementar de forma bem lenta, ao realizar o passo 4, o controlador fará uma nova avaliação incrementando a magnitude de conserto de posição realizando novamente de forma mais brusca a tentativa de posicionar o carro novamente à posição centralizada. Esse processo só é possível por causa do Controle Integral. O efeito de redução de oscilação por tempo é trabalho pelo Controle Derivativo.

O Controlador Proporcional Integral-Derivativo, também chamado de Controlador PID, é uma metodologia de cálculo de controle de processo que une várias outras menores. Utiliza de teorias como as Ações Derivativa (Controle PD), Integral (Controle PI) e Proporcional (Controle P).

Estes quatro modos de controle, incluindo PID, são também designados de ações de controle, cada uma delas reagindo de forma distinta ao erro presente nos sistemas. O controle:

- Proporcional: ajusta a variável de controle de forma proporcional ao erro;
- Integral: ajusta a variável de controle baseando-se no tempo em que o erro acontece;
- Derivativo: ajusta a variável de controle tendo como base a taxa de variação do erro; e
- A combinação destes tipos de controle forma o controlador PID.

São amplamente utilizados em situações que se baseiam em controladores eletrônicos chamados “*single-loop*”.

Como ele é formado por todos os outros controles, será descrito brevemente cada um deles.

### 2.3.2 Teoria de Controle Proporcional (P)

Produz um sinal de saída que é proporcional à parâmetro do erro

$$P_{\text{saída}} = K_p e(t) \quad (5)$$

onde

$e(t)$ : Erro.

$K_p$ : Constante relativa à proporção.

Esse método possui a propriedade de eliminar as oscilações do sinal de saída. Para tal, o sistema permanece sempre ligado e o sinal de saída é diferente de zero.

### 2.3.3 Teoria de Controle Proporcional Integral (PI)

Gera um sinal de saída que é proporcional à magnitude e à duração do erro, ou seja, ao erro acumulado. Isso fornece uma alternativa para acelerar a resposta do sistema, permitindo-o chegar ao valor de referência mais rapidamente.

$$I_{\text{saída}} = K_i \int_0^t e(\tau) d\tau \quad (6)$$

onde

$K_i$ : Constante de ganho integral.

$\tau$ : Tempo integral.

A ação integral corrige o valor da variável manipulada em intervalos regulares (representado como  $\tau$ ). Ele é definido como o inverso do ganho integral.

### 2.3.4 Teoria de Controle Proporcional Derivativo (PD)

Resulta num sinal de saída que é proporcional à velocidade de variação do erro.

$$D_{\text{saída}} = K_d \frac{de(t)}{dt} \quad (7)$$

onde

$K_d$ : Constante de ganho derivativo.

Ela fornece uma correção antecipada do erro, diminuindo o tempo de resposta e melhorando a estabilidade do sistema. Atua em intervalos regulares, chamado tempo derivativo.

### 2.3.5 Teoria de Controle Proporcional Integral-Derivativo (PDI)

Assim, fazendo um casamento de todos os controles P, PI, PD, ditos nas Seções 2.3.2, 2.3.3, 2.3.4, temos a seguinte fórmula

$$u(t) = \underbrace{K_p e(t)}_P + \underbrace{K_i \int_0^t e(\tau) d\tau}_{PI} + \underbrace{K_d \frac{de(t)}{dt}}_{PD} \quad (8)$$

onde

$K_p$ : Ganho Proporcional.

$K_i$ : Ganho Integral.

$K_d$ : Ganho Derivativo.

$e$ : Erro.

$t$ : Tempo.

$\tau$ : Tempo integral.

Aplicando a transformada de Laplace na Equação 8, obtemos:

$$L(s) = K_p + \frac{K_i}{s} + K_d s \quad (9)$$

sendo

$s$ : Frequência complexa.

