

UESB - UNIVERSIDADE ESTADUAL DO SUDOESTE DA BAHIA
Departamento de Ciências Exatas

**Robô para Monitoramento à Distância de Áreas de
Risco e Ambientes Hostis**

Paulo Oliveira Paixão

Vitória da Conquista – BA

Dezembro de 2008.

UESB - UNIVERSIDADE ESTADUAL DO SUDOESTE DA BAHIA
Departamento de Ciências Exatas

**Robô para Monitoramento à Distância de Áreas de
Risco e Ambientes Hostis**

Discente: Paulo Oliveira Paixão

Orientador: Prof. Adilson de Lima Pereira

Monografia apresentada ao Departamento de Ciências Exatas – DCE – UESB para conclusão do curso de Ciência da Computação e obtenção do título de Bacharel em Ciência da Computação.
Área de Concentração: Controle de dispositivos.

Vitória da Conquista – BA

Dezembro de 2008

UESB - UNIVERSIDADE ESTADUAL DO SUDOESTE DA BAHIA
Departamento de Ciências Exatas

Paulo Oliveira Paixão

**Robô para Monitoramento à Distância de Áreas de
Risco e Ambientes Hostis**

Área de Concentração: Robótica e eletrônica

Orientador: Prof. Msc. Adilson de Lima Pereira

Comissão Examinadora

Prof. Msc. Adilson de Lima Pereira
Universidade Estadual do Sudoeste da Bahia

Prof. Esp. Gidevaldo Novais
Universidade Estadual do Sudoeste da Bahia

Prof^a. Msc. Alessandra Oliveira Andrade
Universidade Estadual do Sudoeste da Bahia

Vitória da Conquista – BA

Dezembro de 2008

Dedicatória

Dedico esse trabalho primeiramente a Deus, aos meus pais, meus irmãos, minha esposa, meus professores e meus colegas.

Agradecimentos

As palavras de agradecimentos, não refletem a gratidão por todos aqueles que contribuíram de alguma forma para que esse momento acontecesse.

Agradecer primeiramente a Deus nosso criador, pela vida, saúde, paz e a oportunidade de concluir o curso, depois de tantas dificuldades.

Aos pais, Ângelo e Leozina, que sempre sonharam com o dia da diplomação, depois de verem tantas noites mal dormidas, mas nunca faltaram com o incentivo para que tivesse continuidade essa tarefa e fizeram todos os esforços para proporcionar esse momento.

Aos irmãos eternos admiradores: Alcides, Odília, Léo, Lula (*in memorium*), Telma e João, pelo apoio nos momentos e, por acreditar que o processo um dia chegaria ao fim.

À esposa Margarete: pela compreensão, por abrir mãos de determinados momentos de lazer por entender a necessidade de dedicação de horas de estudo para concluir trabalhos, pela paciência e incentivo.

Aos professores que compartilharam seus conhecimentos e experiências.

Aos colegas da faculdade, aos colegas de trabalho, Alan Lebrão, Gilberto, André e Agnaldo que contribuíram com o projeto.

A todos aqueles que participaram direto ou indiretamente, os sinceros agradecimentos e que Deus illumine a todos.

RESUMO

Este projeto tem como proposta, construir um robô controlado à distância via teclado do computador, para monitoramento de áreas de risco. A porta paralela será usada como interface de comunicação entre o computador e o hardware. Os dados serão transmitidos por ondas de rádio, possibilitando o controle à distância. As imagens do local serão transmitidas por uma câmera de vídeo equipada com transmissor sem fio, possibilitando visualização dessas imagens na tela do computador. As imagens transmitidas serão captadas pelo receptor acoplado à placa de captura de imagens e podem ser gravadas no disco do computador com ajuda de software projetado com esse propósito. Java será a linguagem de programação usada para desenvolver o software de controle. Com esse protótipo em funcionamento, será possível monitorar à distância um local que oferece risco de desabamento; verificação de objetos, suspeitos de conter explosivos; espionagem tática.

ABSTRACT

This project has to offer, a build a robot remotely controlled via computer keyboard, to monitor areas of risk. The parallel port will be used as an interface for communication between the computer and hardware. The data will be transmitted by radio waves, allowing the remote control. The images of the site will be transmitted by a video camera equipped with wireless transmitter, allowing viewing of these images on the computer screen. The images received will be transmitted by the receiver attached to the capture card images and the disc can be recorded with the help of computer software designed for this purpose. Java is the programming language used to develop the software of control. With this prototype in operation, you can remotely monitor a location that offers risk of collapse; verification of objects, suspected to contain explosives; espionage tactics.

Lista de Figuras

Figura 1: Pinagem do Conector DB 25.....	18
Figura 2: Diagrama Interior do motor de passo.....	21
Figura 3: Esquema elétrico de um Transmissor basico	23
Figura 4: Diagrama em blocos de um Receptor.....	23
Figura 5: Diagrama interno do Buffer74LS244	25
Figura 6: Diagrama em blocos do MC145026	27
Figura 7: Diagrama em bloco, layout e encapsulamento CI MC145027	29
Figura 8: Sentido de Fluxo de corrente de uma ponte H.....	31
Figura 9: Diagrama em bloco do CI L298.....	32
Figura 10: Pinagem do CI L298	32
Figura 11: Identificação dos pinos transmissor	34
Figura 12: Transmissor RWS-374	35
Figura 13: Limpeza da superfície de cobre	37
Figura 14: Protótipo transmissor (1) e Receptor (2)	39

Lista de Tabelas

Tabela 1: Endereços da porta paralela.....	17
Tabela 2: Registradores da porta paralela.....	17
Tabela 3: Dos pinos do CI 74LS244	26
Tabela 4: Descrição dos pinos do CI I298.....	33
Tabela 5: Valores de entradas para ponte A	33
Tabela 6: Pinos do RWS-374	35
Tabela 7: Substâncias que compõem a solução.....	38
Tabela 8: Lista de Componentes.....	48

Sumário

INTRODUÇÃO	11
1.1 Motivação	12
1.2 Objetivo	12
1.3 Justificativa	12
1.4 Metodologia.....	12
1.5 Organização da Monografia	Erro! Indicador não definido.
REFERENCIAL TEÓRICO	14
2.1 Robótica	14
2.2 Porta Paralela.....	15
2.3 Funcionamento da porta paralela	16
2.4 Motor de Passo.....	19
2.5 Características do motor de passo	19
2.6 Motor DC (Direct Current)	21
2.7 Funcionamento do motor DC.....	22
2.8 Comunicação sem fio	22
2.9 Java	24
2.10 Conclusão do capítulo	24
CARACTERÍSTICAS DOS COMPONENTES ELETRÔNICOS	25
3.1 Buffer 74LS244.....	25
3.2 Codificador MC145026	26
3.3 Descrição dos pinos do Codificador MC145026	27
3.4 Codificador MC145027	28
3.5 Descrição da pinagem do MC145027	29
3.6 Driver controlador de motor DC	30
3.7 Módulo transmissor TXA1-434-F11	34
3.8 Receptor RWS-374	34
3.9 Configuração do sistema operacional	35
3.10 Conclusão do capítulo 3.....	36
MONTAGEM E FUNCIONAMENTO DO PROJETO	37
4.1 Preparação da placa	37
4.2 Preparação da solução	38
4.3 Montagem do projeto.....	39
4.4 Processo de soldagem	39

4.5	Sistema de redução.....	40
4.6	Trassmissão de dados.....	41
4.7	Recepção de dados.....	41
4.8	Visualização de imagens.....	41
4.9	Conclusão do capítulo 4.....	42
	CONCLUSÃO E PROPOSTA DE TRABALHOS FUTUROS.....	43
	REFERÊNCIAS	45
	APÊNDICE 1 Lista de componentes	Erro! Indicador não definido.
	APENDICE 2 Codigo fonte de controle.....	Erro! Indicador não definido.

CAPÍTULO 1

INTRODUÇÃO

Este projeto descreve o desenvolvimento de um robô controlado à distância. Equipado por um transmissor conectado à porta paralela do computador por onde os comandos de saída serão enviados através do teclado, será possível guiá-lo por locais onde não exista grandes obstáculos. Fará parte da sua estrutura física uma câmera com transmissão sem fio de áudio e vídeo, possibilitando ver as imagens no próprio computador ou mesmo num monitor de TV. Caso essas imagens sejam vistas em um computador, o receptor da câmera de vídeo será conectado a uma placa de captura de vídeo previamente instalada; caso seja num aparelho de TV é suficiente apenas conectar os cabos de áudio e vídeo nas referidas entradas deste aparelho. A manipulação dessas imagens e visualização no computador, será feita utilizando um framework desenvolvido pela *Sun Microsystems*, o JMF (*Java Media Framework*). Este framework permite que estas imagens sejam gravadas em formatos comuns de vídeo, podendo ser analisados posteriormente ou visualizadas em tempo real.

Para o desenvolvimento desse projeto, é necessário ter conhecimento básico, teórico e prático, de eletroeletrônica, programação (Java ou C/C++), conceitos de robótica e telecomunicação. Alguns conceitos como programação e eletrônica serão utilizados e abordados mais amplamente devido a própria estrutura do projeto, uma vez que algumas soluções já encontram-se prontas no mercado, como é o caso do transmissor e do receptor. Com este projeto piloto, é possível agregar algumas funcionalidades e recursos, que poderão ser incorporados em etapas posteriores, como por exemplo, o uso de sensores de proximidade, programação embarcada utilizando micro-controlador, braço mecânico com movimentos autônomos ou controlados, conceitos de inteligência artificial, dentre outras.

1.1 Motivação

Conciliar o hardware e o software na concepção de projetos viáveis com um baixo custo, despertando interesse para o estudo, pesquisa e atuação no ramo da tecnologia. Além disso, idealizar um dispositivo que poderá ser usado em situações de risco, evitando a exposição da máquina humana a essas áreas. Sem dúvida isso é um dos maiores vetores de motivação para iniciar a esse projeto.

