

# IMPLEMENTAÇÃO DE UM ROBÔ

Danilo Sulino S. Pinto, Karina Rocha Gomes da Silva

Universidade Federal de Goiás, Escola de Engenharia Elétrica, Mecânica e Computação, Goiânia – Goiás,  
sulino@hotmai.com, karinarg@gmail.com

**Resumo** - Este artigo tem o objetivo de mostrar a implementação de um robô que pode ser usado, por exemplo, para inspecionar tubulações. Mostrando os passos de desenvolvimento abordando o uso do microcontrolador, o controle dos motores de corrente contínua, a alimentação e o uso de sensores. Apesar de complexo, o desenvolvimento de robôs pode ajudar bastante a encontrar e solucionar diversos tipos de problemas para os seres humanos.

**Palavras-Chave** - Implementação, Inspeção, Robô, Sensores.

## ROBOT IMPLEMENTATION

**Abstract** - The objective of this document is to show a robot implementation that can be used, for example, to inspect pipes. It shows the development steps explaining about the use of microcontrollers, sensors and batteries, and how to control continuous current motors. Despite the fact of being complex, the robot development can really help to find and solve many kinds of human being problems.

**Keywords** - Implementation, Inspection, Robot, Sensors.

## INTRODUÇÃO

Segundo reportagem do Jornal Folha de São Paulo [1], no Brasil, em meio a uma das mais graves crises de abastecimento, cerca de 37% de toda a água tratada para consumo é desperdiçada antes de chegar às torneiras do consumidor, sendo as falhas nas tubulações a principal causa para esse desperdício. O documento mostra também uma estimativa de que, em 2010, os custos dessa água perdida chegaram a impressionantes R\$1,3 bilhão.

Existe, segundo a SANEAGO [2], mais de 2,6 milhões de metros de rede coletora de esgoto apenas na região de Goiânia e, como não há instrumentação para verificação dessas tubulações, a qualidade e a conservação delas são desconhecidas, o que permite grande quantidade de vazamentos e irregularidades, assim como acontece nas tubulações de água tratada.

Uma das formas de detecção desses defeitos seria através de um equipamento, como, por exemplo, um robô de inspeção, capaz de entrar nas tubulações e automaticamente

perceber essas falhas no cano e reportar isso para o inspetor ou permitir por meio de câmeras que o operador identifique essas falhas.

Existem alguns modelos de robôs de inspeção disponíveis no mercado e todos possuem uma câmera de vídeo para que um técnico faça a verificação da existência ou não de falhas. É válido ressaltar que, devido ao alto custo de um robô de inspeção já disponível no mercado que inviabiliza a compra de tais equipamentos por várias empresas.

Outros tipos de problemas reais e ineficiências em processos produtivos podem ser resolvidos com o uso da robótica, como, por exemplo, o desarme de bombas e o processo de paletização de caixas.

A implementação de um robô é então um passo muito importante na obtenção do conhecimento necessário sobre o assunto para o desenvolvimento de uma solução mais economicamente viável.

Com objetivo de mostrar os detalhes de como implementar um robô, na seção II encontra-se o detalhado todos os itens utilizados e na seção III está a montagem e implementação do robô. Alguns exemplos de aplicações reais de robôs desenvolvidos no Brasil estão apresentados na seção IV. E as conclusões se encontram na seção V.

## IMPLEMENTAÇÃO

Um robô é composto pela parte mecânica, que inclui o chassi ou plataforma, os motores e as rodas. Com esses três componentes é possível montar uma estrutura sobre a qual é possível apoiar os componentes eletrônicos e baterias, e fazer as ligações para que o robô possa se locomover.

### A. Plataforma

Foi utilizado o chassi de um robô seguidor de linha da EagleMoss, mostrado nas Figuras 1 e 2. Essa plataforma possui duas rodas tracionadas, com motores de corrente contínua de 5v, e uma roda de giro livre, popularmente conhecida como “roda boba”. Essa é uma configuração bastante comum utilizada em robótica, por ser uma construção relativamente simples, ao se comparar com outras construções que permitem mudar a direção do movimento. Como, por exemplo, o modelo de um carro que possui um único motor e um volante, que muda a direção das rodas dianteiras.