## 3 Elementos do Projeto

### 3.1 Microcontrolador

Microcontrolador utilizado para este trabalho é o NodeMCU. É uma plataforma IoT<sup>1</sup> *open-source*.

Como esperado de um microcontrolador para IoT o sistema possui integrado um componente de comunicação Wi-Fi para troca de dados. Utiliza linguagem de *script* Lua desenvolvida por brasileiros.

Suas especificações são uma CPU ESP8266 possuindo 128 KB de memória, 4 MB de armazenamento e suporta o sistema operacional chamado XTOS. Permite comunicação pelo Wi-Fi e USB onde também é energizada. Possui um total de 10 pinos de entrada e saída de propósito geral (GPIO) onde suportam funções como PWM, comunicação I<sup>2</sup>C e 1-wire. Além da antena Wi-Fi, possui também um conversor USB-TTL para comunicação serial.

Seus microprocessadores podem ser facilmente ligados à um computador e também utilizado a IDE Arduino. A Figura 3 exibe um esquemático de seu protótipo.

---

<sup>1</sup> *Internet of Things*, ou seja, internet das coisas.



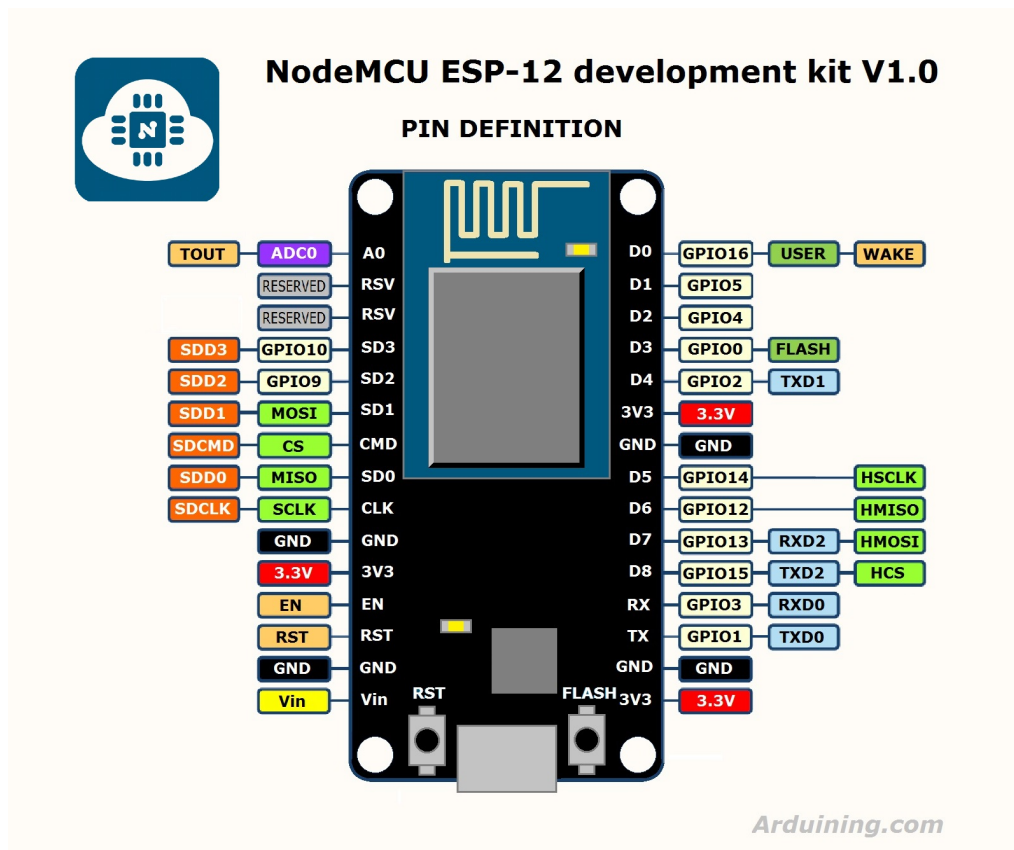


Figura 3: O Microcontrolador NodeMCU e seus Componentes e I/O.

