

Veículo Guiado Automaticamente (AGV)

Igor Gonçalves e Souza – igor-ges@hotmail.com (UNICEP)
Claudio Roberto di Vincenzi – c.vincenzi@gmail.com (UNICEP)

Resumo

Palavra “robot”, que foi pronunciada pela primeira vez em um teatro, que tem como significado “trabalho forçado”, vem surgindo e dominando, principalmente no ramo educacional e tecnológico. O aumento significativo do uso da robótica acabou-se gerando também uma dificuldade no controle do desenvolvimento desses sistemas, criando a necessidade de especialistas para a solução desta dificuldade. Com decorrer dos anos, os robôs AGV’s vem ganhando espaço, principalmente na parte industrial mas, no Brasil a parte de pesquisas ainda estão limitadas. A empresa Xbot, necessitava da implantação do Arduino no robô curumim. Foi feita uma parceria entre o Centro Universitário Central Paulista (UNICEP) e a Xbot para o desenvolvimento do Veículo Guiado Automaticamente (AGV). Com a motivação dos estudos sobre a robótica, foi criado um robô AGV que tenha a autonomia de seguir pessoas ou objetos que são postos a sua frente, sem a necessidade do uso de um controle, ou seja, mestre e escravo. Lógicas e circuitos foram elaborados para que fosse possível a criação do robô autônomo, ou seja, um veículo guiado automaticamente (AGV).

Palavras-chave: Arduino, AGV, Sensor ultrasonico.

1. Introdução

O uso da robótica em diversos ambientes não estruturados, mostra vários desafios que são estudados por pesquisadores, atualmente. Diferente de indústrias e laboratórios, estes ambientes que não são estruturados apresentam uma constante e rápida mudança no seu ambiente e presença humana que, dependendo da aplicação, não tem o conhecimento suficiente para controlar/usar estes sistemas automatizados.

Geralmente, robôs AGV’s são encontrados em indústrias, sistema mestre e escravo ou robôs que seguem linhas para executar sua tarefa, mas, AGV’s também podem ser utilizados em hospitais. O custo de uma implementação dessa é alta, devido aos sensores utilizados e a complexidade da implementação das funcionalidades.

Hoje temos vários exemplos, desde o robô “hospi” até o “RIBA II”, robôs enfermeiros que dão assistência aos pacientes e aos próprios enfermeiros ou médicos, facilitando a locomoção e eliminando esforço humano.

O grande aumento do uso da robótica e a dificuldade nos controles dos sistemas robóticos, acabou criando grandes necessidades de soluções mais robustas e eficientes na área. Existem muitos modelos de desenvolvimento relacionados a AGV mas a solução de problemas ou solução para aplicações nesta área continua muito ampla, com vários desafios. A robótica no Brasil é ainda vista como uma ficção, pois está mais desenvolvida na área de pesquisa e na parte industrial é limitada, muito restrita ou seja, estes projetos acabam não saindo do papel. Apesar da robótica ser restrita, a automação e robôs industriais têm uma grande demanda. Segundo dados da Federação Internacional de Robótica, em 2005 no Brasil, haviam 2.600 unidades de robôs industriais.

Conseqüentemente, com o aumento da utilização de robôs AGV's nas indústrias, foram criados também robôs que atuam em diversas áreas, uma delas é a saúde, que vem chamando atenção, pelo fato de conseguirem fazer membros mecânicos, robôs que operam e a utilização de um AGV como o robô enfermeiro, com finalidade de auxiliar médicos, enfemeiros e até mesmo pacientes.

No ramo da robótica, a empresa Xbot, fundada em 2007 na cidade de São Carlos (SP), pioneira no ramo, tem como objetivo as soluções educacionais, utilizando a metodologia de ensino baseado na robótica, para que seja possível o aprendizado técnico. A empresa Xbot, necessitava da implantação do Arduino no robô curumim. Foi feita uma parceria entre o Centro Universitário Central Paulista (UNICEP) e a Xbot para o desenvolvimento do Veículo Guiado Automaticamente (AGV).

Este projeto consiste em reprogramar e remontar o robô curumim da Empresa Xbot, para a implementação do arduino e sensores ultrasonicos. O robô, denominado como um AGV (Veículos guiados automaticamente) tem a finalidade de seguir as pessoas sem o uso de controles, cabos, etc.

