

**Guilherme José Bissoli e Guilherme Pissaia Esteves**

note final  
9,5 (nove e cinco)  
W.M.

# **Robô Aspirador de Pó (Vacuum Cleaner Robot)**

Monografia do trabalho de formatura  
apresentado a Escola Politécnica da  
Universidade de São Paulo para obtenção do  
Diploma de Engenheiro Mecatrônico.

**São Paulo  
2004**

**Guilherme José Bissoli e Guilherme Pissaia Esteves**

# **Robô Aspirador de Pó (Vacuum Cleaner Robot)**

Monografia do trabalho de formatura  
apresentado a Escola Politécnica da  
Universidade de São Paulo para obtenção do  
Diploma de Engenheiro Mecatrônico.

Área de concentração:  
Engenharia Mecânica Automação e Sistemas

Orientador:  
Prof. Dr. Gilberto Francisco Martha de Souza

**São Paulo  
2004**

Dedicamos este trabalho à todos aqueles que sempre acreditaram no sucesso de nosso projeto, nos empolgando com sua admiração e nos entusiasmando em momentos de desespero. Obrigado por acreditarem em nós.  
Fica aqui o nosso legado!

## Agradecimentos

Ao amigo e orientador Prof. Dr. Gilberto Francisco Martha de Souza, pelo incentivo constante e inigualável apoio moral. Ao Prof. Dr. Jun Okamoto Junior, que nos auxiliou quando tudo parecia perdido. Ao Prof. Dr. Oswaldo Horikawa, o primeiro a acreditar em nós.

Eu, Guilherme Esteves, agradeço o apoio de toda a minha família, que sempre esteve ao meu lado, especialmente à minha namorada Graziela por não medir forças nos momentos mais dificeis e pelo enorme apoio.

Eu, Guilherme Bissoli, agradeço aos meus pais que, durante todo o projeto, acreditaram mais do que eu mesmo no resultado positivo do trabalho, ao meu irmão Eduardo que trouxe componentes essenciais para o projeto eletrônico e ao meu irmão Rodrigo que colaborou com sugestões positivas no meu método de trabalho. E, por fim, mas não menos importante, à Paola pelo incentivo nos momentos finais.

## RESUMO

Neste trabalho executam-se os projeto mecânico e eletro-eletrônico de um robô autônomo microcontrolado de pequeno porte para auxílio na limpeza doméstica através da aspiração de superfícies lisas e planas, capaz de sensoriar seu meio de atuação e otimizar seu deslocamento.

O robô é controlado por um microcontrolador Rabbit – Família RCM2000, seu deslocamento e aspiração são acionados por dois motores DC de 24V e por um motor CC de 35W, respectivamente. O sensoriamento é feito através de botões mecânicos tátteis e de emissores e receptores infravermelhos. A alimentação dos sistemas é feita por 3 baterias de chumbo-ácido de 12V, sendo uma de 9Ah e duas de 2,3Ah. Todos os demais circuitos são compostos de chips comerciais comumente encontrados no mercado.

O protótipo tem suas peças feitas em sua maioria de alumínio e possui um volume total inferior a 13 litros.

A autonomia mínima é de 40 minutos de funcionamento contínuo com todos os sistemas em operação, antecedido por um ciclo de carregamento.

## ABSTRACT

This project presents the electro-electronic, mechanical project and conception of a small-scale autonomous microcontrolled robot for domestic cleaning and floor vacuum of smooth and plane surfaces, capable of sensing its environment, guiding itself and its displacement.

The robot is controlled by a Rabbit microcontroller – Family RCM2000, its displacement and vacuum system are powered by two 24V DC engines and by a 35W continuous current motor , respectively. The environment sensing is done by mechanical touch buttons and by infrared emitters and receivers. The electrical systems are powered by 3 12V lead-acid batteries, one 9Ah and two 2,3Ah batteries. All circuits are structured by commercial chips easily found in the market.

The prototype has mostly aluminum parts and its gross volume should be under 13 liters.

The working autonomy is about 40 minutes of continuous operation, following a cycle of batteries charge.

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>9</b>
1.1	- DEFINIÇÃO DO PROBLEMA (REQUISITOS MÍNIMOS) .....	9
1.2	- ETAPAS DE PROJETO .....	11
1.3	- METAS FUTURAS .....	12
<b>2</b>	<b>ACIONAMENTO.....</b>	<b>14</b>
2.1	- ESCOLHA DO MOTOR .....	14
2.2	- O MOTOR ESCOLHIDO .....	17
2.3	- ACIONAMENTO DOS MOTORES .....	17
2.4	- O DRIVER DE POTÊNCIA .....	19
2.4.1	- Ponte H .....	19
2.4.2	- PWM ( <i>Pulse Width Modulation ou Modulação por Largura de Pulso</i> ) .....	22
2.4.3	- Conversor Digital/Analógico .....	23
2.5	- CIRCUITOS DO DRIVER DE POTÊNCIA .....	24
2.5.1	- Conversor Digital/Analógico .....	24
2.5.2	- Gerador de PWM .....	27
2.5.3	- Amplificador Operacional .....	30
2.5.4	- Inversor .....	32
2.5.5	- Circuito Regulador e Ponte H .....	33
2.5.6	- Conversor de tensão .....	36
2.6	- CIRCUITO FINAL .....	36
<b>3</b>	<b>SENSORIAMENTO.....</b>	<b>39</b>
3.1	- REDE DE SENsoRES .....	39
3.2	- PROPOSTAS DOS SENsoRES .....	42
3.3	- SOLUÇÃO PARA OS SENsoRES .....	48
3.4	- BOUNCING .....	51
3.5	- SENSOR ÓPTICO .....	52
3.5.1	- <i>Solução Escolhida</i> .....	54
<b>4</b>	<b>ENCODER.....</b>	<b>57</b>
4.1	- QUADRATURA .....	58
<b>5</b>	<b>SENSOR DE NÍVEL DE BATERIA .....</b>	<b>62</b>
<b>6</b>	<b>BATERIAS .....</b>	<b>64</b>
6.1	- REQUISITOS QUE A BATERIA DEVE APRESENTAR .....	64
6.2	- CARACTERÍSTICAS DAS BATERIAS .....	67
6.2.1	- <i>Voltagem</i> .....	67
6.2.2	- <i>Capacidade</i> .....	67
6.2.3	- <i>Densidade de energia</i> .....	68
6.2.4	- <i>Curva de descarga</i> .....	68
6.2.5	- <i>Resistência interna</i> .....	69
6.2.6	- <i>Recarga</i> .....	69
6.2.7	- <i>O efeito memória</i> .....	70
6.2.8	- <i>Custo</i> .....	70
6.2.9	- <i>Pacotes de baterias (Packs)</i> .....	71
6.2.10	- <i>Carregador de bateria</i> .....	71
6.3	- ESCOLHA DA BATERIA .....	73
6.4	- BATERIAS DE CHUMBO-ÁCIDO SELADAS .....	73
6.4.1	- <i>Visão geral</i> .....	73
6.4.2	- TAXA DE CARGA E DESCARGA .....	74
6.4.3	- <i>Carregador de qualidade e regulador</i> .....	75
6.4.4	- <i>Descarga profunda</i> .....	76
6.4.5	- <i>Capacidade das Baterias</i> .....	76
6.5	- CURVAS DAS BATERIAS .....	77
6.6	- AS BATERIAS ESCOLHIDAS .....	80

<b>7</b>	<b>MICROCONTROLADOR .....</b>	<b>82</b>
7.1 - RABBIT – FAMÍLIA RCM2000 .....	83	
7.2 - FAMÍLIA 80C51.....	85	
7.3 - ALTERNATIVA ESCOLHIDA .....	85	
<b>8</b>	<b>PROJETO ELETRÔNICO.....</b>	<b>86</b>
8.1 - PLACA DE POTÊNCIA E ACIONAMENTOS .....	88	
8.2 - PLACA DE ENCODER E SENSORES .....	91	
8.3 - PLACAS DOS SENSORES ÓPTICOS .....	93	
8.4 - INTERFACE.....	93	
8.5 - DRIVER DE POTÊNCIA.....	94	
<b>9</b>	<b>PROJETO MECÂNICO .....</b>	<b>95</b>
9.1 - SISTEMA DE LOCOMOÇÃO .....	95	
9.2 - SISTEMA DE LIMPEZA .....	101	
9.2.1 - <i>Sistema de Sucção .....</i>	<i>102</i>	
9.2.2 - <i>Sistema de limpeza com vassoura giratória .....</i>	<i>107</i>	
9.3 - SENSORES .....	113	
9.4 - BATERIAS .....	114	
9.5 - PROJETO MECÂNICO FINAL .....	116	
<b>10</b>	<b>PROJETO DE SOFTWARE .....</b>	<b>119</b>
10.1 - ANÁLISE DO SOFTWARE.....	119	
<b>DISCUSSÕES .....</b>	<b>121</b>	
<b>CONCLUSÕES.....</b>	<b>124</b>	
<b>REFERÊNCIAS.....</b>	<b>125</b>	
<b>ANEXO A -CONCEPÇÃO DO ROBÔ ASPIRADOR DE PÓ .....</b>	<b>128</b>	
<b>ANEXO B - OPÇÕES DE MOTORES ENCONTRADOS.....</b>	<b>130</b>	
<b>ANEXO C - DIAGRAMA ESQUEMÁTICO DOS CIRCUITOS .....</b>	<b>133</b>	
<b>ANEXO D - CIRCUITO DE POTÊNCIA E ACIONAMENTOS DO SISTEMA DE LIMPEZA E DO DRIVER DE POTÊNCIA .....</b>	<b>135</b>	
<b>ANEXO E - CIRCUITO DO DRIVER DE POTÊNCIA DOS MOTORES CC .....</b>	<b>137</b>	
<b>ANEXO F - CIRCUITO DOS ENCODERS E SENSORES .....</b>	<b>139</b>	
<b>ANEXO G - CIRCUITO DO CHICOTE DE SENSORES .....</b>	<b>141</b>	
<b>ANEXO H - CIRCUITO DE NÍVEL DE BATERIA E INTERFACE .....</b>	<b>143</b>	
<b>ANEXO I - SEQÜÊNCIA DE MONTAGEM .....</b>	<b>145</b>	
<b>ANEXO J - PROGRAMA EM C .....</b>	<b>151</b>	
<b>ANEXO K - DESENHOS TÉCNICOS .....</b>	<b>161</b>	

# 1 INTRODUÇÃO

## 1.1 - Definição do Problema (Requisitos mínimos)

Como idéia inicial desejamos desenvolver um robô capaz de se locomover autonomamente aspirando o pó de superfícies domésticas. A concepção inicial pode ser vista no último anexo do relatório (Anexo A – Concepção do Robô Aspirador de Pó).

A locomoção do robô deveria ser feita independentemente de qualquer atuação externa, tanto de controle, como fonte de alimentação ou ajuda humana. Durante a locomoção ele deveria desviar das paredes limites do ambiente em que estiver atuando ou de escadas, evitando sua queda. O robô também seria capaz de desviar de objetos que estejam em seu caminho principal de limpeza, retornando à sua rota inicial, ou garantindo a limpeza ao redor de tal obstáculo.

A locomoção seria garantida, a princípio, apenas em superfícies lisas, livres de buracos, reentrâncias ou líquidos, descartando assim a sua atuação em pisos que impliquem em dificuldades de locomoção, como por exemplo carpetes e tapetes, pedras, areia, terra, grama, ou qualquer outro relevo muito acidentado. A linearidade durante a locomoção não foi colocada como requisito mínimo, devido a fatores aleatórios de difícil controle, tendo-se que, para contornar tal problema, optou-se por trabalhar com controle em malha fechada, não só com o sensoriamento de presença de obstáculos, como também com um sistema auxiliar de identificação de posição em relação a um referencial fixo. Assim, a cobertura de limpeza do ambiente não seria garantida caso tal linearidade não fosse atendida e, para melhoria de tal erro, seria necessário incorporar na programação alguma correção para o acúmulo de erros que tal fato ocasionaria.

O sensoriamento seria realizado por sensores de toque, como chaves de fim de curso, ou através de sensores infravermelho, solução que será escolhida após análise de viabilidade.

O pó deveria ser coletado através de dois sistemas: um sistema de sucção presente nos aspiradores de pó comuns, e outro sistema para recolher partículas maiores através de uma vassoura giratória que utiliza o sistema conhecido popularmente por “vassoura mágica”. Os dois atuariam simultaneamente, independentemente das condições de sujeira oferecidas pela superfície a ser limpa. Os dois sistemas teriam controle apenas para ligar e desligar, e nenhum tipo de controle de intensidade. O pó recolhido seria ser armazenado em um compartimento no próprio robô, de fácil acesso para futura limpeza e esvaziamento do compartimento. Não seria necessário que os dois sistemas de limpeza utilizassem o mesmo compartimento para armazenagem da sujeira. O motor e ventoinha do sistema de sucção, bem como sua tubulação principal, seriam adquiridos a partir de um aspirador para carros existente no mercado, já que o desenvolvimento de tal sistema não é o foco do trabalho.

Para dar início à limpeza, o robô partiria de uma base, responsável pela recarga das baterias. Após a limpeza, ocorre seu retorno à base e, através de uma conexão, recarrega suas baterias. A conexão seria estabelecida de maneira totalmente independente.

A alimentação dos motores de locomoção, de sucção, da vassoura giratória, e do sistema eletrônico deveria ser feita por baterias, não necessariamente uma única.

O controle de todos os sistemas realizar-se-ia por um controlador, através de um circuito adquirido para o projeto, que seria definido conforme as necessidades que surgissem durante a fase de escolha dos motores e sensoriamento do robô. Tal circuito seria algo já existente no mercado. Os outros circuitos, como drivers, eventuais encoders, sensores infravermelhos e sensores de nível de bateria seriam construídos, visando minimizar a quantidade de componentes, deixando o sistema mais específico para o projeto, diminuindo o custo e complexidade do sistema.

A velocidade do robô seria definida pela potência do motor, o peso do robô e a transmissão utilizada, mas calculava-se aproximadamente de 8 a 20 cm/s. O robô deveria contar com um interruptor liga/desliga e outro para início da limpeza.

O projeto teria como objetivo final a construção do protótipo por inteiro, com todos seus sistemas (locomoção, sucção, controle, sensoriamento, alimentação, reservatórios, base, recarga), além de sua programação, chegando assim, ao final do ano, com um produto completo.

## 1.2 - Etapas de projeto

Para a realização do projeto, listou-se as etapas iniciais que deveriam ser realizadas até o projeto físico do robô.

1. Definir requisitos mínimos (limitações)
2. Definir metas futuras (requisitos que possam ser incorporados ao projeto após o alcance dos requisitos mínimos)
3. Estudo de motores de passo e motores CC
4. Escolha dos motores
5. Estudo dos circuitos de drivers
6. Estudo dos circuitos de encoders
7. Estudo dos circuitos de sensoriamento
8. Estudo dos circuitos de alimentação (conversão de tensão)
9. Estudo dos circuitos de nível de bateria
10. Estudo dos circuitos de recarregamento de baterias
11. Estudo do microcontrolador
12. Escolha do microcontrolador
13. Estudo e escolha de engrenagens (transmissão – redução)
14. Estudo e escolha de eixos, rodas, mancais
15. Analisar sistema de sucção
16. Estudar e escolher vassoura giratória
17. Definir circuitos de alimentação
18. Escolher baterias (uma ou mais dependendo da necessidade dos circuitos e motores)
19. Projeto dos circuitos
20. Projetar tubulação/reservatório
21. Projetar sistema da vassoura giratória

22. Projetar Chassi
23. Projetar Base de recarregamento
24. Desenhos técnicos
25. Construção do protótipo
26. Construção dos circuitos
27. Testes dos circuitos
28. Estudo de comunicação Microcontrolador-PC
29. Estudo do endereçamento de ports
30. Testes com o protótipo
31. Programação
32. Testes
33. Aperfeiçoamento (adição de funções)

A partir das tarefas listadas, foi possível criar um cronograma para a execução das tarefas realizadas no primeiro e segundo semestres de 2004.

### **1.3 - Metas futuras**

Após o alcance dos requisitos mínimos de projeto, algumas funcionalidades poderiam ser incorporadas ao robô aspirador.

Uma delas seria a modificação da programação e dos interruptores que formam a interface com o usuário de maneira a possibilitar a escolha de diferentes tipos de intensidade de limpeza desejada. Tais níveis seriam definidos como “limpeza rápida”, “limpeza normal” e “limpeza pesada”.

Na limpeza rápida, o robô adotaria um algoritmo em que não seria realizado um reconhecimento prévio da área a ser aspirada, mas sim, algum algoritmo de locomoção que tentasse cobrir a maior área possível, mas de maneira aleatória, sem a necessidade de se aspirar o local perfeitamente. A limpeza seria encerrada levando-se em conta alguns critérios de parada, tais como tempo de aspiração ou número máximo de desvios pela interferência com obstáculos.

Na limpeza normal, o robô faria o reconhecimento da área previamente, adotando, em seguida, uma estratégia de limpeza e, com isso, tentando passar por toda a extensão da área a ser aspirada. Deveria desviar de obstáculos que não estavam previstos e voltar à rota principal.

Na limpeza pesada, o robô faria, assim como na normal, o reconhecimento e aspiração, mas nesse caso, alguns parâmetros seriam alterados, tais como a velocidade de locomoção, ou aumento do valor de sobreposição entre as linhas da trajetória percorrida, ou até mesmo fazer com que a área seja aspirada por duas vezes seguidas.

Em todos os casos o robô deveria voltar à base ao término da limpeza ou se a bateria estiver com nível abaixo do normal. Caso a aspiração não tenha sido terminada, ele automaticamente, após o recarregamento, voltaria à posição em que havia parado e reiniciaria a aspiração.

Outra meta seria a adição de um sensor embaixo do robô que atuasse juntamente como sensor de detecção de escadas. Tal sensor teria, também, a finalidade de perceber quando o robô fosse levantado, ou seja, tirado do chão por inteiro. Com isso ele desligaria o sistema de sucção e a vassoura giratória e entenderia que a limpeza deveria ser terminada. O robô deveria, então, ser encaixado à base, para recarga.

Outra modificação, mais complexa, seria a implementação de um sistema de comunicação entre o robô e a base, através da emissão de sinais eletromagnéticos, como ultrassom ou infravermelho, para melhorar o posicionamento relativo do robô em relação ao ambiente, fechando, com isso, a malha responsável pela localização. A todo instante o robô poderia corrigir erros de trajetória que não são detectados, tal como escorregamento das rodas no piso. Assim, a afirmação de uma limpeza eficaz seria ainda mais garantida.

Poderia também ser adicionado um sensor no compartimento de pó e partículas que indicasse a necessidade de seu esvaziamento.

Mais uma alteração seria o uso de controle remoto para controlar o robô à distância. Tal controle teria diversas funções como iniciar a aspiração, escolher o tipo de limpeza, parar o aspirador ou o controle de qualquer outra função que venha a ser adicionada ao robô.

Outra adição seria a colocação de um sistema de rádio para a sintonia de emissoras AM/FM. Durante a limpeza o robô poderia, então, tocar o som de uma rádio local tentando, com isso, amenizar os ruídos emitidos pela aspiração, além de entreter o usuário, através de música ambiente.

## 2 ACIONAMENTO

A locomoção do robô aspirador é realizada através de um sistema baseado em rodas. Como a forma básica do robô já está pré-definida e se resume a um cilindro de cerca de 40 centímetros de diâmetro e altura de cerca de 10 centímetros, a melhor maneira que se encontrou para realizar a locomoção foi a adoção de duas rodas independentemente motorizadas, centradas ao robô, e mais duas pequenas rodas de apoio.

Levou-se em conta que se deseja ter a melhor linearidade em trajetórias retilíneas e grande capacidade de manobra sem que, para a realização de curvas, o robô faça movimentos que cubram grandes áreas, o que prejudicaria sua movimentação em lugares pequenos. Logo, a melhor maneira encontrada foi o posicionamento de duas rodas centradas ao corpo do robô.

### 2.1 - Escolha do Motor

Cada roda deve ter um acionamento independente e que possa ser controlado com precisão pois, apenas assim, o robô poderá ser direcionado para a rota desejada. Tal acionamento não poderia ser feito a não ser com motores elétricos, já que são motores que não produzem resíduos (o robô aspirador destina-se a ambientes fechados residenciais), sua fonte de energia, no caso baterias, pode ser recarregada de forma facilitada em qualquer fonte elétrica de 110 ou 220V, encontradas com facilidade em qualquer residência, são acionadores de fácil controle, baratos, de manutenção praticamente zero, de grande oferta de modelos com torques e velocidades variadas e alta eficiência.

A partir disso, algumas opções puderam ser levantadas. Tipos possíveis de motores elétricos que podem ser acionados por baterias são motores de passo e os motores CC. Para a escolha de qual seria a melhor opção, levou-se em conta alguns critérios. Para cada critério foi associado um peso que equivale ao quanto importante tal critério se mostra ao projeto. O peso foi dado em uma escala de 1 a 5, onde 5 equivale a um item mais relevante. Os critérios e seus pesos são:

Critério	Peso
01 - Torque	4
02 - Constância na Rotação	5
03 - Presença de Redução	4
04 - Preço	2
05 - Massa	5
06 - Precisão	5
07 - Consumo	5
08 - Facilidade de controle (Software)	4
09 - Simplicidade dos circuitos de controle (Hardware)	3
10 - Simplicidade dos circuitos de potência (Hardware)	3

Tabela 01 – Critérios para a escolha do tipo de motor

A tabela de decisão 1 foi, então, montada para a escolha da melhor alternativa de motor, com especial atenção para as relações torque X massa do motor (ou, em última instância, volume do motor) e torque X consumo.

Critério	Peso	Motor de Passo	Motor CC
01 - Torque	4	2	5
02 - Constância na Rotação	5	5	4
03 - Presença de Redução	4	1	3
04 - Preço	2	2	4
05 - Massa	5	3	5
06 - Precisão	5	5	4
07 - Consumo	5	3	4
08 - Facilidade de controle (Software)	4	4	4
09 - Simplicidade dos circuitos de controle (Hardware)	3	5	3
10 - Simplicidade dos circuitos de potência (Hardware)	3	4	5
<b>Total</b>		<b>139</b>	<b>165</b>

Tabela 02 – Tabela de decisão para escolha do tipo de motor

A partir do resultado da tabela, optou-se pelo uso de um motor CC.

O próximo passo foi a pesquisa de motores disponíveis no mercado que pudessem atender às necessidades de projeto.

Encontrou-se 7 motores que, a princípio, atenderam os requisitos de projeto (Anexo B – Opções de motores encontrados). A partir dessas possíveis alternativas, levantou-se uma segunda tabela de decisão com os seguintes critérios e pesos:

Critério	Peso
01 - Torque	5
02 - Redução	4
03 - Preço	2
04 - Massa	4
05 - Consumo	5
06 - Volume Ocupado	5
07 - Facilidade montagem	4
08 - Facilidade no acoplamento	5

Tabela 03 – Critérios para a escolha do motor CC

E com isso montou-se a tabela de decisão:

Critério	Peso	Motor 1	Motor 2	Motor 3	Motor 4	Motor 5	Motor 6	Motor 7
01 - Torque	5	5	1	2	4	4	4	2
02 - Redução	4	2	3	4	5	5	4	2
03 - Preço	2	4	4	4	4	4	4	4
04 - Massa	4	1	2	5	5	2	4	5
05 - Consumo	5	1	5	2	5	1	2	5
06 - Volume ocupado	5	3	3	1	4	3	4	3
07 - Facilidade de montagem	4	5	3	2	4	4	4	4
08 - Facilidade no acoplamento	5	4	5	4	5	2	3	2
<b>Total</b>		<b>105</b>	<b>110</b>	<b>97</b>	<b>154</b>	<b>102</b>	<b>121</b>	<b>112</b>

Tabela 04 – Tabela de decisão para escolha do motor CC

## 2.2 - O motor escolhido

O motor escolhido, com base nos critérios citados, foi o motor 4, que é encontrado com facilidade no mercado a R\$ 15,00 com disponibilidade de pedidos em grande quantidade.

Motor escolhido para atuar na locomoção do robô aspirador de pó:

### **24 RPM 24 VDC COM CAIXA DE REDUÇÃO EM LINHA**

Novo ITT AUTOMOTIVE modelo 402.757



<b>Especificações:</b>
24 RPM
24 V CC
50 mA @ sem carga
30 in-oz torque (calculado)
Rotação reversível
Totalmente encapsulado
Corrente contínua
6 mm (0.235") x 18 mm (0.71") eixo longo com chamfro
Quatro furos rosqueados na face com 11/16"x 1-1/8" centros
Tamanho total 3-1/4" x 1-1/2" x 1-3/8"
Massa 3/4 lb

Figura 2.1 – Motor escolhido para atuar na locomoção do robô

## 2.3 - Acionamento dos motores

Após a escolha do motor que seria utilizado para a movimentação do robô, foi necessário decidir como seria feito o seu controle. Devido às funções que foram definidas ao robô aspirador, ele deveria ter a capacidade de saber sua posição em relação a um ponto de partida para que, com isso, pudesse traçar uma rota que seria seguida para a realização da limpeza no ambiente. Além disso, ele deveria estar sempre “ciente” de sua posição quando estivesse seguindo a rota planejada, para que a área coberta fosse a maior possível.

Em vista disso, duas alternativas principais foram analisadas. Uma delas foi fazer com que o robô se comunicasse constantemente com uma base fixa que lhe informaria as coordenadas da posição que ele ocupa. Outra alternativa seria que o próprio robô fizesse uma contagem dos giros do motor e, impondo a hipótese de que não há escorregamento, ele acumulasse as informações de sua posição conforme fosse se movimentando e, assim, calculasse sua posição em relação ao ponto de partida.

A primeira alternativa, apesar de parecer mais simples, envolveria uma grande complexidade em relação à comunicação *wireless*. A base teria que calcular a posição do robô através de sistemas de ondas, sonoras ou luminosas, que poderiam sofrer interferências do meio externo. Outro fator decisivo foi que a implementação de tal sistema de localização envolveria uma eletrônica muito avançada já que a base deveria obter a posição do robô móvel, definir a distância as dimensões e informá-lo de sua posição sempre através de um sistema de comunicação *wireless*. Como tal alternativa apresentou grande dificuldade de implementação, optou-se pela segunda.

Nesta segunda alternativa, o próprio robô deve contar os giros do motor. Baseado no tamanho conhecido da roda e pela posição angular do motor ele tem a possibilidade de calcular o quanto andou e, assim, saber sua localização.

Para contar os giros do motor o robô possui um disco de encoder com 50 ranhuras. O encoder é, então, acoplado diretamente ao eixo do motor, não havendo então nenhuma redução. Com isso, o controle de giro tem uma precisão de  $7,2^\circ$  ( $360^\circ / 50 = 7,2^\circ$ ). A princípio, adotou-se que o diâmetro da roda é 100 mm, fixo.

Assim o arco mínimo de controle, ou seja, a precisão do posicionamento devido ao número de ranhuras do encoder é de  $\frac{7,2\pi}{180} \cdot 50 = 6,3\text{ mm}$ . Tal precisão é o suficiente para o posicionamento do robô levando-se em conta que seus deslocamentos são da ordem de 10 m. Além disso, como dito anteriormente, o robô deve armazenar suas posições para definir a rota e posteriormente segui-la, logo não se pode ter um número muito elevado de incrementos sendo armazenados.

Com isso definido, deve-se escolher qual a melhor maneira de acionar o motor para que ele tenha uma resposta compatível com a resolução do encoder. A maneira como isso é feito usualmente, e com grande precisão, é a utilização de uma driver de potência baseado na tecnologia de PWM (Pulse Width Modulation ou Modulação por Largura de Pulso).

## 2.4 - O Driver de Potência

O driver baseado na tecnologia de PWM funciona da seguinte maneira:

A idéia básica é gerar duas ondas quadradas que entrem cada uma em um pólo do motor. As ondas quadradas devem ter um *duty-cycle* (ciclo de trabalho) e uma freqüência tal que imponham ao motor um sentido de rotação e uma velocidade controlada. A partir do valor médio dessas ondas o motor responderá com um movimento no sentido e velocidade desejado. Impondo ondas iguais nos dois pólos do motor este fica parado.

### 2.4.1 - Ponte H

Para impormos essas duas ondas quadradas no motor, deve-se ter um circuito de alta potência (potência do motor). O circuito que faz com que uma onda quadrada digital seja transformada em um sinal analógico é o conhecido circuito Ponte H.