Sua programação pode ser feita por diversas maneiras. As duas principais são utilizando um terminal serial para envio de comandos diretamente à placa usando a porta USB do controlador. Para isso, deve-se configurar a comunicação serial para 9600 *baud-rate*. Ao iniciar o controlador, já iniciará informações de comunicação. Feito isso, já é possível enviar comandos ao controlador. Outra forma é utilizando programas para envio de dados até a placa. Eles realizarão toda a ‘interfaceação’ do desenvolvedor com a placa e um exemplo deles é a IDE Arduino.

Como ela possui placa de comunicação Wi-Fi e um controlador poderoso, é possível também transformá-la num *Web Service* para que realize toda a comunicação sem fio.

### 3.2 Energização

Sua energização será realizada por meio de fonte externa. Isso é necessário pois o sistema completo possui vários componentes que necessitam de uma fonte de energia estável e constante para não afetar no desempenho e também pelo fato do controlador não conseguir fornecer mais que 3.3V e 12mA em suas saídas GPIO. Qualquer valor fora deste intervalo pode gerar em não corretude do sistema, inclusive danificação.

Utilizar uma fonte por meio da interface USB não seria suficiente para energizar todos os atuadores e sensores do sistema justificando assim o uso da fonte externa.

### 3.3 Sensores

Para o carro, será utilizado somente um único tipo de sensor para seu funcionamento, sendo este será o sensor fototransistor. Em sua superfície, existirá um LED que fará a iluminação da área no qual refletirá sobre a superfície e chegará até o sensor fototransistor. Como o seguidor de linha move sobre dois tipos de superfície (branca e preta), é possível identificar quando ele sairá da direção correta. Essa ideia melhor visualizada com a Figura 4.

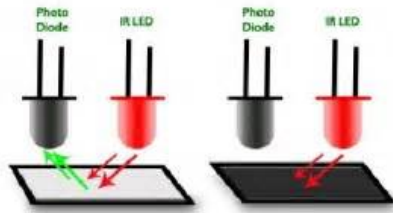


Figura 4: Sensor Fototransistor.

O projeto utilizará não somente dois sensores mas uma série deles, posicionados logo à extremidade da faixa de direção como é exibido na Figura 5.

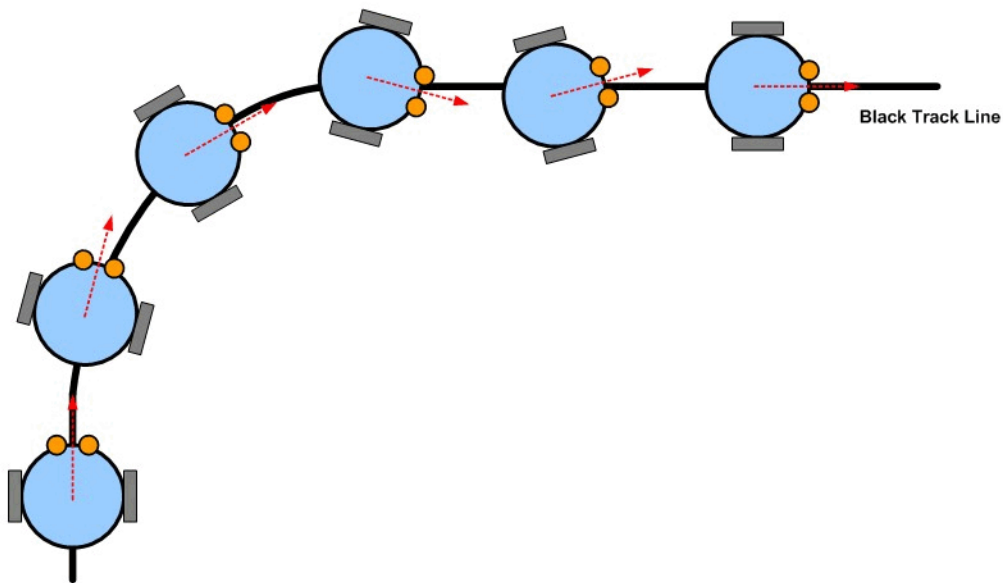


Figura 5: Posição dos Sensores. Exemplo utilizando dois sensores.