Ter uma roda tracionada de cada lado permite que o robô seja capaz de andar para frente e para trás, quando os dois motores tem velocidades iguais, e permite também fazer curvas, quando um lado está mais rápido que o outro.

Ao escolher essa plataforma simples, a vantagem é a facilidade de encontrar no mercado, porém ela só permite utilizar o robô em ambientes secos.



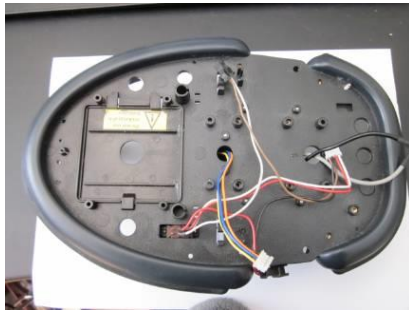


Fig. 1. Vista superior do chassi Eaglemoss utilizado no robô



Fig. 2. Vista inferior do chassi Eaglemoss utilizado no robô

### B. Controlador

Para fazer o controle do robô foi utilizada uma plataforma eletrônica da família Arduino, o Arduino Uno [3], mostrado na Figura 3, uma plataforma de desenvolvimento, que possui porta USB para conexão com um computador e possui um ambiente de codificação próprio com linguagem de programação bem parecido com a linguagem C. Esse ambiente de programação já faz a compilação do código para o formato a ser gravado e o upload no microcontrolador da Atmel de 16MHz que ele possui.

Com o Arduino é possível controlar e fazer leitura de sensores, controlar o movimento do robô e fazer alguns tipos de cálculos ou rotinas necessárias para a aplicação.

O microcontrolador foi utilizado para acionar e controlar a velocidade dos motores e fazer a leitura dos sensores de distância e do acelerômetro.

Para fazer leituras de um sensor basta conectar seus pinos de sinal nas portas I/O do Arduino, e criar uma rotina de leitura específica de cada sensor [4].

Para controlar os motores é necessário o uso de uma ponte H e da modulação de pulso PWM, para controle da velocidade do motor. Essas informações estão mais detalhadamente descritas na seção Controle dos motores.

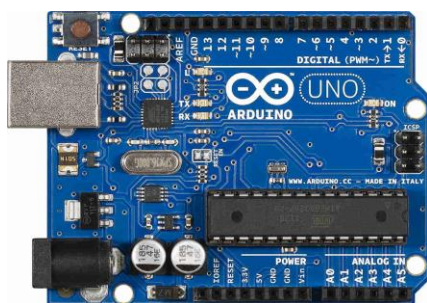


Fig. 3. Arduino UNO

### C. Acionamento dos motores

Para controlar motores não é possível apenas ligar o fio do motor em um pino do Arduino mesmo que o robô trabalhe com 5v, que é a tensão dos pinos. Um motor drena uma corrente muito maior do que qualquer microcontrolador consegue fornecer. Dependendo do motor, ele pode consumir de 60mA até mais de 1A para poder funcionar. E a porta do microcontrolador consegue fornecer no máximo 40mA, se tentar drenar mais corrente a porta queima.

Para controlar os motores é necessário o uso de uma ponte H. Esse circuito é composto por transistores, funcionando como chaves, que se acionados na base por um sinal digital do microcontrolador da maneira correta permite a passagem de corrente por eles, e em consequência a passagem de corrente elétrica pelos motores.

O esquema do circuito de uma ponte H, ilustrando os transistores como se fossem chaves, está mostrado na Figura 4. Em 'a' todos os transistores estão abertos logo o motor está parado. Em 'b' dois transistores permitiram a passagem de corrente pelo motor, da esquerda para a direita, fazendo o motor rotacionar para um lado. Em 'c' acontece o inverso de 'b' e logo o motor é acionado também no sentido contrário.

Pela Figura 4, é possível perceber que o nome ponte H veio do formato do circuito, onde os quatro transistores formam as duas linhas verticais da letra H, e o motor a linha horizontal.