1.2 Objetivo

Montar um robô utilizando recursos disponíveis de software e hardware, para utilização na inspeção de áreas de risco e ambientes hostis, diminuindo a exposição de vidas humanas. Alguns desses problemas podem ser a inspeção de ambientes com risco de desabamento, ameaças de bombas e monitoramento a distância.

1.3 Justificativa

Nos dias de hoje, as pessoas convivem com ameaças terroristas, construções irregulares, mal conservadas e mal fiscalizadas. É comum em países subdesenvolvidos o pouco uso da tecnologia para executar algumas tarefas de risco, utilizando de forma perigosa a “máquina humana”. O desenvolvimento desse equipamento tem por finalidade o aproveitamento de tecnologias baratas para criação de soluções viáveis.

1.4 Metodologia

Como metodologia, inicialmente será feito um levantamento de requisitos no que diz respeito aos componentes que poderão ser utilizados, interface de comunicação, viabilidade de custo e tempo necessário para construção. Revisão bibliográfica de conceitos de eletrônica, lógica, circuitos digitais, bibliotecas disponíveis para manipulação de interfaces, linguagem de programação adequada para a proposta com possibilidades de expansão e inclusão de outras funcionalidades.

O projeto é composto de módulo veículo, módulo eletrônico, câmera de vídeo e a programação para controle dos motores de passo ou motor linear. No veículo, os motores são responsáveis pelo deslocamento. A comunicação será feita com módulos de transmissão e recepção, bem como a captação das imagens de vídeo. A linguagem de programação utilizada será a linguagem Java, uma vez que sua máquina virtual possibilita a execução dos controles em qualquer plataforma independente do sistema operacional. É uma linguagem versátil e robusta, segura e eficiente e tem suporte para programação embarcada e bibliotecas para manipulação de imagens, áudio e vídeo.

CAPITULO 2

REFERENCIAL TEÓRICO

Para facilitar o entendimento desse projeto é importante explicar alguns conceitos e, teorias de tecnologias e termos utilizados.

2.1 Robótica

O termo robótica foi criado pelo escritor de ficção científica Isaac Asimov em 1948 no seu romance "Eu Robô". Robótica é um ramo da tecnologia que engloba mecânica, eletricidade, eletrônica e computação, que atualmente trata de sistemas compostos por motores e partes mecânicas automatizadas e controlados por circuitos integrados. Esses sistemas são controlados manual ou automaticamente por circuitos eletroeletrônicos. Os robôs são apenas máquinas; não sonham nem sentem. Cada vez mais as pessoas utilizam dispositivos automáticos para suas tarefas. Algumas dessas tarefas são vistas cotidianamente quando aperta-se um botão, os vidros do carro fecham e travas das portas são acionadas, com outro, o portão da garagem se abre, dentre muitas outras existentes. Esta tecnologia, hoje adotada por muitas fábricas e indústrias, tem obtido de um modo geral, êxito em questões como a redução de custos, aumento de produtividade, mas por outro lado, diminui a mão-de-obra humana nas grandes linhas de produção, trazendo como conseqüência um acréscimo nas taxas de desemprego. O problema da mão de obra sendo substituída por máquinas tem proporcionado um grande debate entre as classes operárias, sindicatos e classe patronal. Sem dúvida é um assunto que continuará a ser debatido e gerando uma ampla discussão.

A implantação de sistemas automatizados deve ter sua implantação planejada visando produzir o menor impacto social possível. A mecanização de

algumas atividades como a colheita de cana, laranjas e café, se torna a cada dia um caminho sem volta e alguns acordos entre sindicatos são feitos para que parte da produção seja feita por mão de obra humana. Atualmente, devido aos inúmeros recursos que os sistemas computacionais nos oferecem, a robótica atravessa um momento de contínuo crescimento, que permitirá em um curto período de tempo, o desenvolvimento de robôs inteligentes tornando a ficção de outrora, uma realidade. Universidades e escolas técnicas têm incentivado a criação de robôs inteligentes e outras soluções que podem dentro de um curto prazo estar disponível para a maioria da população.

2.2 Porta Paralela

Quando um usuário passa a ter contato com o mundo da informática, aprende que o computador possui interfaces e dispositivos de entradas e saídas. A porta paralela nada mais é do que uma dessas interfaces onde é possível usufruir das funcionalidades de alguns dispositivos como Impressoras, escaner ou outros dispositivos conectados. Por que paralela? Há duas formas de se transmitir dados. Uma delas é a forma serial, outra paralela. Serial, como o próprio nome diz, é a transmissão seqüencial de bits. Transmitir de forma paralela é transmitir por um byte de uma só vez.

A transmissão em paralelo, ao contrário da transmissão em série, precisa de vários fios para transportar a informação. Neste tipo de transmissão, a grande vantagem é a sua velocidade de transmissão. Em vez de enviar bit a bit os dados, transmite-se uma quantidade de bits de uma só vez. No entanto, esta é melhor apenas quando o receptor estiver a uma curta distância, visto que o grande inimigo da transmissão em paralelo é a interferência. Isso quer dizer que quanto menor a distância, maior a taxa de transferência de bits, o que não acontece na comunicação em série que é ideal para longas distâncias. Desta forma, a utilização da transmissão em paralelo é melhor aplicável na transmissão interna de dados de um computador através do respectivo barramento, entre o processador e memória de acesso randômico ou então na transmissão entre computador e periféricos (a uma distância de até, aproximadamente, 6m). A velocidade da transmissão depende da frequência

do relógio (ciclos por segundo). Em cada ciclo de relógio é possível transmitir um dado. Aumentando a frequência, aumenta-se a velocidade de transmissão dos dados. Para comparar a velocidade de transmissão em diferentes sistemas que usam diferentes quantidades de bits, a velocidade de transmissão em paralelo foi padronizado em *bytes* por segundo (B/s).

2.3 Funcionamento da porta paralela

A Porta Paralela, assim como as outras portas do computador (*COM*, *USB*, controladora de *Joystick*) é uma interface digital por onde o computador envia e recebe dados do mundo externo. Na porta paralela é possível conectar outros dispositivos que não seja somente a impressora, podendo controlá-los, enviando ou recebendo dados através desse dispositivo. A conexão de qualquer equipamento à porta paralela é feita através do conector DB25, que é um conector com padrão universal e contém 25 pinos. É através desse conector que se estabelece a comunicação com interfaces conectadas a ela, permitindo o controle de dispositivos e aquisição de dados. A porta paralela possui três configurações de comunicação, sendo uma unidirecional e duas bidirecionais.

A porta paralela na configuração SPP (*Standard Parallel Port*) pode chegar a uma taxa de transmissão de dados de 150KB/s. Comunica-se com a CPU utilizando um barramento de dados de 8 bits. Para a transmissão de dados entre periféricos são usado 4 bits por vez. Esta configuração apenas envia dados à porta. Estes modos de configuração podem ser modificados na bios do computador.

A porta avançada EPP (*Enhanced Parallel Port*) chega a atingir uma taxa de transferência de 2 MB/s. Para atingir essa velocidade, será necessário um cabo especial. Comunica-se com a CPU utilizando um barramento de dados de 32 bits. Para a transmissão de dados entre periféricos são usado oito bits por vez. Indicado para dispositivos que requerem alta taxa de transferência num canal paralelo.

A porta avançada ECP (*Enhanced Capabilities Port*) tem as mesmas características que a EPP, porém, utiliza DMA (*Direct Memory Access*), sem a necessidade do uso do processador, para a transferência de dados. Utiliza também um buffer FIFO de 16 bytes.

Para acessar a porta LPT é necessário endereçar corretamente. A tabela 1 apresenta os endereços das portas paralelas.

Tabela 1: Endereços da porta paralela

Nome da porta	Endereço	Endereço da porta	Descrição
LPT1	0000:4008	378 h	Endereço base
LPT2	0000:400A	278 h	Endereço base

A tabela 2 define a função dos registradores, com os quais se define o comportamento para cada tipo de registro.

Tabela 2: Registradores da porta paralela

Nome	Endereço LPT1	Endereço LPT2	Descrição
Registro de dados	378 h	278 h	Envia um byte pra porta
Registro de Status	379 h	279 h	Lê o status da porta
Registro de Controle	37A h	27A h	Envia dados de controle

Como mostra a figura 1, referente ao conector DB 25 fêmea, a porta paralela é formada por 17 pinos de comunicação e 8 pinos que são ligados em comum ao terra (*gnd*).

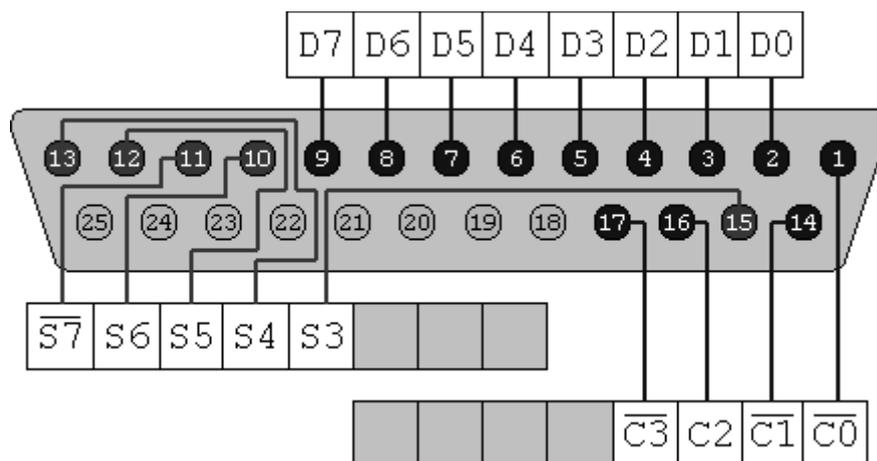


Figura 1: Pinagem do Conector DB 25

- Os pinos 2 a 9, denominados, da direita para a esquerda, de D0, D1, D2, D3, D4, D5, D6 e D7 são controlados pelo Registrador de Dados (saída de dados), e a letra D indica isso. O número que vem após a letra corresponde à posição do bit dentro do byte. D0, que corresponde ao pino 2, representa o bit menos significativo e D7, conectado ao pino 9, representa o bit mais significativo. O acesso a estes pinos é feito pelo endereço 0x378 para porta paralela padrão.
- Os pinos 10, 11, 12, 13 e 15, denominados S6 (Ack), S7 (Busy), S5 (Paper End), S4 (Select Out), S3 (Error), são controlados pelo Registrador de Status (entrada de dados), e a letra S indica isso. O número que vem após a letra corresponde à posição do bit dentro do Byte. O acesso a estes pinos é feito pelo endereço 0x379 para porta paralela padrão.
- Os pinos 1, 14, 16 e 17, denominados C0 (Strob), C1 (Auto Feed), C2 (Init) e C3 (Select In), são usados pelo registrador para controlar a porta paralela e a letra C faz referência a isso.. O número que vem após a letra corresponde à posição do bit dentro do byte. Estes pinos podem ser usados como pinos de saída. O acesso destes pinos é feito pelo endereço 0x37A para porta paralela padrão.
- Os pinos 18 a 25 são pinos de aterramento (*ground ou gnd*).