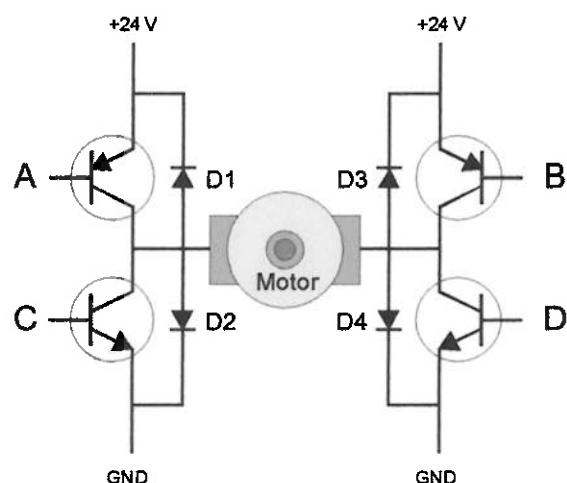


Figura 2.2 – Ponte H

A onda quadrada deve ser amplificada através de transistores ou um CI driver de potência. Para essa amplificação, utiliza-se a onda quadrada gerada pelo CI PWM (explicado mais adiante) para habilitar e desabilitar os transistores conforme as Figuras 03 e 04.

Para o funcionamento da Ponte H, coloca-se a onda quadrada gerada pelo PWM nas entradas A e D e o inverso dessa mesma onda em B e C. Ou seja, quando A e D estão em *High*, B e C estão em *Low*. Quando B e C estão em *High*, A e D estão em *Low*.

Com isso, adotando-se que A e D estejam habilitando a passagem de corrente em seus respectivos transistores (onda quadrada em *High*), a corrente irá passar da esquerda para a direita, o que faria com que o motor fosse acionado em um sentido de rotação.

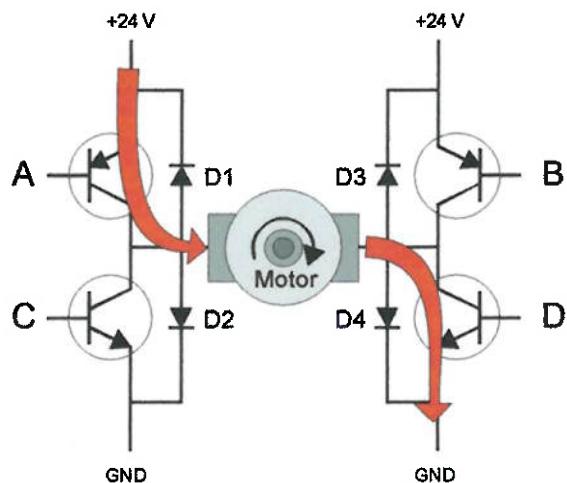


Figura 2.3 – Ponte H com sinais A e D em *High*

Agora, adotando-se que B e C estejam habilitando a passagem de corrente em seus respectivos transistores (onda quadrada em *High*), a corrente irá passar da direita para a esquerda, o que faria com que o motor fosse acionado em um sentido de rotação contrário.

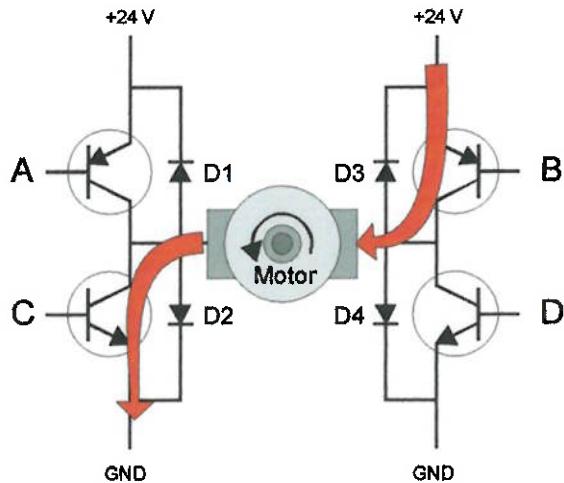


Figura 2.4 – Ponte H com sinais B e C em High

Logo, dependendo do tempo e que a onda fica em High nos sinais A e D, ou B e C, o motor se comportará de maneira a tirar uma média temporal da corrente que entra em cada sentido e será acionado com uma certa velocidade em um certo sentido.

Assim, fica evidente que, quando as ondas tiverem um *dead-time* de 50%, será aplicada ao motor uma tensão média igual nos dois pólos. Esse *dead-time* é o tempo em que a onda fica em tensão High.

Quando o *dead-time* for diferente de 50 %, um dos lados do motor terá uma tensão média aplicada diferente em cada pólo, fazendo com que se move para um dos lados. Com *dead-time* de 0 a 50 %, o motor gira em um sentido, com *dead-time* de 50 a 100%, o motor gira no sentido oposto.

Um fator importante que deve ser considerado é que não se podem ter os quatro sinais em High, pois isso ligaria a tensão de 24V ao GND, curto-circuitando a fonte de tensão. Logo, deve-se ter um circuito que regula os sinais A, B, C e D de modo que não permita tal situação. Este circuito será chamado de **Circuito Regulador**. Os diodos presentes na ponte H servem para proteção desse circuito. Esses diodos que ligam os motores à tensão devem responder a três questões:

Os diodos funcionam com uma voltagem menor que a voltagem base dos transistores?

Os diodos conseguem agüentar a máxima voltagem reversa (24V)?

Os diodos conseguem agüentar a máxima corrente (200mA)?

Esses diodos devem estar presentes pois, quando o motor acelera ou desacelera por qualquer razão (sinal ,carga ou atrito) há uma relutância do campo elétrico presente nos enrolamentos do motor em mudar. Mais propriamente, essa mudança do campo induz energia elétrica. Essa “devolução” de energia pode voltar para os chips. Os diodos D1 e D3 protegem os chips de sobrevoltagem funcionando quando se tem tensão maior vindo do motor do que das baterias. As baterias absorvem então essa energia. A voltagem nominal do diodo deve ser menor que a voltagem nominal dos transistores, ou o diodo não funcionará cedo o bastante para protegê-los. Pelo fato dos diodos estarem instalados reversamente, a energia não pode fluir das baterias para o motor. Se o diodo estivesse instalado em oposição, a energia iria fluir imediatamente pelo motor, ignorando a saída dos transistores (ou pior, curto-circuitando através deles).

Já os diodos D2 e D4 protegem os chips de sobvoltagem (menor que o terra) funcionando quando a voltagem no motor é menor que do terra. Mais uma vez as baterias cuidam do problema, antes que a energia flua voltando para o chip.

#### 2.4.2 - PWM (Pulse Width Modulation ou Modulação por Largura de Pulso)

Voltando a linha principal de raciocínio, para gerar a onda quadrada que será colocada, responsável pela habilitação dos transistores da Ponte H, utiliza-se um CI PWM. Este CI funciona da seguinte maneira:

Internamente ele gera uma onda em formato dente de serra e a compara com um nível de tensão, que deve estar entre o máximo e o mínimo dessa onda dente de serra. A partir dessa comparação o CI gera uma onda quadrada onde o *dead-time* varia entre 0 e 100 %, conforme o valor da tensão de comparação se altera.

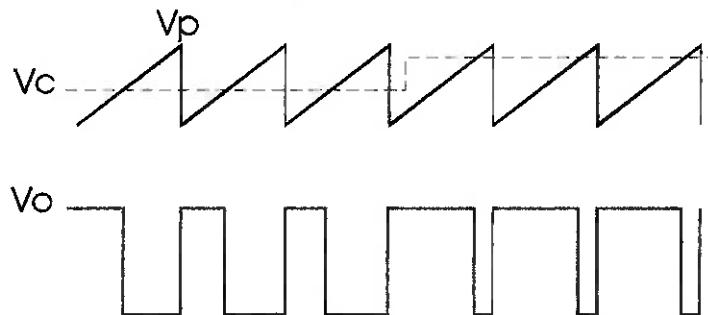


Figura 2.5 – Modulação por Largura de Pulso

Pela figura acima, conforme o valor da tensão de comparação se altera ( $V_c$ ), o *dead-time* da onda quadrada também se altera.

A freqüência da onda dente de serra é dada pela conexão de um resistor (RT) e de um capacitor (CT) ao CI de PWM. Essa freqüência deve ser tal que o motor consiga segui-la e não apresentar vibrações excessivas.

Se necessário, a onda quadrada gerada pelo PWM deverá passar por um amplificador operacional para eliminação de um possível *offset* (nível lógico *Low* acima de 0V) e para seu condicionamento para tensão TTL (nível lógico *High* igual a 5V).

### **2.4.3 - Conversor Digital/Analógico**

Precisa-se agora gerar a tensão de comparação que se altere conforme desejado. Isso é feito utilizando-se um conversor digital/analógico.

A partir de uma entrada digital deve-se informar uma palavra que definirá o sentido de rotação e a velocidade do motor.

Por exemplo, se utilizada uma entrada de 8 bits no driver do motor, as palavras de 0000 0000 até 0111 1111 definem 128 níveis de velocidade em um sentido.

A palavra 1000 0000 define que o motor estará parado.

E as palavras de 1000 0001 até 1111 1111 definem mais 127 níveis de velocidade no outro sentido de rotação.

As palavras 0000 0000 e 1111 1111 são as velocidades máximas, em sentidos opostos.

Tal palavra deve ser transformada em um sinal analógico, através de um conversor digital analógico, onde cada palavra indicará um nível de tensão diferente. Essa tensão agora irá para o CI gerador de PWM.

Com isso, um diagrama simplificado do driver de potência do motor fica:

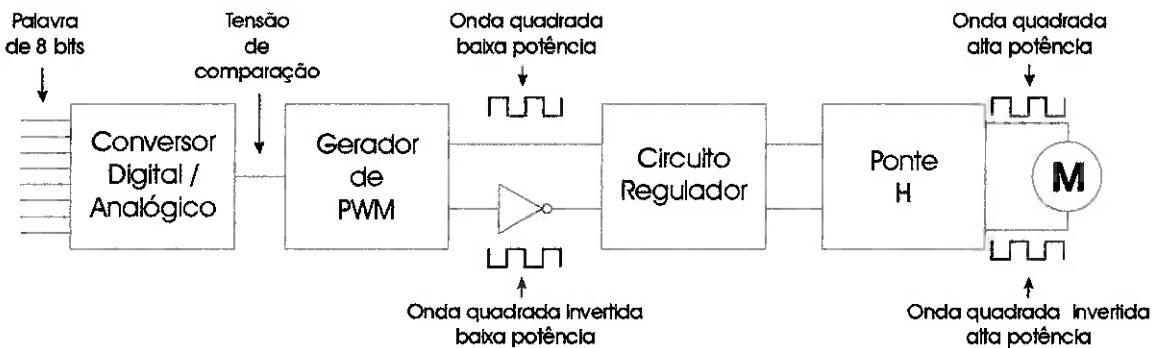


Figura 2.6 – Esquema simplificado do Driver de Potência do motor CC

Logo, para o acionamento do robô, são utilizados dois drivers de potência, um para cada motor. Será analisado agora, cada bloco separadamente.

## 2.5 - Circuitos do driver de potência

### 2.5.1 -Conversor Digital/Analógico

Visando a diminuição de portas que serão ocupadas do controlador, optou-se por um conversor digital/analógico quádruplo de 8 bits (quatro saídas independentes, das quais serão utilizadas duas). Assim, utilizando apenas um conversor, podem-se acionar os dois motores a partir de mais três bits. Dois bits selecionam qual das quatro portas disponíveis será utilizada, e o outro habilita a porta. Desses dois bits que selecionam as portas, um deles poderá ficar fixo, já que, variando-se apenas um, pode-se habilitar duas saídas diferentes.

Mesmo tendo que acionar os dois motores exatamente ao mesmo tempo, tem-se uma defasagem de apenas alguns clocks do controlador, o que não é perceptível no movimento. Para movimentar os dois motores necessita-se das seguintes instruções:

- Enviar uma palavra para o conversor D/A para definição de velocidade;
- Escrever um bit para escolha do motor;
- Escrever um bit para habilitar o motor;

Assim, com apenas três instruções, um dos motores é acionado. Com mais três instruções, o outro motor também pode ser acionado. Uma instrução leva aproximadamente 50 µs para ser executada, causando um atraso entre os acionamentos dos motores de cerca de 150 µs, o que não será perceptível para a precisão do movimento.

O número de bits dá a quantidade de níveis de velocidade que o motor atingirá. Nesse caso, como já explicado anteriormente, 8 bits geram 127 níveis para um sentido de rotação e 128 níveis para o outro sentido, e um nível para deixar o motor parado. Esse número de níveis de controle é importante pois só assim pode-se garantir que, quando da implementação do protótipo, não estaria-se em falta com um controle mais preciso, o que é muito difícil de se prever nessa fase do projeto.

O CI escolhido foi o **TLC7226**, da Texas Instruments (Ver datasheet em CD anexo).

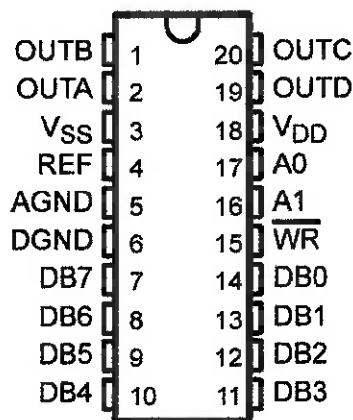


Figura 2.7 – Conversor Digital/Analógico TLC7226

Os pinos 7 a 14 (DB0....DB7) recebem a palavra de 8 bits que definirá o sentido e a velocidade do motor. Os pinos 15 a 17 (WR, A0 e A1) definem qual porta de saída será utilizada e a habilita, conforme a tabela abaixo:

CONTROL INPUTS			OPERATION
WR	A1	A0	
H	X	X	No operation Device not selected
L	L	L	DAC A transparent
↑	L	L	DAC A latched
L	L	H	DAC B transparent
↑	L	H	DAC B latched
L	H	L	DAC C transparent
↑	H	L	DAC C latched
L	H	H	DAC D transparent
↑	H	H	DAC D latched

L = low, H = high, X = irrelevant

Tabela 05 – Habilitação das portas de saída do conversor D/A

Assim, a porta A1 pode ser fixada em GND, deixando a porta A0 responsável pela troca da porta de saída (OUTA ou OUTB). A porta WR (invertido) funciona como *Enable*. Quando uma borda de subida é detectada a saída selecionada por A0 recebe a palavra escrita em DB0 a DB7.

Os pinos 3 e 6 ( $V_{SS}$  e DGND) devem ser ligado ao terra. O pino 18 ( $V_{DD}$ ) deve ser ligado a 12V.

Os pinos 4 e 5 ( $V_{REF}$  e AGND) definem os limites superior e inferior do sinal de saída do D/A. Nesse caso, esses limites devem estar de acordo com o CI gerador de PWM. Experimentalmente esses valores foram obtidos e chegou-se a  $V_{REF} = 3,4V$  e AGND = 1,2V. Esses valores são dependentes apenas do PWM utilizado (no caso utilizou-se o LM3524D) e são os valores limites da onda dente de serra gerada pelo PWM para comparação com esta tensão de saída do conversor D/A.

Utilizando esses valores na entrada do PWM como tensão de comparação, obtêm-se os *dead-times* mínimo e máximo respectivamente e, com isso, aproveitando todos os níveis de tensão (256) para o controle de velocidade.

Se não fossem escolhidos os valores limites adequados alguns níveis de tensão seriam perdidos, ou os *dead-times* mínimo e máximo não seriam atingidos. Ou seja:

Caso o limite inferior escolhido fosse 0V, ao invés de 1,2V, as palavras que fornecessem níveis de tensão entre 0V e 1,2V não teriam influência sobre a onda quadrada gerada pelo PWM.

Caso o limite inferior fosse maior que 1,2V, o *dead-time* máximo jamais seria alcançado.

Caso o limite superior fosse 5V, ao invés de 3,4V, as palavras que fornecessem níveis de tensão entre 3,4V e 5 V não teriam influencia sobre a onda quadrada gerada pelo PWM.

Caso o limite superior fosse menor que 3,4V, o *dead-time* mínimo jamais seria alcançado.

Para gerar as tensões de 1,2V e 3,4V a partir da tensão de 5V, pode-se utilizar uma seqüência de resistores como da figura abaixo:

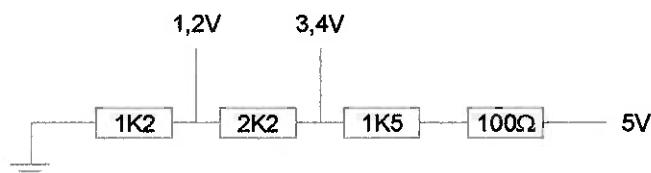


Figura 2.8 – Conjunto de resistores para geração de 1,2V e 3,4V

Com isso, o conversor D/A fornecerá uma tensão de saída que varia entre 1,2 V e 3,4V. Essa tensão será a utilizada pelo PWM para a definição da onda quadrada com o *dead-time* desejado.

### 2.5.2 - Gerador de PWM

Para a geração da onda quadrada, optou-se por um CI gerador de PWM muito conhecido e bem simplificado, o **LM3524D**, da National Semiconductor (Ver datasheet em CD anexo).

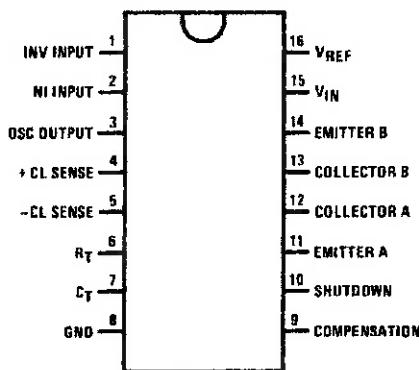


Figura 2.9 – PWM LM3524D

Inicialmente, devemos definir os valores do resistor RT e do capacitor CT (Pinos 6 e 7). Esses componentes irão definir a freqüência da onda quadrada que será gerada.

Para determinar essa freqüência foi realizado um teste onde se ligou o motor a um gerador de onda quadrada. A freqüência da onda foi sendo alterada até que o ruído emitido pelo motor fosse o menor possível, indicando menor vibração. Assim, chegou-se que, com uma freqüência de 3 KHz, o motor estava em sua melhor performance.

Para determinar os valores de CT e RT deve-se utilizar a fórmula:

$$f_{osc} = \frac{1}{RT \cdot CT}$$

Partindo do valor máximo recomendado para CT = 0,1  $\mu\text{F}$ , temos:

$$3000 = \frac{1}{RT \cdot 1.10^{-7}} \Rightarrow RT = 3333 \Rightarrow RT = 3K3\Omega$$

Verificou-se experimentalmente que, apenas com um resistor de 3K9 $\Omega$ , a freqüência de 3KHz foi atingida.

O pino 2 (NI Input) receberá o sinal vindo do conversor D/A que será utilizado para comparação com a onda dente de serra interna ao PWM para geração da onda quadrada.

O pino 1 e 9 (Inv Input e Compensation) devem ser ligados entre si. Os pinos 4, 5, 8, 10, 11 e 14 (+CL Sense, -CL Sense, GND, Shutdown, Emitter A e Emitter B) devem ser ligados ao terra.

O pino 15 ( $V_{IN}$ ) deve receber uma tensão de 5V. O pino 16 ( $V_{REF}$ ) deve ser ligado ao terra com um capacitor de  $10\mu F$ .

Os pinos de saída 12 e 13 (Collector A e Collector B) devem ser ligados entre si e a uma tensão de 12V com um resistor de  $3K9\Omega$ .

Dessa maneira consegue-se gerar uma onda quadrada, com *dead-time* definido pela tensão do pino 2 (NI Input), com freqüência de 3KHz que deveria variar entre 0 e 12V. Experiências comprovaram que a variação se dá entre 2,5 e 12V.

Foi possível, experimentalmente, obter as seguintes ondas:

Impõe-se uma tensão de comparação de 1,2V, o *dead-time* ficou em 0%, e a amplitude ficou em 12V, como pode ser notado pela figura abaixo:

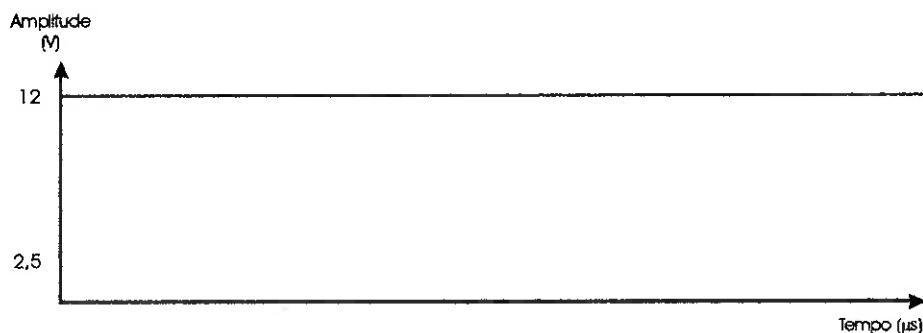


Figura 2.10 – Onda gerada pelo PWM, com tensão de comparação de 1,2V.

Impõe-se uma tensão de 2,3V, o *dead-time* ficou em 50%, como esperado, e a amplitude da onda varia de 2,5V a 12V, como mostra a figura abaixo:

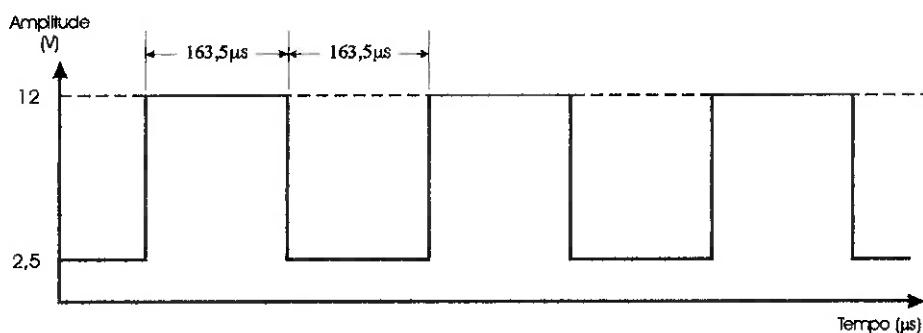


Figura 2.11 - Onda gerada pelo PWM, com tensão de comparação de 2,3V, freqüência de 3058Hz, 1 *dead-time* de 50%.

Nota-se pelo período da onda, que a freqüência atingida ficou próxima dos 3KHz desejados, com algo em torno de 3,058 KHz.

Agora, impondo-se uma tensão de 3,4V, o *dead-time*, que deveria ser de 100%, ficou na faixa dos 96,6%, já que a onda continua em *High* por um período de 11 $\mu$ s. Mesmo mudando-se a tensão de comparação, o *dead-time* de 100% nunca foi atingido, sendo um limite do CI gerador da onda. O efeito é notado na figura abaixo:

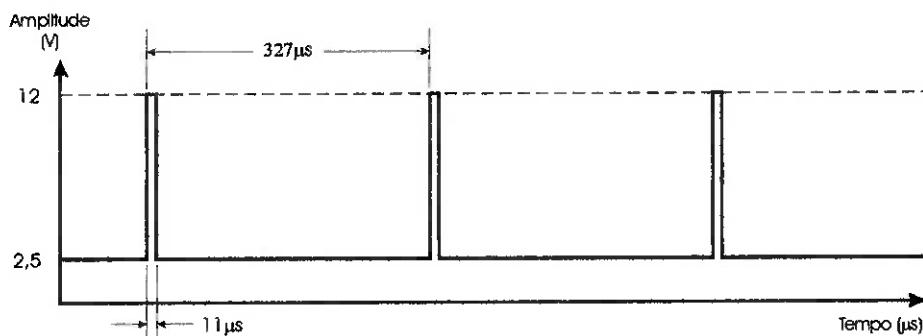


Figura 2.12 - Onda gerada pelo PWM, com tensão de comparação de 3,4V, freqüência de 3058Hz, *dead-time* de 96,6%.

Tal efeito não é relevante já que o motor não têm uma resposta tão rápida a um pico de apenas 11 $\mu$ s.

### 2.5.3 - Amplificador Operacional

Para eliminar esse problema, em que a variação da onda que sai do PWM se dá entre 2,5V e 12V, ao invés de 0V e 5V, deve ser adicionado ao circuito um amplificador operacional na forma de um comparador. Além de regular os limites da onda quadrada, o amplificador operacional também serve para isolar o PWM do resto do circuito. A ligação deve ser feita da seguinte maneira:

A saída do PWM deve entrar no “-“ do Amp Op, e uma tensão de 5V deve ser conectada ao “+” do Amp Op. O  $V_{CC}^+$  deve ter uma tensão de 5V e o  $V_{CC}^-$  deve ter 0V, como mostrado na figura abaixo.

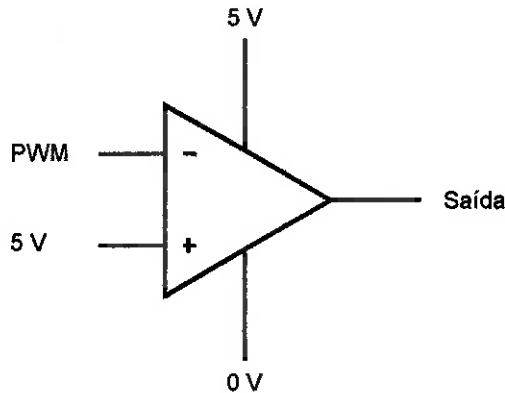


Figura 2.13 – Regulagem da saída do PWM através de Amp Op comparador

O CI utilizado para essa regulagem é o **OPA743PA**, da Texas Instruments (Ver datasheet em CD anexo).

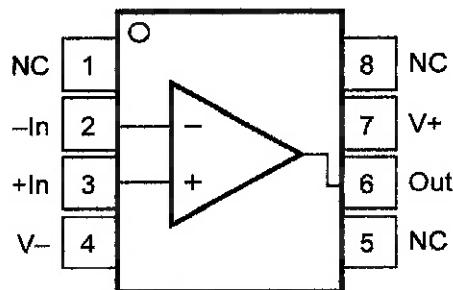


Figura 2.14 – Amplificador Operacional OPA743PA

A saída do PWM deve ser ligada ao pino 2.

A tensão de 5V deve ser ligado aos pinos 3 e 7.

O pino 4 deve ser ligado ao terra.

O pino 6 é a saída do Amp Op.

Assim, quando o sinal do PWM for maior que 5V, a saída do Amp Op será de 5V.

Quando a saída do PWM for menor que 5V, a saída do Amp Op será de 0V.

Agora se tem uma onda quadrada que varia entre 0V e 5V, com o *dead-time* desejado, controlado pela tensão de comparação vinda do conversor D/A.

A adição do Amp Op implicou, apenas, numa redução da freqüência da onda de 3058Hz para 3037Hz, o que não trará problemas para o correto funcionamento do motor.

## 2.5.4 - Inversor

Para a inversão da onda quadrada deve-se utilizar um CI inversor. O CI escolhido é o **CD74HCT04E**, da Texas Instruments (Ver datasheet em CD anexo).

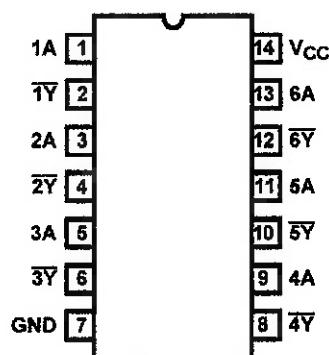


Figura 2.15 – Inversor CD74HCT04E

Este CI possui 6 inversores, onde 2 são utilizados, um para cada motor.

A saída do Amp Op deve ser ligada ao pino 1, que tem como saída o pino 2. A outra entrada, utilizada para o outro motor, se dá pelo pino 3 que tem como saída o pino 4.

O pino 7 deve ser ligado ao GND e o pino 14 deve ser ligado à tensão de 5V.

Para isolar o inversor do resto do circuito pode-se colocar outro Amp Op, mas desta vez apenas como seguidor, conforme a figura abaixo:

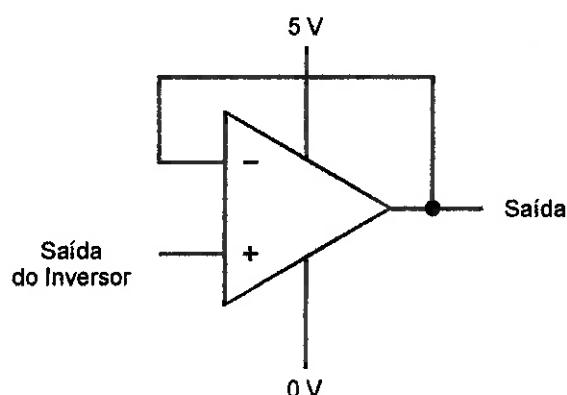


Figura 2.16 – Amp Op seguidor para isolar o Inversor

O CI utilizado é o mesmo **OPA743PA**. O pino 2 deve ser ligado à saída (pino 6). A saída do inversor deve ser ligada ao pino 3. A tensão de 5V deve ser ligado ao pinos 7. O pino 4 deve ser ligado ao terra. O pino 6 é a saída do Amp Op.

Com isso têm-se agora as duas ondas quadradas prontas para entrar na ponte H.

Através de experiências, verificou-se que a presença deste segundo Amp Op não tinha influência sobre a performance do circuito, portanto este componente foi eliminado no driver.

## 2.5.5 - Circuito Regulador e Ponte H

Com as duas ondas quadradas geradas, uma vinda do Amp Op e a outra vinda do Inversor, têm-se os sinais necessários para utilização da Ponte H.