2. Revisão Bibliográfica

2.1 Robôs Móveis

Segundo Secchi (2008) “Aplicações da robótica estavam centradas nos setores manufatureiros mais desenvolvidos para a produção em massa: indústria automobilística, indústria química etc.

No princípio dos anos sessenta foi introduzido na indústria, de modo significativo, os robôs manipuladores. Essa proliferação, motivada pela ampla gama de possibilidades que oferecia, levantou o interesse dos pesquisadores para conseguir manipuladores mais rápidos, precisos e fáceis de programar. A consequência direta desse avanço originou um novo passo na automação industrial, que tornou mais flexível a produção com o nascimento da noção de célula de fabricação robotizada.

O trabalho desenvolvido pelos robôs manipuladores consiste frequentemente em tarefas repetitivas, como a alimentação das diferentes máquinas componentes da célula de fabricação robotizada. Para isso, é necessário situar as máquinas no interior de uma área acessível para o manipulador, caracterizada pela máxima extensão de suas articulações, o que poderia ser impossível na medida em que as células sofriam progressivas ampliações.

Uma solução para esse problema foi o desenvolvimento de um veículo móvel sobre trilhos para proporcionar um transporte eficaz dos materiais entre as diferentes zonas da cadeia de produção. Dessa forma, aparecem nos anos 80 os primeiros Veículos Guiados Automaticamente (AGV).

Uma melhoria com respeito à sua concepção inicial se baseia na substituição dos trilhos como referência guia na navegação pelos cabos enterrados, reduzindo, com ele, os custos de instalação.

A possibilidade de estruturar o ambiente industrial permite a navegação de veículos com capacidades sensoriais e de raciocínio mínimas. Desse modo, a tarefa se ordena em uma sequência de ações que, ao serem concluídas, o veículo supõe que alcançou o objetivo para o qual está programado.

Diante de qualquer mudança inesperada na área de trabalho que afete o desenvolvimento normal da navegação, o sistema de navegação do veículo se encontrará impossibilitado para executar ações alternativas que o permitam retomar sua atividade. Por outro lado, por suas potenciais aplicações fora do âmbito industrial, onde é caro ou impossível estruturar o entorno, dotou-se os veículos de um maior grau de inteligência e percepção no intuito de obter um veículo de propósito geral apto para desenvolver-se em qualquer classe de ambiente. Assim, nos anos 90, surgem os robôs móveis. (Idem, 2008).

Uma definição correta de robô móvel propõe um conhecimento incerto, mediante a interpretação da informação fornecida através de seus sensores e do estado atual do veículo.

Os robôs móveis são dispositivos de transporte automático, ou seja, são plataformas mecânicas dotadas de um sistema de locomoção capazes de navegar através de um determinado ambiente de trabalho, dotados de certo nível de autonomia para sua locomoção, portando cargas. Suas aplicações podem ser muito variadas e estão sempre relacionadas com tarefas que normalmente são arriscadas ou nocivas para a saúde humana, em áreas como a agricultura, no transporte de cargas perigosas ou em tarefas de exploração solitárias ou cooperativas junto a outros veículos não tripulados.

Exemplos clássicos são o traslado e coleta de materiais, as tarefas de manutenção de reatores nucleares, a manipulação de materiais explosivos, a exploração subterrânea etc.

É necessário esclarecer que o conceito de autonomia não se relaciona apenas com questões energéticas, mas também se refere à capacidade de perceber, modelar, planejar e atuar para alcançar determinados objetivos, sem a intervenção (ou com uma intervenção muito pequena) do operador humano, já que o robô pode se locomover em ambientes estruturados ou não estruturados, total ou parcialmente conhecidos.

O papel deste deve ser desempenhado pelo próprio sistema de controle do veículo, que o deve suprir com a inteligência necessária para fazê-lo mover corretamente.

A denominação do robô móvel faz referência a essa capacidade para alcançar um ou vários objetivos com uma intervenção muito pequena de supervisores humanos. Por outro lado, a denominação de veículo auto-guiado faz referência às estruturas móveis que só se limitam a seguir caminhos preestabelecidos (linhas pintadas no chão, bandas magnéticas, bandas refletoras). ” (Secchi, 2008).