Obs.: Os pinos C0, C1, C3 e S7 da porta paralela têm a lógica invertida, isto é, quando se deseja ativar o pino envia-se o valor "0", para desativá-lo, envia-se o valor "1".

2.4 Motor de Passo

Motor de passo é um tipo de motor elétrico que pode ser controlado por sinais digitais, tornando-o preciso e de recomendável utilização em aplicações que venham a requerer um ajuste fino de posicionamento. Para entender o conceito de motor de passo, é necessário um breve comentário sobre motores elétricos em geral e, em seguida serão o modo de funcionamento e de como controlá-lo, pois, o controle de seus movimentos é sem dúvida o que o diferencia dos demais motores.

Já um motor convencional, é um dispositivo que transforma energia elétrica em energia mecânica, em geral energia cinética. Ou seja, num motor, a simples presença da corrente elétrica, seja corrente contínua ou corrente alternada garante movimento em um eixo, que pode ser aproveitado de diversas maneiras dependendo da aplicação do motor. O acionamento de máquinas e equipamentos mecânicos por motores elétricos é um assunto de grande importância econômica.

2.5 Características do motor de passo

O motor de passo é um transdutor que converte energia elétrica em movimento controlado através de pulsos, o que possibilita o deslocamento por passo, onde passo é o menor deslocamento angular. Outra vantagem do motor de passo em relação aos outros motores é a estabilidade. Quando se deseja obter uma rotação específica, calcula-se o número de rotação por pulsos o que nos possibilita uma boa precisão no movimento. Aspectos relevantes de um motor de passo que poderá ser utilizado:

- Graus por passo – sem dúvida, a característica mais importante ao se escolher o motor, o número de graus por passo está intimamente

vinculado com o número de passos por volta, como por exemplo 90 graus.

- Momento de Frenagem – momento máximo com o rotor bloqueado, sem perda de passos.
- Momento (torque) – efeito rotativo de uma força, medindo a partir do produto da mesma pela distância perpendicular até o ponto em que ela atua partindo de sua linha de ação.
- Resposta de passo – é tempo que o motor gasta para executar o comando.
- Ressonância – como todo material, o motor de passos tem sua frequência natural. Quando o motor gira com uma frequência igual a sua, ele começa a oscilar e a perder passos.
- Tensão de trabalho – normalmente impresso no próprio chassi do motor, a tensão em que trabalha o motor é fundamental na obtenção do torque do componente. Tensões acima do estipulado pelo fabricante aumenta o torque do motor, porém, tal procedimento resulta na diminuição da vida útil do mesmo. Destaca-se que a tensão de trabalho do motor não necessariamente deve ser a tensão utilizada na lógica do circuito. Os valores normalmente encontrados variam de +5V à +48V.

Existem dois tipos de motor de passo: unipolar e bipolar.

Motor unipolar são caracterizados por possuírem um *center tape* entre o enrolamento de suas bobinas. Normalmente utiliza-se este *Center tape* para alimentar o motor, que é controlado aterrando-se as extremidades dos enrolamentos como mostra a figura 2.

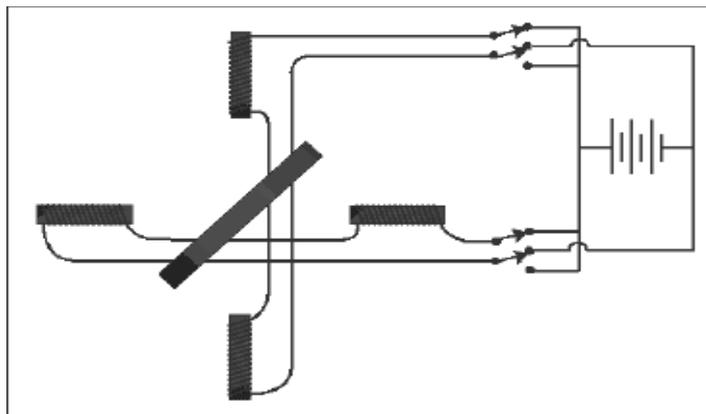


Figura 2: Diagrama Interior do motor de passo

Motores bipolares exigem circuitos mais complexos. A grande vantagem em se usar os bipolares é prover maior torque, além de ter uma maior proporção entre tamanho e torque. Fisicamente os motores têm enrolamentos separados, sendo necessário uma polarização reversa durante a operação para o passo acontecer. Para que um motor de passo funcione, é necessário que sua alimentação seja feita de forma seqüencial e repetida. Não basta apenas ligar os fios do motor de passo a uma fonte de energia e sim ligá-los a um circuito que execute a seqüência requerida pelo motor. Essa forma de alimentação das bobinas também define o tipo de movimento do motor.

Existem três tipos básicos de movimentos: o de passo inteiro, meio passo e o micro passo, tanto para o motor bipolar como para o unipolar.

2.6 Motor DC (Direct Current)

São conhecidos por seu controle preciso de velocidade e por seu ajuste fino e são, portanto, largamente utilizados em aplicações que exigem tais características. A utilização dos motores de corrente contínua teve um grande incremento nos últimos anos, graças à eletrônica de potência. Fontes estáticas de corrente contínua com tiristores confiáveis, de baixo custo e manutenção simples, substituíram os grupos conversores rotativos. Com isso, motores de corrente contínua passaram a constituir alternativa mais atrativa em uma série de aplicações.

2.7 Funcionamento do motor DC

Todos os motores elétricos valem-se dos princípios do eletromagnetismo, mediante os quais condutores situados num campo magnético e atravessados por correntes elétricas sofrem a ação de uma força mecânica, ou eletroímãs exercem forças de atração ou repulsão sobre outros materiais magnéticos. Na verdade, um campo magnético pode exercer força sobre cargas elétricas em movimento. Como uma corrente elétrica é um fluxo de cargas elétricas em movimento num condutor, conclui-se que todo condutor percorrido por uma corrente elétrica, imerso num campo magnético, pode sofrer a ação de uma força. Num motor há dois eletroímãs em que um impulsiona o outro.

2.8 Comunicação sem fio

A comunicação sem fio proporciona maior flexibilidade que os meios de comunicação baseados em cabos possibilitando a transmissão entre pontos equidistantes. Contudo, há algumas desvantagens. É limitada pelo alcance do transmissor (a que distância o sinal pode ser recebido). Na comunicação sem fio, a mensagem ou o dado, são convertidos em sinais elétricos por um equipamento chamado transmissor e transmitido por ondas eletromagnéticas. Este utiliza um oscilador para gerar ondas de rádio que se propagam pelo espaço. O transmissor modula a onda para que essa carregue o sinal eletrônico modificado, através do espaço, onde é captado pela antena de um receptor. O receptor decodifica ou demodula a onda de rádio e um dispositivo irá interpretá-la.

Sistemas de comunicação sem fio na maioria dos casos envolvem transmissões unidirecionais, em que os dados são enviados por um dispositivo e recebido por outro, como por exemplo as redes de televisão. Transmissões bidirecionais exigem equipamentos que funcionam tanto como um transmissor, quanto um receptor, para o envio e recepção de sinais. O dispositivo que funciona como transmissor e receptor é chamado de transceptor. Telefones celulares e rádios bidirecionais utilizam transceptores, a fim de que possam

manter a comunicação em duas vias. Os primeiros transceptores eram muito grandes, mas seu tamanho diminuiu devido aos avanços tecnológicos e a miniaturização de componentes eletrônicos. Sistemas de comunicação sem fio cresce e mudam à medida em que a tecnologia avança. Hoje, utilizam-se vários sistemas, todos operando em frequências de rádio distintas. Estão em desenvolvimento novas tecnologias para proporcionar mais serviços com maior confiabilidade. A figura 3 mostra o esquema elétrico de um transmissor simples de comunicação unidirecional.

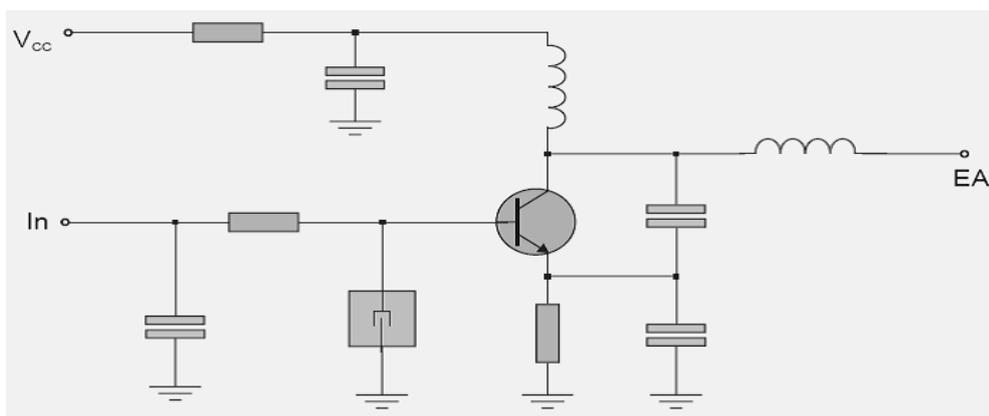


Figura 3: Esquema elétrico de um Transmissor básico

A figura 4 mostra o diagrama em blocos de um receptor simples.