Tentou-se a implementação da Ponte H utilizando-se quatro transistores de potência **IRF840**, sem a implementação do circuito regulador. Tal experiência comprovou que, sem um sincronismo dos sinais de habilitação dos transistores, a Ponte H não é eficiente.

Tentou-se também a implementação da Ponte H utilizando-se CI's drivers de potência, mas, da mesma maneira, seu correto funcionamento só se dava quando os sinais de habilitação não tinham oscilação, ou seja, se tratavam de sinais constantes, que não necessitavam de sincronização para evitar que os dois transistores de um mesmo lado da ponte H fossem habilitados ao mesmo tempo gerando um curto circuito da fonte de tensão.

Verificou-se também, nessa fase experimental, a importância dos diodos para proteção dos chips contra a sobrevoltagem e a sobvoltagem.

Assim, optou-se por um CI que incorpora o circuito regulador e a Ponte H em um mesmo encapsulamento. Trata-se do CI **A3968SA**, da Allegro (Ver datasheet em CD anexo).

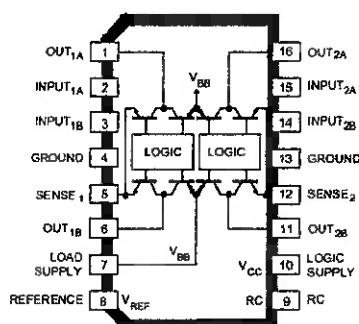


Figura 2.17 – Driver utilizado como Ponte H A3968SA

Além disso, este CI possui duas Pontes H independentes, ideais para a construção de um driver para dois motores.

A partir das ondas vindas do Amp Op e do Inversor, este driver de potência amplifica o sinal para uma amplitude que varia entre 0V e 24V, com uma corrente que pode atingir até 1A.

As entradas do driver que se dão pelos pinos 9 e 10, têm como saídas os pinos 11 e 7.

As entradas do driver que se dão pelos pinos 15 e 16, têm como saídas os pinos 14 e 2.

Os pinos 1, 8, 12 e 13 devem ser ligados ao GND.

O pino 3 recebe a tensão de 24V.

O pino 6 recebe a tensão de 5V.

E ao pino 4 deve-se impor uma tensão de 1V obtida pela conexão central de dois resistores, um de  $39K\Omega$  e um de  $10K\Omega$ , ligados em série, que são ligados à tensão de 5V e ao terra.

Das saídas 11, 7, 14 e 2 deve-se conectar os diodos responsáveis pela proteção dos chips.

Infelizmente, tais CI's foram danificados nas fases de testes. Como foram adquiridos como amostras grátis vindas dos Estados Unidos, escolheu-se outro CI em seu lugar.

Fez-se a opção pelo CI UC3717AN, da Texas Instruments (Ver datasheet em CD anexo).

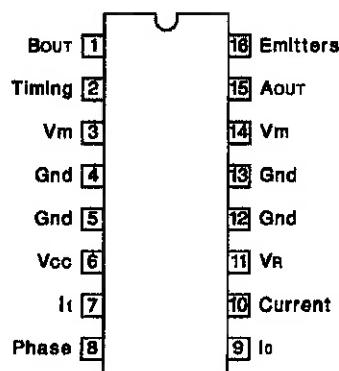


Figura 2.18 – Driver utilizado como Ponte H UC3717AN

Apesar deste CI incorporar um gerador de PWM, não se obteve sucesso em sua utilização completa, apenas como ponte H. No entanto tal CI possui a ponte H apenas para um

motor, sendo então necessários dois deles para a movimentação dos dois motores de locomoção do robô.

A ponte H presente neste CI já incorpora os diodos de proteção ligados ao terra, sendo necessários apenas os diodos de proteção ligados a V<sub>CC</sub>.

Outra diferença é que este CI também incorpora o inversor, sendo necessário apenas o sinal direto do PWM. Este sinal entra como uma fase que se altera e faz a comutação dos transistores.

Os pinos 2, 4, 5, 7, 9, 10, 12, 13 e 16 devem ser ligados ao terra.

Os pinos 6 e 11 devem ser ligados a 5V.

Os pinos 3 e 14 devem ser ligados a 24V.

O pino 8 é a fase que recebe o sinal do PWM.

Os pinos 1 e 15 são as saídas para os motores. São nesses pinos que devem ser conectados os diodos de proteção para 24V.

O diodo escolhido foi o BYV26D, da Philips (Ver datasheet em CD anexo). Trata-se de um diodo com resposta rápida, evitando atrasos na proteção.

Assim, pode-se montar um driver de potência que pode ser representado pelo seguinte esquema simplificado:

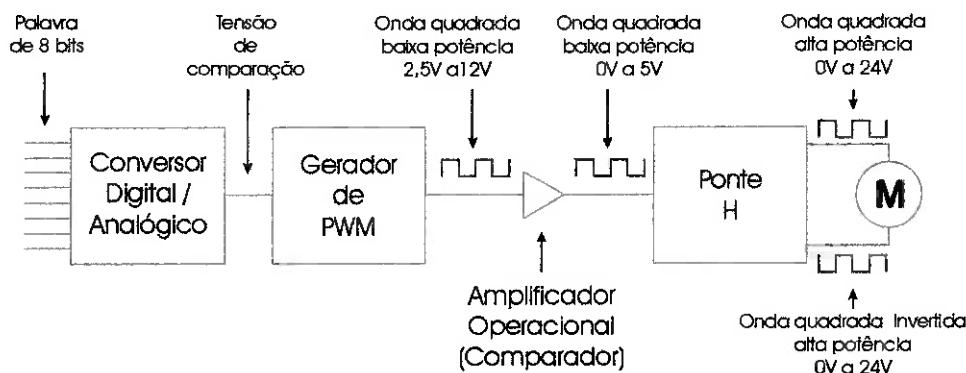


Figura 2.19 – Esquema simplificado do Driver de Potência do motor CC após experiências em laboratório

## 2.5.6 - Conversor de tensão

O circuito completo envolve também um conversor de tensão. Como os CI's do Conversor D/A, os AmpOp's e os drivers necessitam de uma alimentação de 5V, utilizou-se um CI para transformar uma entrada de 12V para 5V. As outras entradas de 12V e 24 V serão provenientes das próprias baterias.

O CI utilizado para conversão de tensão é o L7805CV, da STMicroelectronics (Ver datasheet em CD anexo).

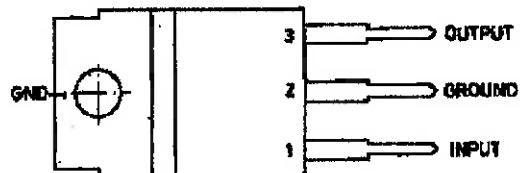


Figura 2.20 – Conversor de tensão de 12V para 5V L7805CV

O pino 1 é ligado a 12V, o pino 2 ao terra e o pino 3 fornece 5V.

## 2.6 - Circuito final

Com isso, foi possível gerar o circuito final do driver de potência dos motores de locomoção do aspirador de pó robô (Ver anexo E – Circuito do driver de potência dos motores CC). Adicionou-se a cada fornecedor de tensão (12V, 24V, 5V, 1,2V e 3,4V) um capacitor de  $10\mu F$  para a estabilização da tensão.

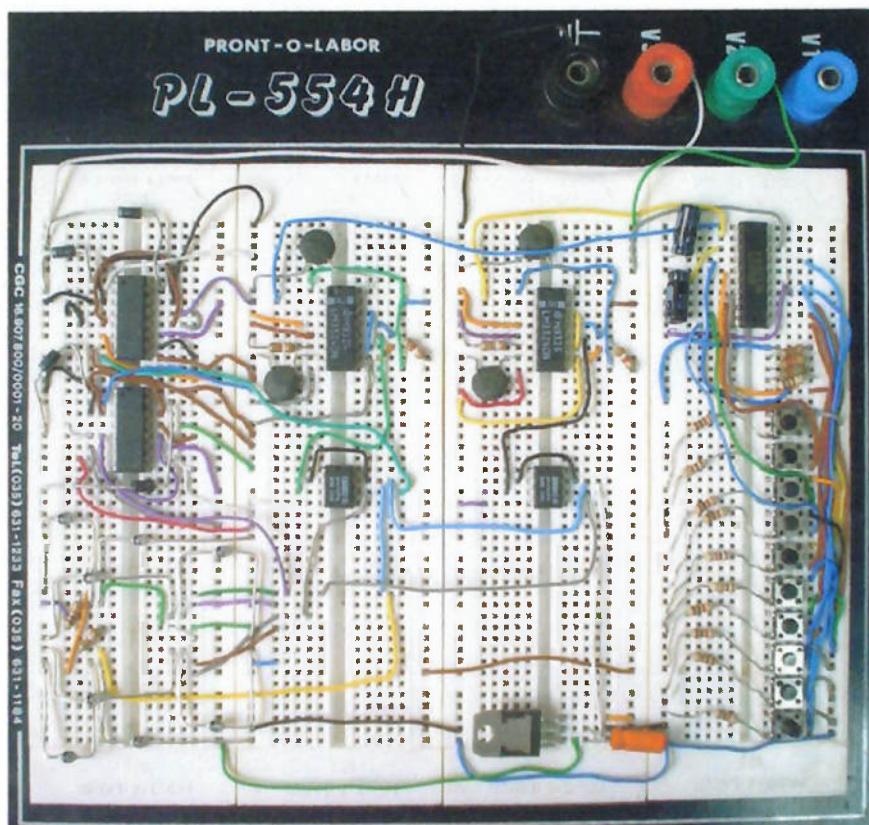


Figura 2.21 – Foto do circuito do driver de potência montado em protoboard.

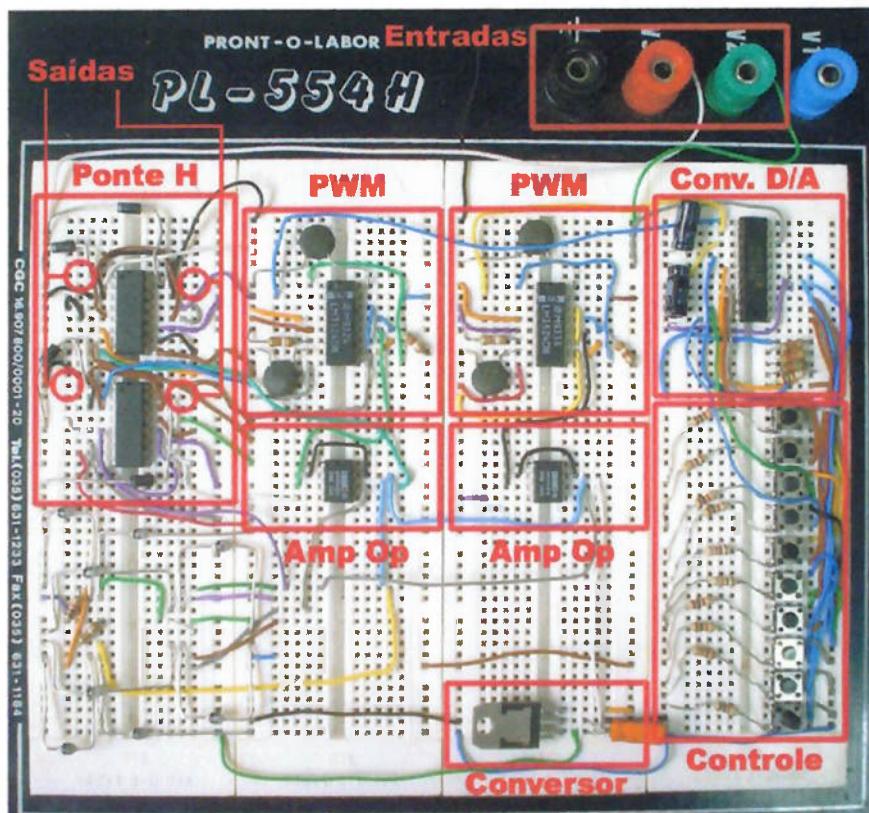


Figura 2.22 – Foto do circuito do driver de potência – separação de blocos.

Pela foto, pode-se perceber, separadamente, os blocos que compõe o circuito do driver de potência. No espaço abaixo da Ponte H e dos Amp Op's estavam montados, respectivamente, o driver **A3968SA** e o inversor **CD74HCT04E** que, após terem sido danificados, foram substituídos pelo driver **UC3717AN**.

Uma particularidade deste driver de potência se dá em relação à entrada de 24V. Devido ao fato do CI **UC3717AN** operar a Ponte H de maneira que, para que se tenha o motor parado, é necessária uma onda quadrada, com *duty-cycle* de 50%, ocorre um problema durante a inicialização do programa de controle do robô. Ao iniciar o programa, as saídas da porta que escrevem a palavra de 8 bits no conversor D/A estarão com valor zero. Isso implica em uma condição do motor de velocidade máxima em um sentido de rotação. Além disso, o bit de controle que habilita o motor (/WR) que é ativo em Low, também estará em zero. Dessa maneira, um dos motores começaria a se movimentar.

Para evitar tal acontecimento, a tensão de 24V, que alimenta os motores, deve inicializar “cortada”. Assim, mesmo com o motor habilitado para rodar, sua alimentação estaria cortada, evitando o movimento. Para solucionar tal problema, deve ser adicionado um relê para controle dessa tensão de 24V, que estará inicialmente aberto.

Finalmente, tal relê só fecharia o contato após a estabilização das portas do controlador, e escrita de uma palavra que defina velocidade zero para os motores. Tal relê será disposto em uma placa responsável pelo fornecimento de tensão para todos os circuitos do robô. Com isso, será necessária mais uma saída de I/O do controlador, para habilitar o relê e, consequentemente, o driver de potência.

## 3 SENSORIAMENTO

Um dos requisitos de projeto é que o robô seja autônomo e, portanto, se move pelo ambiente sem qualquer auxílio humano, ainda que remoto, devendo fazer os ajustes e correções de curso necessários para “varrer” toda a área destinada à aspiração e detectar barreiras, como paredes, móveis e objetos.

### 3.1 - Rede de sensores

Os sensores do robô estarão distribuídos ao longo de toda sua frente projetada, ou seja, toda a região passível de colisão com objetos, paredes e móveis do ambiente de operação do robô.

Há a necessidade de sensores na parte traseira do robô, no caso de este se movimentar para trás? Essa pergunta foi levantada durante o projeto. De fato, há momentos em que o robô retrocede em seu movimento, porém, ele retorna percorrendo a mesma direção na qual se deslocou para frente anteriormente, agora no sentido oposto. Portanto se até chegar no ponto em que o robô, por um motivo de fim de curso de aspiração ou pela detecção de alguma barreira física, parou e, em seguida, deu início ao processo de retorno, ele não foi interrompido por nenhum objeto, ou ainda, não se chocou a qualquer elemento, não há a necessidade de sensoriar este retorno, uma vez que a superfície a ser percorrida por ele agora, já foi atravessada anteriormente.

Para compreender melhor este fenômeno, segue um modelo da distribuição da rede de sensores no aspirador:

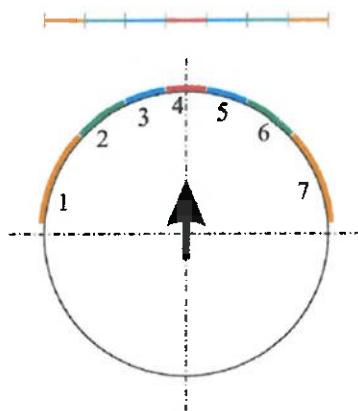


Figura 3.1 – Rede de Sensores

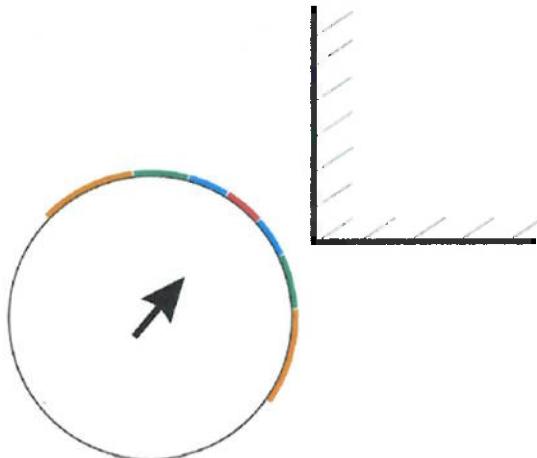


Figura 3.2 – Contato do Sensor com barreira

A seta aponta o sentido de deslocamento ou, ainda, à frente do robô. Nota-se que toda a projeção da frente do robô é sensoriada com o uso de 7 sensores, indicados pelos números de 1 a 7 e representados por cores distintas no esquema acima.

Cada sensor é uma combinação de dois sensores distintos, um mecânico e um óptico, trabalhando em paralelo, com seus sinais digitais combinados em um único sinal de saída que é enviado ao microcontrolador. Quando um objeto é detectado, por um sensor, na verdade, pode ter sido percebido pelo sensor óptico, sem a necessidade de contato físico, ou pelo mecânico, dado por uma chave. Mais adiante serão melhor detalhados os motivos pelos quais este sistema combinado é usado.

Note que cada sensor tem um tamanho diferente. A medida que se afastam do da região central do robô ficam maiores. Isto se dá para que haja uma distribuição homogênea da secção frontal “coberta” por cada sensor.

Digamos agora que o sensor caminha em uma trajetória de choque com um obstáculo como indicado na parte (b) da figura. No caso, uma parede, mais precisamente, uma quina com a qual o robô se chocará e detectará através de seu sensor 6.

Ao detectar a parede, o robô interrompe seu movimento e inicia um retorno suficiente para que seu sensor deixe de detectar a parede, ou seja um deslocamento mínimo. Ressalta-se, mais uma vez, que essa trajetória de retorno se dá ao longo da região pela qual o robô já se deslocou anteriormente quando em movimento de avanço, daí o fato de se dispensar o uso de sensores em sua região traseira.

Uma vez retornado ligeiramente para trás, o robô está livre para rotacionar, alternando seus atuadores e acionando-os um para frente e outro para trás, livrando seu região frontal do objeto detectado e liberando seu movimento de avanço.

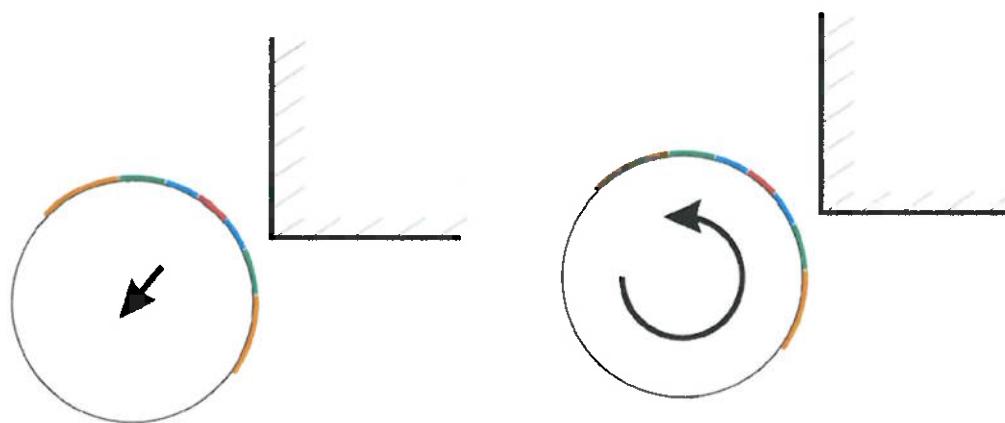


Figura 3.3 – Retorno do robô

Figura 3.4 – Rotação do robô

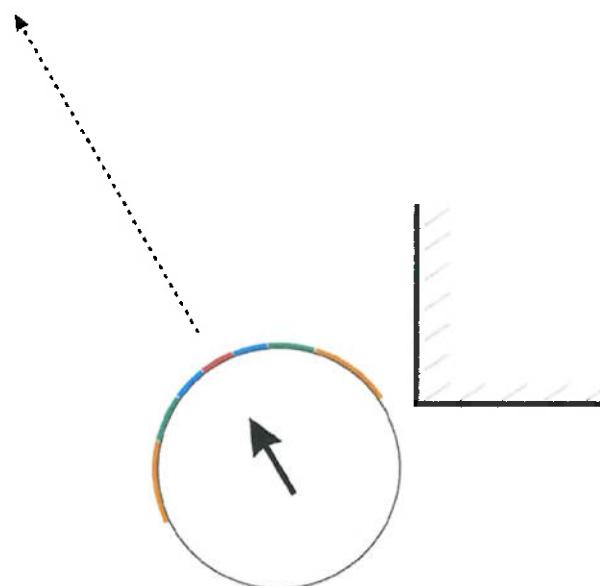


Figura 3.5 – Nova trajetória

A correção de rota, ou rotação do robô, é proporcional ao sensor ativado quando houve a percepção da barreira. Isso é possível pois cada sensor é interpretado singularmente, tendo seu sinal de saída conectado a um pino diferente das entradas digitais do microcontrolador.

Com esta proposta, o robô consegue se deslocar pelo ambiente com a capacidade de corrigir sua trajetória com bastante precisão, uma vez que sua parte frontal está dividida em 7 regiões percebidas discriminadamente.

### **3.2 - Propostas dos sensores**

Há diversos métodos de sensoriamento do ambiente de trabalho do robô, entre eles o uso de sensores ultra-sônicos, ópticos, mecânicos, magnéticos, indutivos e captação e tratamento de imagens.

Os requisitos dos sensores para o robô aspirador são:

- Capacidade de detectar todo tipo de superfície, regular ou irregular.
- Detectar todo tipo de material ou, pelo menos, a maioria dos materiais comumente encontrados no interior de residências e escritórios, ambientes aos quais se destina a aplicação do robô.
- Baixo custo. O robô, por se tratar de uma utilidade doméstica, precisa ter custo competitivo no mercado e, portanto, um sistema sensorial caro e de implementação complexa é desfavorável por encarecer o projeto.
- Resolução suficiente para detectar obstáculos a uma distância pequena de modo a corrigir sua posição e deslocamento sem que haja uma grande área “cega” até o obstáculo, ou seja, o robô deve chegar suficientemente perto do obstáculo para, então, detectá-lo tendo aspirado o máximo do espaço possível.
- Havendo a necessidade de tocar a superfície a ser detectada, este toque deve se dar de maneira suave, não danificando de forma alguma a superfície e, ainda, evitando movimentá-la e deslocá-la de sua posição original.
- Deve ser compacto, economizando espaço no interior do robô.
- Ter interface compatível com níveis lógicos do microcontrolador.
- Baixo consumo de energia.

A tabela abaixo avalia cada um dos métodos para a aplicação desejada. Os critérios considerados foram *Resolução* (Res), *Confiabilidade* (Conf), *Complexidade* (Complex), *Flexibilidade* (Flex), *Consumo* (Cons) e *Custo*. Os números entre parênteses indicam os pesos de cada critério.

Método	Res (1)	Conf (2)	Complex (1)	Flex (2)	Cons (2)	Custo (3)	Total
Ultra-sônico	5	5	1	5	2	1	33
Óptico	4	3	4	4	4	4	42
Mecânico	2	5	5	5	5	5	52
Magnéticos	4	2	3	1	3	4	31
Indutivos	4	2	3	1	3	4	31
Tratamento de Imagens	5	4	1	4	1	1	27

Tabela 06 – Alternativas de sensores

Os sensores ultra-sônicos apresentam excelente resolução e são largamente utilizados em robôs AGV. Seu funcionamento está baseado na emissão de ondas, pulsos, ultra-sônicos através de um emissor acoplado ao robô e que, após refletidos, retornam ao robô onde são detectados por um receptor sincronizado que, por sua vez, registra o intervalo de tempo decorrido entre a emissão do pulso e seu retorno.

Dada a velocidade de propagação do pulso ultra-sônico no ar, a distância do robô ao objeto ou obstáculo detectado é determinada pelo produto dessa velocidade pelo tempo decorrido.

É extremamente preciso, detecta objetos feitos de praticamente qualquer material e possui flexibilidade suficiente para não só detectar obstáculos mas também para posicionar o robô no ambiente aferindo as distâncias às paredes e mapeando todo o espaço ao seu redor.

Porém, apresenta características que impossibilitam sua aplicação no projeto. Possui altíssimo custo, encarecendo o produto final e tornando-o inviável no mercado, o emissor ultra-sônico é grande demais e muito pesado para ser instalado no interior do robô.

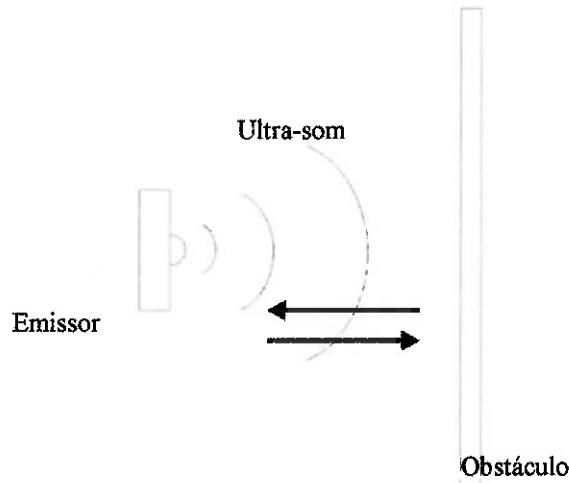


Figura 3.6 – Sensor por ultra-som

Os sensores magnéticos e inductivos são muito utilizados em automação. São compactos, baratos e seus sinais de resposta são facilmente tratados de modo a fazer sua interface com o microcontrolador.

A desvantagem destes métodos de sensoriamento é o número restrito de materiais detectáveis. Os sensores percebem materiais metálicos os quais, apesar de presentes em domicílios e escritórios, não são os obstáculos de maior interesse para a aplicação e funcionamento do robô. O objetivo é detectar paredes e móveis, barreiras mais comuns nos ambientes nos quais o robô será utilizado.

A captação e tratamento de imagens são feitos por uma câmera de vídeo e um software de interpretação de imagens. É bastante complexo e passível de erros, além de extremamente caro. Exige um grande número de itens de hardware o que é responsável pelo alto custo do sistema. Apresenta, também, um nível de detalhe muito superior ao necessário no projeto.

Uma das técnicas de tratamento de imagens é o corte de luz. O funcionamento mais detalhado do sistema é dado a seguir:

Enquanto que na medição pela triangulação, os dados referentes à forma do objeto coletados em diversos pontos são obtidos, projetando-se um único spot de laser sobre a superfície do objeto, a forma de toda uma secção pode ser obtida simultaneamente projetando

um feixe de luz na forma de uma faixa estreita. Neste método, a forma de uma secção é medida como se uma faixa de luz estivesse cortando o objeto, e daí o seu nome de método de corte por luz.

Uma faixa de luz de alta intensidade, como por exemplo o laser, é projetada na superfície do objeto. Observando-se esta faixa de luz, por uma direção diferente daquela do sistema de projeção, é observada uma imagem da faixa que é deformada de acordo com a forma do objeto. Esta imagem é então detectada por meio de um dispositivo de coleta de imagem, por exemplo uma câmera de TV e com base na grandeza das deformações da faixa e nos dados relativos ao arranjo geométrico dos sistemas de projeção e de observação, é determinada a forma do objeto.

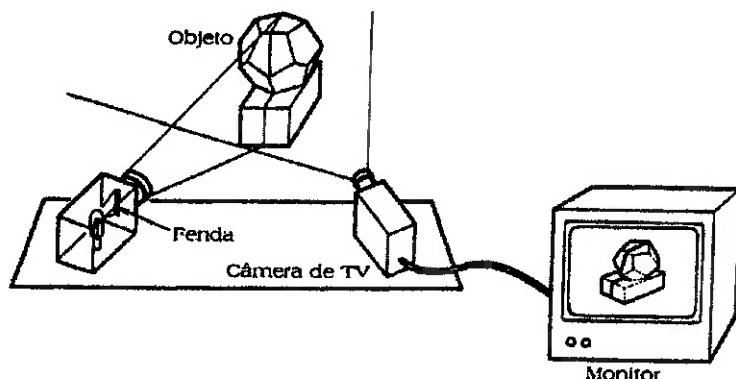


Figura 3.7 – Sensor por corte de luz

Este método utiliza um sistema óptico relativamente simples, porém ainda superior ao requisitado pelo projeto, sendo que a sua sensibilidade pode ser alterada dentro de uma larga faixa mediante modificações no arranjo do sistema óptico. Ainda, a medição da posição da faixa de luz pode ser feita por meio de técnicas simples de processamento de sinal ou de imagem, sendo por isso, uma técnica bastante visada para ser utilizada em sistemas de coleta de formas tridimensionais de robôs (visão de robô).

Os métodos mais indicados, como demonstra a tabela, são os baseados em sistemas mecânicos e ópticos. O sistema mecânico pode ser feito através de chaves, switches, botões, contatos elétricos, entre outros. Em todos estes dispositivos se faz necessário o contato do sensor com o objeto detectado.