Utilizando uma série de sensores, cada sensor será responsável por avaliar se o carro saiu da linha original. Quanto mais à extremidade a faixa estiver em relação à série de sensores, maior será o erro dela e assim, a consequente uma correção mais rápida. Quanto maior a quantidade de arranjo de sensores melhor será a performance do carro pelo motivo que mais sensores representarão melhor o nível do erro. Por este motivo, será utilizado seis sensores, posicionando três em cada extremidade da faixa. Eles serão dispostos o mais próximo possível de seu adjacente pois a resolução

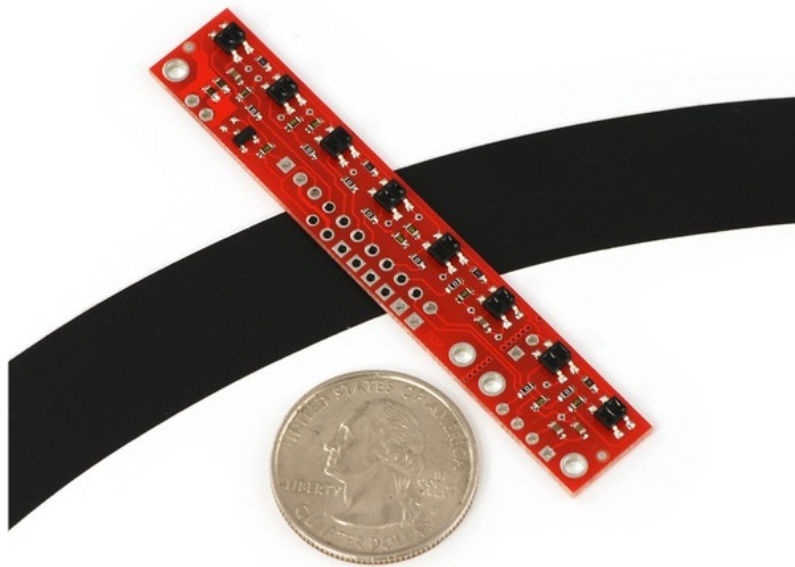


Figura 6: Sensores em série.

de operação será melhor do que distâncias grandes como três centímetros ou mais. Um exemplo com 8 sensores é exibido na Figura 6.

O carro andará numa velocidade máxima fixa em um valor. Quando algum de seus sensores perceber que o carro saiu da linha, a fórmula de correção de direção será acionada para avaliar os dados dos sensores e assim verificar os procedimentos a serem realizados nos atuadores para que o carro volte a operar normalmente na direção correta em cima da faixa. Nesta parte que existe o grande diferencial deste trabalho proposto.

Utilizando controle simples, sem realimentação, é possível perceber que seu erro após à primeiro desvio será grande e sua forma de conserto fará com que ele aumente ainda mais sua margem de erro. isso causará grande distúrbio em seu movimento criando assim um padrão geométrico construído por uma sequência de segmentos lineares alternados quanto à direção, formando linhas quebradas com alternância de ângulos salientes e reentrantes como mostrado na Figura 7.