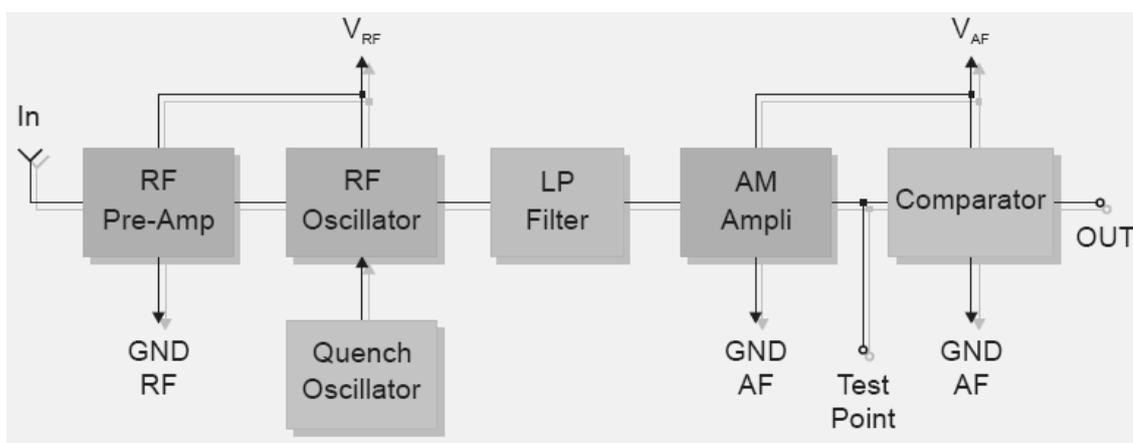


Figura 4: Diagrama em blocos de um Receptor

2.9 Java

Java é uma linguagem de programação orientada a objetos, desenvolvida por uma equipe na *Sun Microsystems*. Inicialmente foi elaborada para ser a linguagem base de projetos de software para produtos eletrônicos, mas com a explosão da Internet à partir de 1995 Java passou a ser amplamente devido ao sucesso mundial da internet. É uma linguagem robusta, multiplataforma e eficiente para programação de dispositivos móveis e eletrônicos. Seu sucesso se deve, sobretudo, a grande colaboração de desenvolvedores de todo mundo que escrevem artigos, tutoriais bem como a inclusão pelas instituições de ensino dessa linguagem em algumas disciplinas nos cursos de graduação.

2.10 Conclusão do capítulo

Nesse capítulo foi mostrado algumas características da porta paralela que fará comunicação entre o computador e a interface de hardware, bem como endereços, registradores e a funções de cada pino. Características e funcionamento de motores de passo e motores linear, vantagens e tipo de controle. Foi apresentado também o meio de transmissão através de ondas de rádio e a justificativa da linguagem de programação escolhida.

CAPÍTULO 3

CARACTERÍSTICAS DOS COMPONENTES ELETRÔNICOS

Para que haja um melhor entendimento do projeto, é necessário uma breve descrição do funcionamento básico dos componentes eletrônicos.

3.1 Buffer 74LS244

O circuito integrado 74LS244, conforme folha de dados, [13] é um buffer de oito linhas de dados e três estados não invertidos, projetado para ser empregado como *drivers* de endereçamento de memória, sendo muito utilizado na indústria e equipamentos de consumo. No projeto é utilizado como um *driver* de proteção para a porta paralela. Possui 20 pinos sendo 8 pinos de entrada (pinos pares exceto 10 e 20), 8 pinos de saída (pinos ímpares exceto 1 e 19), 2 pinos de controle (pinos 1 e 19) e 2 pinos para alimentação do circuito (pinos 10 e 20) como mostrado na figura 5.

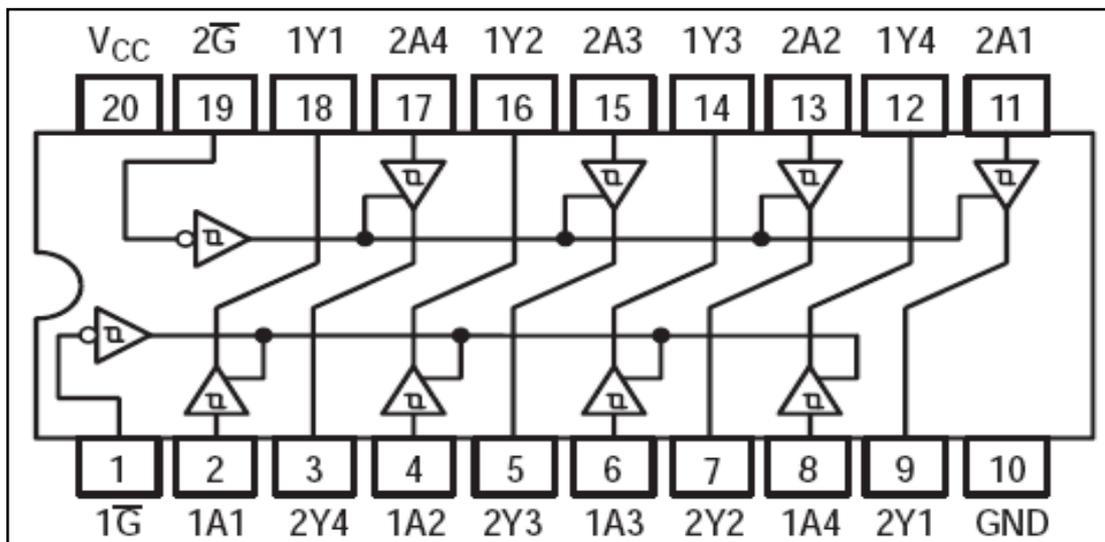


Figura 5: Diagrama interno do Buffer 74LS244

Os valores dos pinos de saídas são dependentes das entradas dos pinos 1 e 19. Para que os valores dos pinos de entradas estejam presentes nos pinos de saídas é necessário que o pino 1 e 19 estejam em nível baixo. O pino 1 controla as entradas pares de 2 a 8 e o pino 19 controla as entradas pares de 12 a 18, caso seja levado a nível alto (vcc), independente dos valores de entrada, as saídas estarão em estado de alta impedância como mostra a tabela 3. Para o projeto proposto este estado é irrelevante, portanto esse estado não será usado.

Tabela 3: Dos pinos do CI 74LS244

Controle 1G ; 2G	Entradas 1A _x , 2A _y	Saída
L	L	L
L	H	H
H	X	(Z)

Nota:

- L = Low Level – Nível baixo
- H = High Level – Nível Alto
- Z = High Impendace – Alta Impedância
- X = Don't Care – Valor Irrelevante

3.2 Codificador MC145026

O MC145026 é um codificador utilizado largamente em projetos de aplicações controladas remotamente. Ele possui nove linhas de informações paralelas que são enviadas serialmente. As nove linhas são codificadas em modo binário com estados (baixo e alto) podendo combinar $2^5 = 32$ endereços com quatro bits de dados ou $2^9 = 512$ endereços sem nenhum bit de dados. No modo trinário com estados (baixo, alto e aberto) é possível combinar $3^5 = 243$

endereços com quatro bits de dados ou combinar $3^9 = 19.683$ endereços sem nenhum bit para dados. Para aumentar a segurança e confiança na transmissão de dados, cada palavra é transmitida duas vezes. A transmissão é efetuada três pulsos de relógio após primeira palavra ser transmitida.

3.3 Descrição dos pinos do Codificador MC145026

Para um melhor compreensão das descrições dos pinos a figura 6 ilustra o diagramas em blocos do MC145026. Todas suas características e especificações estão contidas na folha de dados do fabricante.

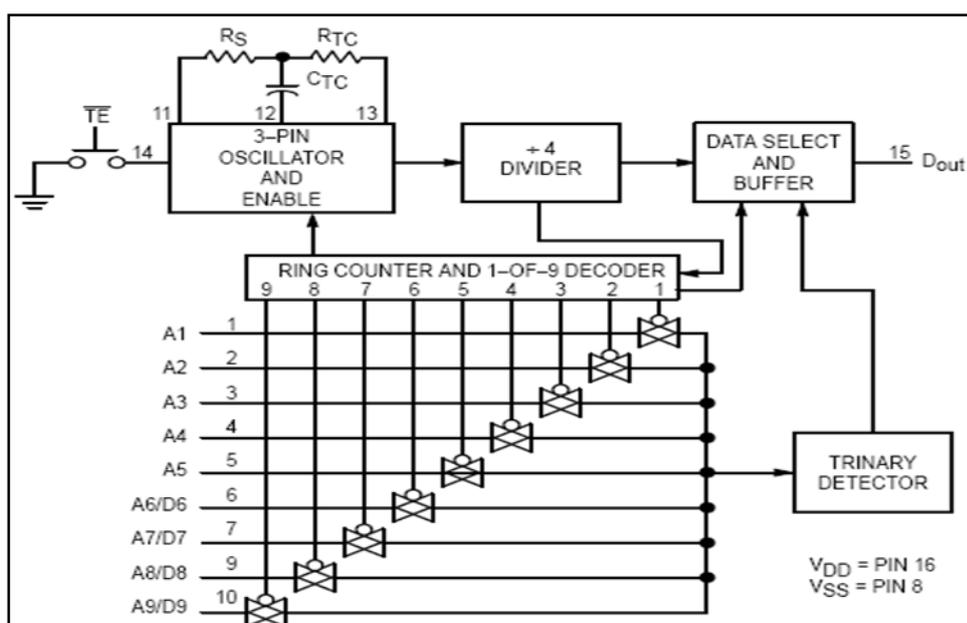


Figura 6: Diagrama em blocos do MC145026

- Os pinos de A1-A5 são específicos para endereços, os pinos A6/D6 – A9/D9 podem ser utilizados tanto para endereços quanto para dados. A letra “A” significa “adress” (endereço), a letra “D” significa “data” (dados) e os números significa a posição do bit.