Os sensores ópticos são baseados na emissão de feixes infravermelhos através de leds

e sua recepção após refletidos na barreira física que deseja-se detectar, semelhante ao sistema que utiliza ultra-som, porém muito mais barato, simples e compacto, com a desvantagem de que dependendo da textura da superfície a ser detectada, os feixes infravermelhos podem ser espalhados, não retornando ao receptor e, consequentemente, evitando o sensoriamento e percepção do objeto.

Por essa razão optou-se por um sistema conjunto de sensores mecânicos e ópticos, combinando a confiabilidade e segurança do sensor mecânico com a sensibilidade e ausência da necessidade de contato físico do sensor óptico.

O objetivo dessa combinação é que todo o sensoriamento seja feito de maneira óptica, evitando o desgaste das partes externas do robô e o choque com objetos e paredes do ambiente aspirado, diminuindo o risco de possíveis danos materiais. Mas, para que o robô tenha flexibilidade suficiente para atuar em qualquer ambiente, o sensor mecânico é necessário, pois como comentado anteriormente, os sensores ópticos são passíveis de falhas na percepção, principalmente devido à irregularidades nas superfícies detectáveis.

Há a necessidade de que este contato seja suave para não danificar os objetos tocados.

Os micro-switches foram os primeiros sistemas testados, são robustos, possuem um curso relativamente grande durante seu acionamento, proporcionando um intervalo de parada para o robô sem que este entre em contato com a barreira detectada. A desvantagem deste *offset* no acionamento é o fato de aumentar o período de contato entre o robô e a barreira física, aumentando o risco de danificá-la.

A variedade de tamanhos e modelos de switches auxilia sua implementação no sistema, podendo ser facilmente instalado no interior do robô, possuem roldanas nas extremidades dos contatos que diminuem o atrito no toque, além de prover furos de fixação que facilitam sua montagem e aumentam a rigidez do sistema.

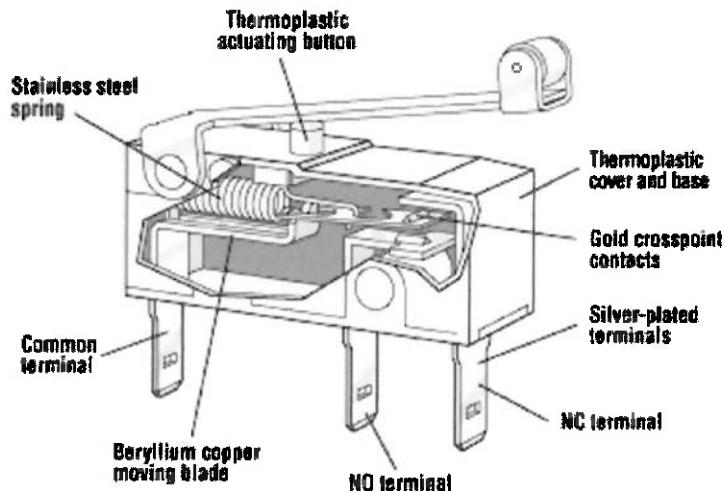


Figura 3.8 – Micro-Switch

Um problema no sensoriamento mecânico é que, devido a sua natureza de necessitar do contato físico com o objeto detectado, seu posicionamento e fixação no robô limita sua atuação.

Colocá-lo na região central do robô no sentido vertical implica em estar passível ao robô não detectar um sofá, ou uma cama, por exemplo, pois são objetos que estão a uma determinada altura da superfície onde o robô se desloca. A princípio isto não seria problema, ainda mais, faz parte dos requisitos de projeto o robô ser capaz de entrar em lugares de difícil acesso, como sob camas e sofás, sendo assim, é necessário que tais objetos não sejam detectados e, portanto, não interpretados como obstáculos. Porém, um sério problema é o sofá ter uma altura insuficiente para o acesso do robô na região sob sua estrutura e, no entanto, devido ao switch estar localizado à meia altura do chão, não detectá-lo, podendo ficar preso sob este sofá ou após ficar tocá-lo e ter seu deslocamento interrompido, continuar o acionamento do motor indefinidamente supondo ainda estar em movimento.

Colocar o switch na região superior do robô pode causar um efeito indesejado, quando há em seu sentido de deslocamento um objeto mais baixo que a altura do sensor em relação ao solo, como um rodapé, por exemplo, bastante comum em residências e escritórios, ocorrendo, em última instância, o mesmo fenômeno observado anteriormente, quando o robô falhou em detectar um sofá ou cama de altura inferior à sua em relação ao solo.

A alternativa para contornar o problema foi colocar os sensores aos pares, um superior e outro inferior, no limite viável para sua fixação nas extremidades do robô. Isso resolveria o problema, mas também dobraria a quantidade necessária de switches, aumentando o espaço interno do robô requerido à sua instalação. Isso torna o sistema inadequado do ponto de vista de otimização de espaço e inviável financeiramente, uma vez que aumenta seu custo de produção.

Mas a necessidade de sensoriar tanto o nível mais próximo ao chão quanto o limite superior do robô continua existindo.

Uma segunda alternativa considerada foi utilizar um par de contatos, um mais acima e outro localizado mais abaixo, porém, conectados a uma única chave (switch). Isso resolveu o problema do uso degenerado de chaves, mas diminuiu em muito a resistência e rigidez mecânica da montagem. O problema observado foi que, dependendo do sentido e intensidade do contato entre o objeto detectado e o contato, este carregava o switch com um momento fletor que o torcia, deslocando o contato até o final de seu curso sem que fosse observada sua comutação de estado.

### **3.3 - Solução para os sensores**

Projetou-se, então, um sistema de contatos mais robusto, capaz de assimilar momentos torções, mas que permite deslocamentos em uma direção e sentido preferenciais, de modo a converter contatos em direções com ângulos distintos em relação à normal do botão ao resultado de movimentação no sentido de promover o contato entre os terminais do sensor.

O sistema é composto por dois botões de alumínio, um localizado na parte inferior e outro na parte superior do robô, com suas faces apontadas para fora. Sustentando esses botões está uma lâmina de aço mola (1060 ou 1080) ou bronze que permite uma deformação quase linear do botão para trás. Esta lâmina está fixa e isolada a um suporte parafusado ao chassi do robô que sustenta, também, dois terminais anteriores aos botões que, no contato com os botões, fecham o circuito de detecção.

Estes terminais são de cobre mas, mesmo assim, permitem uma deformação elástica, absorvendo um deslocamento inercial do robô mesmo após o comando de frear os motores depois de detectado um obstáculo.

Na região central do sistema, há uma reentrância onde o sistema de sensoriamento óptico se localiza. Este, por sua vez, por não necessitar de contato físico, é implementado na região central do robô em relação ao eixo vertical.

O sistema foi projetado em software de modelagem 3D da seguinte forma:

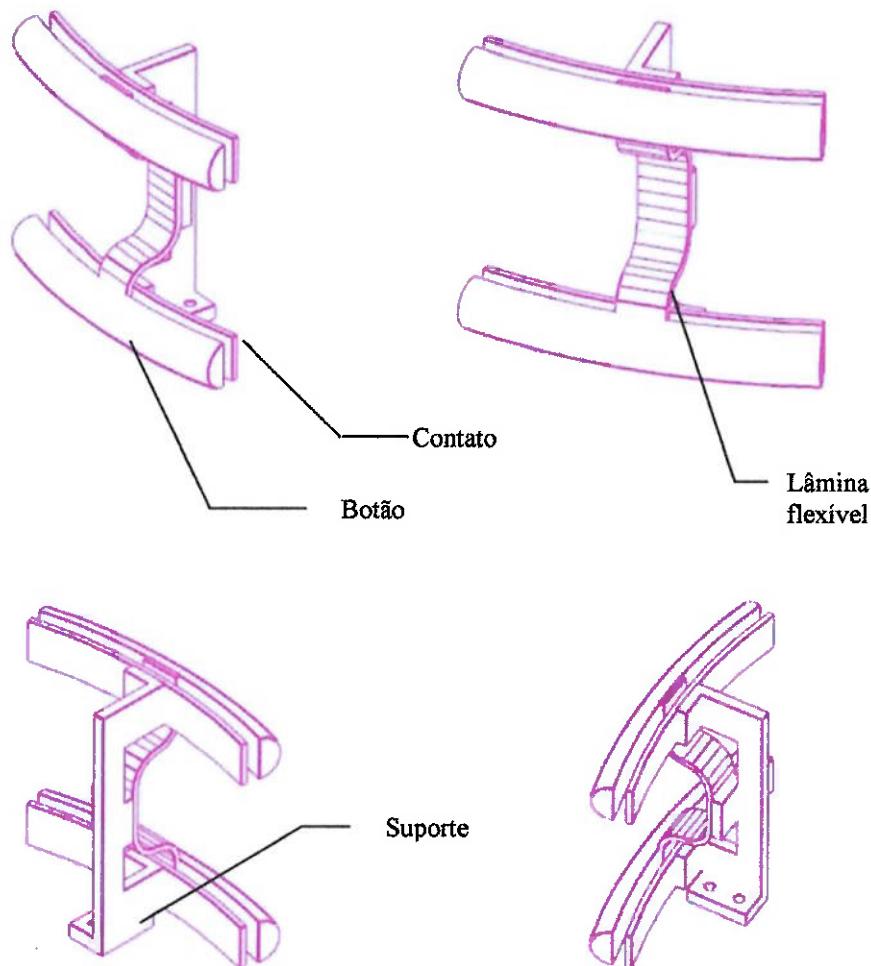


Figura 3.9 – Sensor mecânico

E artisticamente concebido como:



Figura 3.10 – Sensor mecânico de posição

O botão de alumínio é visto em amarelo e a lâmina de aço mola ou bronze é curva de modo a proporcionar um espaço para o emissor e receptor ópticos responsáveis pelo sensoriamento remoto do robô.

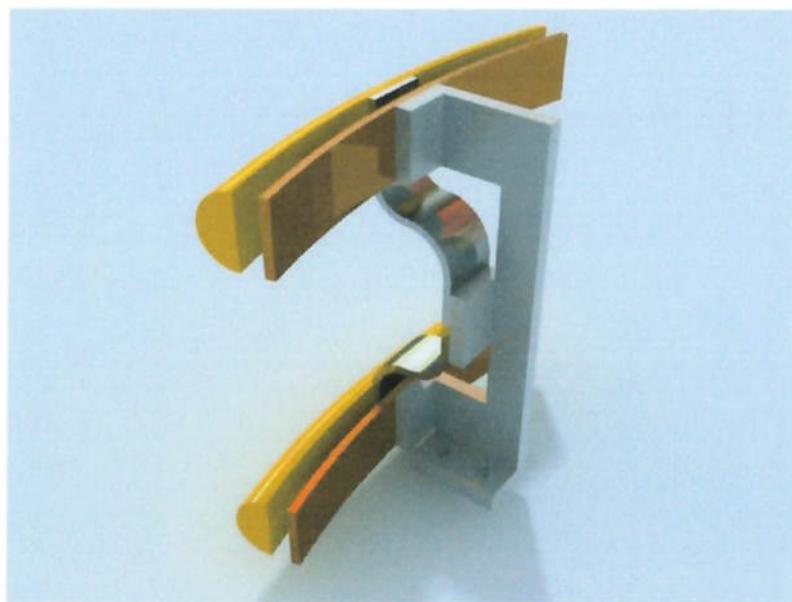


Figura 3.11 – Sensor mecânico de posição

### 3.4 - Bouncing

Como toda chave mecânica, há no sinal de saída do sensor o problema de bouncing, ou seja, intervalo de tempo no qual o sinal oscila entre os níveis lógicos 0 e 1 (0 e 5V respectivamente).

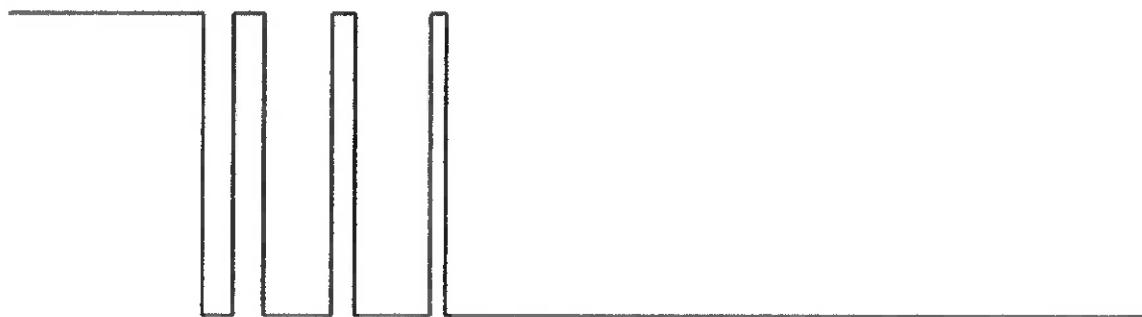


Figura 3.12 – Sinal do sensor com efeito de bouncing (ativo em *Low*)

Este problema é amenizado com o uso de capacitores que regulam a tensão durante o período de oscilação para uma tensão média, porém que já é entendida como nível lógico 0 pelo microcontrolador.

Entretanto, este fenômeno, ainda que presente, não terá efeitos negativos no sistema, pois não há a necessidade de um sinal limpo, sem qualquer problema de bouncing pois, a freqüência de trabalho do microcontrolador é suficientemente grande e muito superior à freqüência do bounce de sinal, para que já nas primeiras oscilações seja detectado o nível lógico 0 e, portanto, a indicação de que há um obstáculo na trajetória do robô. Se o sinal continua oscilando por um período de mais alguns milisegundos, não terá efeito algum, podendo ser descartado por software.

### 3.5 - Sensor óptico

O funcionamento do sensor óptico é bastante simples. Como já mencionado, ele emite luz infravermelha através de um led. Essa luz, quando refletida por uma superfície, retorna ao robô onde há um receptor infravermelho. Para que não haja interferência da luz externa (ambiente), o ideal é que este receptor seja síncrono, ou seja, tenha uma freqüência de trabalho semelhante à freqüência de emissão dos leds. Tal fato age como um filtro passabanda, onde somente a luz emitida pelos leds, portanto de interesse, é aproveitada.

Além dos 7 sensores ópticos que funcionarão em paralelo com os mecânicos, dois outros serão adicionados. Ambos voltados para baixo, detectando o chão, um na parte frontal e outro na parte posterior. O frontal tem com objetivo detectar se há uma escada ou depressão no chão e, caso positivo, envia um sinal ao microcontrolador que o interpreta como um novo obstáculo. O posterior está sempre detectando o chão e sua função é, uma vez retirado o robô do chão, ele envia um sinal de não detecção indicando que o robô foi erguido e, portanto, deve parar o funcionamento dos sistemas de aspiração e locomoção. Isso facilita o desligamento do robô em casos excepcionais, evitando transtornos tentando localizar um botão com o robô em movimento.

Um modelo fácil de se construir utiliza basicamente emissores infravermelhos, um receptor que trabalha com freqüência entre 38KHz e 40KHz, um oscilador para gerar a freqüência de emissão dos leds, e portas lógicas para combinar o sinal de resposta.

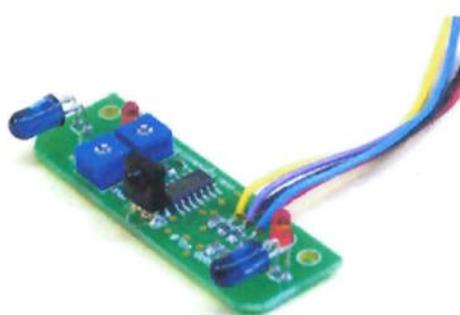


Figura 3.13 – Sensor óptico

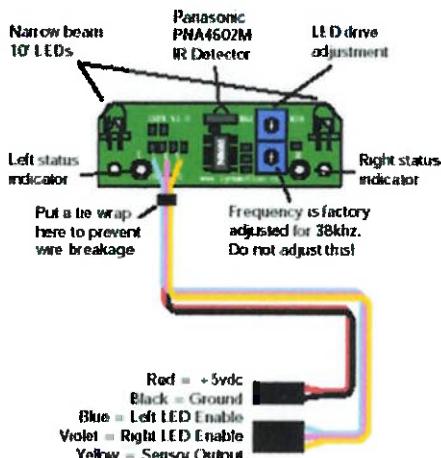


Figura 3.14 – Visão esquemática

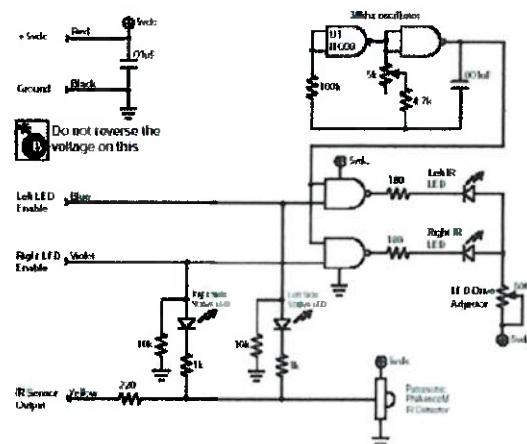


Figura 3.15 – Circuito do sensor óptico

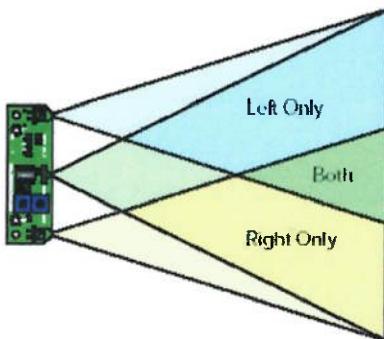


Figura 3.16 – Campo de visão e resposta do Sensor

Uma segunda opção é o uso de sensores industriais de proximidade. Trabalham de forma análoga ao descrito anteriormente, também com o uso de infravermelho e filtros.



Figura 3.17 – Sensor industrial

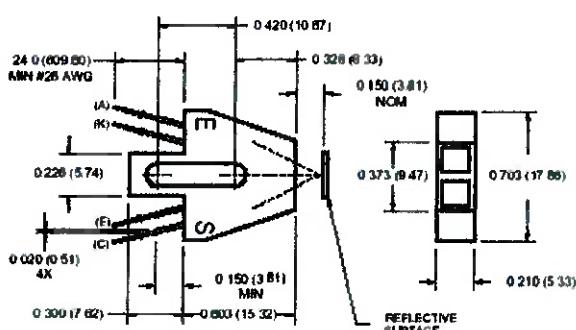


Figura 3.18 – Sensor industrial - dimensões

Os sensores disponíveis têm seu alcance fixo, porém, variam de modelo para modelo. Este atua a uma distância mínima de 3mm, suficientemente perto para dar qualidade para o posicionamento do robô e, ao mesmo tempo, dar tempo para que ele chegue a uma parada total.

A desvantagem dessas duas opções é a complexidade do circuito, onde há a necessidade de se produzir uma onda com uma dada freqüência no emissor e com mesma freqüência de leitura no receptor, para que assim se possa filtrar outras fontes luminosas.

### 3.5.1 - Solução Escolhida

Uma outra alternativa, seria utilizar emissores e receptores que atuam em um dado comprimento de onda, diferente da luz visível, evitando assim a interferência da luminosidade do ambiente. Dessa maneira, o circuito seria apenas de controle de corrente. Esse controle, para o emissor, define a distância com que o obstáculo é detectado.

Dois componentes que atendem à essas especificações são, para o emissor, o **TSAL5100**, e para o receptor, o **QSE159**.

O emissor, **TSAL5100**, (Ver Datasheet em CD anexo) opera com comprimento de onda  $\lambda = 940\text{nm}$ . Sua corrente elétrica de operação é de 100mA, a uma tensão de 5V. Como teremos 9 sensores espalhados pelo corpo do robô, teremos quase 1A corrente para alimentá-los. Assim, é conveniente que o conversor de tensão utilizado tenha capacidade para tal uso.



Figura 3.19 – Emissor TSAL5100

O receptor **QSE159**, (Ver Datasheet em CD anexo) opera com um comprimento de

onda  $\lambda = 880\text{nm}$ . Apesar de não ser o mesmo valor do comprimento de onda do emissor deve-se considerar que o emissor terá sua corrente alterada para ajuste da distância de detecção e, com isso, o comprimento de onda será alterado.



Figura 3.20 – Receptor QSE159

Pode-se ainda adicionar, ao circuito do receptor, um capacitor conectado entre a saída e o terra, para melhorar a qualidade do sinal.

É importante notar que tal receptor é ativo em *Low*, ou seja, o sinal vai para nível lógico *Low* ao detectar uma barreira. Tal fato facilita a junção do sinal entre o sensor óptico e o mecânico simplesmente através de uma porta AND (Ver Datasheet em CD anexo).

Situação	Sinal		
	Óptico	Mecânico	AND
Nenhum detectando	1	1	1
Óptico detectando	0	1	0
Mecânico detectando	1	0	0
Os dois detectando	0	0	0

Tabela 07 – Junção dos sinais dos sensores ópticos e mecânicos

Assim, a detecção será dada quando o valor do sinal recebido pelo controlador passar de *High* para *Low*. Logo, deve-se construir o circuito de maneira que o sinal se mantenha em *High* enquanto nenhuma barreira for detectada.

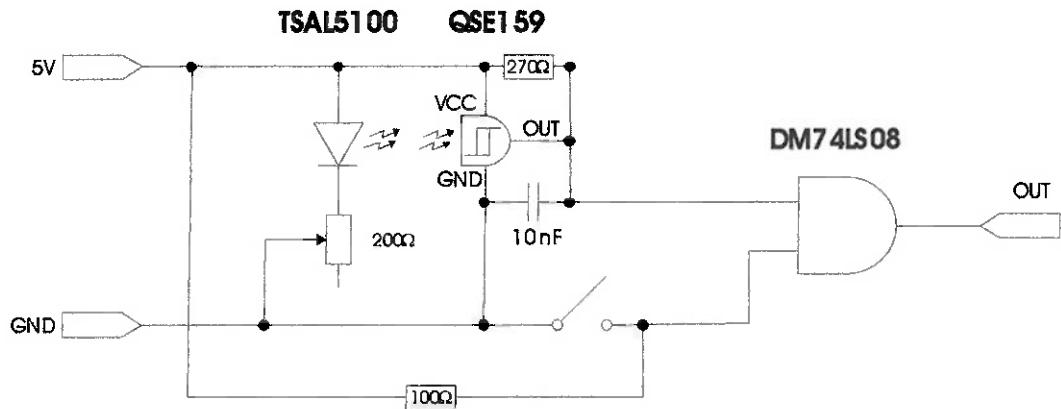


Figura 3.21 – Junção dos sinais dos sensores óptico e mecânico

O potenciômetro do circuito acima é responsável pelo controle da corrente que vai para todos os 7 emissores dos sensores que estão em volta do robô, não apenas para um emissor. Durante os testes, o valor encontrado para o potenciômetro foi de  $30\Omega$ , fazendo com que o receptor detecte uma barreira a cerca de 50 mm de distância. Dessa maneira, alimentados com uma tensão de 5V, os emissores consomem uma corrente de 167mA, ligados constantemente. Os receptores, ligados a um resistor de  $270\Omega$  cada, possuem uma corrente de consumo de até 5mA cada, mais uma corrente de saída, devido à porta AND, de cerca de 0,1mA cada, num total de aproximadamente 35,7mA para os sete. Os sensores mecânicos, ligados cada um a um resistor de  $100\Omega$ , consomem, devido à porta AND, uma corrente de 0,1mA cada, num total de 0,7mA. Assim, todo o sistema de sensores consome uma corrente de cerca de 203,4 mA.

A tensão de 5V fornecida ao sistema provém de um chip L7805, que transforma a tensão de 12V de uma das baterias para 5V. Apesar de suportar uma corrente de até 1A, optou-se por colocar um L7805 exclusivo para alimentação dos sensores, já que esse consumo é constante (167mA dos emissores e 35mA dos receptores). A porta AND se encontra em uma placa separada e é alimentada por outro chip L7805 compartilhado por outros circuitos.

## 4 ENCODER

Para detectar o deslocamento do robô será utilizado um encoder óptico. Seu funcionamento é bastante simples. Um disco com 50 fendas é colocado no eixo do motor solidário a este e, portanto, tendo sua rotação dada juntamente com a rotação do eixo do motor. Um conjunto de photo-emissor e photo-transistor é colocado de modo a ter seu feixe ora interrompido pelo disco e ora livre, quando este atravessa uma das fendas.

Cada motor tem seu encoder, podendo-se assim, determinar a velocidade, ou ainda, o deslocamento de cada motor de maneira independente. Isso é particularmente útil quando quer-se uma rotação do robô em relação ao seu eixo de simetria vertical, maneira com a qual o robô altera seu trajetória de deslocamento.

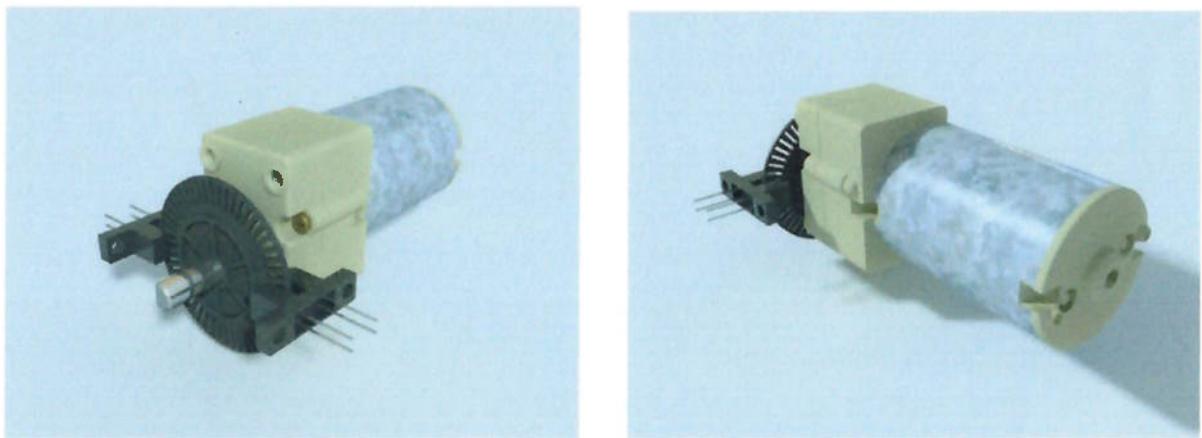


Figura 4.1 – Motor, Disco de encoder e sensores ópticos (Desenho 3D)

Portanto, em uma rotação, o feixe atravessa 50 fendas, ou seja, descreve no sinal de saída uma senóide de 50 picos (pontos de máximo).

Este sinal analógico não é o ideal para a interpretação da rotação do eixo do motor, tendo então de ser tratado, transformado num sinal digital, ora de nível lógico 0 (zero, correspondente ao feixe bloqueado) e ora de nível lógico 1 (5V, correspondente a uma fenda).

Utiliza-se para tanto um comparador, composto por um amplificador operacional com seu sinal  $V^+$  ligado ao sinal vindo do photo-transistor e seu sinal  $V^-$  ligado a uma tensão de comparação e, ainda, seu terminal  $V_{cc}$  ligado a 5V e  $Gnd$  aterrado. Assim, enquanto a tensão do photo-transistor é inferior à de comparação, a saída do amp-op é zero e, enquanto a tensão do photo-transistor é superior à de comparação a saída é 5V, gerando um sinal retangular, digital, podendo ser então contados os pulsos referentes a 1/50 de volta.

## 4.1 - Quadratura

Para determinar o sentido de rotação (horário e anti-horário) do motor utiliza-se dois sensores ópticos deslocados de  $90^\circ$  em relação à fase dos sinais de saída. Com isso, consegue-se determinar qual o sinal que antecede o outro, determinado a direção de rotação. Isto é útil quando um acionamento pode se dar ao contrário do pretendido devido a uma força contrária ao movimento desejado. Na aplicação do robô isso não é necessário, pois a redução é suficientemente grande para travar o eixo quando não acionado por tensão ou, ainda, para evitar que este se move no sentido oposto ao determinado pelo microcontrolador devido às forças externas.

Este fato ajuda a simplificar o sistema, tornando-o mais barato e, consequentemente, mais viável do ponto de vista comercial.

Porém ainda há um problema. Pode ocorrer uma situação onde o motor pára em uma posição em que uma fenda encontra-se parcialmente bloqueando o feixe de luz do encoder. Isso pode causar um mal condicionamento do sinal de saída fazendo com que oscile entre 0 e 1, sem que o motor se move, mas sendo interpretado pelo microcontrolador como pulso e, portanto, como movimento.