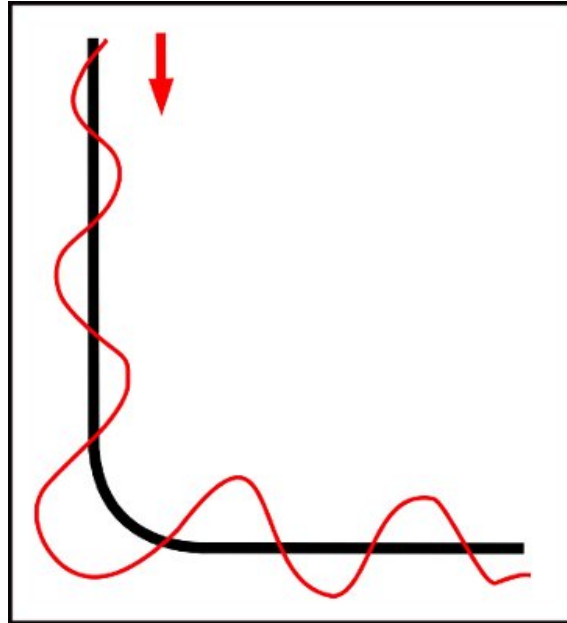


Figura 7: Seguidor de Linha comum.

Esse movimento oscilatório faz com que o carro gaste tempo em completar sua tarefa de percorrer a linha e também energia, item essencial para sistemas embarcados.

Já utilizando o controle proposto é possível que este padrão seja decrementado ao ponto de tornar o movimento do carro mais suave e portanto seu trajeto mais conciso quanto à linha do trajeto. O resultado esperado é exibido na Figura 8.

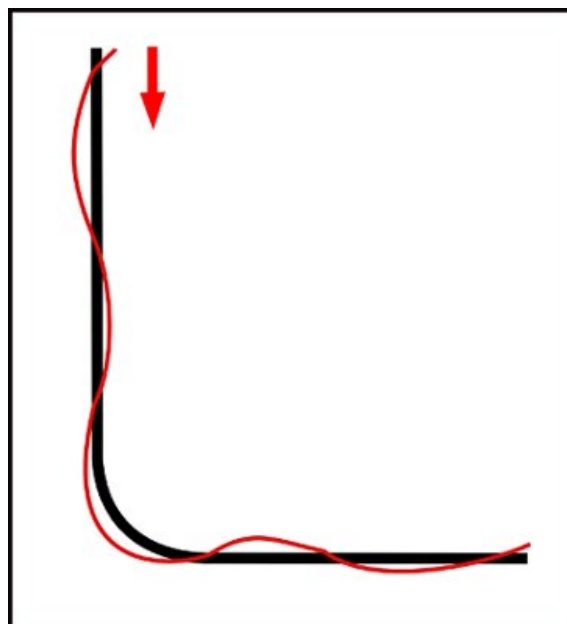


Figura 8: Seguidor de Linha utilizando Controle Proporcional.

Este trajeto faz com que seus movimentos sejam mais suaves, estáveis, ágeis e eficientes em relação ao mostrado na Figura 7.

### 3.4 Atuadores

O sistema possuirá duas unidades de motores atuadores.

Os motores serão do tipo de Corrente-Contínua, propriedade física que não permite o cálculo exato de distância e velocidade do equipamento sem a utilização de sensores de marcação de passo. Para isso, será utilizado vários sensores para captação de dados do ambiente (que no caso será a faixa a ser seguida) e processado para enviar dados de controle aos atuadores para o carro manter o mais centralizado à faixa.

Eles serão acoplados à *board* utilizando um *shield* intermediário entre o microcontrolador e os motores. Inicialmente, serão acionados em uma velocidade máxima *s*. Quando o carro perceber que necessita de uma correção na direção, independente do grau dela, o motor responsável por corrigir o erro será decrementado em sua velocidade, tentando aproximar da direção desejada. Esse processo repetirá enquanto o carro não completar sua meta de trafegar pelo circuito todo.

Tal *shield* realizará o processo de interface de controle e energização dos motores criando assim um sistema estável. Seu nome é L293DD e possui suporte total para interface de pinos NodeMCU. Seu sistema de operação utiliza Ponte-H dupla e com isso é possível controlar até dois motores além de conectores para seleção de interface serial UART, SPI, e entrada analógica, além de conexões para habilitar as opções de *Enable* e *Reset* do microcontrolador. Ele pode ser visto por meio da Figura 9.