Um grande exemplo de um robô AGV na medicina é o Robô enfermeiro Hospi desenvolvido pela empresa Matsushita. (Figura 1)

A função dele é levar informações aos quartos dos pacientes e também transportar medicamentos, sem ter necessidade de que o enfermeiro/medico exerça algum esforço físico e tenha a atenção mais voltada para a resolução do caso em si.



Figura 01 – Robô Hospi

Fonte: <http://tecnogeek.com.br/>

2.2 Arduino

O arduino surgiu em 2005 na Itália, pelo professor Massimo Banzi, que tinha a intenção de ensinar eletrônica e programação de computadores para os seus alunos que cursavam design afim de que usassem nos seus projetos. Para ensinar electrónica e programação para pessoas que não são da área era difícil e na época também era difícil achar placas poderosas com baixo custo. Pensando nisso, Massimo e Dave Cuartielles decidiram criar sua própria placa, David Mellis e Nicholas Zambetti, ficaram responsáveis pela parte de programação do arduino, surgindo assim a grande febre mundial da eletrônica. (novatec.com.br e destacom.ufms.br)

O nome arduino, de origem germânica é composto pelas palavras hard+win, formam hardwin que na tradução quer dizer grande amigo mas os latinos o chamaram de Ardivinus e depois para o italiano Arduino. É um simples projeto com eletrônica e programação embarcada de alto nível. A grande vantagem do arduino é possuir portas digitais, analógicas e pwm. Possui um atmel AVR(ATMega168), oferece um IDE e bibliotecas de programação de alto nível. O arduino é um projeto open source hardware e software.(labdegaragem.com.br)

2.3 O que é arduino? Porque usar arduino?

O Arduino é uma placa de controle I/O baseada no microcontrolador Atmega (Atmel), que serve de controle para diversos outros sistemas, porém o diferencial desta placa é que ela é desenvolvida e aperfeiçoada por uma comunidade que divulga as suas placas e seus códigos de aplicação, pois a concepção da placa é open-source.

Vantagens de se utilizar o arduino:

1. Baixo custo
2. Software para várias plataformas - Microsoft Windows, Mac OS X e Linux.
3. Linguagem simples.

4. Software livre
5. Existe uma comunidade ativa para usuários.

Existem várias versões do arduino, que são: UNO, MEGA 2560, LEONARDO, YÚN, TER, MICRO, ROBOT, ESPLORA, MEGA ADK, ETHERNET, MINI, PRO MINI, NANO, PRO, FIO. (destacom.ufms.br/)

2.4 Arduino Mega 2560

Possui uma considerável quantidade de portas, o que viabiliza a implementação de projetos mais complexos garantindo a eficiência e o baixo custo. Um bom exemplo para essa implementação são os shields, plaquinhas encaixadas em cima do arduino para aumentar a sua funcionalidade, o Ethernet Shield por exemplo conecta o arduino a uma rede Ethernet.

O Arduino Mega 2560 (Figura 02) é uma placa de microcontrolador baseada no ATmega2560. Ele possui 54 pinos de entradas/saídas digitais, 15 portas PWM, 16 entradas analógicas, 4 UARTs (portas seriais), um oscilador de cristal de 16 MHz, uma conexão USB, uma entrada de alimentação, e um botão de reset. Ele contém tudo o que é necessário para dar suporte ao microcontrolador, só é preciso conectar a um computador com um cabo USB ou a uma fonte de alimentação.