Para isso, utilizam-se dois sinais defasados como indicado anteriormente de modo a gerar o seguinte padrão:

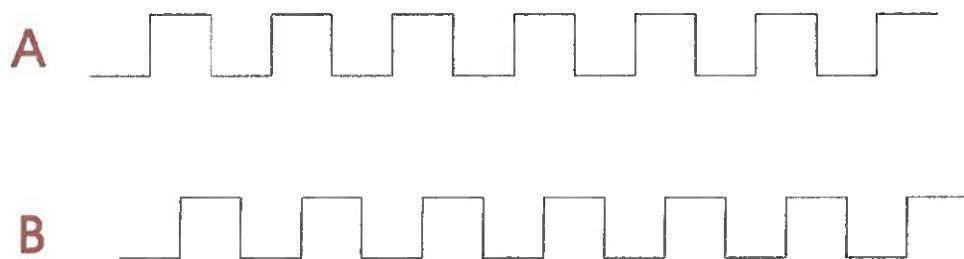


Figura 4.2 – Sinais defasados em 90°

Veja que agora, ainda que um dos sinais esteja mal condicionado, no limite entre zero e 1, o segundo sinal se encontra bem definido em zero ou 5V.

O circuito do encoder é, então, dado pelo esquema:

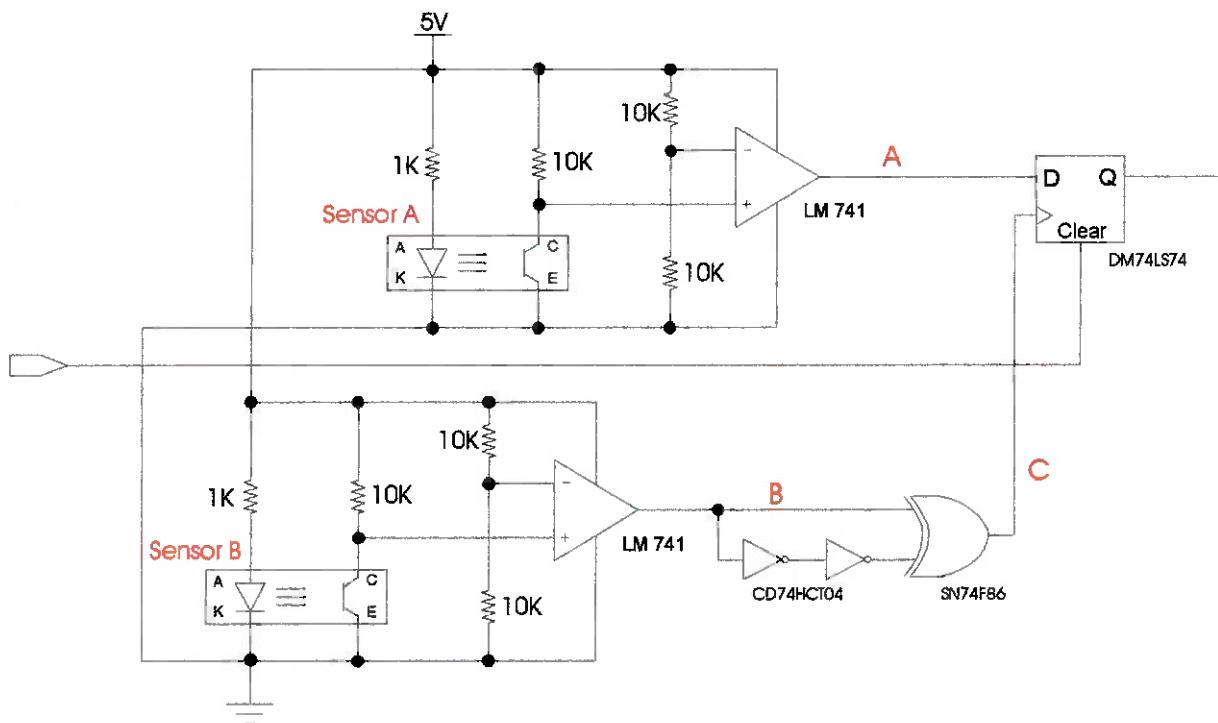


Figura 4.3 –Circuito completo do encoder

Os sinais A B e C assumem a seguinte forma:

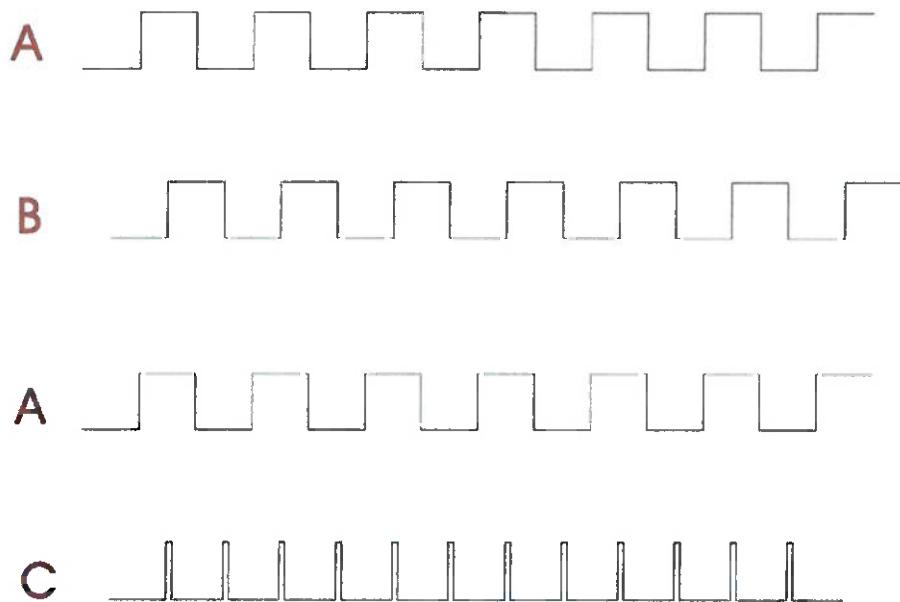


Figura 4.4 – Sinais A e B dos photo-sensores e sinais A e C para o flip-flop

O sinal C é responsável por fornecer os pulsos de clock para o flip-flop que registra em sua saída o nível lógico de A bem condicionado, pronto para ser enviado ao microcontrolador.

Serão utilizados os timers do microcontrolador para registrar os pulsos do encoder. São timers de 12 a 16 bits geralmente disponíveis no mercado, portanto, são capazes de registrar 4096 pulsos e 65536 pulsos respectivamente. A roda que será utilizada possui 90 mm de diâmetro, logo, 282,7 mm de circunferência. O disco do encoder com 50 ranhuras fornece uma precisão de 5,6 mm ( $282,7 / 50 = 5,6$  mm). Dessa maneira, um contador de 16 bits pode contar até 367m de deslocamento( $65536 \times 0,0056 = 367$  m)

Este valor é suficiente para a aplicação do robô, uma vez que é sensivelmente superior às dimensões de salas comerciais e residenciais. Cabe colocar que os timers serão resetados a cada fim de curso do robô (parada ou mudança de direção de deslocamento do robô).

Uma outra alternativa para contagem dos pulsos do encoder seria realizada por software, onde uma interrupção no programa verificaria a mudança no sinal enviado pelo encoder e incrementaria uma variável. Deve-se apenas tomar um tempo de interrupção que respeite o princípio de Nyquist para amostragem de sinais. O princípio diz que o período de amostragem (no caso, o intervalo entre cada interrupção), deve ser duas vezes menor ou igual ao menor período do sinal amostrado. Na prática, esse valor deve ser pelo menos dez vezes menor.

Para o nosso caso, a velocidade máxima do motor é de 24 rpm. Com um disco de encoder de 50 ranhuras, temos 1200 pulsos por minuto, ou 20 pulsos por segundo. Dessa maneira, um pulso ocorre a cada 50 ms. Adotando uma freqüência de amostragem dez vezes maior, a interrupção precisa ocorrer a cada 5 ms ou com freqüência de 200Hz. Assim, garante-se que nenhum pulso será perdido, e a contagem do encoder será eficaz.

Os controladores atuais operam com uma freqüência muito maior, cerca de 20 MHz o que garante a implementação de tal alternativa.

## 5 SENSOR DE NÍVEL DE BATERIA

O sensor de nível da bateria foi pesquisado para que se possa monitorar a carga de cada uma das baterias de 12V empregadas no robô.

Utiliza-se um circuito integrado (Ver datasheet em CD anexo) que monitora a tensão da bateria continuamente. A tensão por ela fornecida irá passar por um amp-op que converte a tensão para um valor entre 0 e 4V, que alimenta o pino 5 ( $V_{REF}$ ) como mostra a figura a seguir. Os valores de R1 e R2 são ajustados de modo a se obter uma interface pontual ou em barra.

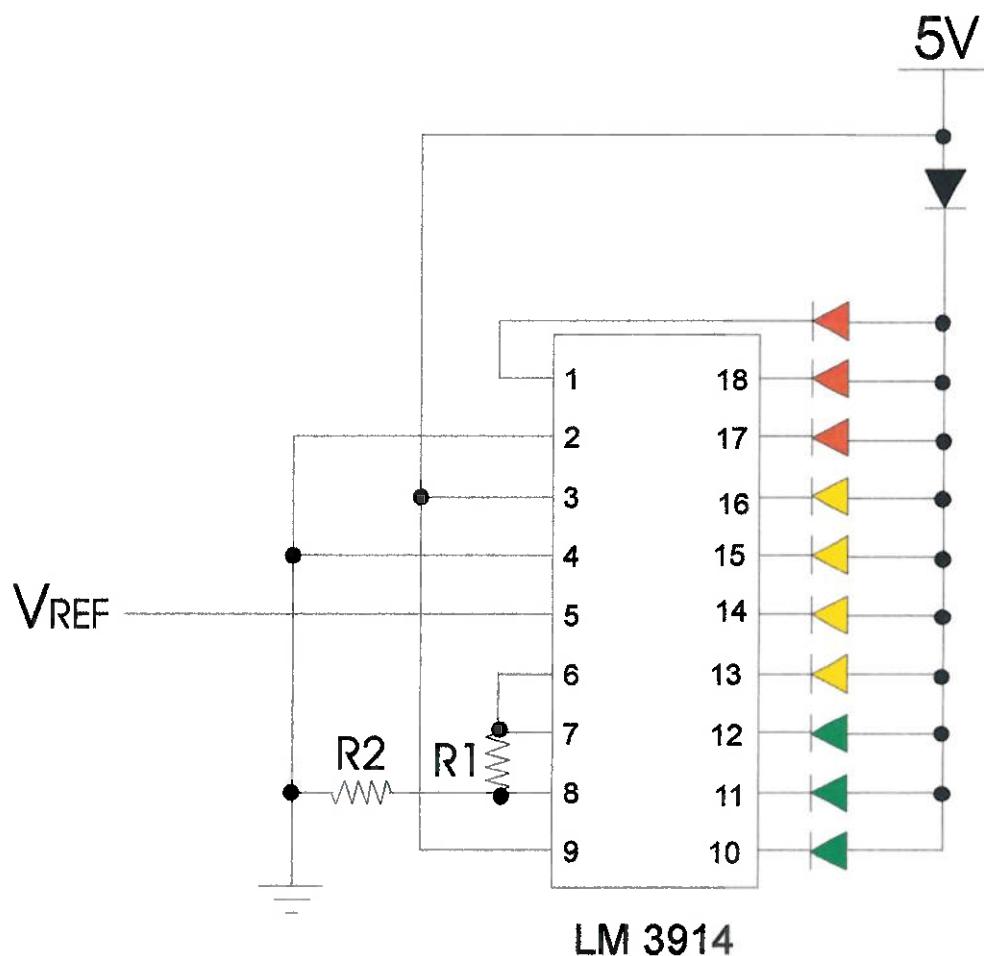


Figura 5.1 – Circuito do sensor de nível de bateria

A interface pontual é dada com um único led aceso ed cada vez. Indicando do primeiro led verde (carga máxima) passando led a led até o último led vermelho (carga mínima).

A interface em barra acende todos os leds no início indicando carga máxima na bateria. A partir daí, os leds vão apagando um a um indicando queda de tensão da bateria, no sentido de apagar primeiro os verdes, depois os amarelos e, finalmente, os vermelhos.

A saída 18 (penúltimo led vermelho) será conectada, também, ao microcontrolador que, quando detectada, indicará a necessidade de retorno do robô à base para o recarregamento das baterias. Será utilizada a penúltima saída de modo a se garantir que ainda resta energia suficiente para acionar os motores e trazer o robô à base.

O mesmo sistema será implantado para monitorar à bateria que alimenta o aspirador, dando ao sistema flexibilidade suficiente para desligar o aspirador antes do término da carga da bateria, evitando o descarregamento total desta e aumentando, assim, sua vida útil.

Monitorando também a bateria que alimenta o aspirador, tem-se o aproveitamento melhor do tempo útil do robô, pois quando já não há potência suficiente para aspirar, o robô retorna à base para se recarregar, evitando que ele continue sua operação normal com baixa eficiência de limpeza.

## 6 BATERIAS

Existem diversas maneiras de fornecer energia a um robô. Alguns robôs de grande porte utilizam máquinas de combustão para gerar energia elétrica ou acionar atuadores hidráulicos ou pneumáticos.

Para um robô pequeno, no entanto, a energia proveniente de baterias traz uma série de vantagens, já que baterias são baratas, seguras, de tamanhos relativamente reduzidos e de fácil utilização. Além disso, os motores conseguem converter a energia elétrica em energia mecânica com uma alta eficiência. Existem muitos tipos diferentes de bateria, cada uma com suas vantagens e desvantagens.

Para que se possa escolher qual o melhor tipo de bateria para o aspirador robô, deve-se primeiramente realizar um estudo sobre elas. Antes disso é necessário saber quais os requisitos que as baterias devem possuir.

### 6.1 - Requisitos que a bateria deve apresentar

A escolha da bateria está diretamente relacionada como consumo total dos sistemas elétricos do aspirador robô. Quatro sistemas principais são os consumidores de energia: O sistema de aspiração, o sistema de movimentação da vassoura giratória, o sistema de locomoção do aspirador robô e o sistema de controle.

O sistema de aspiração é o responsável pelo maior consumo do aspirador robô. O motor elétrico responsável pela aspiração provém de um aspirador comercial para carros. O modelo utilizado para experiências iniciais de consumo é acionado a partir de uma tensão de 12V. Tal modelo de aspirador foi testado no laboratório para verificação da corrente de consumo, para com isso definir sua potência. Obteve-se um valor de corrente em regime permanente de uso de cerca de 2,5Amp, com um pico de corrente no início do funcionamento de 4 Amp. Assim, a potência do motor de aspiração se dá por:

$$P = V \cdot I$$

$$P = 12 \cdot 2,5$$

$$\mathbf{P = 30W}$$

O motor do aspirador testado poderia ainda ser substituído por um com maior potência, algo que só poderia ser verificado após a construção do protótipo e teste do sistema de aspiração. Caso fosse necessário, seria utilizado um motor com o dobro da potência e, consequentemente, o dobro do consumo de corrente, algo em torno de 5A. Assim, tem-se:  $P = 12 \cdot 5$       **P = 60W**

O consumo do sistema de locomoção foi testado após a montagem do driver de potência. Os motores, já pré-selecionados, são acionados por uma tensão de 24V. Verificou-se que, sem carga sobre os eixos dos dois motores responsáveis pela locomoção do robô e operando a velocidade máxima eles consomem, juntos, uma corrente de 0,3 Amp. A carga que será imposta sobre o eixo desses motores está relacionada apenas com a massa final no aspirador de pó robô, já que a velocidade será a máxima permitida pelo motor. Como ainda não se pode prever tal massa, principalmente pelo fato de que essa é uma fase de escolha de bateria (a maior responsável pelo aumento de massa), não se pode ter uma idéia muito clara de qual será o torque imposto sobre o eixo do motor.

Mas para fim de testes iniciais, adicionou-se uma carga (sem medição de magnitude) mas que se considera superior a que os motores estarão submetidos. Tal imposição de carga foi realizada tentando-se segurar o eixo do motor. Notou-se que sua velocidade praticamente não era alterada com a imposição da carga. Tal fato pode ser explicado já que a redução presente na caixa do motor é de cerca de 85 vezes. Assim, mesmo com a imposição da carga, a corrente de consumo não ultrapassou o valor de 0,35 Amp. Como já dito, pelo fato de não se ter o valor do torque que será imposto sobre o eixo dos motores, o valor da corrente pode ser admitido, para se ter um fator de segurança, como 0,4 Amp.

Com isso, temos:

$$P = V \cdot I \quad P = 24 \cdot 0,4 \quad \mathbf{P = 9,6W}$$

O sistema de movimentação da vassoura giratória é feito por um pequeno motor de 4,5V. Tal motor foi testado e, alimentado com esta tensão, consome uma corrente de cerca de 0,15 Amp em regime permanente sem carga sobre o eixo. Aplicando-se uma carga sobre o eixo até atingir uma velocidade angular igual a zero, obteve-se um consumo de corrente de 0,4 Amp.

Assim, o consumo máximo do motor é dado por:

$$P = V \cdot I \quad P = 4,5 \cdot 0,4 \quad \mathbf{P = 1,8W}$$

Já o consumo do sistema de controle depende de todos os circuitos presentes no robô. Por se tratar de circuitos de baixa potência não serão considerados para a escolha da bateria. Eles utilizarão a mesma bateria que acionará os motores.

Além do consumo, a bateria deve atender a algumas restrições, como:

O tamanho deve obedecer à restrição que impõe a altura máxima do robô de 100 mm.

Além dessa dimensão, ela deve ser o menor possível para que sua distribuição dentro do robô lhe permita uma otimização espacial que reduza seu tamanho ao máximo, proporcionando um produto mais compacto.

Um ponto muito importante é qual a divisão das baterias para os diversos sistemas, ou seja, qual sistema utilizará qual bateria. É sensato pensar que os sistemas podem ser agrupados da seguinte forma: sistema de limpeza e sistema de locomoção. O sistema de limpeza é composto pelo sistema de aspiração e de movimentação da vassoura giratória. O sistema de locomoção é composto pelos motores de locomoção e todo o sistema de controle.

Assim fica evidente que, para se manter ativo, ou seja, se movendo, desviando de obstáculos e com suficiência para retornar ao ponto de partida, o robô não necessita que o sistema de limpeza esteja em funcionamento. Logo, o fornecimento de energia para estes dois grupos pode ser separado. Com isso, teríamos uma bateria de 12V alimentando o motor do aspirador e o motor de movimentação da vassoura giratória, e mais uma bateria de 24V para alimentar os motores de locomoção e os circuitos de controle. Defini-se com isso também que a bateria de 12V deverá possuir maior capacidade, já que alimenta o sistema com maior consumo de potência. E a bateria de 24V poderá ter uma menor capacidade, já que os motores de locomoção e os circuitos não consomem tanta potência.

Agora, definido os requisitos mínimos, pode-se fazer um estudo sobre as baterias disponíveis.

## 6.2 - Características das baterias

Motores CC típicos podem operar com uma tensão entre 1.5V e 100V ou mais. Para a fabricação de robôs, normalmente se utilizam motores que operam com 6, 12 ou 24V, já que as baterias são normalmente encontradas com esses valores de tensão. No caso do robô aspirador têm-se motores de 12 e 24V. Os termos *Bateria* e *Célula*, são normalmente usados se discernimento, mas eles têm significados diferentes. Tecnicamente, uma célula é uma caixa em que uma reação química é responsável pela geração de energia elétrica, enquanto que uma bateria é um conjunto de células conectadas.

### 6.2.1 - Voltagem

A voltagem produzida pela reação química de uma célula depende dos materiais utilizados na reação. Essa voltagem é chamada de *voltagem nominal da célula*, e varia de acordo com a tecnologia usada na bateria. Uma célula para luzes de flash utiliza uma reação de Corbono-Zinco e possui uma voltagem de 1.5V. Já a maioria das baterias de carro possui seis células de chumbo-ácido, cada uma com uma voltagem de 2V, totalizando os 12V da bateria.

### 6.2.2 - Capacidade

Em geral, quanto maior a célula, mais corrente e energia ela pode fornecer. A capacidade da célula é medida em Ampere-Hora, que significa o número de horas que uma célula pode fornecer uma certa quantidade de corrente antes que sua voltagem fique menor que um valor limite pré-determinado. Por exemplo, uma bateria alcalina de 9V (que consiste internamente de seis células alcalinas de 1.5V) geralmente possui 1 AmpH, o que significa que a bateria pode fornecer continuamente uma corrente de um ampere por uma hora antes de “morrer”. Na medição de capacidade, uma bateria de 9V é considerada não-funcional quando sua voltagem fica menor que 5.4V (valor este determinado por um comitê).

Entretanto, a capacidade da bateria depende de como ela é usada. O padrão de medição AmpH assume um tempo de descarga de 20 horas. Isto é, uma bateria de 9V deve ser testada mantendo um fornecimento de 1/20 de sua capacidade (o que seria 50 miliamps), por 20 horas. Se ela for drenada rapidamente, como num teste de uma hora, ou com uma alta

corrente aplicada, a capacidade efetiva será muito menor. A voltagem irá descer abaixo do limite antes que o produto corrente por tempo atinja 1 AmpH.

A curva de descarga de voltagem de uma bateria de Chumbo-ácido é praticamente uma linha inclinada para baixo. Baterias de Níquel-Cádmio mantém sua voltagem por mais tempo e decaem próximo do fim de uso. Apesar das baterias de Chumbo-ácido ter mais energia por peso (densidade de energia) grande parte da energia não é utilizada pois, a voltagem de 12V da bateria decai abaixo do limite do equipamento que está sendo suprido, normalmente, para uma câmera esse valor é 11.5V. As filmadoras desligam automaticamente a 11.5V. Uma bateria de 14V teria uma descarga mais profunda, mas isso é ruim para a bateria e encurtaria sua vida útil.

### **6.2.3 - Densidade de energia**

Em muitas aplicações a razão mais importante de uma célula é a capacidade por unidade de peso, ou *densidade de energia*. A densidade de energia difere completamente entre os tipos de bateria. Células baratas de Corbono-Zinco têm a menor densidade de energia entre todos os tipos de células. Células alcalinas têm algo em torno de 10 vezes a densidade de energia de uma célula de Corbono-Zinco, já as células de Níquel-Cádmio possuem uma densidade de energia menor que as alcalinas, mas são recarregáveis.

### **6.2.4 - Curva de descarga**

A voltagem de uma célula decresce com a vida da bateria conforme ela vai se descarregando. A curva característica de descarga varia consideravelmente com os tipos de células. Por exemplo, as células alcalinas possuem uma queda quase linear entre a célula cheia e zero volt. Desde que a queda seja gradual, é fácil dizer quando a capacidade da bateria foi superada e ela precisa ser trocada. Células de Níquel-Cádmio possuem uma região com queda linear bem suave até um ponto em que sua voltagem cai drasticamente. Uma tensão constante é muito importante em certas aplicações, mas as células de Níquel-Cádmio têm uma morte súbita sem avisos.

### 6.2.5 - Resistência interna

Uma célula pode ser modelada como uma fonte de tensão perfeita em série com um resistor. Essa característica importante das células, chamada de resistência interna, determina a razão com a qual a energia pode fluir para fora da célula, juntamente com uma máxima voltagem e corrente.

Células de Chumbo-ácido têm uma resistência interna muito baixa, o que faz com que elas tenham um grande uso em baterias de carro, já que uma grande corrente pode ser drenada da bateria para operar o motor de arranque. Durante o tempo de recarga de um flash de uma câmera, as células devem fornecer carga o mais rápido possível. Novamente, a relação é limitada largamente pela resistência interna da célula. Células alcalinas têm resistência interna maior que as células de Níquel-Cádmio, por essa razão os flashes terão um tempo de recarga maior quando usa células alcalinas.

Células com baixa resistência interna (especialmente Níquel-Cádmio e Chumbo-ácido) podem ser perigosas para se trabalhar, já que grandes correntes podem fluir da célula quando se curto-circuita externamente, já que a resistência interna é tão baixa. Essas correntes aquecem o metal pelo qual estão passando, derretendo facilmente a isolação. As células podem aquecer até explodirem. Por essa razão é muito importante não curto-circuitar uma bateria de Chumbo-ácido ou Níquel-Cádmio. Células alcalinas e de Carbono-Zinco também não devem ser curto-circuitadas, mas sua alta resistência interna fará com que a corrente tenda ao limite. Ainda assim, elas podem aquecer e causar incêndios.

### 6.2.6 - Recarga

Uma outra característica importante das células é se elas são ou não recarregáveis, e se forem, quantas vezes podem ser recarregadas. O recarregamento é mais barato que a troca, e muitas vezes mais conveniente. A química nas células são muito tóxicas ao ambiente, o que torna o recarregamento mais apropriado. Infelizmente, as células com maior densidade de energia (alcalinas e lítio) não são recarregáveis, embora as características de células recarregáveis sejam propícias.

### 6.2.7 - O efeito memória

O termo efeito memória se refere a um fenômeno que ocorre em células de Níquel-Cádmio recarregáveis. Se uma célula for parcialmente descarregada antes da recarga, ela tende a “lembra” o nível de descarga, e, com o tempo, se torna útil apenas até aquele nível de descarga. Há um desacordo entre os fabricantes de células quanto à existência atual do efeito memória, mas a maioria concorda que as células de Níquel-Cádmio devem ser completamente descarregadas antes da recarga. Algumas tecnologias de células, como Chumbo-ácido e as novas células de Níquel-hidreto, não exibem esse efeito. Células de Chumbo-ácido normalmente duram centenas de ciclos de descarga completa e milhares de ciclos de descarga parcial.

### 6.2.8 - Custo

Por último mas não menos importante é o custo: como muitas coisas na vida (mas não todas), células com alta performance custam muito mais que células com baixa performance. Para propósitos de consumo, é geralmente aceito que células de Níquel-Cádmio, que custam muito mais que células alcalinas, são menos caras que o número de vidas da célula. Células de Níquel-Cádmio podem ser recarregadas centenas de vezes, enquanto que células alcalinas são descartadas após um uso. Em contrapartida, células de Níquel-Cádmio exibem a “morte súbita”, descrita anteriormente.

A tabela abaixo demonstra a maioria das tecnologias mais comuns de células. O advento dos notebooks e dos carros elétricos têm criado uma grande demanda para o aperfeiçoamento das baterias. Novos tipos de bateria incluem as de Níquel Metal Hidreto, Lítio-polímero, Sódio Sulfato, Zinco-Ar e Zinco-Bromo. A alta capacidade de ser recarregável faz com que as células de Níquel hidreto sejam muito caras, mas oferecem o dobro da capacidade se comparadas com as células de Níquel-Cádmio e chumbo-ácido. O problema das novas baterias é que nenhuma delas oferece todas as características de uma bateria ideal: baratas, longa-vida, alta densidade de energia e seguras.

Tipo de célula	Voltagem (V)	Densidade de energia	Resistência interna	Recarregáveis	Custo
Carbono-Zinco	1,5	baixa	alta	não	baixo
Alcalina	1,5	alta	alta	não	moderado
Lítio	3,0	muito alta	baixa	sim	alto
Níquel-Cádmio	1,2	moderada	baixa	sim	moderado
Chumbo-ácido	2,0	moderada	baixa	sim	moderado
Níquel Hidreto	1,2	alta	?	sim	muito alto
Lítio Polímero	?	alta	?	sim	muito alto
Sódio Sulfato	?	alta	?	sim	muito alto
Zinco-Ar	1,5	muito alta	alta	sim	muito alto
Zinco-Bromo	?	alta	?	sim	muito alto

Tabela 08 – Características das células mais comuns

### 6.2.9 - Pacotes de baterias (*Packs*)

Existem duas maneiras que as células podem ser combinadas: conexão em série ou paralelo. Quando as células são conectadas em série sua voltagem é somada mas sua capacidade em AmpH não. Baterias em série devem ser compostas por células de mesma capacidade. Quando as células são conectadas em paralelo, a voltagem total continua a mesma, mas a capacidade é somada.

### 6.2.10 - Carregador de bateria

Uma regra para o recarregamento de baterias é carregar com uma relação igual a um décimo da capacidade AmpH da bateria. Por exemplo, se a bateria tem uma capacidade de 1,3 AmpH, ela deve ser carregada com uma corrente de 130mA. No entanto, alguns fabricantes recomendam relações de carga específicas para suas baterias. Por exemplo, a bateria Panasonic de 12V usada no carregador ELEC 201 tem uma relação de carga máxima recomendada de 520mA.

O carregador da figura abaixo foi desenvolvido para carregar com uma corrente de no máximo 520mA, como recomendado pelo fabricante. O carregador também foi desenvolvido para prevenir a bateria de sobrecarga, e para indicar quando a bateria estiver completamente carregada. Essas tarefas são feitas pelo uso de um transistor no circuito com um switch que controla a corrente para a bateria e um diodo zener para perceber quando a bateria estiver carregada. O diodo zener requer uma tensão de 12V para que uma corrente flua. Assim, se a voltagem da bateria estiver abaixo desse valor o resistor de polarização ligará o transistor

(conduzirá) e toda a corrente da fonte CC irá fluir para a bateria, carregando-a. Com a bateria carregada, sua voltagem sobe, e o diodo zener começará a conduzir corrente, desviando-a da base do transistor, desligando-o. O transistor desligará completamente quando a voltagem da bateria atingir cerca de 13V: 12V sobre o zener, 1,7V sobre o LED indicador, menos 0,7V da tensão de polarização reversa do transistor. Nesse ponto, a corrente da fonte CC flui através do diodo zener e do LED. O resistor do circuito limita a máxima corrente de carga na bateria, e controla a corrente que vai para a base do transistor.