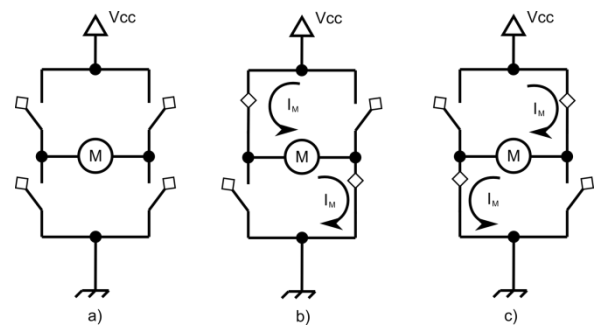


Fig. 4. Esquema do circuito de uma ponte H

Ao invés de montar uma ponte H com transistores foi utilizado um módulo ponte H L298N, que possui duas pontes H capaz então de controlar 2 motores, e já é montada com os conectores de forma que fica fácil de conectá-la ao microcontrolador e à fonte de alimentação. Esse módulo é facilmente encontrado no mercado e tem preço baixo. O módulo está mostrado na Figura 5.

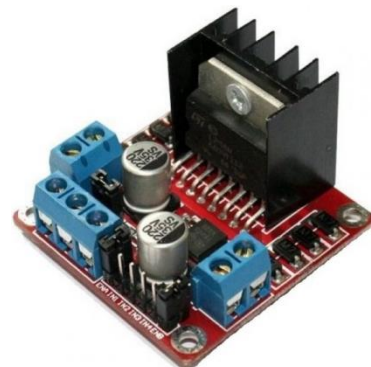


Fig. 5. Módulo Ponte H L298N

#### D. Controle de velocidade dos motores

Para controlar a velocidade dos motores é utilizada a modulação PWM (sigla em inglês para Modulação por Largura de Pulso). Com essa modulação é possível obter um resultado analógico com um sinal digital.

O controle digital é utilizado para formar uma onda quadrada ao trocar o sinal digital entre ligado e desligado de uma maneira rápida e coordenada. A Figura 6 ilustra essa modulação. A troca do sinal digital de alto para baixo é feito dentro de um mesmo ciclo de clock do microcontrolador, mostrando o quão rápida é essa troca. Quanto maior o tempo de sinal alto dentro de um ciclo, maior será a tensão resultante percebida pela carga.

Na Figura 6, podemos ver, por exemplo, que quando o *Duty Cycle* (a porcentagem do período na qual o sinal permanece ativo) é de 50%, vemos que a linha tracejada, a qual corresponde a tensão resultante percebida pela carga, está na metade, mostrando que no caso, os motores estariam recebendo metade da tensão da fonte.

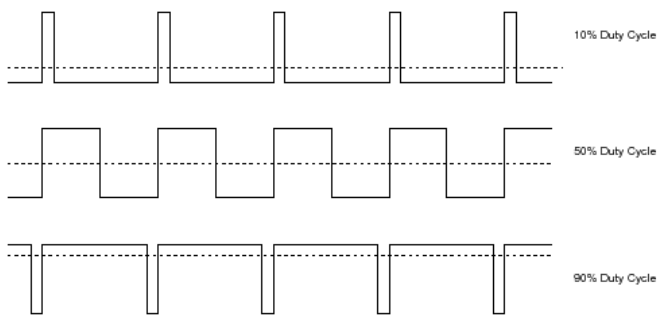


Fig. 6. Modulação PWM

#### E. Alimentação

Para que o robô possa funcionar é necessária uma fonte de alimentação para fornecer energia ao Arduino, aos motores e a ponte H. Apesar de o Arduino trabalhar com 5v, sua alimentação deve ser entre 6 e 20v [3], e ele possui um regulador interno para gerar os 5v.

Para que tudo funcione corretamente é preciso não só ligar a alimentação, mas que o gnd do Arduino e do módulo ponte H estejam conectados, para que tenham a mesma referência e assim os sinais digitais com PWM do Arduino possam ser corretamente entendidos pela ponte H.

Foram utilizadas 4 pilhas AA, que fornecem cada uma 1,5v, gerando no total 6v, que alimentaram tanto o Arduino como a ponte H.