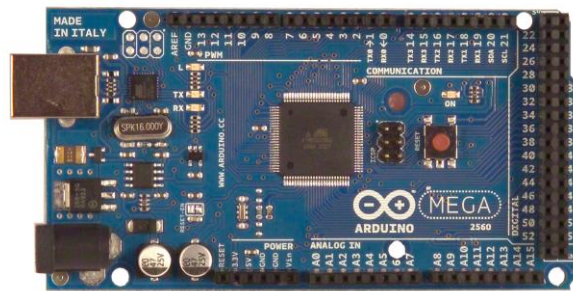


Figura 02 – Arduino Mega 2560

Fonte: <http://flipmu.com/>

O arduino é uma plataforma de prototipagem que faz a robótica ficar mais acessível, surgindo assim diversos projetos como:

1. Casa inteligente
2. Semáforo interativo
3. Alarma com sonorizador
4. Projeto com sensores ultrasonicos

A versatilidade de escrever e apagar o código que é gravado no microcontrolador faz com que o arduino seja uma excelente ferramenta. Um dos pontos positivos do arduino, ele é open-source, podemos modifica-lo sem nenhum problema. (playground.arduino.cc/)

Criado para fins educacionais, porém devido ao seu grande sucesso, é utilizado hoje por inúmeras pessoas, cada uma utilizando da maneira que o convém.

2.5 Sensor HC –SR04

O sensor ultrassônico HC-SR04 (Figura 03) funciona como um detector de objetos permitindo medir distâncias entre 2 centímetros e podendo chegar a distâncias máximas de até 4 metros de distância com uma precisão de 3 milímetros. Isso permite que criar alarmes, sensores de presença, sensores de estacionamento, sensor de obstáculo para que possa ser usado em robôs, e etc. (www.micropik.com/)

2.5.1 Funcionamento:

1. Necessita de um gatilho (trigger) de pelo menos 10 μ s de nível alto.
2. O módulo envia automaticamente 8 pulsos de 40kHz e automaticamente detecta se houve retorno de algum pulso.
3. Se houver um sinal de retorno a partir dos disparos o nível de saída é modificado.

O funcionamento é simples, é enviado um sinal ultrasonico pelo modulo (Chirp), o mesmo detecta a volta desse sinal (Echo) e com base nesses dois acontecimentos, gera um sinal que é possível medir a distancia até o objeto. (<http://www.micropik.com/>)

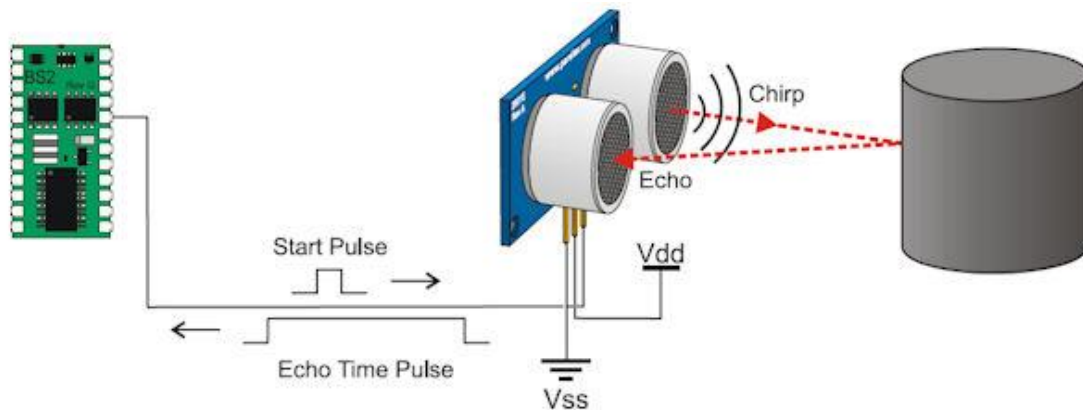


Figura 03 – Funcionamento do sensor

Fonte: www.nubiasouza.com.br/

2.6 O software

A linguagem de programação do arduino, é baseada em “wiring”, ou seja, uma plataforma computacional física que é baseada em um ambiente multimídia de programação “processing”.

O arduino surgiu para que pudesse simplificar a programação dos microcontroladores, além de não precisar de um gravador, porquê ele já possui um bootloader. O seu “IDE” (ambiente de desenvolvimento) autoriza o uso de C/C++, assim, aumentando muito mais o leque de opções para projetos. Este ambiente traz inúmeras bibliotecas, facilitando o uso e diminuindo o tempo gasto para a elaboração de projetos. Resumidamente, a linguagem de programação padrão tem origem do wiring e é essencialmente C/C++. (playground.arduino.cc/)