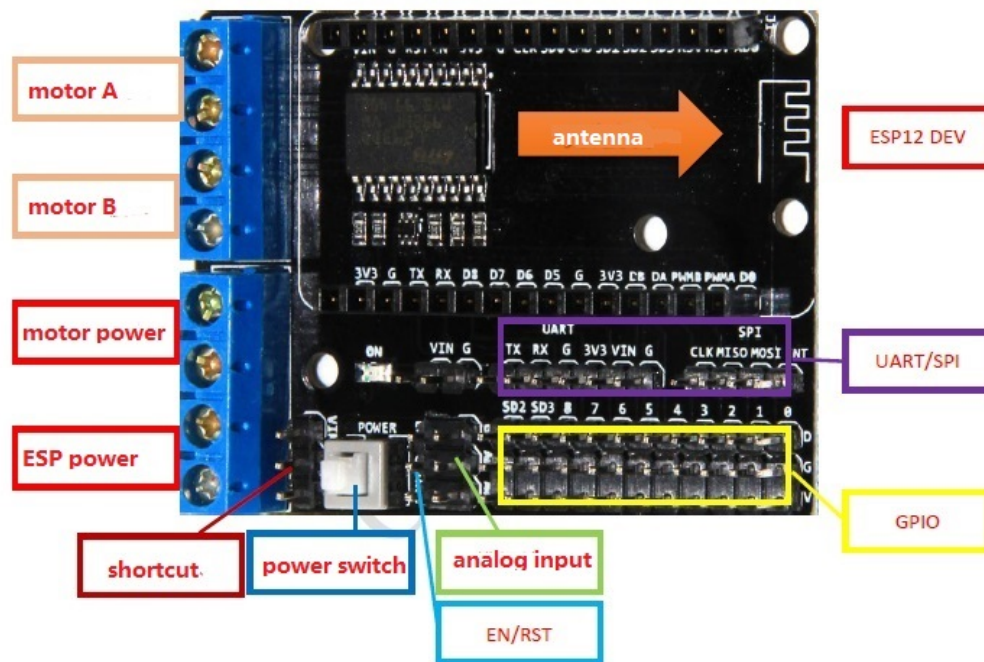


Figura 9: Motor *Shield* a ser Utilizado.

## 4 Implementação

Aqui será descrito uma representação de como será a implementação de toda a teoria descrita neste trabalho. Assim, de forma geral, teremos o seguinte algoritmo

implementado para tratamento dos valores obtidos dos sensores e ajuste da rotações dos motores.

---

**Algorithm 1** Controle Proporcional de Correção de Angulação

---

```

1: procedure PROPORTIONAL_CONTROL( $position, set\_point, \Delta, \int, last\_ \Delta, y', \Phi$ )
2:    $\Delta \leftarrow position - set\_point$ 
3:    $\int \leftarrow \int + \Delta$ 
4:    $y' \leftarrow \Delta - last\_ \Delta$ 
5:    $\Phi \leftarrow \Delta * K_p + \int * K_i + y' * K_d$ 

```

---

onde:

***position***: Posição atual do carro.

***set\_point***: Ponto central onde encontra-se a faixa.

**$\Delta$** : Representa o erro proporcional da posição atual contra o ponto central da faixa.

**$\int$** : Representa o acúmulo de erro atual.

**$last\_ \Delta$** : Representa o erro proporcional da posição da iteração passada.

**$y'$** : Retêm a diferença entre os erros atual e da iteração anterior.

**$\Phi$** : Correção a ser realizada nesta iteração.

Como é de se esperar, a correção proporcional  $\Phi$  será aplicada à velocidade motor para conserto de posição do carro.