#### F. Sensor de distância

Para tentar reconhecer buracos na tubulação, foi utilizado um sensor de distância, de modo que se passar por um buraco a distância entre o sensor e a superfície aumentará, sendo assim reconhecido um buraco. O sensor de distância utilizado foi o sensor SHARP GP2Y0A51SK0F, mostrado na Figura 7, um sensor de saída analógica que utiliza um LED emissor de luz infravermelha e um receptor sensível a posição. O modelo utilizado faz leituras de 2 até 15cm.

O sensor utiliza o método de triangulação para fazer a medida da distância. Basicamente ele emite a luz infravermelha e o ângulo de reflexão na superfície muda e de acordo com o local que o sensor percebe a luz recebida

um valor de tensão que varia de 0,3v até 2,5v é emitido como sinal.



Fig. 7. Sensor de distância SHARP

#### G. Acelerômetro

Um sensor que mede a angulação do robô, também foi usado, para poder verificar qual a posição do robô na tubulação para poder fazer um controle tentando manter o robô sempre na horizontal, permitindo melhor precisão na medida do sensor de distância e para que o robô não capote.

O sensor utilizado foi o módulo MPU6050 [5], mostrado na Figura 8, um modelo de baixo custo facilmente encontrado no mercado. Esse módulo possui um acelerômetro e um giroscópio, e já vem com um hardware que contém um conversor analógico para digital de 16 bits, conferindo a ele boa precisão.

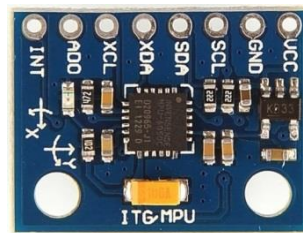


Fig. 8. Acelerômetro

### IMPLEMENTAÇÃO DO ROBÔ

Para implementar o robô, o primeiro passo é ter todos os componentes e fios ou jumpers para poder fazer as conexões. O segundo passo é fixar todos os elementos na plataforma. Os motores e rodas serem fixados é imprescindível. Apesar de o robô funcionar sem as outras partes estarem fixadas, não é recomendado, pois os sensores podem acumular erros de leitura, e pode haver erros por mau contato dos fios durante a movimentação do robô. E se os fios e conectores não estiverem em perfeito estado de conservação, pode haver mau contato mesmo com todos os componentes fixados.

Utilizou-se 4 baterias AA em série resultando em uma alimentação em 6v. Para alimentar o microcontrolador basta ligar o positivo das baterias na entrada Vin, e o negativo no pino gnd. É necessário também fazer a mesma conexão no módulo ponte H.

Estando os componentes alimentados os motores devem ser conectados nas saídas do módulo ponte H, e deve ser observada a ordem de ligar os dois motores para que ambos rodem no mesmo sentido quando receber mesmo sinal. Para

cada ponte H é necessário dois sinais de PWM, um para controlar os dois transistores, conforme Figura 4(b) e outro para controlar os outros dois transistores, conforme Figura 4(c). Se forem acionados os 4 transistores ao mesmo tempo o motor não irá rodar e pode ser que gere uma corrente muito alta por a corrente fluir direto pelos transistores até a referência.

O sensor de distância e o acelerômetro precisam de alimentação em 5v, e como precisam de pouca corrente, podem ser alimentados diretamente com os pinos do Arduino.

O sensor de distância possui sinal analógico transmitido por um único fio. Esse sinal varia de 0,3v até 2,5v o que corresponde a distâncias de 15 até 2 cm, respectivamente. Basta então receber esse sinal com uma entrada analógica do Arduino e interpretar, no tempo, se a distância corresponde a distância normal da superfície ou a um buraco. Se o sinal for conectado a um pino unicamente digital do Arduino, o valor da leitura simplesmente não será feita com sucesso, podendo levar a pensar em mal funcionamento dos sensores.

Esse reconhecimento é relativamente simples apenas no caso de buracos grandes e profundos, de tal modo que a variação na leitura do sensor seja muito grande, maior do que a imprecisão de leitura, que chega a 11%. Também é importante ressaltar que o sensor de distância faz a leitura de forma pontual, portanto para verificar toda a circunferência da tubulação se faz necessário o uso de vários sensores.