2.7 Linguagem C e C++

Versão criada pelo Dennis Ritchie em 1972, tinha a intenção de distribuir como um software junto com o Sistema Operacional Unix. A linguagem C, teve um início graças a linguagem ALGOL 60 em 1960. O ALGOL era uma linguagem de alto nível, permitindo que os programadores pudessem trabalhar “afastado da máquina”, sem ter

que ficar se preocupando com os comandos, dados armazenados ou processados. A linguagem ALGOL, foi criada com a intenção de substituir o FORTRAN mas, não obteve sucesso, porque na época era necessário um grande conhecimento de hardware e não de uma linguagem de alto nível. Em 1967, surgiu no mercado o CPL (*combined Programming Language*), teve o objetivo de trazer o ALGOL de volta, porém, não obteve muito sucesso e teve um final igual ao ALGO, no mesmo ano foi criado o BCPL que era uma simplificação do CPL, feito apenas para tentar manter uma boa impressão. No ano de 1970, Ken Thompson projetou o UNIX, implementando um compilador para uma versão mais reduzida do CPL, chamando assim de linguagem B. As linguagens BCPL e B, demonstravam ser bastante limitadas, podendo apenas ser usada em determinadas classes de problemas. Estes problemas, levou o projetista Dennis Ritchie a criar outro tipo de linguagem, uma que seria a sucessora do B, que no caso se chamaria C. A linguagem quis manter o tema “contato com o computador real” e implementar tbm novas condições para que o programador pudesse desenvolver programas de diversas áreas, seja comercia e engenharia. Em 1985 a *American National Standards Institute* (ANSI), foi estabelecido um padrão oficial do C, que passaria a ser chamado “C ANSI”.

A linguagem C++, é uma versão expandida do C. Essas extenções foram inventadas pelo Bjarne Stroustrup, em 1979. Inicialmente a linguagem foi chamada de “C com Classes” porém, em 1983 o nome mudou para C++. (ANA F.G. ASCENCIO e EDILENE A.V. DE CAMPOS, 2002)

A necessidade da invenção do C++ foi por motivos de programação, que era: aumento da complexidade da linguagem. Muito tempo foi se passando e os programas dos computadores foram se tornando mais complexos. Mesmo sendo uma ótima linguagem de programação, o C tem seus limites, uma vez que o programa atinge 25 mil a 100 mil linhas códiicadas, se torna muito complexo e difícil analisa-lo, a manutenção acaba se tornando complicada. O C++ apareceu para quebrar esse paradigma, a essencia é permitir que vários programadores compreendam e gerenciem programas mais complexos.

Desde o momento em que a linguagem C++ foi desenvolvida, foi passada por várias revisões, ocorrendo modificações e algumas adições. A ISO e a ANSI se juntaram para padroniza-la. (ANA F.G. ASCENCIO e EDILENE A.V. DE CAMPOS, 2002 e playground.arduino.cc/)

2.7.1 Diferenças entre C e C++

São linguagens diferentes mas, com sintaxes semelhantes. A grande diferença entre estas duas linguagens, está no paradigma usado, enquanto o C é uma linguagem funcional, o C++ é uma programação orientada para objetos e também para a funcional. O fato de terem sintaxes parecidas, torna o C++ quando não utilizado uma programação orientada a objeto, igual ao C e podendo até compilar em uma compilador C. (ANA F.G. ASCENCIO e EDILENE A.V. DE CAMPOS, 2002)

Há também outras diferenças, que são:

1. Em C não pode usar objetos, já que não existem mas, em C++ existem.
2. A extensão do código-fonte em C é “.c” e para C++ é “.cpp” (que vem do C plus plus).
3. A programação em C, tem um conjunto de instrução menor, facilitando o aprendizado porém em caso de programas mais complexos, C++ tem um desempenho melhor, contendo um leque de funções/instruções maiores que C.

3. Desenvolvimento do Trabalho

Construir um robô controlado por Arduino, que segue as pessoas onde quer que elas fossem, contendo sensores que indicariam temperatura ambiente, umidade do ar, distancia em que o robô se encontra do operante, tudo isso seria transmitido para um tablet ou até mesmo o celular do usuário.

Optou-se fazer com que o robô apenas siga as pessoas, utilizando o Arduino Mega2560 e o sensor HC-SR04, pois o tempo de desenvolvimento não seria suficiente.