- RS, CTC, e RTC (pinos 11, 12 e 13) fazem partes do oscilador do decodificador. As fórmulas para cálculos desses componentes encontram-se na folha de dados do fabricante. Os valores dependem da largura de banda do canal de comunicação, transmissor/receptor.
- TE *transmit enable* (pino 14), este pino tem a lógica invertida, ou seja, quando está em nível baixo, habilita a transmissão dos dados de entrada quando, levado para nível alto, desabilita a transmissão.
- Data *out* (pino 15) é por esse pino que saem de forma serial as palavras de dados disponíveis nos pinos de entrada.
- V_{SS} , *Negative Power Supply* (pino 8) este é o pino ligado ao potencial mais negativo da fonte, conhecido também com Gnd (ground).
- V_{DD} , *Positive Power Supply* (pino 16) este pino é ligado ao potencial mais positivo da fonte de alimentação.

3.4 Codificador MC145027

Esse decodificador é um dispositivo eletrônico projetado para ser usado em aplicações onde haja utilização de endereçamento como controle remoto. O decodificador recebe um dado serial de 9 bits e interpreta os cinco primeiros bits como um código de endereço. Se o endereçamento for trinário é possível combinar $3^5 = 243$ endereços, caso seja binários, é possível combinar $2^5 = 32$ endereços. Os outros 4 bits é interpretado como sendo dados. Ao contrário do MC145026, no MC145027 não é possível utilizar todos os 9 pinos para endereçamento.

Quando uma sequência de nove bits é enviada para o decodificador, é verificado se o endereço (5 primeiros bits) contido na sequência é o mesmo, que está definido no decodificador. Esta configuração é estática, ou seja, não muda.

3.5 Descrição da pinagem do MC145027

A figura 7 ilustra o diagrama em bloco do MC145027, para que se tenha uma noção mais clara das descrições de cada pino do CI, entretanto, todas as especificações podem ser encontradas na folha de dados do fabricante.

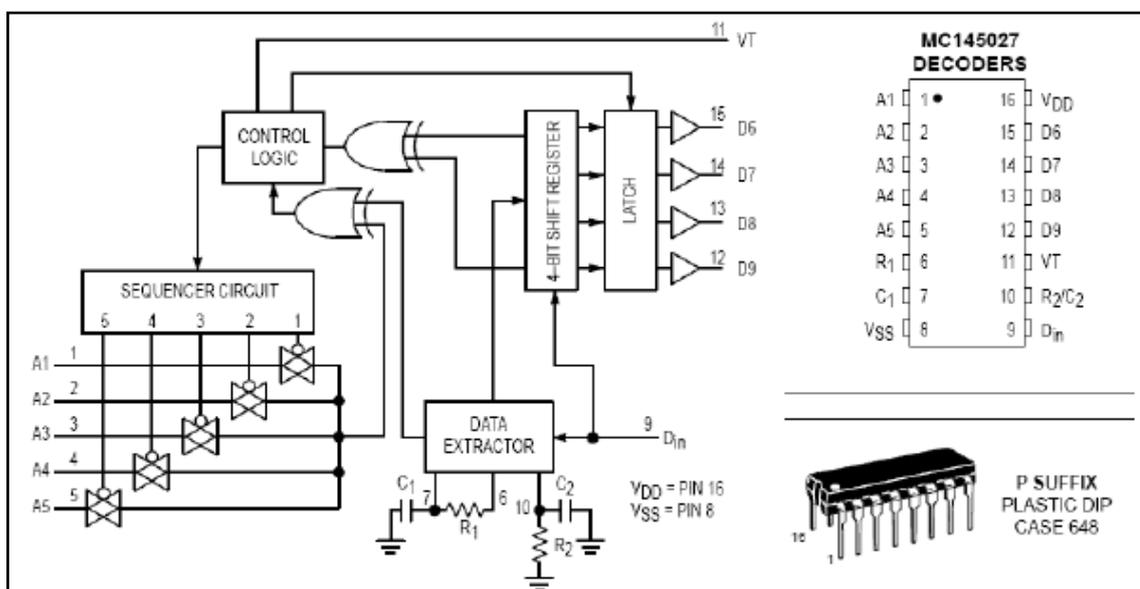


Figura 7: Diagrama em bloco, layout e encapsulamento CI MC145027

- A1-A5 (pinos de 1-5), são os pinos de entradas de endereço. Estes pinos precisam estar devidamente codificados na entrada para que o pino VT (*valid transmission*) seja ativado. Os endereços podem ser tantos binários (baixo, alto) como trinários (baixo, alto, aberto).
- D6-D9 (pinos 15-12), Estes pinos recebem as informações binárias que estão na entrada de dados do codificador. Estes pinos só aceitam dados binários, caso um dado trinário esteja na entrada do codificador é aceito como nível alto (lógico 1).
- Din – Data in (pino 9). Este pino é entrada serial de dados para decodificador. A entrada do sinal precisa estar com nível de amplitude TTL.

- R_1, C_1 – Resistor 1, Capacitor 1 (pinos 6 e 7). Estes capacitores são usados para determinar a largura de pulso dos dados que estão sendo recebidos. O cálculo desses valores está relacionado com a taxa de transmissão dos dados. As fórmulas para cálculos desses componentes estão na folha dados do fabricante.
- R_2/C_2 – Resistor 2 / Capacitor 2 (pinos 10). Neste pino são conectados o resistor e capacitor que serão usados para detectar o fim da palavra transmitida como também o fim da transmissão. As fórmulas para cálculos desses componentes estão na folha dados do fabricante e tem relação com os componentes do oscilador do codificador.
- VT – *Valid trasmission output* (pino 11). Para que uma transmissão seja válida é necessário duas condições:
- Primeiro, 2 endereços precisam ser recebidos consecutivamente e que ambos contenha o mesmo endereço. Segundo, que os 4 bits de dados precisam ser válidos na último dado recebido. Sendo assim, o pino VT é ativado e indica que a informação nos pinos saída de dados está sendo atualizada.
- VSS, *Negative Power Supply* (pino 8) este é o pino ligado ao potencial mais negativo da fonte, conhecido também com Gnd (*ground*).
- VDD, *Positive Power Supply* (pino 16) este pino é ligado ao potencial mais positivo da fonte de alimentação.

3.6 Driver controlador de motor DC

Trata-se de um circuito utilizado para controlar um motor DC a partir de sinais gerados por um micro-controlador ou um circuito com tal função. Devido à disposição dos seus componentes, torna-se extremamente fácil selecionar o sentido da rotação de um motor, apenas invertendo a polaridade sobre seus terminais. Também é importante para a utilização com circuitos digitais, no caso desse projeto, como os sinais de saída do decodificador não

suportam a corrente necessária e nem possui a tensão adequada para acionar um motor, é necessária uma unidade de potência que possa alimentá-lo de forma correta.

Quando um motor DC é ligado a uma bateria, observa-se que ele gira em uma única direção. Para alterar o sentido da rotação do motor, basta apenas ligar os terminais do motor de forma invertida. Para que não seja necessário fazer essa operação manualmente, o circuito em forma de uma ponte H é utilizado. Uma ponte H básica é composta por 4 chaves posicionadas formando a letra “H” como mostra a figura 8, sendo que cada uma localiza-se num extremo e o motor é posicionado no meio. Quando o fluxo de corrente vai de “A” para “D” o motor gira num sentido, horário. Quando o fluxo de corrente vai de “B” para “C”, o motor gira no sentido inverso.

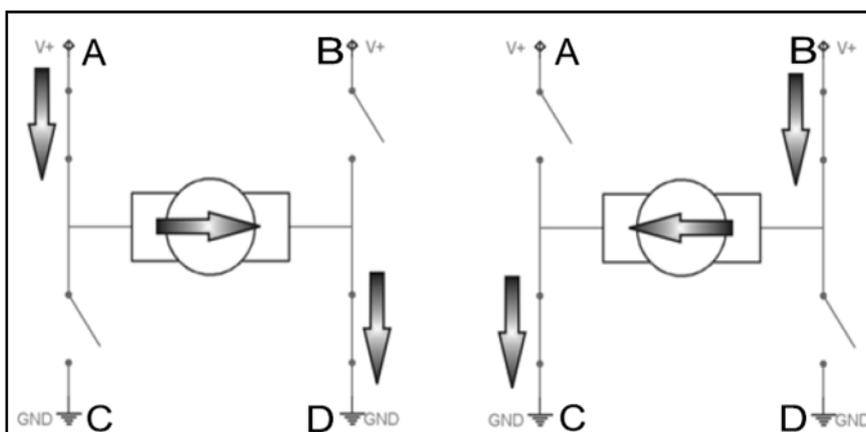


Figura 8: Sentido de Fluxo de corrente de uma ponte H

Para controle dos motores de deslocamento do veículo proposto, o CI L298 é usado. Esse CI possui duas pontes H que vai controlar dois motores. O CI L298 é um circuito integrado que pode ser alimentado com até 46 volts e suporta até 4 ampers de corrente contínua. As figuras 9 e 10 mostram a disposição dos pinos desse CI e em seguida a descrição da função de cada um. Mais detalhes podem ser encontrados na folha de dados do fabricante.