O sensor de proximidade infravermelho já foi testado para o fim de inspeção de falhas [6], porém quando o robô se desvia um pouco da trajetória retilínea dentro da tubulação, o sensor acusaria como uma falha, já que o ajuste de distância é manual.

O MPU6050 faz a comunicação via I2C, (sigla em inglês para Circuito Inter-Integrado), um barramento serial multi-mestre e multi-escravo desenvolvido pela Philips [7], que necessita de 2 fios para realizar a comunicação. O sensor sempre se comporta como um escravo perante o Arduino com os pinos de SDA e SCL conectados ao barramento I2C. Se não seguir corretamente todos os passos e detalhes do protocolo, pode haver erros na recepção dos dados, e como os dados não são passados no formato que entendemos, que no caso é em graus, é muito difícil encontrar tardiamente onde está o problema.

É então possível fazer a leitura dos valores crus lidos pelo sensor, fazer os ajustes e conversões e então obter a angulação do robô. Dessa forma é possível, então, fazer a programação do controle do robô para que ele se mantenha sempre com o chassi sempre na horizontal.

Esse controle não é simples de se fazer, mas um exemplo pode ser usando o Q-Learning, um método de controle baseado no aprendizado por reforço ou repetição (Reinforcement Learning). Nesse método faz-se a programação de algumas regras e objetivos para que o robô possa realizar as ações, pelo sensor obter a angulação resultante da ação e pelas regras receber ou não uma recompensa e então considerar se aquela ação é melhor ou pior que outra ação para a mesma situação. Repetindo essas ações muitas vezes, ele chega a um ponto que seleciona a melhor combinação de ações para que ele se mantenha no objetivo ou que chegue até o objetivo.

Para fazer uma vídeo inspeção, o desenvolvimento é um pouco mais complexo, pois além de todos os passos já informados é necessário instalar uma câmera e fazer a transmissão do vídeo em tempo real para que um operador possa verificar a existência de falhas. É necessário também que haja conexão com o microcontrolador para que o operador controle o robô à distância. Podendo assim percorrer o tubo na velocidade que permita uma boa inspeção, que é de no máximo 15cm/s [8], e que permita o operador parar o robô para verificar algum ponto de dúvida.

Toda essa construção pode ser feita através de um cabo, que pode inclusive servir para alimentar o robô para que o mesmo não dependa de sua bateria interna permitindo maior autonomia. Porém várias são as dificuldades: a primeira é o uso de um cabo longo para que o robô percorra boa extensão da tubulação, segundo o guia do profissional da Secretaria Nacional de Saneamento Ambiental [9], é necessário haver ponto de acesso a uma tubulação de esgoto a cada 100m; a transmissão de vídeo digital em longas distâncias tem grandes atenuações e susceptibilidade a ruídos; Há queda de tensão ao utilizar cabos extensos; para fazer comunicação com o microcontrolador em longas distâncias é necessário um protocolo de comunicação mais apropriado, pois o USB, por exemplo é aconselhado a não ultrapassar 5m; Com tantas vias dentro de um mesmo cabo, para sinal de câmera, alimentação e comunicação, uma via pode gerar interferência em outra; E por último, um ponto muito importante a ser considerado é que o cabo extenso e com muitas vias pode ser muito pesado, podendo chegar a um ponto no qual o robô, apesar de usar um motor potente, não tenha peso o suficiente para ter atrito com o chão e conseguir puxar esse cabo.

## EXEMPLO DE APLICAÇÕES

É possível citar vários exemplos de robôs, que possuem completo ou parte do seu desenvolvimento no Brasil. Temos robôs de inspeção de tubulações, mostrado na Figura 9, desenvolvidos pela empresa RYD Engenharia em Goiânia.



Fig. 9. Robô de inspeção de tubulações VX1-300 (RYD)

Podemos citar também o robô paletizador, desenvolvido pela empresa Redutep Soluções Industriais, mostrado na Figura 10, que por não possuir rodas e não ter formato humanoide é citado muitas vezes como um braço robótico.