Portas digitais são essenciais para o projeto. O acionamento dos sensores será através das portas digitais, ele envia sinais de nível alto e baixo (0 ou 1/0 ou 5 volts).

Ao chegar o sinal do sensor ultrassônico, que seja nível lógico alto (1), é enviado um pulso do Arduino e do próprio é enviado outro sinal a placa mãe do robô curumim, passando por circuitos que contem pontes H, fazendo assim o acionamento dos motores.

Exemplo: Se a pessoa está indo para a direita, o sensor do meio e o da esquerda ficam em nível 0 e o da direita em nível 1, fazendo o robô se locomover para a direita.

3.1 Etapas para a construção do Robô AGV

Robô Curumim, da empresa Xbot que foi cedido para fins acadêmicos, na Figura 04, ele se encontra montado e pronto para uso.



Figura 04 – Robô Curumim

Fonte: www.xbot.com.br/

Foram retiradas a placa mãe e a placa driver do Curumim, instalando no lugar o arduino mega 2560. (Figura 05)

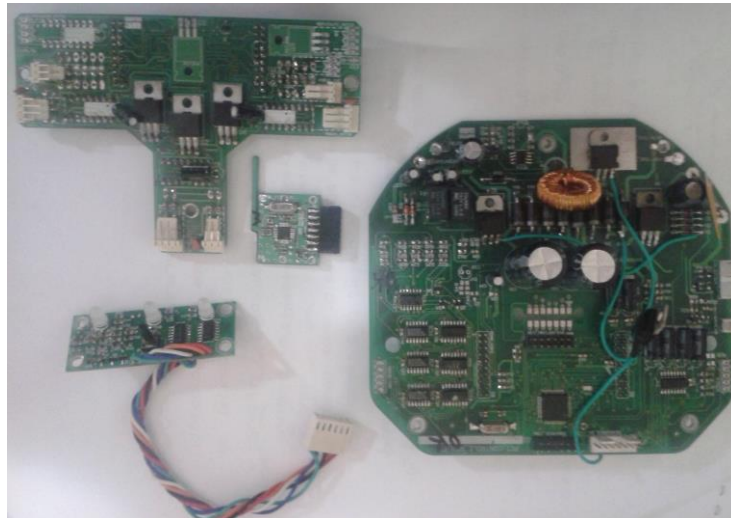


Figura 05 – Placa Mãe e Placa Driver

Fonte: O autor.

Aproveitando os componentes da parte motora do Curumim, foi necessária a retirada da placa mãe e da placa driver para que a implementação do arduino fosse possível (Figura 06). O arduino, sensores e as pontes H's foram compradas a parte.

São 3 motoredutores, sendo que cada um deles são independentes, alimentados por 2 baterias, cada uma contendo 14.4V e uma corrente nominal de 2500mA/h.

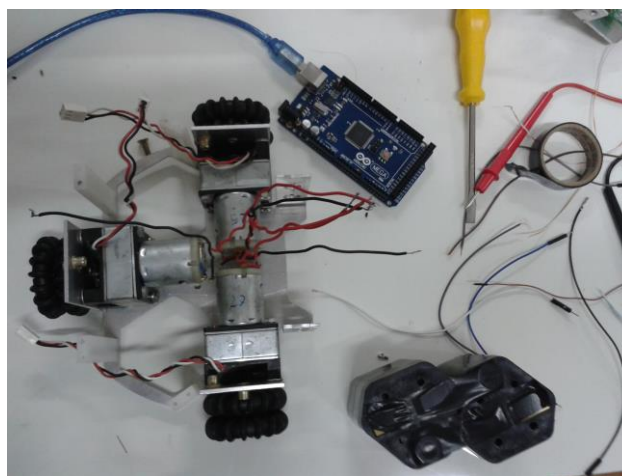


Figura 06 – Motores, baterias e arduino.

Fonte: O autor.