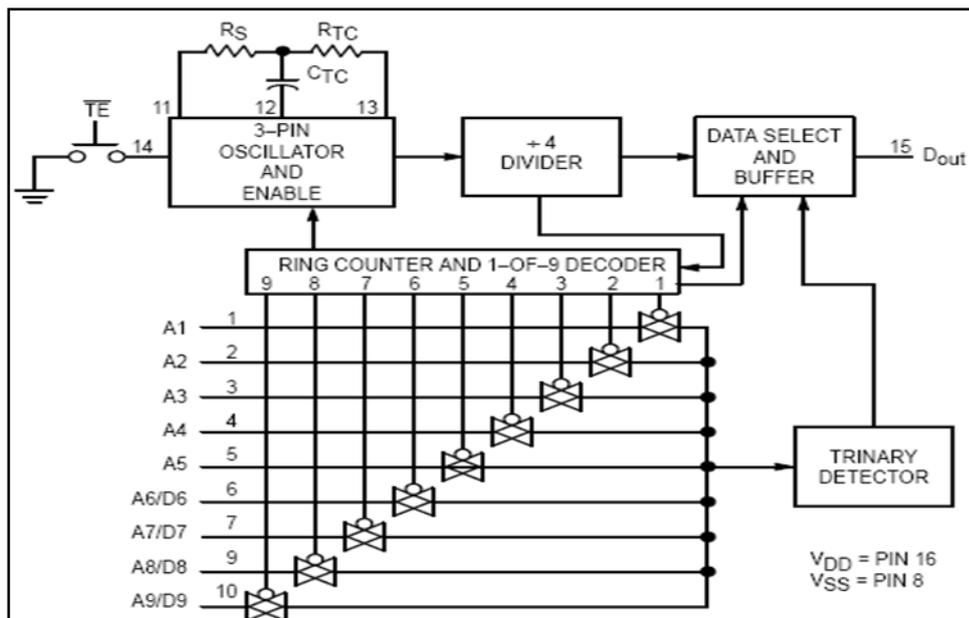


Figura 9: Diagrama em bloco do CI L298

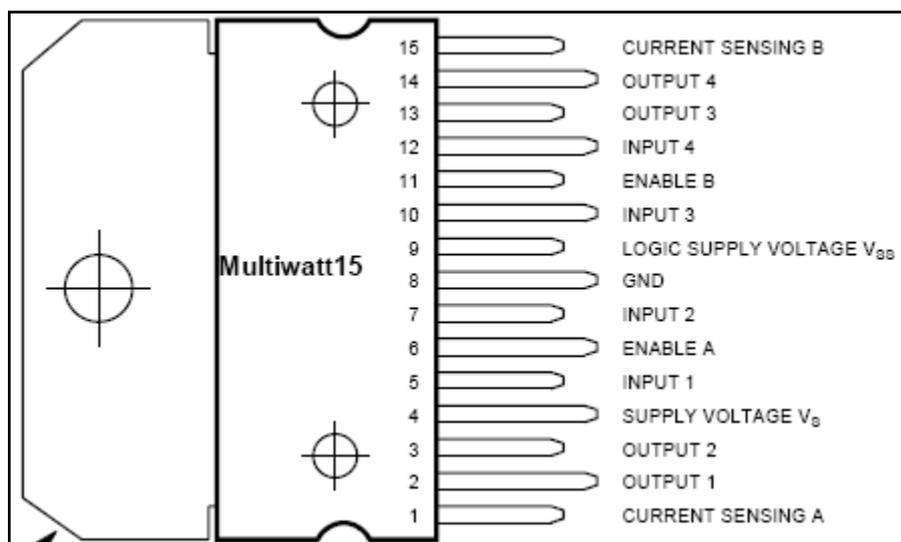


Figura 10: Pinagem do CI L298

A tabela 4 apresenta uma descrição sucinta das funções de cada pino do CI L298.

Tabela 4: Descrição dos pinos do CI I298

Nº do pino	Descrição e função do pino
1;15	Sensor A ; Sensor B. Pinos que controlam a corrente de carga.
2;3	Saída 1 ; Saída 2. Saída da ponte A. A corrente de carga desses pinos é controlada no pino 1
4	Entrada para alimentação do dispositivo controlado pelos pinos 2 e 3
5;7	Entradas que controlam a ponte A
6;11	Entrada que habilita e desabilita a ponte A (pino 6) e ponte B (pino 11)
8	Ligado ao pólo negativo da fonte de alimentação
9	Alimentação de 5 V para circuito interno da ponte H
10;12	Sensor A ; Sensor B. Pinos que controlam a corrente de carga.
13;14	Saída 1 ; Saída 2. Saída da ponte B. A corrente de carga desses pinos é controlada no pino 15

A tabela 5 apresenta comportamento do motor DC em função dos valores que devem ser colocados nas entradas IN1, IN2 e En1. A tabela representa os valores para a ponte A, mas o comportamento da ponte B é o mesmo.

Tabela 5: Valores de entradas para ponte A

En	Entradas IN	Saídas
	$IN_1 = 1; IN_2 = 0$	Motor cc gira em determinado sentido
$EN_1 = 1$	$IN_1 = 0; IN_2 = 1$	Motor cc gira em sentido oposto
	$IN_1 = IN_2$	Motor para de forma mais rápida
$EN_1 = 0$	$IN_1 = 1 \text{ ou } 0; IN_2 = 1 \text{ ou } 0$	Motor para de forma rápida

3.7 Módulo transmissor TXA1-434-F11

Módulo de Rádio com ressonador SAW e antena externa. É um circuito híbrido que permite transmissão completa de rádio ligado a um circuito codificador. Aplicações: acionamento remoto, alarmes automotivos, sistemas de segurança residencial, monitoramento.

O transmissor da figura 11 opera na freqüência de 433,92 MHz, e possui uma taxa de transmissão de no máximo 3 KB/s. A fonte de alimentação é de 3V -12V.

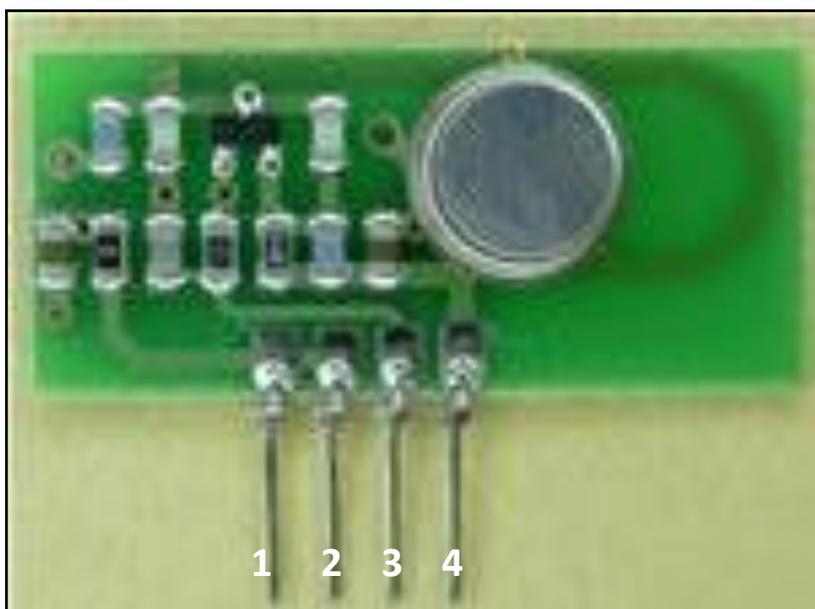


Figura 11: Identificação dos pinos do transmissor

Os pinos vem definidos com a seguinte configuração: Pino 1 – Antena; Pino 2 GND; Pino 3 – VCC; Pino 4 – Dados recebidos da saída serial do codificador.

3.8 Receptor RWS-374

A figura 13 mostra o RWS-374 que é um dispositivo de recepção que opera com uma freqüência de 433,92 MHz. Possui uma taxa de recepção de no máximo 3 KB/s e uma sensibilidade típica de -108 dBm, modulação em

Amplitude Modulada e requer alimentação de 5 volts DC conforme dados do fabricante. A figura 12 mostra a disposição dos pinos.



Figura 12: Transmissor RWS-374

A tabela 6 mostra a descrição e funções de cada pino.

Tabela 6: Pinos do RWS-374

Nº do pino	Descrição
1,6 e 7	Conectados ao terminal mais negativo da fonte alimentação
2	Pino de saída de dados em modo digital
3	Pino para saída linear de dados
4 e 5	Alimentação de 5 volts DC
8	Antena de recepção de sinal

3.9 Configuração do sistema operacional

Nas versões XP, NT e 2000 dos sistemas operacionais da Microsoft é implementado uma política de segurança que impede o usuário de trabalhar diretamente com a porta paralela. Neste texto, será mostrado como liberar o acesso para que o projeto funcione devidamente. A solução para esta situação dar-se através de *software*. Bibliotecas diferentes devem ser utilizadas na hora de implementar as rotinas para que não haja problemas de segurança. No

entanto, vários softwares já estão prontos com versões desenvolvidas especificamente para suprir estas necessidades. Um programa utilizado para esse propósito é o "Userport", desenvolvido por Tomas Franzon. Para facilitar, segue os passos para que os programas funcionem corretamente:

- 1º) Baixe o arquivo em <http://www.eletronica.org/arquivos/userport.zip>.
- 2º) Descompacte o arquivo em um diretório (pasta) de sua preferência.
- 3º) Acesse o computador com privilégio administrativo, abra o diretório onde o arquivo foi descompactado.
- 4º) Copie o arquivo *UserPort.sys* para o diretório `c:/windows/system32/drivers` (ou `c:/winnt/system32/drivers` para o Windows NT).
- 5º) Execute o programa *UserPort.exe*.
- 6º) Preencha o campo para endereços com `0x378-0x37A`, clique em "ADD".
- 7º) Selecione os outros endereços e clique no botão "Remove".
- 8º) Clique em "Start" para iniciar .

Com estes passos, o seu sistema operacional estará "desbloqueado" para acesso direto a porta paralela. O procedimento só é necessário uma única vez. Caso deseje voltar a condição anterior, basta executar o 5º passo novamente e clicar em "STOP".

3.10 Conclusão do capítulo 3

Neste capítulo foi apresentado as características e funcionamento básico de cada componente de hardware utilizado no projeto. A descrição e funcionalidade de cada pino que compõe cada circuito integrado, a configuração do sistema operacional para que ter acesso direto à porta paralela.