Fig. 10. Robô paletizador (Redutep)

Temos também um robô para uso policial e militar para, por exemplo, desarmamento de bombas, como o robô batizado de CAIPORA, mostrado na Figura 11.



Fig. 11. Robô CAIPORA

## CONCLUSÕES

Com o trabalho pode-se concluir que a implementação de um robô, por mais simples que seja a aplicação, envolve vários conhecimentos, desde eletrônica até programação. A quantidade de detalhes e especificações a serem consideradas torna ainda mais difícil a implementação, pois se não corretamente cumpridas todas as etapas e conferidos todos os detalhes o produto final pode simplesmente não funcionar e o diagnóstico do erro pode não ser tão simples quanto seria feito durante o desenvolvimento.

Com um conhecimento mais abrangente e atenção aos detalhes é possível construir robôs que possam eficientemente ser usados para ajudar o ser humano, a reconhecer e solucionar vários tipos de problemas, como mostrou os exemplos.

## AGRADECIMENTOS

Este projeto foi financiado pela FAPEG por meio de bolsa ao pesquisador.

## REFERÊNCIAS

[1] F. Lobel, “Folha de São Paulo,” 21 Janeiro 2015. [Online]. Available: <http://www1.folha.uol.com.br/>

cotidiano/2015/01/1578007-brasil-desperdica-37-da-agua-tratada-aponta-relatorio-do-governo-federal.shtml [Acesso em 31 Março 2015].

- [2] SANEAGO, “SANEAGO,” Agosto 2004. [Online]. Available: <http://www.saneago.com.br/site/?id=esgoto6&tit=esgoto>. [Acesso em 31 Março 2015].
- [3] Arduino, “Arduino UNO,” [Online]. Available: <https://www.arduino.cc/en/Main/ArduinoBoardUno>. [Acesso em 2 February 2016].
- [4] M. McRoberts, Arduino básico, São Paulo: Novatec Editora, 2011.
- [5] Arduino, “MPU-6050 Accelerometer + Gyro,” Arduino, [Online]. Available: <http://playground.arduino.cc/Main/MPU-6050>. [Acesso em 2 February 2016].
- [6] D. S. S. Pinto, K. R. G. Silva e G. S. Mazzoccante, “ESTUDO DO SENSOR DE PROXIMIDADE INFRAVERMELHO PARA PEQUENAS DISTÂNCIAS E DESCONTINUIDADES EM SUPERFÍCIES PLANAS,” em XIII CEEL, Uberlândia, 2015.
- [7] NXP Semiconductors, *I2C-bus specification and user manual*, 2014.
- [8] L. Zheng e Y. Kleiner, “State of the art review of inspection technologies for condition assessment,” *Measurement*, vol. 46, n. 1, pp. 1-5, 2013.
- [9] Secretaria Nacional de Saneamento Ambiental, “Esgotamento Sanitário: Operação e Manutenção de Redes Coletoras de Esgoto: guia profissional em treinamento,” Brasília, 2008.

## DADOS BIOGRÁFICOS

**Danilo Sulino Silveira Pinto**, nascido em 21/12/1990 em Goiânia-GO, é engenheiro eletricista (2014) e mestrando em engenharia elétrica pela Universidade Federal de Goiás, com bolsa da FAPEG. Foi co-fundador e primeiro presidente do Núcleo de Robótica Pequi Mecânico da UFG. 2012-2013 bolsista em intercâmbio acadêmico internacional na State University of New York at Buffalo.

**Karina Rocha Gomes da Silva**, nascida em Jaú - SP, possui graduação em Ciência da Computação pela Universidade Federal de Goiás (1999), mestrado em Informática pela Universidade Federal da Paraíba (2001) e doutorado em Engenharia Elétrica pela Universidade Federal de Campina Grande (2007). Atualmente é professora Adjunta IV da Universidade Federal de Goiás. Tem experiência na área de Engenharia de Computação, com ênfase em Sistemas Embarcados. Suas áreas de interesse são: Verificação funcional, desenvolvimento de IP-Cores, FPGAs e Sistemas embarcados.