Foram usadas 2 pontes H (L298N), uma sendo o módulo L298 H-bridge Motor Driver, que tem como função acionar 2 motores, controlando a velocidade e o sentido do seu giro. A outra ponte, L298N que contém um CI (Circuito Integrado), possui 2 pontes H's, servindo como um driver para alta tensão e alta corrente, podendo ser alimentada até 46V e conduzindo até 2A por canal. A ponte L298N com o CI foi usada para utilizar os níveis lógicos padrão TTL e acionar uma carga indutiva, que no caso são os motores. Basicamente, essas duas pontes estão convertendo os sinais elétricos para sinais digitais. Para o controle dos motores, com o uso da ponte H L298 H-bridge Motor Driver (Figura 07) ligando no arduino:

- Porta “In1” e “In2” - Conecta na entrada digital do arduino (Portas 40 e 42) para que o motor da direita ande para frente e para trás.
- Porta “In3” e “In4” – Conecta na entrada digital do arduino (Portas 41 e 43) para que o motor da esquerda ande para frente e para trás.

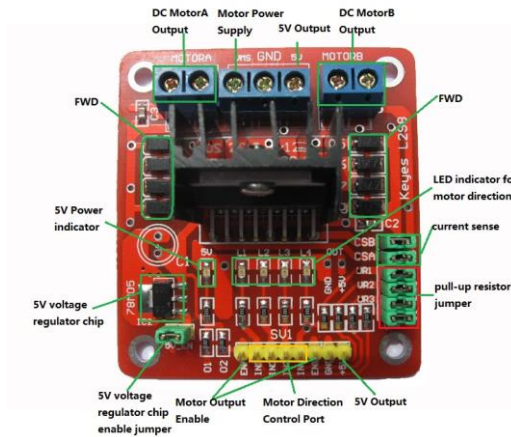


Figura – 07 Dual H-Bridge Motor Driver

Fonte: O autor.

Para controlar o motor do meio, a ponte H utilizada foi o L298N que contém um CI (Figura 08). Setando a pinagem do motor em “In3” e In4” e entrando no arduino nas portas 8 e 9 da PWM.

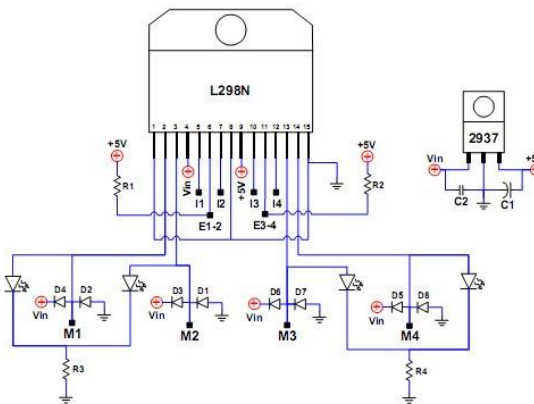


Figura – 08 Ponte H com CI

Fonte: O autor.

Robô Curumim montado, com o arduino, as pontes e os sensores. (Figura 09 e 10)

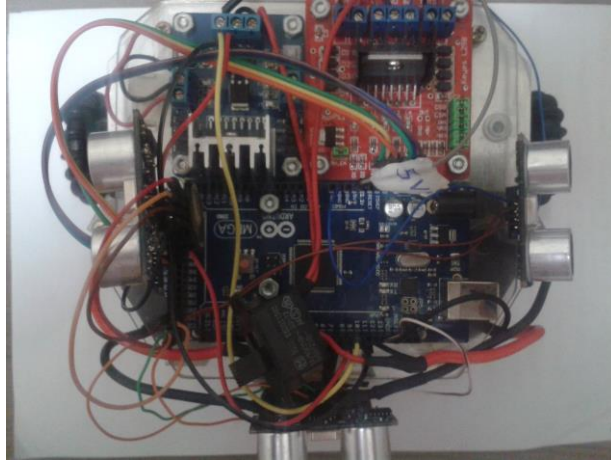


Figura – 09 Robô Curumim

Fonte: O autor.