Devido ao circuito de sensibilidade de tensão, esse carregador não irá impor uma sobre carga na bateria. O LED provê um indicador para carga completa.

Battery Charger V4.1  
(c) 1998 Rice University

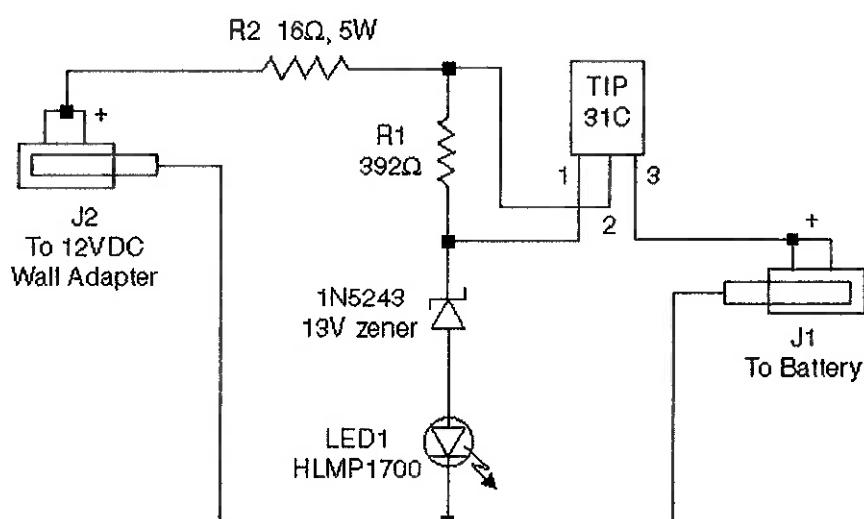


Figura 6.1 – Carregador ELEC 201

Baterias de Chumbo-ácido e seus recarregadores são relativamente baratos e disponíveis em todo mundo. Quanto maior a bateria, por mais tempo o equipamento irá operar. Uma bateria de 10 AH durará mais que duas de 5 AH já que a maior consegue segurar melhor a sua voltagem.

Carregadores “inteligentes” (que carregam rapidamente e então param a carga evitando danos) estão cada vez mais sendo utilizados. Carregadores de 16 horas são baratos, mas levam muito tempo para recarregar, mas pelo menos não causam danos à bateria se for deixado ligado após a completa carga da mesma.

## 6.3 - Escolha da bateria

Analisado esses aspectos iniciais, pode-se perceber que duas opções são as mais evidentes para a utilização nesse projeto. Pela tabela apresentada (Tabela 1 – Características das células mas comuns), as duas opções mais viáveis devido, principalmente, ao custo e ao fato de serem recarregáveis, são as baterias de Chumbo-ácido e de Níquel-Cádmio. São baterias facilmente encontradas no mercado por um preço relativamente acessível.

Realizou-se então uma pesquisa para obter mais informações sobre tais baterias. Verificou-se que as baterias de NiCd não são encontradas para o fornecimento da potência requerida pelo robô. Poderiam ser montados pacotes (packs) mas seu custo seria muito alto, além da massa e dimensão que não atenderiam os requisitos necessários.

Com isso, a bateria escolhida para suprir as necessidades do robô é a bateria de Chumbo-ácido. Neste ponto, é necessário se fazer um estudo mais aprofundado de tal tipo de bateria para que se possa fazer a escolha do valor de AmpH da bateria.

## 6.4 - Baterias de Chumbo-ácido seladas

### 6.4.1 - Visão geral

Baterias Chumbo-ácido de glass matte ou de gel são freqüentemente usadas em aparelhos de vídeo já que o líquido das células das baterias de Chumbo-ácido (como nas baterias de carro) podem vazar ácido por possuírem orifícios (não são seladas). Células de glass matte e gel são baratas se comparadas a baterias de NiCd além de não possuírem o efeito memória. Infelizmente, diferentemente das de NiCd, as baterias de Chumbo-ácido podem ser facilmente danificadas se forem deixadas em uma condição de descarga completa, ou se forem sobrecarregadas, assim, é importante carregá-las sempre antes do uso, e utilizar um carregar especial para baterias de Chumbo-ácido.

Baterias de 12V de glass matte ou gel são feitas de seis células dentro da bateria, cada uma com uma voltagem de 2.15V, totalizando 12.9V. Baterias de glass matte usam uma pasta

grossa (grossa como cera de vela ou pasta de dente) de eletrólito de ácido sulfúrico entre as placas positivas (Dióxido de Chumbo) e as placas negativas (esponja de Chumbo) da bateria, com um isolador poroso entre as placas evitando o curto. Células de gel são similares exceto pelo eletrólito ser como uma “geléia” que pode ser danificada pelo congelamento. Nos dois tipos as placas são relativamente finas se comparadas com as baterias úmidas, logo elas descarregam um pouco mais rápido.

Células de glass matte ou gel não possuem fluido se movimentando no interior da bateria como ocorre nas baterias úmidas de Chumbo-ácido e por isso não necessitam de manutenção já que elas não podem ser cobertas pelo eletrólito. Por essa razão, é muito importante ter um carregador de alta qualidade que não ferva o eletrólito, o que poderia secar o eletrólito e danificar a bateria. Após muitos anos de uso, todas as baterias recarregáveis não deveriam ser dispostas em lixo comum que vai para um aterro, mas sim para locais apropriados de lixo tóxico.

#### **6.4.2 - Taxa de carga e descarga**

As baterias de Chumbo-ácido possuem uma resistência interna tal que as mantêm carregadas por um tempo maior que as baterias de NiCd antes de se auto-descarregarem, ainda que as baterias de Chumbo-ácido de glass matte ou gel não se descarregam tão rápido quanto as baterias úmidas de Chumbo-ácido. Mas essa resistência interna implica que essas baterias não conseguem fornecer tanta corrente por um pequeno período de tempo como uma de mesmo tamanho de NiCd, e mais, elas levam mais tempo para serem carregadas do que as baterias de NiCd.

Uma regra prática é que as baterias de Chumbo-ácido úmidas não aceitam uma carga maior que 25% da sua taxa AmpH até 80% do seu nível de carga (chamado “volume de carga”), e por isso carregam mais devagar de 80 a 100% (o que se chama de absorção ou aceitação de carga), isto é, levam no mínimo 4-5 horas para carregar uma bateria completamente descarregada de Chumbo-ácido úmida. Já uma bateria de Chumbo-ácido de glass matte ou gel possuem uma taxa de carga/descarga maior que as de Chumbo-ácido úmidas devido as suas placas finas bem próximas e podem ser carregadas a uma taxa de 40% de sua AmpH até 80% do seu nível de carga, e então carregar mais devagar de 80 a 100%, isto

é, levam um mínimo de 2,5 a 3,12 horas para carregar uma bateria completamente descarregada de Chumbo-ácido de glass matte ou gel. O padrão de carga não é linear e uma bateria de Chumbo-ácido será melhor carregada quando estiver próxima do vazio e o carregador irá gradualmente cobrir a corrente de carga conforme a bateria começa a ser preenchida (80%) e atinja a voltagem limite, ou ela será sobre-carregada fazendo a bateria esquentar e produzir gás, o que danificaria a bateria. Um aquecimento leve durante a carga ou descarga é aceitável, mas uma bateria realmente quente indica que um sério dano está ocorrendo. A temperatura máxima que a bateria pode atingir é cerca de 52° C.

#### **6.4.3 - Carregador de qualidade e regulador**

As baterias de Chumbo-ácido necessitam de um carregador com uma leve diferença no nível de voltagem e um método diferente de carga que os carregadores de baterias de NiCd. Como as baterias de glass matte e gel podem ser facilmente danificadas pela sobre-carga comparadas às baterias de NiCd, o carregador deve cuidadosamente limitar o nível de voltagem o nível de corrente quando estiver completando a carga. Níveis de voltagem em baterias de Chumbo-ácido durante a carga devem ser reguladas entre 13,6 – 13,8V, algumas vezes indo até 14,2V (máximo 14,4V). Uma vez que a célula de gel esteja quase carregada, a variação do nível deve estar entre 13,2 – 13,6V e pode ser carregada por “gotejamento” ou pulsos com uma corrente bem pequena por muitas horas sem danos.

Usando um carregador próprio, baterias de glass matte ou gel não produzem vapores corrosivos (gases explosivos). Alguns poucos carregadores de baterias de carro de 12V que trabalham nas faixas de 10-12A fazem um trabalho cuidadoso de carregar as baterias de Chumbo-ácido de glass matte ou gel de 6-14AmpH, e alguns não controlam nada (eles sobreaquecem e destroem a bateria). Alguns modelos de carregadores não convertem a corrente para zero Amps conforme vão atingindo a carga total. Isso pode danificar baterias de pequeno porte. Lembrando que 40% da taxa AmpH é o máximo que as baterias de glass matte ou gel podem agüentar. Logo uma bateria de 6AmpH não pode receber uma corrente maior que 2,4A do carregador, uma de 14AmpH não deve receber correntes maiores que 5,6A. Assim, deve-se sempre tentar escolher um carregador que automaticamente leve a corrente a zero, e com isso, ele nunca fornecerá mais corrente do que a bateria pode agüentar.

#### **6.4.4 - Descarga profunda**

Como regra geral, quanto mais profunda for a descarga da bateria de Chumbo-ácido e por quanto mais tempo a bateria for deixada descarregada, mais isso diminuirá os ciclos úteis de vida da bateria. Muitas vezes é uma boa idéia não descarregar a bateria de Chumbo-ácido além de 50% de seu nível de carga, mas se for feita uma descarga profunda além dos 50%, é muito importante que ela seja carregada imediatamente, pois a sulfatação pode começar a ocorrer quando a bateria de Chumbo-ácido é deixada em uma condição descarregada (especialmente baterias de Chumbo-ácido úmidas) e isso irá drasticamente reduzir a vida útil da bateria.

Um exemplo para não descarregar uma bateria de glass matte ou gel além dos 50% seria usar uma lâmpada de 100W e 12 V continuamente por no máximo 22 minutos usando uma bateria de 12 AmpH antes de colocá-la de volta ao recarregador. Não é incomum se chegar a uma vida de 4 a 7 anos de uma bateria de Chumbo-ácido se ela for devidamente utilizada, dependendo do número e magnitude dos ciclos de carga e descarga. Baterias de Chumbo-ácido úmidas são mais susceptíveis a sulfatação do que as baterias de glass matte ou gel, e por essa razão as baterias de glass matte ou gel são mais requeridas para descargas profundas.

#### **6.4.5 - Capacidade das Baterias**

Baterias de Chumbo-ácido úmidas são medidas em “cold cranking Amps” (que significa o número de Amps que pode ser fornecido a -17,8°C, para mover uma máquina por 30 segundos sem cair para uma tensão abaixo de 7,2V), e medidas em “capacidade de reserva” (que significa o número de minutos que a bateria pode fornecer 25A com uma tensão acima de 10,5V a 26,7°C).

Mas essas medidas não têm sentido para algumas outras aplicações como equipamentos de vídeo por exemplo. Assim, as baterias de glass matte ou gel são medidas em “AmpHora”, que significa o número de Amps por hora a uma tensão nominal de 12V e 26,7°C que uma bateria pode continuamente fornecer por um período de 20 horas e não cair para menos de 10,5V. Por exemplo, uma bateria de 20 AmpH deve ser apta a fornecer 1 Amp

a 12V (12 Watts) continuamente por 20 horas antes que esteja completamente descarregada. A medida é feita para 20 horas pois baterias de glass matte ou gel têm uma alta resistência interna. Quanto mais rápido se descarrega uma bateria de Chumbo-ácido, menor será o total de AmpH que se conseguirá tirar dela. Por exemplo, se for descarregada essa bateria de 20 AmpH a 2 Amp (24 Watts) se conseguirá apenas 16,8 AmpH ou 8,4 horas (84% da capacidade). Descarregando a 4 Amps (48 Watts) se terá 14 AmpH ou 3,5 horas (70% da capacidade). Descarregando a 10 Amps (120 Watts) se terá 10,8 AmpH ou 1 hora (54% da capacidade) antes da bateria estar completamente descarregada ou atingir um nível mínimo aceitável de 10,5V. Por essa razão, para aplicações de vídeo e luzes é geralmente uma boa idéia ter baterias de glass matte ou gel maiores, do que se estivesse usando baterias de NiCd.

Mais informações sobre o funcionamento das baterias seladas de Chumbo-ácido pode ser visto no CD em anexo - “Technical Handbook Valve-Regulated Lead-Acid Batteries”.

## 6.5 - Curvas das baterias

Após o conhecimento das principais características das baterias, principalmente das suas capacidades, realizou-se uma pesquisa para que se pudesse observar as curvas características das baterias, e com isso, pudesse ser feita a seleção das baterias a serem utilizadas.

Obtiveram-se as curvas de três fabricantes: BB Battery (Ver no CD em anexo – “BP4.5-12”), Power Sonic (Ver no CD em anexo – “Power Sonic – Sealed Lead-Acid Batteries”) e Fiamm-GS.

Primeiramente, será feita uma análise para a escolha da bateria de 12V para o sistema de limpeza (motor do aspirador e da vassoura giratória).

Pelo fabricante Power Sonic tem-se:

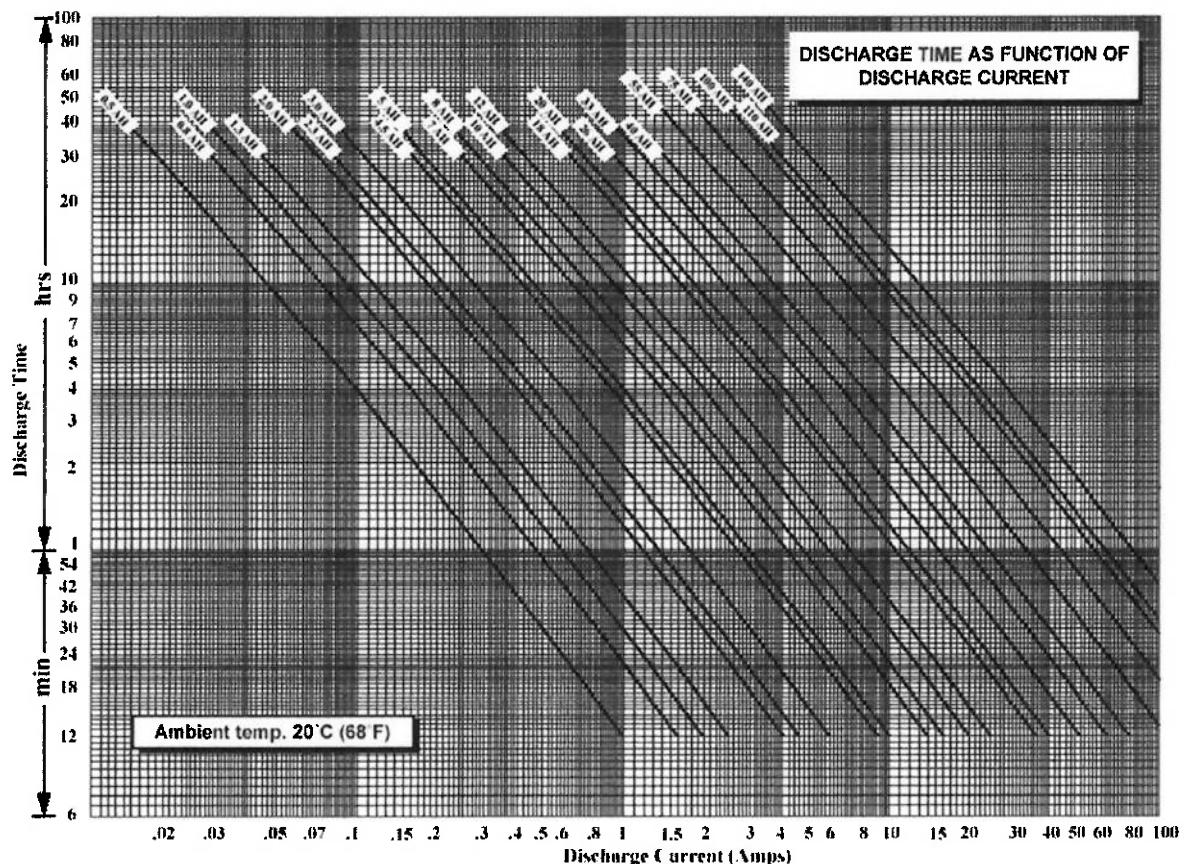


Figura 6.2 – Curva Tempo de descarga em função da corrente de descarga – fabricante Power Sonic

Pelo gráfico pode-se observar que, impondo uma corrente de 2,5 A (aproximadamente a corrente consumida pelo motor do aspirador de pó), e para a operação durante um período de 1 hora, a bateria mais indicada deve possuir 4,5 AH, ou seja, a corrente representa cerca de 0,55 vezes a capacidade da bateria.

O fabricante informa que, para que a tensão final de cada uma das seis células que compõe a bateria não seja inferior a 1,67V (totalizando 10,0V), a corrente imposta seja no máximo 0,5 vezes a capacidade da bateria. Assim, essa bateria de 4,5AH, deve operar com corrente máxima de  $0,5 \times 4,5 = 2,25$ A, para que sua tensão final não seja inferior a 10,0V após 1 hora. Mas, como a meta de operação imposta para o projeto não era de 1 hora, mas sim de 30 minutos, tal bateria se mostra adequada para o uso.

Agora, pela análise da curva demonstrada pelo fabricante Fiamm-GS, que também informa que a corrente imposta seja no máximo 0,5 vez a capacidade da bateria para que sua tensão final não seja inferior a 10,0V, tem-se que a duração é de 75 minutos para uma bateria de 4,5 AH, utilizando a curva de 0,5C, já que  $0,5 \times 4,5 = 2,25$ A.

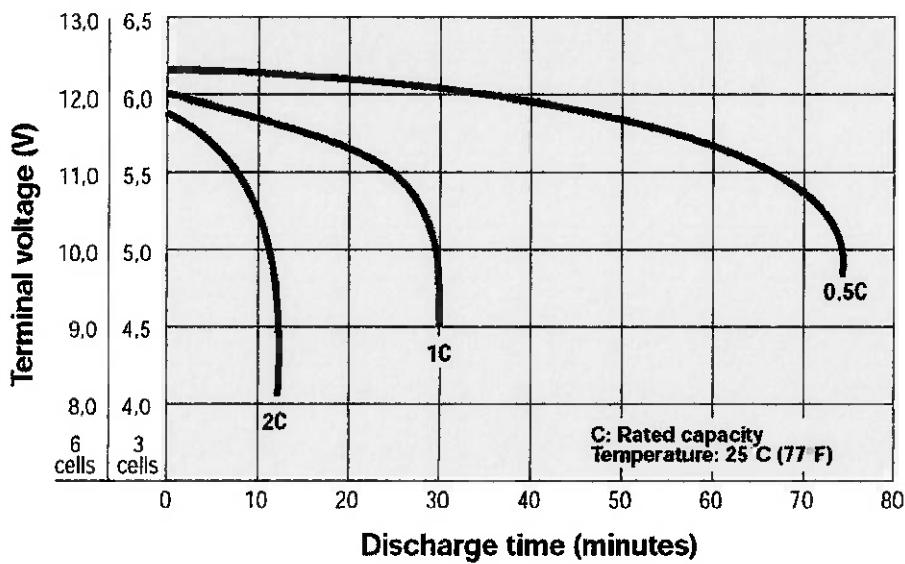


Figura 6.3 – Curva de voltagem terminal da bateria em função do tempo de descarga para uma dada corrente através da capacidade de carga da bateria – fabricante Fiamm-GS

Analisando a curva do fabricante BB Battery, percebe-se que com a utilização de uma corrente de cerca de 2,5 A, a duração da bateria de 4,5AH é de cerca de 1,2 hora, até atingir uma tensão terminal de 10V.

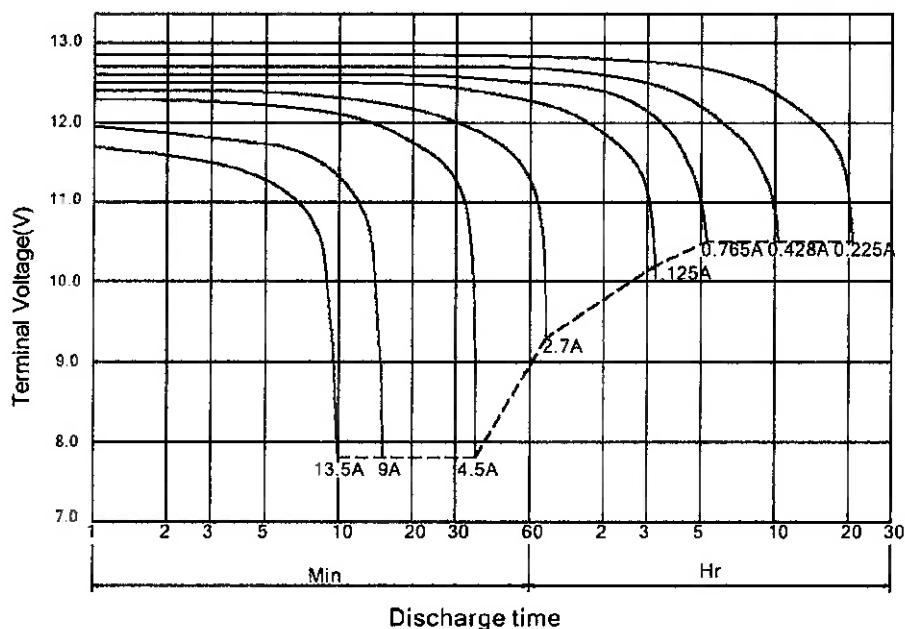


Figura 6.4 – Curva de voltagem terminal da bateria em função do tempo de descarga para uma dada corrente – fabricante BB Battery.

Assim, percebeu-se que o tempo médio de duração da bateria de 4,5AH fornecendo 2,5 A é de 1 hora.

## 6.6 - As Baterias escolhidas

A partir dessa primeira aproximação, partiu-se para uma pesquisa de preços. Constatou-se que a bateria mais comum possui tensão de 12V e capacidade de 7,2 AH. Seu preço varia entre R\$ 30,00 e R\$ 45,00, dependendo do fabricante. Tal variação está relacionada com a corrente inicial que a bateria pode fornecer. Quanto maior esse valor, mais cara é a bateria.

O fabricante que está presente na maioria dos pontos de venda e que possui a maior corrente inicial é a UNIPOWER. Tal fornecedor possui três alternativas viáveis. Todas possuem as mesmas dimensões externas de 151mm x 65 mm x 95 mm, mesma massa igual a 2,66kg e a mesma tensão nominal de 12V:

Alternativa	Capacidade (AH)	Corrente inicial (A)	Preço (R\$)
1	7,2	2,8	40,00
2	7,2	4,2	45,00
3	9	3,6	66,00

Tabela 09 – Tabela de decisão para escolha da bateria de 12V

A partir dessas considerações, optou-se pela bateria de 9AH (Ver no CD em anexo “BP8-12”) como a mais indicada para o robô. Isso se deve ao fato já citado de que o motor de aspiração de 30W utilizado nos testes pode vir a ser substituído por um outro motor de 60W, consumindo o dobro da corrente e, assim, requerendo uma bateria com o dobro da capacidade.

Assim, como se pode verificar pela curva do fabricante Power Sonic , utilizando-se a bateria de 9AH para fornecer no máximo uma corrente de 5A, ela terá uma duração de aproximadamente 1 hora.

Pensando-se agora na bateria para a alimentação do sistema de locomoção e circuitos, temos que, pela pesquisa realizada nos pontos de venda, é preferível a formação de uma tensão de 24V pela junção de duas baterias de 12V em série. Com isso, a tensão será de 24V e a corrente fornecida será a mesma de uma única bateria. Tal corrente engloba os motores de locomoção (0,4 A) e os circuitos. Como aproximação inicial, a corrente total estimada é de 1A. A tensão de 24V será utilizada para alimentar os motores de locomoção e, utilizando-se a tensão de apenas uma das baterias de 12V, serão alimentados todos os circuitos.

Assim, pelo mesmo gráfico do fabricante Power Sonic, impondo uma corrente de 1A pode-se verificar que, para a operação durante um período de 1 hora, a bateria mais indicada é a de 2AH. Novamente, como o fabricante informa que, para que a tensão final de cada uma das seis células que compõe a bateria não seja inferior a 1,67V (totalizando 10,0V), a corrente imposta seja no máximo 0,5 vez a capacidade da bateria, tem-se que a corrente máxima deve ser  $0,5 \times 2 = 1\text{A}$ , para que a bateria dure por 1 hora.

Pela pesquisa feita nos pontos de venda, encontrou-se apenas uma bateria com capacidade de 2,3AH, também da UNIPOWER (Ver no CD em anexo – “BP2.3-12”), o que está de pleno acordo com as requisitos necessários. Tal bateria possui dimensões externas de 180mm x 60mm x 35 mm e massa igual a 1kg.

Com isso, definiram-se quais as baterias serão utilizadas:

- **Uma bateria da UNIPOWER, com tensão nominal de 12V, capacidade de 9AH, dimensões externas de 151mm x 65 mm x 95 mm, massa igual a 2,66kg e preço de R\$ 66,00.**
- **Duas baterias da UNIPOWER com tensão nominal de 12V, capacidade de 2,3AH, dimensões externas de 180mm x 60mm x 35 mm, massa igual a 1,0kg e preço de R\$ 38,00.**

## 7 Microcontrolador

O microcontrolador a ser utilizado não será projetado nesse trabalho, mas sim, adquirido de um fabricante especializado. Não se tem por objetivo estudar todo o funcionamento do microcontrolador, e sim seu uso na prática. Dessa maneira, a aquisição de um microcontrolador montado, com saídas mapeadas, memória, e todo o circuito que dá base ao microcontrolador será feito frente à disposição de modelos do mercado.

A escolha do microcontrolador é dependente de uma série de fatores. Os principais fatores que influenciam sua escolha, para o caso desse projeto, são: número de pinos de I/O e quantidade de memória.

A velocidade do clock do microcontrolador não é uma barreira, pois, a maioria deles possui um clock muitas vezes maior que o necessário, sendo difícil encontrar microcontroladores com velocidade menor, visando a diminuição de custos. O programa que será executado não terá forte dependência da velocidade, já que possuirá comandos simples, o que não exige um grande processamento. Assim, as limitações se dão no número de pinos de I/O e na quantidade de memória.

A quantidade de memória necessária, não só para armazenamento do programa que será executado, mas também para o armazenamento de dados que serão coletados conforme o robô mapeia a região a ser aspirada, não tem uma estimativa muito clara. O tamanho do programa só será definido durante sua programação, tarefa essa a ser efetuada na próxima fase de projeto. Também não se tem conhecimento sobre a quantidade de informações que o robô deverá coletar para que possa definir sua posição e, com isso, ter a região mapeada, para posterior processamento de dados e definição da trajetória a ser percorrida.

Em vista disso, a memória requerida pelo microcontrolador será a maior possível, visando uma condição de segurança para a próxima fase de projeto.

A quantidade de pinos de I/O será então, a grande limitadora na escolha do microcontrolador. Esse número de pinos é definido a partir dos componentes que serão controlados pelo microcontrolador. Assim, a quantidade de pinos fica em:

- 8 pinos para controle de velocidade dos motores de locomoção;
- 1 pino para seleção de qual motor de locomoção receberá a velocidade definida;
- 1 pino para habilitar o motor de locomoção selecionado a receber a velocidade definida;
- 1 pino para habilitar o relê para liberação dos 24V para o driver de potência
- 2 pinos para receber o sinal dos dois encoders;
- 9 pinos para os sensores espalhados pelo robô;
- 2 pinos para os sensores de nível de bateria (12V e 24V);
- 2 pinos para acionamento do sistema de limpeza (aspiração e vassoura giratória);
- 4 pinos para a interface com o usuário (selecionar programa, executar, e dois led's)

Com isso, obteve-se um total de 30 pinos de I/O necessários para o controle do robô aspirador.

Pesquisou-se, então, uma série de alternativas para que se pudesse escolher o microcontrolador que suprisse essas necessidades e que fosse o mais barato possível já que, para a viabilização de um projeto, o custo é primordial.