CAPITULO 4

MONTAGEM E FUNCIONAMENTO DO PROJETO

Existem algumas técnicas para se fazer placas de circuito impresso. Desenhando manualmente as trilhas na placa de fenolite com uma caneta especial, (um pouco ultrapassado) ou transferindo o desenho do circuito para a referida placa, através do processo de aquecimento da impressão feita em papel *glossy* ou folha de transparência sobre a placa. Neste projeto foi usado o processo de transferência do desenho do circuito utilizando a impressão sob transparência. Para gerar o *layout* da placa utilizar-se-á qualquer *software* que realize este trabalho. O PCBWizard 3.6 é um excelente *software* para essa atividade. Após gerar o layout da placa e imprima o circuito. Para a impressão, utilize uma impressora laser ou fotocópia do circuito (qualidade inferior) sobre uma transparência apropriada para esse tipo de impressora, ou papel *glossy*. Nas configurações da impressora é necessário alterar para transparência ou alta gramatura o tipo de papel.

4.1 Preparação da placa

Limpe a superfície recoberta de cobre, utilizando uma esponja de aço umedecida em álcool isopropílico, como na figura 13.



Figura 13: Limpeza da superfície de cobre

Após terminar a limpeza, enxágüe a placa com água, seque-a com um pano macio (uma toalha limpa é perfeita para esse serviço) e guardar em local onde não haja sujeira ou poeira. Tenha certeza que não haja nenhum resquício de gordura na placa, senão a impressão não irá fixar perfeitamente, surgindo falhas posteriores à corrosão. Recorte o desenho do circuito, cole sobre a placa, cubra a transparência com uma folha de papel ofício sobre essa transparência e aquecer com ferro de passar roupa em temperatura média por uns 02 (dois) minutos. Deixe a temperatura da placa baixar e então mergulhe-a na água e retire a transparência ou papel.

4.2 Preparação da solução

Para preparar a solução são necessárias as substâncias com as seguintes características:

Tabela 7: Substâncias que compõem a solução

Quantidade	Substancias
350 ml	Água
100 ml	HCl – 45% de pureza (Ácido Muriático)
20 ml	H ₂ O ₂ – 50% de pureza (Água Oxigenada)

Para lidar com substâncias químicas é necessário tomar algumas medidas de segurança para evitar acidentes. Use equipamentos de proteção como luva e máscara.

Para preparar a solução, coloque a água em um recipiente (de preferência que comporte placa a ser corroída) acrescente o HCl (ácido muriático), em seguida adicione H₂O₂ (água oxigenada) e agite bem a mistura. Coloque a placa dentro da solução, aguarde entre 5 (cinco) a 10 (dez) minutos. Após esse período, verifique se realmente o processo de corrosão foi concluído então limpe a placa com esponja de aço para retirada da tinta que recobre as trilhas. Utilizando um perfurador ou furadeira (com broca de 9mm) fure os locais onde serão inseridos os componentes.

4.3 Montagem do projeto

Depois de preparar a placa de circuito impresso, separe os componentes que serão utilizados em cada um dos módulos (transmissor / receptor), observando os valores corretamente e quantidades de cada um. Com a ajuda do software que utilizou para projetar o circuito, imprima um protótipo do projeto, para auxiliar o processo de montagem. Para evitar qualquer imprevisto, é recomendado o uso de soquetes com a mesma quantidade de pinos dos CI's, evitando que os componentes sejam soldados diretamente na placa. Este processo facilita a remoção quando for o caso e evitando exposição dos terminais do CI à alta temperatura do ferro de solda, o que pode levar a queima e inutilização do componente.

A figura 14 mostra o protótipo da placa de circuito impresso do transmissor e do receptor.

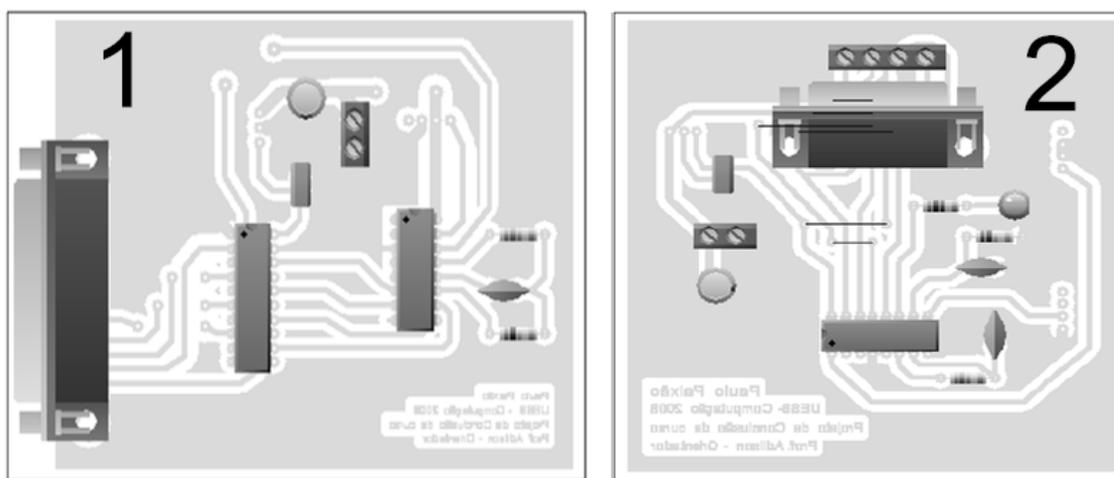


Figura 14: Protótipo transmissor (figura 1) e Receptor (figura 2)

4.4 Processo de soldagem

Coloque cada componente no devido lugar observando a polaridade e posição, aqueça os terminais e as trilhas, encoste a solda entre o terminal do componente e a trilha, aplique a quantidade de solda suficiente e então retire o

ferro. Repita para todos os componentes e após certificar que todos os componentes foram soldados, verificar se todos estão na posição correta, bem como inspecionar os terminais para certificar de que não existe nenhum terminal com excesso de solda (o que pode provocar curto-circuito) ou sem solda (acarretando mal contato).

Encaixe cada componente no respectivo soquete ligue e verifique o se está funcionando corretamente

4.5 Sistema de redução

Nesta seção será descrito o sistema de redução dos motores utilizado neste projeto. O motor será controlado pelo circuito do componenete L298. Para a montagem mecânica, será necessário definir o princípio do funcionamento do mecanismo.

Motores elétricos de pequeno porte encontrados em impressoras e brinquedos ou aparelhos eletrônicos rodam a uma velocidade muito grande, e possui pouco torque. Nessas condições caso o motor seja montado diretamente na roda do veículo, conseguirá uma velocidade considerável, porém, pode não haver força suficiente para proporcionar o movimento. Nessa situação é necessário utilizar um mecanismo de redução, podendo ser sistema de engrenagem ou sistema de polia. O princípio da redução é bem simples, quando diminuir a velocidade e aumentar o torque, usa-se uma polia ou engrenagem pequena (motor) conjugada a uma polia ou engrenagem grande (roda). O aumento de torque é devido ao efeito de alavanca; quanto maior o diâmetro da polia, maior a força de torque. O princípio básico do uso de uma polia é transferir uma força aplicada em uma corda para um vetor diferente, utilizando-se uma polia fixada num determinado ponto. As polias são basicamente um disco que gira livremente em um eixo, com um sulco no meio, por onde passa uma correia.

4.6 Transmissão de dados

A biblioteca Parallport possui apenas dois métodos que pode ser acessado pelo usuário, os métodos “read () e write ()” passando como argumento um número inteiro. Toda a parte de acesso de baixo nível à porta paralela é feito pela biblioteca “parport.dll” que ficará na mesma pasta do projeto. As teclas de direcionamento do teclado serão usadas para dar a direção do robô. Ao pressionar umas das teclas, o programa envia para o para a porta paralela o equivalente em bits para controle de cada motor. Somente os pinos 1, 2, 3, 4 e 5 da porta paralela são usados. O dado enviado através do pino 1, habilita o envio dos dados recebidos pelo codificador tornando-os disponíveis na saída, enquanto os outros nos pinos 2, 3, 4, e 5 são os dados de controle. Esses valores serão enviados ao 74ls244 pelos pinos 2, 4, 6 e 8. Novamente os dados são repassados ao codificador MC145026 através dos pinos 12, 14, 16 e 18. Os bits são concatenados aos bits de endereço e enviados ao transmissor serialmente. O dado de nove bits recebido pelo transmissor se propaga pelo espaço através de ondas eletromagnéticas.

4.7 Recepção de dados

O sinal propagado chega ao receptor, é convertido em um dado digital de nove bits e encaminhado ao decodificador através do pino 9. Os cinco bits de endereço são comparados ao configurado no decodificador (A1-A5 que correspondem aos pinos 1, 2, 3, 4 e 5), se forem iguais após duas transmissões, os bits de dados são enviados ao CI L298 responsável pelo controle dos motores, ou seja, movimentação do robô.

Para controlar as duas pontes do CI L298 é necessário 6 bits, 2 para habilitar o “enable” de cada porta H e quatro para controlar as entradas IN₁, IN₂, da ponte H1 e IN₃, IN₄ da ponte H2.

4.8 Visualização de imagens

A câmera colocada no robô permite a visualização das imagens em tempo real, bem como a gravação do conteúdo transmitido para visualização e

análise posterior. A transmissão se dará através de transmissor próprio contido na câmera. Um receptor com saídas de áudio e vídeo envia a imagens recebidas para o computador ou um aparelho de TV. As imagens enviadas para o computador é visualizadas com um software específico do fabricante da câmera ou através softwares específicos por terceiros como o proposto nesse projeto. Para reproduzir a imagens, o software poderá ser desenvolvido com o framework JMF (Java Media Framework).

4.9 Conclusão do capítulo 4

Neste capítulo, foi apresentada uma descrição sucinta do processo de motagem dos componentes, a transmissão das informações, visualização das imagens e funcionamento do dispositivo.

CAPITULO 5

CONCLUSÃO E PROPOSTA DE TRABALHOS FUTUROS

Desenvolver um projeto de final de curso não é tarefa fácil. Discutir conceitos e aplicações, já traduz o sentido de complexidade de um projeto. Idealizar e implementar dispositivos de hardware fomenta o sentimento de que não será possível finalizá-lo ou mesmo completá-lo com as mínimas funções pretendidas no tempo predeterminado. Pensar que o projeto não irá dar certo, é ter quase certeza de que realmente não terá êxito. É preciso pensar positivo, acreditar, ter persistência, confiança, apoio, pesquisar e dedicar algumas horas de trabalho. As dificuldades desse trabalho de final de curso, foram superadas com muita confiança.