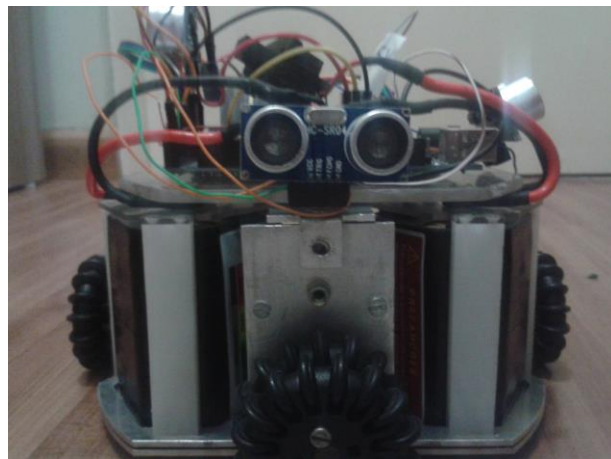


Figura – 10 Robô Curumim

Fonte: O autor.

3.1.1 Programação

Para que o robô pudesse andar para frente, para trás e dos lados, foi necessário o uso da lógica de programação, utilizando a condicional “if” em c++. Essa condicional, trata-se de condições que sejam verdadeiras ou falsas, podendo estar ligada a estruturas de repetições. A estrutura “if”, é a estrutura mais básica de teste de lógica, encontrada em todas as linguagens, mas de maneiras diferentes. É executada uma série de instruções caso uma condição seja verdadeira ou falsa.

Declarar portas do arduino, da ponte H, dos sensores e montar a lógica para que o robô pudesse se movimentar, comparando valores em cm, ou seja, comparando valores de distâncias. Na própria programação, é feita uma conta que calcula o tempo de transmissão e recepção do sensor e converte para centímetros.

4. Proposta para Trabalho Futuro

A hipótese de melhorar este projeto é com a implementação de vários sensores (umidade do ar, temperatura, etc.), uso do sistema mestre escravo e verificar o comportamento do robô em uma instalação hospitalar. Mudar as dimensões do robô, motores com maiores potências, manipulação ágil e fácil.

5. Conclusão

Conclui-se que o objetivo do trabalho foi alcançado com sucesso pois o robô executou as funções determinadas pelo programa. A montagem da parte mecânica (motores), a implantação do arduino, dos sensores e a programação foram realizados e o robô Curumim executou a tarefa de seguir pessoas ou objetos sem a necessidade de controles, cabos e etc.

Este protótipo, utilizando o robô Curumim da Xbot, apresentou um sistema autonomo, sem a necessidade de um controle.

Referências Bibliográficas

- ASCENCIO, A., F., G. e CAMPOS, E., A. V., Fundamentos da Programação de Computadores, 2 Edição, São Paulo, Editora Pearson, 2002.
- BANZI, MASSIMO, Primeiros Passos com o Arduino, Edição 1, São Paulo, Editora Novatec, 2011.
- EVANS.MARTIN, NOBLE, JOSHUA, HOCHENBAUM, JORDAN, Arduino em Ação, Edição 1, São Paulo, Editora Novatec, 2013.
- JOHN J. CRAIG, ROBÓTICA, 3ª Edição, São Paulo, Editora Pearson, 2013.
- MCROBERTS, MICHAEL; Arduino Básico, Edição 1, São Paulo, Editora Novatec, 2011.
- NIKU, B., SAEED, Introdução a Robótica – Análise, Controle, Aplicações, 2ª Edição, São Paulo, Editora LTC (GRUPO GEN), 2013.
- PAZOS, FERNANDO, AUTOMAÇÃO DE SISTEMAS E ROBÓTICA, 1ª Edição, São Paulo, Editora Axcel, 2002.
- SILVA, FLÁVIO SOARES CORRÊA DA; MELO, ANA CRISTINA, Princípios de Linguagens de Programação, São Paulo, Editora Edgard Blucher, 2003.
- ARDUINO, CC. Disponível:< playground.arduino.cc/>. Acesso: 20/02/2014
- BANZI, MASSIMO, Primeiros Passos com o Arduino, Edição 1, São Paulo, Editora Novatec, 2011. Disponível: <novatec.com.br>. Acesso: 15/03/2014
- DESTACOM, UFMS, Disponível: <destacom.ufms.br/>. Acesso: 10/03/2014
- LABORATÓRIO DE GARAGEM, Disponível: <labdegaragem.com.br>. Acesso: 20/03/2014
- XBOT, Disponível: <xbot.com.br/externo>. Acesso: 21/02/2014