Chegou-se às seguintes alternativas:

## 7.1 - Rabbit – Família RCM2000

Feature	RCM2000	RCM2010	RCM2020
<b>Microprocessor</b>	Rabbit 2000 at 25.8 MHz		Rabbit 2000 at 18.432 MHz
<b>Flash</b>		256K	
<b>SRAM</b>	512K		128K
<b>Backup Battery</b>	Connection for user-supplied battery (to support RTC and SRAM)		
<b>General Purpose I/O</b>	40 parallel I/O include: • 26 configurable I/O • 8 fixed inputs • 6 fixed outputs (grouped in five 8-bit ports and		

	shared with serial ports)
<b>Additional Inputs</b>	2 Startup Mode, Reset In
<b>Additional Outputs</b>	Status, Clock, Watchdog, Reset Out
<b>Memory I/O</b>	13 address, 8 data, I/O Read-Write, Buffer Enable
<b>Serial Ports</b>	Four 5 V CMOS-compatible, 2 configurable as clocked ports
<b>Serial Rate</b>	Max. burst rate = CLK/32 Max. sustained rate = burst/2
<b>Connectors</b>	Two 2 x 20, 2 mm IDC headers
<b>Slave Interface</b>	Slave port permits use as master or intelligent peripheral with Rabbit-based or other master controller
<b>Real-Time Clock</b>	Yes
<b>Timers</b>	Five 8-bit timers (four cascadable from the first) and one 10-bit timer with 2 match registers
<b>Watchdog/Supervisor</b>	Yes
<b>Power</b>	4.75–5.25 V DC, 130 mA
<b>Operating Temp.</b>	4.75–5.25 V DC, 98 mA –40°C to +85°C
<b>Humidity</b>	5–95%, non-condensing
<b>Board Size</b>	2.3" x 1.9" x 0.55" (58 x 48 x 14 mm)

Tabela 10 – características dos microcontroladores Rabbit – Família RCM2000



Figura 7.1 – Microcontrolador Rabbit - Família RCM2000

## 7.2 - Família 80C51

A CPU que acompanha o kit é a 80C31. Possui além disso, RAM: 32K, EPROM: 32K, 4 ports com conectores, Conexão Serial RS232 e cabo "COM2" para o PC, 8 saídas de I/O Mapeado (podendo assim controlar até 8 periféricos por I/O Mapeado ao mesmo tempo), Cls com soquetes torneados, pequena área de Prototipagem, Clock: 12 MHz, Reg. 5V Interno, Kit aceita: 80C31 / 80C32 / 87C51 / 87C52 / ATMEL(DIP-40 pinos). O programa desenvolvido roda na RAM do Kit (chave "load-run" com auto-reset). Assim, não se necessita gravar EPROM toda hora para desenvolver programas. Software de comunicação direta com PC. Vem com 2 cabos padrão DECN para os Ports e Conexão para I/O Mapeado. No site estão disponíveis programas-exemplo e documentação de utilização do mesmo.

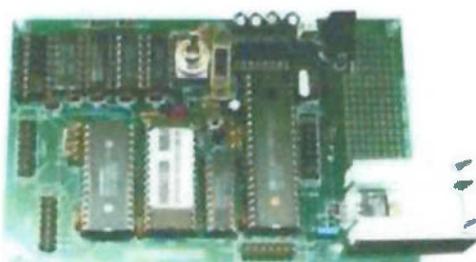


Figura 7.2 – Microcontrolador Família 80C51

## 7.3 - Alternativa escolhida

A opção mais completa é o microcontrolador Rabbit – RCM2020, pois sua velocidade de clock não é tão elevada, o que faz seu custo menor. Em dois revendedores pesquisados, o valor variou entre US\$ 150,00 e US\$ 300,00. Já o microcontrolador da família 80C51, foi encontrado por um valor bem inferior, em torno de R\$ 200,00. Mesmo com essa variação de preços, optou-se pelo controlador Rabbit, por possuir mais memória e tratar-se de um kit mais completo, já que na fase do projeto em que ele foi escolhido não se tinha tantos parâmetros para decisão.

## 8 Projeto Eletrônico

Inicialmente, a partir de todos os subsistemas eletrônicos já citados, pode-se perceber a ocorrência de uma divisão dos circuitos em 7 circuitos principais. São eles:

1. Driver de potência – (10 pinos de entrada, ligados ao  $\mu$ C, e 4 pinos de saída, ligados aos 2 motores);
2. Encoder – (2 pinos de saída, ligados ao  $\mu$ C, e 4 pinos de entrada, ligados aos receptores dos 2 encoders, onde cada encoder possui 2 receptores);
3. Sensoriamento (óptico e mecânico) – (8 pinos de saída, 7 ligados ao  $\mu$ C e 1 aos sensores para ajuste de sensibilidade, e 14 pinos de entrada, ligados aos sensores, 7 ópticos e 7 mecânicos);
4. Interface (chaves e led's) – (4 pinos de saída, sendo 2 ligados ao  $\mu$ C e 2 ligados aos led's, e 4 pinos de entrada, 2 ligados ao  $\mu$ C e 2 ligados às chaves);
5. Nível de bateria – (2 pinos de entrada, ligados um em cada bateria (12V e 24V), e 20 pinos de saída, ligados aos led's, sendo que 2 são ligados ao  $\mu$ C);
6. Acionamento (sistema de limpeza e driver) – (3 pinos de entrada, ligados ao  $\mu$ C, e 3 pinos de saída, ligados aos motores do aspirador, ao da escova giratória, e ao driver de potência);
7. Conversão de tensão – (6 pinos de entrada, ligados às baterias (12V 9AH, 12V 2,3AH e 12V 2,3AH), e 3 tipos de saídas, ligados aos outros circuitos (5V, 12V e 24 V))

Com isso, gerou-se um diagrama em blocos básico dos circuitos que compõe do o sistema eletrônico do robô:

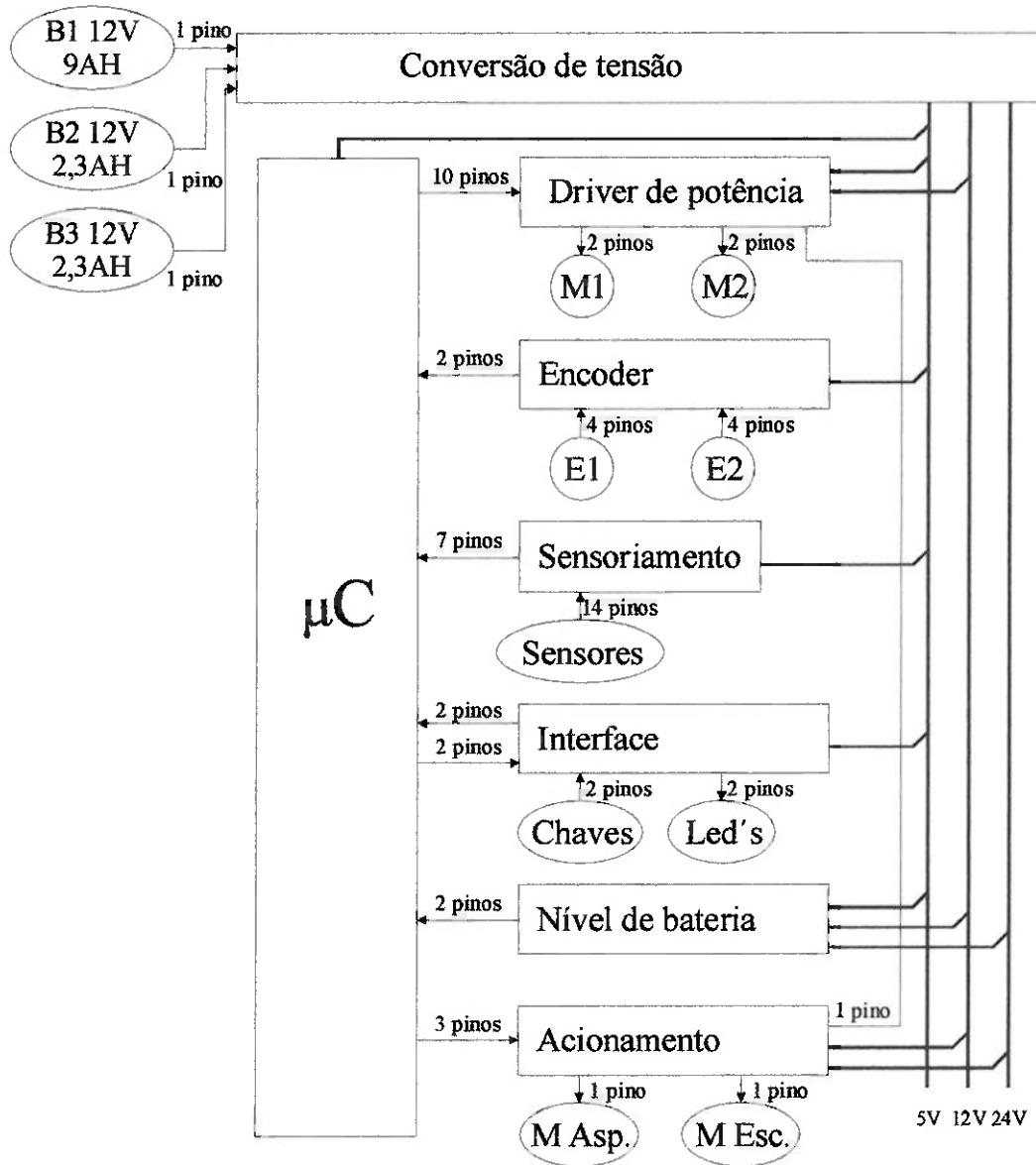


Figura 8.1 – Diagrama de blocos dos circuitos principais do robô

Os circuitos eletrônicos foram divididos em blocos funcionais de modo a facilitar suas concepções e fabricações. Cada subsistema foi implementado em uma placa distinta para que, ao final, pudesse ser testados independentemente, intercambiáveis e substituíveis.

Foi desenvolvida uma placa de potência e acionamentos responsável pelo recebimento das tensões das baterias de chumbo-ácido de 12V e geração das demais tensões utilizadas pelos sistemas e componentes do robô, 5V, 12V, 24V.

Uma segunda placa foi construída para receber os sinais dos sensores mecânicos e ópticos e combiná-los dois a dois em paralelo, de modo que cada sensor mecânico tem seu correspondente óptico, gerando em sua saída um único sinal para cada par indicando a detecção ou não de um obstáculo por este.

Esta placa ainda trata os sinais dos encoders ópticos que detectam a posição angular dos discos dos eixos dos motores de locomoção e, portanto, são utilizados para avaliar a velocidade e o deslocamento do robô.

A placa de drivers para os motores DC é responsável por receber os sinais digitais que formam um byte indicando a velocidade desejada para cada motor, convertê-los para sinais analógicos e, finalmente, gerar um perfil de onda que, após amplificado, dá a cada motor um nível médio de tensão correspondente à velocidade requerida à placa pelo sinais digitais supracitados.

Há sete pequenas placas de sensores ópticos compostas de um emissor infravermelho e um receptor sincronizado que filtra a luz ambiente e detecta a onda infravermelha emitida após sua reflexão em uma barreira física.

O microcontrolador está conectado a uma placa de prototipagem que carrega também o circuito de interface com o usuário.

Chegou-se então a um diagrama esquemático de todos os circuitos eletrônicos do aspirador robô. (Ver Anexo C - Diagrama esquemático dos circuitos).

A partir daí tem-se cada uma das 5 placas desenvolvidas.

## 8.1 - Placa de potência e acionamentos

Esta placa é responsável pelos acionamentos do motor de aspiração que opera com 12V de tensão e consome uma corrente nominal de aproximadamente 2A, do motor do

sistema de vassoura giratória, pelo fornecimento da tensão de 24V comutado por relé à placa do driver para os motores principais, responsáveis pela locomoção do robô, além de fornecer as tensões de alimentação pra as demais placas, 5V para a de encoder e leitura dos sensores, 5V para as placas dos sensores ópticos, 12V para a placa do driver e para o microcontrolador.

O sistema de aspiração é comutado pelo acionamento de um relé, acionado pela saída digital do microcontrolador. O sinal em nível lógico da porta paralela é conectado à base do transistor 2N2222 através de um resistor de  $1K\Omega$ , que garante que a corrente na porta não ultrapasse 12mA, corrente máxima que o microcontrolador pode fornecer em cada pino das portas de saída.

O transistor aciona a bobina do relé que fornece 12V para o motor de aspiração.

Semelhante ao sistema de potência para o motor de aspiração, a placa fornece 24V para os drivers dos motores de locomoção do robô. Um transistor 2n2222 é acionado por um dos pinos de saída do microcontrolador através de um resistor de  $1K\Omega$ , fornecendo corrente suficiente a uma tensão de 5V para acionar um segundo relé que fornece comuta e envia 24V para a placa dos drivers.

A vassoura giratória é acionada por um motor CC que opera em série com um resistor de  $47\Omega$  a uma tensão de 12V. Esse resistor foi utilizado para reduzir a tensão sobre o motor de modo que ele operasse sem riscos ao seu enrolamento e sem a necessidade de mais um regulador de tensão. O microcontrolador envia um sinal digital que, também através de um resistor de  $1K\Omega$ , gera a corrente de base necessária para liberar a corrente do emissor de um terceiro transistor 2N2222, que fornece a tensão de 12V para o sistema.

A placa recebe as entradas de 3 baterias de 12V de chumbo-ácido, duas de 2,3Ah e uma de 9Ah. As primeiras são ligadas em série na placa e servem exclusivamente para fornecer a tensão de 24V para os motores de locomoção. A bateria de 9Ah alimenta o microcontrolador e as demais placas e sensores do robô. O polo negativo (terra) desta bateria é ligado ao polo negativo das baterias ligadas em série.

A tensões de 5V são geradas por dois chips L7805, um exclusivamente para alimentar os emissores infravermelhos dos sensores ópticos. O segundo gera a tensão para a

alimentação das demais placas, com exceção da placa do driver que já possui um chip semelhante para gerar tal tensão internamente.

O microcontrolador recebe a tensão de 12V e a regula para 5V também através de um chip L7805, presente na sua placa de prototipagem.

Diodos foram colocados apontando do terra para as tensões geradas e recebidas pela placa para a proteção contra picos de tensão nos acionamentos dos relés e tensões reversas que possam ser geradas nos drivers de potência.

Em suma, a placa de potência possui:

- 3 entradas de alimentação

- 2 baterias de 12V e 2,3Ah

- 1 bateria de 9Ah

- 3 entradas digitais

- Comutação do motor de aspiração

- Comutação do motor da vassoura giratória

- Comutação de 24V para a placa dos drivers

- 2 saídas de 5V

- Alimentação dos sensores ópticos e mecânicos

- Alimentação da placa de encoder e sensores

- 3 saídas de 12V

- Alimentação da placa de prototipagem do microcontrolador

- Alimentação do motor de aspiração

- Alimentação do motor da vassoura giratória

- 1 saída de 24V

- Alimentação dos drivers dos motores de locomoção

Ver desenho esquemático da placa no Anexo D - Circuito de potência e acionamentos do sistema de limpeza e do driver de potência.

## 8.2 - Placa de encoder e sensores

A placa recebe como entradas as saídas dos receptores infravermelhos em nível lógico *high* ou *low*, que é ativo 5V (nível lógico 1). Os botões táteis, também ativos em *high*, ou seja, quando detectam um obstáculo têm suas saídas em nível lógico 1. São 7 entradas para os sensores mecânicos (táteis).

Essas entradas são combinadas com as 7 entradas dos sensores ópticos de modo que operem em paralelo, ou seja, a saída de um determinado sensor vai para zero quando um sensor óptico ou mecânico ou, ainda, ambos detectam um obstáculo. São 7 pares de sensores óptico/mecânico que utilizam 7 pinos de *input* do microcontrolador.

Os sinais desses sensores são combinados através de portas *AND*, implementadas através de dois chips 74LS08, que possuem 4 portas *AND* de duas entradas cada.

Quando qualquer um dos sensores, seja mecânico ou óptico, de um determinado par de sensores vai a zero (nível lógico *low*), independentemente da saída de seu sensor complementar, a saída do conjunto vai a zero. Com isso, garante-se que esses pares operam em paralelo.

Há, ainda, uma entrada que, após passar por um potenciômetro de  $200\Omega$  é ligada ao terra. Esta entrada recebe os emissores infravermelhos ligados em paralelo. Variando o valor da resistência deste potenciômetro para um valor menor tem-se uma intensidade maior na emissão desses leds devido à maior corrente que circula no circuito. Quando a variação é inversa, aumenta-se o valor desta resistência, a intensidade da emissão é reduzida. Em geral, este valor é mantido próximo de  $40\Omega$ , o que garante uma intensidade de emissão suficiente para que o detector óptico ative a uma distância que varia de 2cm a 5cm do obstáculo.

São 16 pinos de comunicação com os conjuntos de emissores e receptores dedicados à detecção da posição angular dos eixos das rodas do robô (encoder).

São 4 pinos para cada conjunto emissor/receptor (K, E, A, C) e cada eixo possui dois pares deslocados de 90°, responsáveis pela detecção da posição angular do disco de encoder em fases defasadas, possibilitando a detecção de quadratura e evitando sinais oscilantes e ruídos quando os eixos estão parados.

As saídas dos receptores (C) são comparadas em 4 amplificadores operacionais (LM741) distintos a tensões fixas de 2,5V. Quando em operação, tais saídas apresentam um formato senoidal com valor mínimo 0V e máximo 5V. Comparados esses sinais com 2,5V nos amplificadores operacionais, suas saídas apresentam formas de onda quadradas nos mesmos níveis de tensão máxima e mínima.

Para cada par de sinais das saídas dos amplificadores, tem-se um conectado na entrada D de um flipflop tipo D (74LS74) e o segundo é invertido duas vezes, utilizando o chip 74HCT04, antes de ser conectado a uma porta *XOR* (ou exclusivo) implementada pelo chip CD4030. A segunda entrada desta porta possui o sinal original antes da dupla inversão, responsável simplesmente por atrasá-lo. A saída do *XOR* é, portanto, sempre zero com exceção de pequenos pulsos em nível 1 quando o sinal de entrada é comutado de zero para 1 ou vice-versa. Por fim, esta saída dá o *clock* no flipflop supracitado que grava em seu sinal Q o atual estado da entrada D, filtrando-a de sinais oscilatórios indesejados.

Em suma, a placa encoder/leitura dos sensores possui:

- 7 entradas dos sensores ópticos
- 7 entradas dos sensores ópticos
- 1 entrada dos emissores infravermelhos
- 16 pinos para os conjuntos de emissores/receptores de encoder
- 4 conjuntos K, E, A, C

- 7 saídas dos sensores com os sinais combinados
- 2 saídas de leitura dos pulsos do encoder
- 2 entradas de alimentação

Ver desenho esquemático da placa no Anexo F - Circuito dos encoders e sensores.

### **8.3 - Placas dos sensores ópticos**

São placas muito simples totalizando 7 pequenos conjuntos de emissores/receptores infravermelhos.

Os receptores são sincronizados para que filtrem a luz ambiente e detectem unicamente ondas no espectro do infravermelho.

Cada placa possui um emissor infravermelho de 940nm comprimento de onda, um receptor sincronizado, um resistor de  $270\Omega$  responsável por dissipar a potência quando o conjunto está ativo e um capacitor para filtrar ruídos.

Ver desenho esquemático da placa no Anexo G - Circuito do chicote de sensores.

### **8.4 - Interface**

O circuito de interface, apesar de ter sido projetado, juntamente com o circuito de nível de bateria (Ver desenho esquemático da placa no Anexo H - Circuito de nível de bateria e interface), não foi implementada por um atraso no cronograma.

A interface do microcontrolador com o usuário é feita através dos botões e leds presentes na placa de prototipagem do Rabbitcore RCM2000.

São três leds e três botões ligados a três pinos de saída e três pinos de entrada do microcontrolador que podem ainda, entretanto, ser utilizados para comunicação I/O.

O primeiro led indica que o microcontrolador possui alimentação e, portanto, está ligado. Os dois leds seguintes ficam acessos indicando que é necessário fazer o *setup* inicial do sistema. Quando este é realizado, o segundo led apaga. Quando o programa de varredura é iniciado, o terceiro led apaga e o robô entra em operação.

A cada parada do robô, o terceiro led acende indicando que o programa pode ser interrompido pelo usuário durante aproximadamente 1 segundo, quando o robô está imóvel.

Ao final do programa os leds 2 e 3 piscam alternadamente indicando que o processo foi finalizado e um novo *setup* e reinicio podem ser feitos.

O primeiro botão é o *reset* do sistema que reinicia o software no microcontrolador como se este acabara de ser carregado em sua memória.

O segundo botão é utilizado para fazer o *setup* da máquina, que pára os motores e prepara os sistemas para o início do processo de limpeza.

O terceiro botão é responsável por iniciar o programa de varredura da superfície do ambiente, após feito o *setup* ou, ainda, interrompe o processo de varredura quando o robô faz paradas sistemáticas e temporizadas a cada obstáculo detectado e consequente rotação.

## 8.5 - Driver de potência

A placa do driver de potência já está detalhada do capítulo 2.

Ver desenho esquemático da placa no Anexo E - Circuito do driver de potência dos motores CC.

## 9 Projeto Mecânico

### 9.1 - Sistema de Locomoção

Como já comentado no tópico sobre acionamento dos motores, o disco de enconder não terá nenhuma redução para a contagem dos giros do motor. Seu acoplamento será feito diretamente no eixo do motor, já que a contagem feita dessa maneira é suficiente para dar a precisão necessária para o controle de movimentação do robô.

As rodas também serão acopladas diretamente, já que a rotação proveniente do motor é exatamente a necessária para dar ao robô a velocidade especificada. Essa rotação é de 24 RPM, rotação essa adquirida experimentalmente. Pela limitação imposta de altura do robô em 100 mm, o valor do diâmetro da roda será de 90mm.

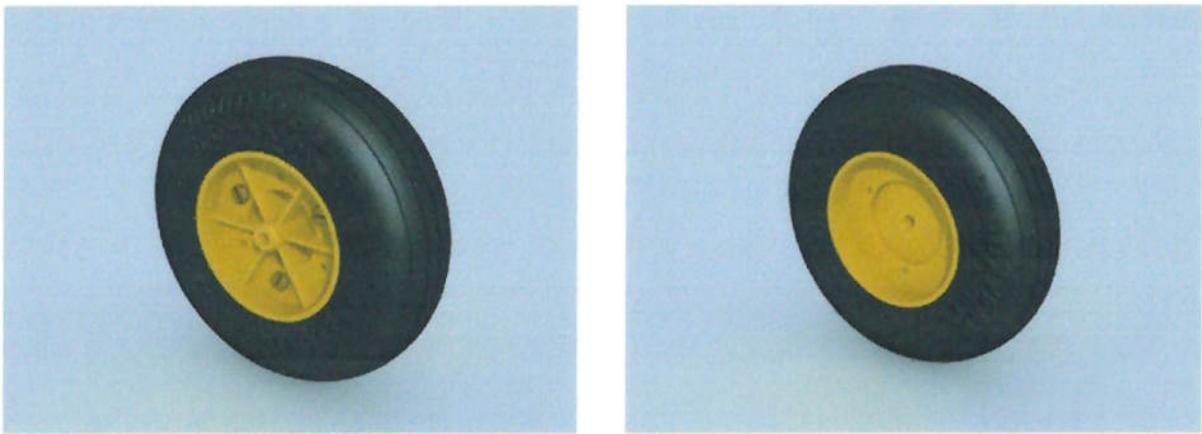


Figura 9.1 – Rodas para locomoção – 90 mm de diâmetro. (Desenho 3D com dimensões reais)

A roda escolhida foi encontrada em casa especializada em modelismo. Possui roda em plástico e pneu de borracha. O par pode ser encontrado por um valor de R\$ 45,00.

Dessa maneira, a velocidade máxima do robô será de:

$$V = \frac{n}{60} \cdot 2\pi \cdot R \Rightarrow V = \frac{24}{60} \cdot 2\pi \cdot 45 \Rightarrow V = 114 \text{ mm/s}$$

Pelo fato do eixo do motor possuir um comprimento de apenas 18mm, o acoplamento com o disco de encoder e a roda não pode ser feito sem a adição de um prolongador. Em vista disso, fez-se um projeto do eixo que deve ser acoplado ao motor para permitir essa transmissão.

Levou-se em conta, também, a fixação do disco de encoder. Essa fixação será realizada por uma placa de alumínio fixada ao disco através de cola epóxi. Essa placa será posicionada de tal maneira que se encaixe ao chanfro presente no eixo do motor, e imponha um atrito suficiente para que o disco não saia com tanta facilidade. As figuras abaixo mostram o acoplamento do disco de encoder ao eixo do motor.

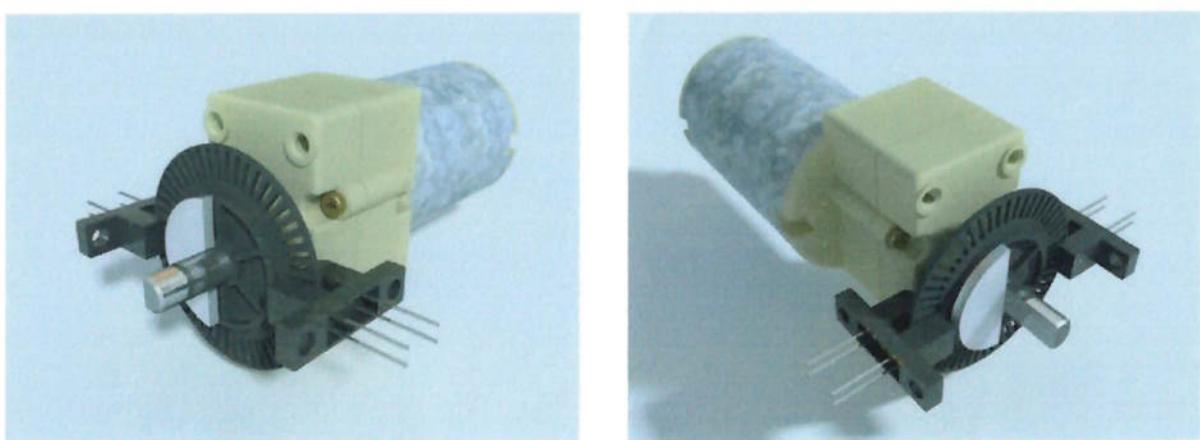


Figura 9.2 – Acoplamento do disco de encoder ao eixo do motor juntamente com os sensores (Desenho 3D)

Após o encaixe do disco de encoder com a placa ao eixo do motor, o prolongador será então fixado através de um pequeno parafuso prisioneiro em sua lateral. A fixação do eixo à roda será feita por três parafusos.

As figuras abaixo mostram o prolongador do eixo, feito em alumínio com o parafuso lateral para fixação.

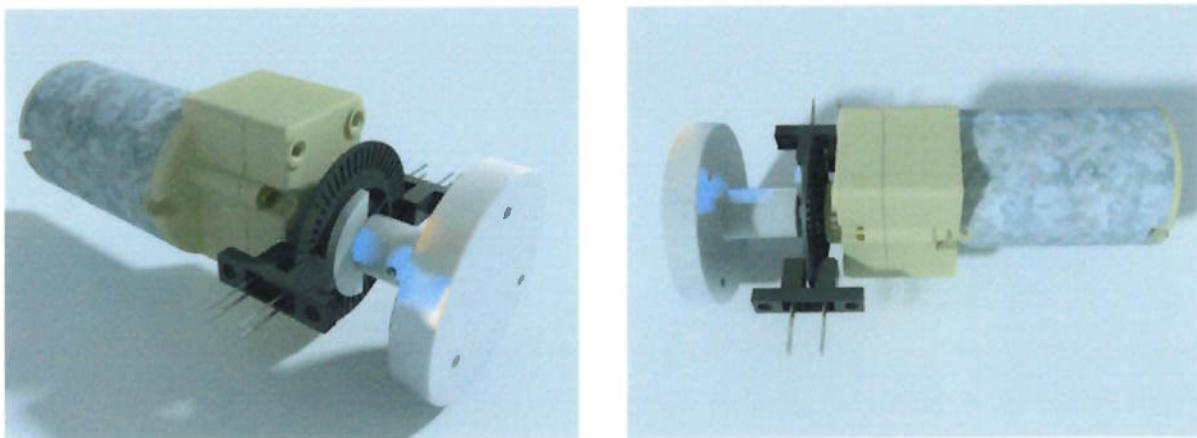


Figura 9.3 – Montagem do prolongador ao eixo do motor, juntamente com o disco de encoder e os sensores  
(Desenho 3D)

Dessa maneira, evitou-se a utilização de mancais, facilitando o projeto, a fabricação, montagem e diminuindo custos de produção. A redução presente na caixa do motor é suficiente para fornecer a velocidade e o torque necessários. Percebe-se então como a escolha de um bom motor é eficiente para todas as fases do projeto.

Para a fixação do motor ao chassi do aspirador, e fixação dos sensores do encoder, projetou-se um suporte que realiza as duas funções. O projeto de tal suporte levou em conta alguns fatores de extrema importância, tais como:

1 - Volume: o suporte deveria ter o menor volume possível, já que uma das restrições do projeto é o volume total do robô, e para isso, todas as peças devem ter sempre um tamanho reduzido. Chegou-se a uma solução com dimensões de 50 x 49 x 25 mm.

2 - Rígidez: como estará sustentando o motor, que é umas das partes mais importantes do robô, responsável pela sua locomoção, além de ser o apoio principal que irá receber todas as forças “peso” de tudo que estiver no robô, a rigidez é um dos quesitos primordiais. Para tal, não optou-se pelo uso de chapas dobradas, que não possuem uma rigidez tão alta, e sim por um bloco de alumínio usinado, formando um perfil em “U” com uma base.