Inicialmente, foi feito um estudo do que seria necessário para compor a estrutura física do *hardware*, tecnologias viáveis, custo e onde comprar os componentes necessários. No mercado local só se comercializa componentes básicos, o que obriga a comprá-los em mercado de outro estado. Entre as possibilidades de interface de comunicação, utilizar a porta paralela tornaria o custo mais viável. A utilização da interface *USB* nos dias atuais é mais relevante devido ao fato de todos os computadores modernos possuírem várias interfaces desse tipo disponíveis, entretanto, aumentaria o custo em dinheiro e demandaria mais tempo de estudo para poder manipulá-la. Os dispositivos para captura de imagens foram conseguidos por empréstimo reduzindo substancialmente o custo final. Os motores e engrenagens foram retirados de sucata de impressoras.

Na montagem do hardware não houve problemas relevantes. Na fase de projeto houve alguns equívocos quanto ao desenho do circuito elétrico que gerou mal funcionamento. As alterações foram feitas e os problemas foram minimizados. A grande dificuldade foi escrever a monografia. O conteúdo sempre fica aquém do esperado para cada tópico. Quanto a desenvolvimento

do *software*, não houve tempo hábil para implementação da interface de captura, controle e visualização das imagens. Os dispositivos como câmeras e placas de captura não são padronizadas e o programa básico feito na linguagem Java não conseguiu capturar o dispositivo. A utilização de Java como linguagem de programação amplia as possibilidades de controle através da internet e software para manipulação das imagens capturadas.

Durante a implementação de um trabalho surgem outras idéias capaz de agregar novas funcionalidades. É preciso manter a idéia original e fazer mudanças conscientemente, para não fugir da proposta inicial, pois o descontrole nas mudanças torna o trabalho dispendioso e de custo elevado. Este trabalho amplia os conhecimentos de eletrônica e desperta o interesse por outras áreas afim, com programação embarcada e desenvolvimento de sistemas para coleta remota de dados

As propostas de trabalho futuro poderão viabilizar estudos mais aprofundados em mecatrônica, inteligência artificial, desenvolvimento e programação de hardware.

- Desenvolver um sistema de deslocamento autônomo dotando-o de inteligência artificial.
- Adequar aos projetos de futebol de robôs ou robôs gladiadores.
- Agregar sensores de proximidades, substância e calor para coletar informações.
- Braços manipuladores capazes de agarrar e mover objetos.
- Manipulação pela internet transmissão de imagens pela internet.
- Desenvolvimento do aplicativo de visualização das imagens junto com o de controle do robô.

REFERÊNCIAS

BARTKOWIAK, Robert A. **Circuitos elétricos / Robert A. Bartkowiak**; tradução Moema Sant'Anna Belo ; revisão técnica Antônio Pertence Junior. – São Paulo: Makron Books, 1994.

BARRADAS, Ovidio Cesar Machado. **Você e as telecomunicações / Ovídio César Machado Barradas**.- Rio de Janeiro : Interciência, 1995.

GROOVER, Mikell; WEISS, R. Mitchell. **Robótica: tecnologia e programação**. McGraw-Hill, São Paulo, 1988.

Harvey M. Deitel & Paul J. Deitel. **Java: Como Programar – 6. Ed – São Paulo: Pearson Prentice Hall, 2005.**

Malvino, Albert Paul. **Eletrônica Volume I – 1ª ed – São Paulo : McGraw-Hill, 1987.**

Lobosco, Orlando Silvio & Dias, José Luiz P. da Costa; **Seleção e Aplicação de motores Elétricos**; Makron Books – São Paulo 1997.

Apostilas e Ebooks . Disponível em:

<http://www.eletronica.org/arq_artigos/paralela.zip>. Acesso em Dez de 2007.

Interfaces e Protocolos – Eletrônica.org.- Disponível em:

<www2.eletronica.org/apostilas-e-ebooks/interfaces/apostila-paralela.pdf>.

Acesso em: Dez. 2007.

Porta Paralela – Disponível em:

<<http://www.jobtecltda.com.br/paralela/paralela.htm#Porta%20Paralela>>.

Acesso em Dez. 2007

Eletronica.org – **Estudo do Motor de Passo e seu controle digital – Motor AC**, disponível em:

<<http://www.eletronica.org/modules.php?name=News&file=article&sid=15>>.

Acesso: Jan de 2008.

Introdução a Porta Paralela – TUGATRÔNICA – Disponível em :

<http://www.tugatronica.com/iii-introducao-a-porta-paralela.php>>. Acesso em jan. 2008.

Filho, Antonio de S. R.; Ferreira, Davi Almeida; Sobral, Victor A. Leal;

Neto, Pedro W. de A. Farias - Motor AC – Disponível em:

<http://www.dee.eng.ufba.br/trabalho/HPs_ENG335_20062/Motor_AC/intro.htm
> Acesso em: Dez. de 2007.

Motor de passo – Wikipédia - disponível em:

<http://pt.wikipedia.org/wiki/Motor_de_passo>. Acesso em Dez. de 2008.

Robotica – Motor de passo: Disponível em:

<http://www.mrshp.hpg.ig.com.br/rob/m_passo.htm>. Acesso em Dez de 2008.

Silveira, Jader Garcia da – **Transmissão de dados via rádio**, Disponível em: <

<http://www.das.ufsc.br/gpri/pub/seminars/tdvr.pdf>> acesso em Jan 2008.

Abrindo LPT com Userport – Disponível em:

< <http://www.eletronica.org/arquivos/userport.zip>>. Acesso em Ago. de 2008.

Octal 3-STATE Buffer/Line Driver/Line Receiver - Disponível em:

<http://www.datasheetcatalog.org/datasheets/70/216095_DS.pdf>. Acesso em Ago. de 2008.

Encoder MC145026 / Decoder MC145027 - Disponível em:

<<http://www.datasheetcatalog.org/datasheet/motorola/SC41343DW.pdf>>.

Acesso em Ago. de 2008.

L298 - Dual Full-bridge Driver - Disponível em:

<<http://www.datasheetcatalog.org/datasheet2/2/052daje928cw7pc0uqs1ipyryppy.pdf>>. Acesso em ago. de 2008.

Radio Transmitter Module with SAW Resonator - Disponível em: <
<http://php2.twinner.com.tw/files/onshine/TXA1-434-F11.pdf>>. Acesso em Dez.
de 2008.

Wireless Hi Sensitivity Receiver Module – RWS-374 – Disponível em:
<<http://www.wenshing.com.tw/DATA/RF-MODULE/RWS-374-6.pdf>>. Acesso
em Dez. de 2008.

Apendice 1 – Lista de componentes

A Tabela 8 apresenta a lista de componentes utilizadas no projeto.

Tabela 8: Lista de Componentes

Quantidade	Descrição
01	Conector DB 25 fêmea 90°
01	CI SN74LS244N
01	Codificador MC145026
01	Decodificador MC145027
02	Capacitor Eletrolítico 100 µF x 35V
01	Capacitor Cerâmico 5,6 nF
01	Capacitor Cerâmico 22 nF
10	Capacitor Cerâmico 100 nF
02	Regulador de tensão L7805CV
01	Resistor ¼ W 200 Kohms
01	Resistor ¼ W 100 Kohms
02	Resistor ¼ W 51 Kohms
02	Resistor ¼ W 330 ohms
02	Led de 5 mm
02	Soquete para CI de 16 pinos
01	Soquete para CI de 20 pinos
01	Borne de 4 polos
01	Kit Transmissor/Receptor
01	CI Driver L298N
02	Rabicho p/ Bateria de 9V
01	Metro de fio de cabo de rede
50	Centímetros de Solda
02	Baterias de 9V
02	Motores CC

APENDICE 2 – Código fonte de controle

```

package meurobo;
import parport.ParallelPort;
    import java.awt.event.*;
    import java.lang.Thread;
    import java.awt.*;
    import javax.swing.*;
public class RoboParalelo extends JFrame implements KeyListener{

    ParallelPort bitport = new ParallelPort (0x378);
    ParallelPort bitenable = new ParallelPort (0x37A);
    Thread time ;

    JLabel l1 , l2;
    public RoboParalelo(){

        super("Controle RobOP");
        setSize(300,300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        l1 = new JLabel("");
        getContentPane().setLayout(new FlowLayout());
        addKeyListener(this); // aqui é importante
        getContentPane().add(l1,l2);
        setVisible(true);
    }
    public void keyPressed(KeyEvent e){
        // int keyState = getKeyStates();
        // bitenable.write(0);

        if (e.getKeyCode() == KeyEvent.VK_UP) {
            l1.setText("Movimento Progressivo");
            bitenable.write(1);
            bitport.write(14);

            // bitport.write(22);
        }
        if (e.getKeyCode() == KeyEvent.VK_DOWN) {
            l1.setText(" Movimento Regressivo");
            bitport.write(13);
            bitenable.write(1);
            // bitport.write(21);
            // bitenable.write(1);
        }
        if (e.getKeyCode() == KeyEvent.VK_RIGHT) {
            l1.setText("Curva para Direita -->");
            bitport.write(6);
            bitenable.write(1);
        }
    }
}

```

```

        if (e.getKeyCode() == KeyEvent.VK_LEFT)
        {
            l1.setText("Curva para
            Esquerda <--");
            bitport.write(5);
            bitenable.write(1);
        }
    }
    public void keyReleased(KeyEvent e){
        l1.setText("Estou parando... ");
        bitport.write(24);
        bitenable.write(1);
    }
    public void keyTyped(KeyEvent e){
    }
    public static void porta (){
    // bitnaporta.write(2);
    }
    public static void main(String[]args){
        RoboParalelo teste = new RoboParalelo();
        teste.bitport.write(0);
        teste.bitenable.write(1);
    }
}

```