3 - Facilidade construtiva: Apesar de complexa, essa peça não tem uma construção tão problemática. Praticamente utiliza-se de uma série de passadas de fresagem, e algumas furações. A dificuldade maior na construção desse suporte está nos rasgos para os sensores do

encoder. Como tais sensores devem estar posicionados de uma maneira tal que proporcionem uma defasagem de sinais, enquanto um deve detectar uma janela, o outro deve detectar a transição entre uma janela e uma parte fechada. O deslocamento relativo ao centro do eixo do motor deve ser de então de 0,5 mm, já que o rasgo do disco do encoder possui apenas 1 mm. Uma alternativa seria abrir um rasgo maior para os dois sensores e adicionar calços para alcançar a defasagem.

Dessa maneira chegou-se a um suporte que contém todas as características acima, facilitando a montagem e construção do protótipo. Percebe-se novamente como um bom motor pode ajudar o projeto, já que nesse caso a caixa de redução já possuía furos para fixação.

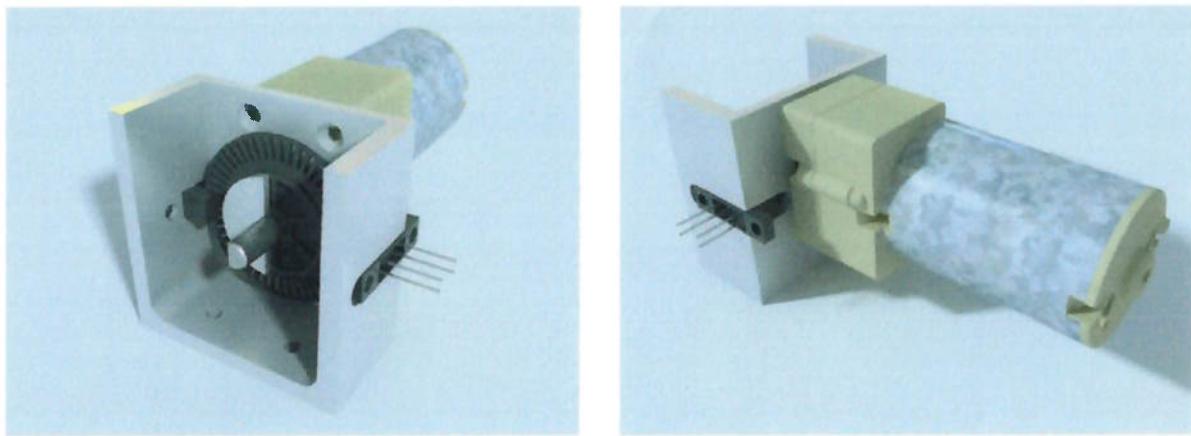


Figura 9.4 – Suporte do motor de locomoção e dos sensores do encoder (Desenho 3D)

Para fixação do suporte ao chassi do aspirador, idealizou-se uma certa flexibilidade de ajuste de altura. Tal flexibilidade é importante para que se possa ajustar a distância do chassi ao piso que está sendo aspirado.

Esse ajuste é necessário já que não se pode prever com muita exatidão a força de sucção que o aspirador terá, e qual a distância mínima que o chassi deve ter do piso. Assim, entre o suporte do motor e o chassi, foram colocados calços de borracha densa, que além de permitir o ajuste de elevação também servem como absorvedoras de vibrações de alta freqüência, normalmente produzidas pelo motor.

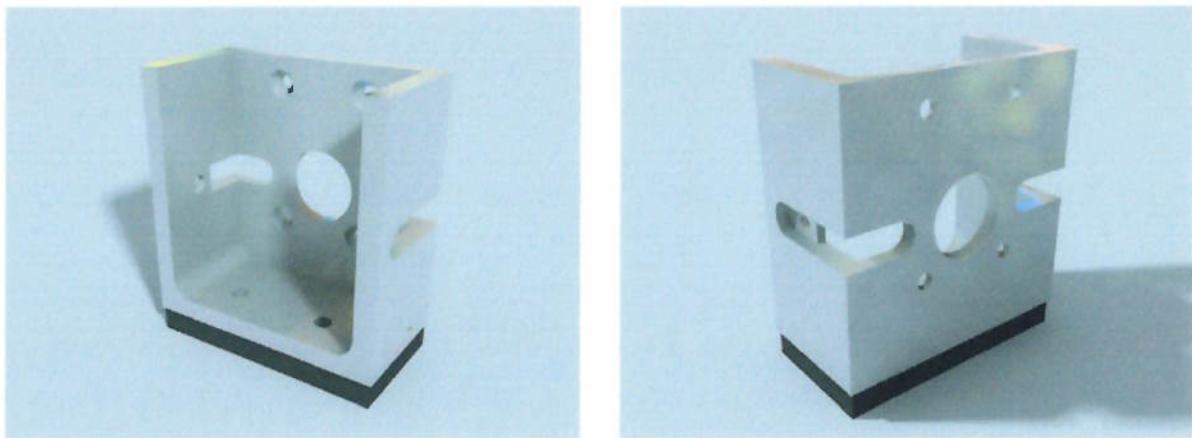


Figura 9.5 – Suporte do motor com calço de borracha (Desenho 3D)

Com isso, chegou-se ao sistema motor do sistema de locomoção formado pelo motor, suporte do motor, disco de encoder, placa de fixação do disco, dois sensores ópticos, eixo de acoplamento, roda e calço de borracha. O sistema se torna então uma peça única, de fácil montagem e independente de qualquer outro sistema do robô para sua montagem ao chassi.

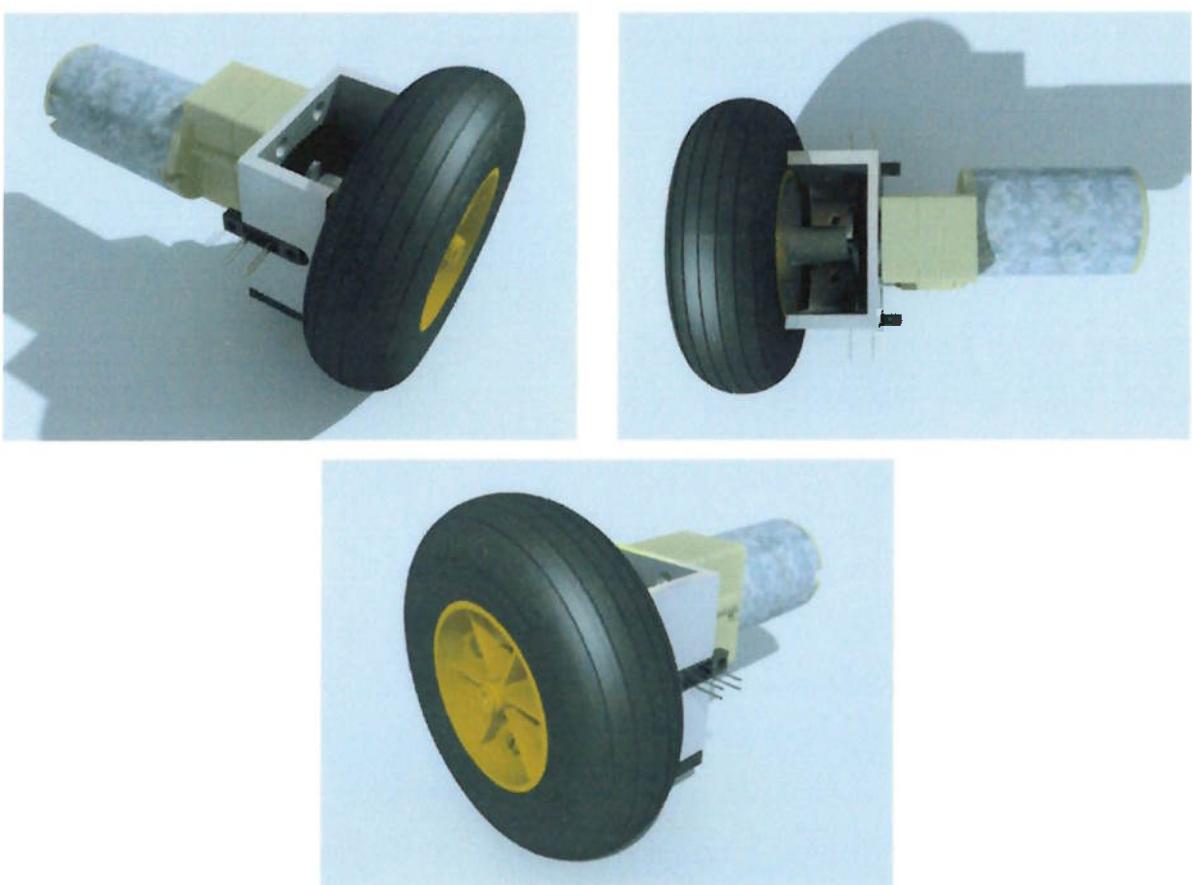


Figura 9.6 – Representação do sistema motor do sistema de locomoção (Desenho 3D)

Para auxílio no sistema de locomoção foi necessária a adição de duas pequenas rodas de apoio, já que o sistema motor é composto por apenas duas rodas centradas ao chassi do robô.

Tais rodas devem seguir o sentido e direção de movimento de deslocamento do robô. Para isso projetaram-se duas pequenas rodas com um suporte que lhe permite girar livremente para acompanhar o movimento. Tal acompanhamento se deve ao fato do pivô de rotação vertical das rodas estar deslocado em relação ao seu centro. Com isso, quando ocorre uma mudança na direção do movimento do robô, o chão exerce uma força lateral à roda, e o desalinhamento do pivô gera um momento que a faz se alinhar à direção do movimento.

Tais rodas foram envolvidas por borracha para evitar riscos nas superfícies a serem aspiradas. Sua concepção é apresentada a seguir.

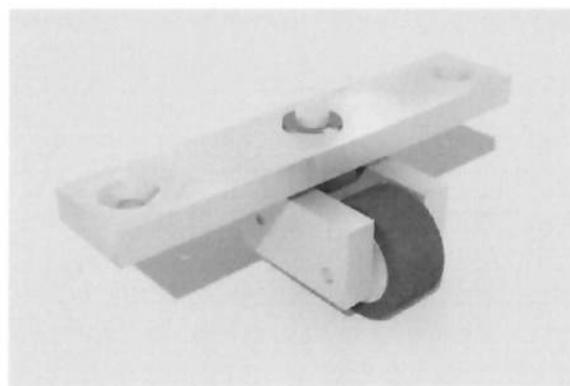


Figura 9.7 – Rodas de apoio (Desenho 3D)

As rodas de apoio necessitam ainda de um possível ajuste de altura devido ao fato de que um apoio de quatro pontos (duas rodas motoras e duas de apoio) pode trazer um problema quando o peso do que está sendo apoiado está deslocado do centro geométrico.

O que ocorre é que o objeto pode tender a se apoiar apenas em três pontos. Como temos que garantir que dois desses apoios sejam as rodas motoras, uma das rodas de apoio pode ser não utilizada, e para isso um ajuste de altura seria necessário. Para tanto, pode-se colocar arruelas por baixo do suporte das rodas entre este e o chassi, elevando a roda de apoio e consequentemente abaixando o chassi. Com isso a roda de apoio oposta é retirada do chão, garantindo o apoio nas rodas motoras.

Dessa forma, concluiu-se o projeto final do sistema de locomoção, com os acionamentos de locomoção de duas rodas motoras e as duas rodas de apoio.

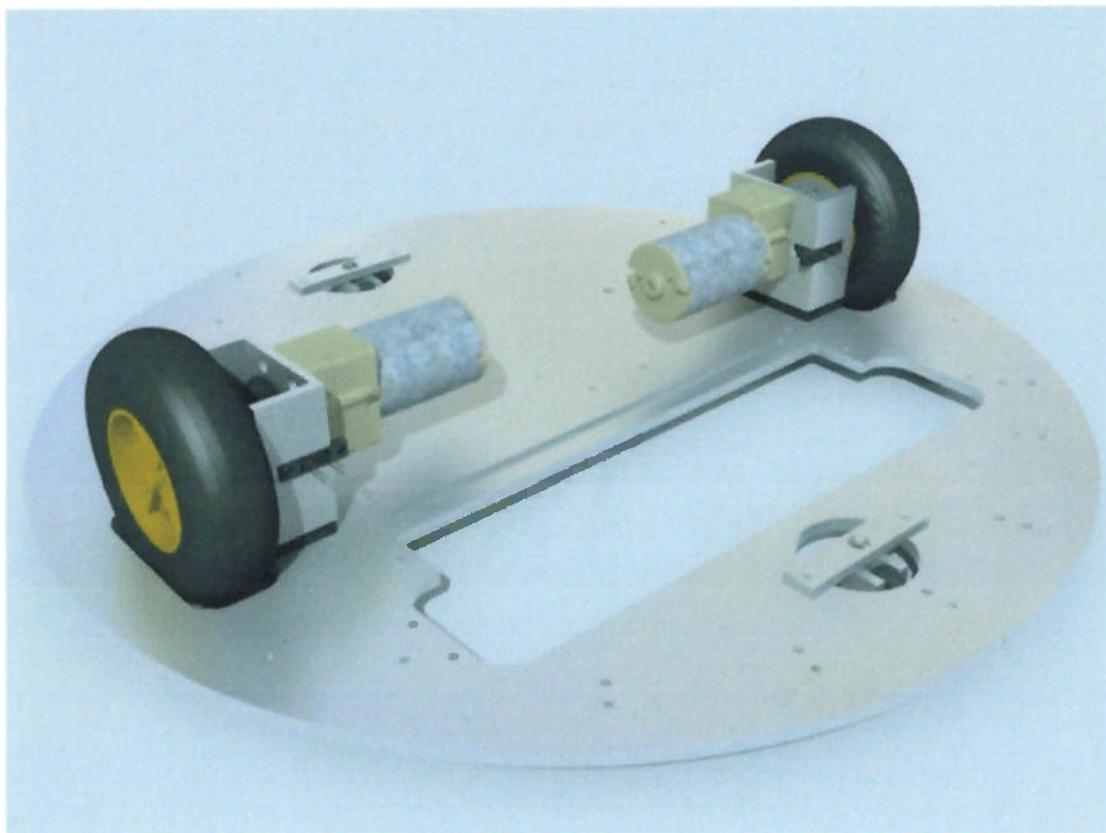


Figura 9.8 – Sistema de locomoção completo. Sistemas motores e rodas de apoio (Desenho 3D)

## 9.2 - Sistema de Limpeza

O sistema de limpeza do robô aspirador é formado por um sistema de sucção e um sistema de limpeza com vassoura giratória.

O sistema de sucção é responsável pela aspiração de sujeiras de menor tamanho, como o pó, cabelos, e pequenas partículas de sujeira.. O sistema de vassoura giratória é responsável pela coleta de sujeiras com maiores dimensões. Como este sistema se encontra à frente do sistema de sucção, a vassoura ajuda a levar as partículas de sujeira para trás, aumentando a eficiência do sistema de sucção.

### 9.2.1 - Sistema de Sucção

O sistema de sucção, como já dito no tópico sobre baterias, é composto por um motor elétrico proveniente de um aspirador de pó para carros, vendido comercialmente. Tal escolha foi feita pois, juntamente com o motor elétrico, está acoplado o sistema de sucção composto por uma pequena ventoinha, um suporte para o motor, um suporte para o filtro, e o filtro em si.

Como o intuito do trabalho não é desenvolver um sistema de aspiração, e sim desenvolver um produto que tenha como foco a facilidade da automação, este sistema será incorporado de um produto já desenvolvido.

No sistema de sucção, além do motor, dos suportes e do filtro, existem mais três componentes importantes: o reservatório de pó, a tubulação, e o bocal da aspiração. Tais componentes foram fabricados especialmente para o robô.

O aparelho utilizado para testes iniciais foi um aspirador com potência nominal de 35W. Para comprovar tal potência, foi realizado um teste de consumo de corrente. A tensão nominal do motor elétrico, como deve funcionar com uma bateria de carro, é de 12V. Obteve-se um valor de corrente em regime permanente de uso de cerca de 2,5A, com um pico de corrente no início do funcionamento de 4 Amp. Assim, a potência do motor de aspiração se dá por:

$$P = V \cdot I \quad P = 12 \cdot 2,5 \quad P = 30W$$

Um pouco abaixo do valor estabelecido pelo fabricante.

O aspirador testado está apresentado a seguir:



Figura 9.9 – Aspirador de pó para carro testado

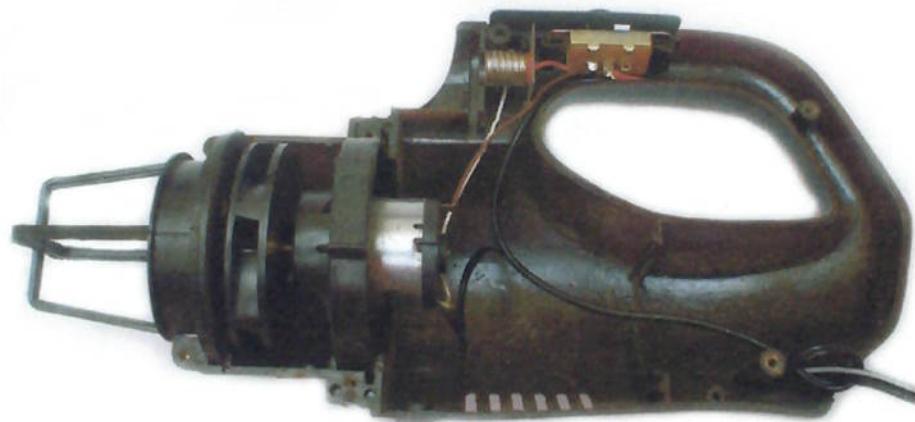


Figura 9.10 – Aspirador, visão interna



Figura 9.11 – Detalhe da ventoinha e suporte do motor

Testou-se outro motor com o dobro da potência:

$$P = 12 \cdot 5 \quad P = 60W$$

Onde a corrente passa de 2,5A para 5A.

Durante os testes, o aspirador de carro com maior potência não apresentou uma sucção muito forte, o que não era esperado. Assim, optou-se pela primeira opção, já que possui um consumo bem menor (a metade) e praticamente mesmo poder de sucção.

Para a utilização de tal aspirador, apenas a parte onde se encontra a câmara de sucção foi utilizada, descartando-se a outra parte do corpo do aspirador. Com isso, obteve-se um sistema de aspiração muito compacto, ideal para um sistema autônomo.

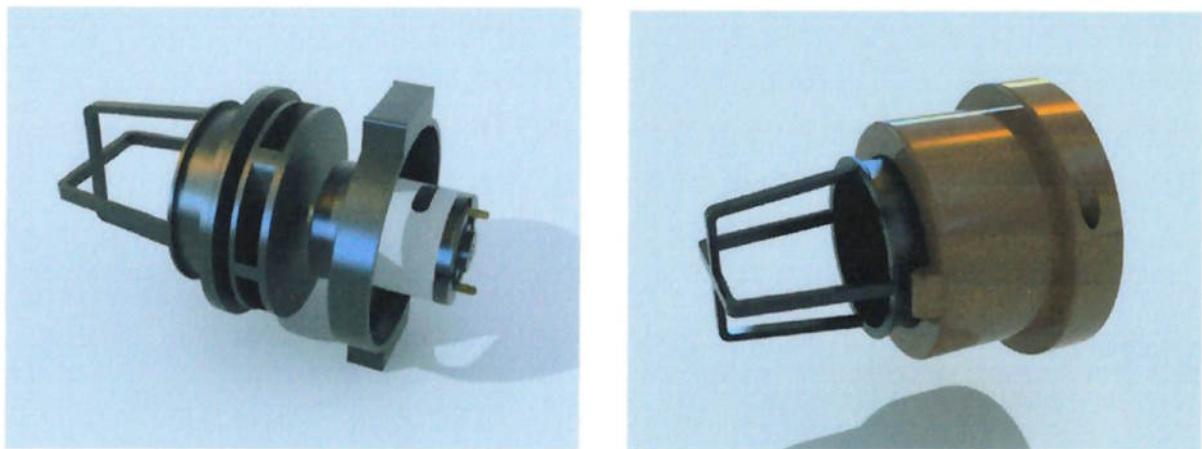


Figura 9.12 – Sistema de Aspiração. Primeira imagem: suporte de filtro, ventoinha, suporte de motor e motor.

Segunda imagem: suporte de filtro e câmara de sucção

(Desenho 3D)

Para a fixação do sistema de aspiração ao chassi do robô, projetou-se um suporte no formato de uma abraçadeira. O suporte possui uma base, que é fixada ao chassi através de parafusos, e uma abraçadeira, que se prende à base lateralmente, também através de parafusos.

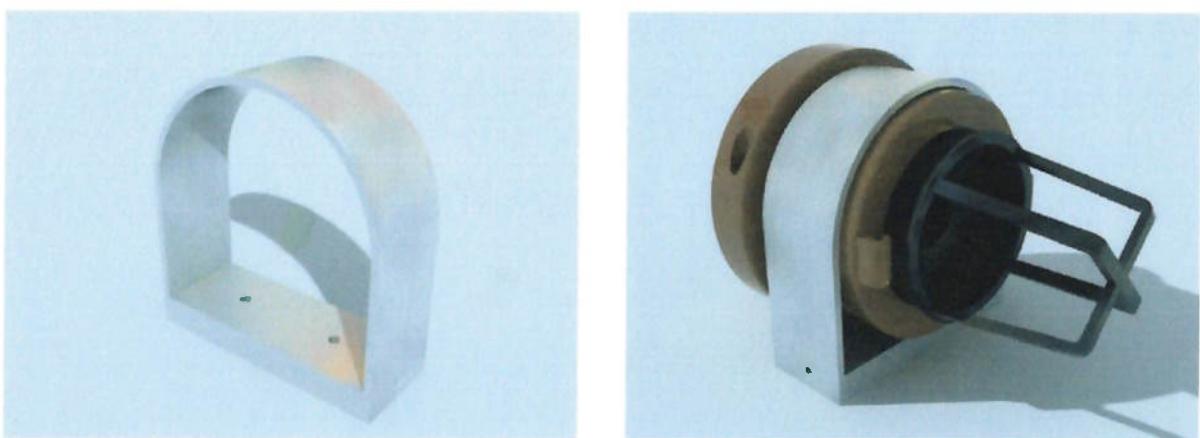


Figura 9.13 – Suporte de fixação do sistema de succão (Desenho 3D)

À frente do sistema de aspiração, engatado na parte onde se encontra o filtro, está posicionado o reservatório de sujeira. A vedação entre eles é feita através de um anel de

borracha. Apesar das limitações de espaço, conseguiu-se projetar um reservatório com capacidade para 1,5 litro de sujeira.

O reservatório foi construído com chapa de alumínio de 1mm de espessura, rebitado nas extremidades, e pode ser removido e lavado. Sua vedação foi feita utilizando-se uma vedação de silicone, encontrado facilmente em lojas de material de construção. A fixação se dá através de um pequeno parafuso, localizado na extremidade oposta ao sistema de aspiração, acima do suporte da tubulação. Por tratar-se apenas de um protótipo, tal sistema não se mostra muito prático, havendo a necessidade de se possuir uma chave de fenda para sua remoção. Mas para um projeto final, tal fixador poderia ser um fecho plástico de pressão. Por baixo do reservatório existem dois apoios de borracha que além de apoarem o reservatório para a posição correta servem de calços para que o reservatório não entre em contato com a roda de apoio que se encontra logo abaixo dele.

Na extremidade oposta ao sistema de aspiração, encontra-se o suporte da tubulação, que traz a sujeira do bocal até o reservatório. Como dito anteriormente, em tal suporte também está presente a fixação do reservatório. Entre o suporte e o reservatório existe um outro anel de borracha para vedação. O suporte é fixado ao chassi através de parafusos.

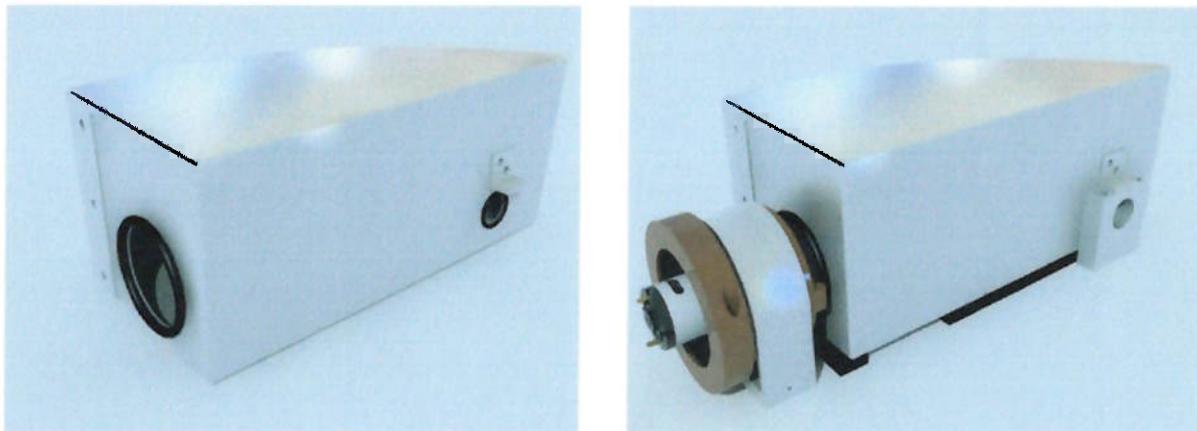


Figura 9.14 – Reservatório do sistema de sucção. Primeira imagem: Reservatório com anéis de vedação e fixador. Segunda imagem: reservatório com apoios, sistema de aspiração e suporte de tubulação. (Desenho 3D)

Segue-se então, a partir do suporte, uma mangueira com diâmetro interno de 18 mm até o bocal de aspiração. Nas extremidades dessa mangueira existem dois pequenos pedaços

de mangueira com menor diâmetro para encaixe das pontas ao suporte de um lado e ao bocal do outro.

O bocal de aspiração é constituído por duas partes que se encaixam, construído por chapas de alumínio de 0,5 mm de espessura, dobradas de forma a cobrir o rasgo presente no chassi e formar uma entrada na qual a mangueira é engatada. Percebe-se aqui que tal sistema possui uma perda de carga um pouco elevada, mas como já mencionado, o intuito do trabalho não é o de desenvolver um sistema de sucção, e sim um sistema autônomo funcional. O bocal é preso ao chassi por parafusos e vedado com silicone.

Dessa maneira, chegou-se ao seguinte sistema de sucção, formado pelo sistema de aspiração com o filtro, seu suporte, reservatório de sujeira, apoios, suporte da tubulação, dois anéis de borracha para vedação, tubulação e bocal.

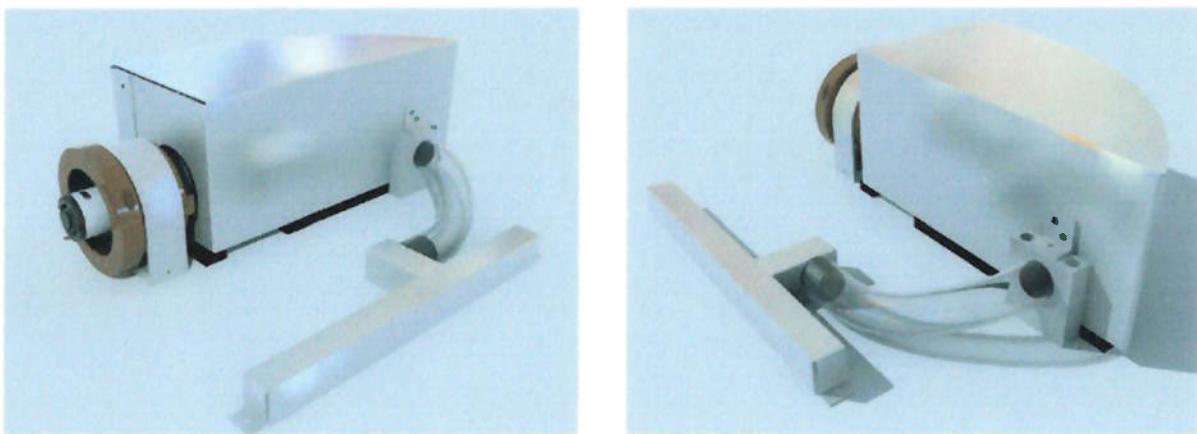


Figura 9.15 – Sistema de sucção completo (Desenho 3D)

Abaixo do bocal encontra-se o rasgo por onde a sujeira é aspirada. Tal rasgo possui dimensões que fazem com que sua área seja próxima à área do bocal do aspirador de carro original de onde o sistema de aspiração foi retirado. As dimensões do rasgo são 200mm x 3,5mm resultando em uma área de 700 mm<sup>2</sup>. A seção originalmente redonda com diâmetro de 30mm possuía uma área de 706 mm<sup>2</sup>. A seguir apresenta-se o detalhe do rasgo e o posicionamento do sistema de sucção no chassi.

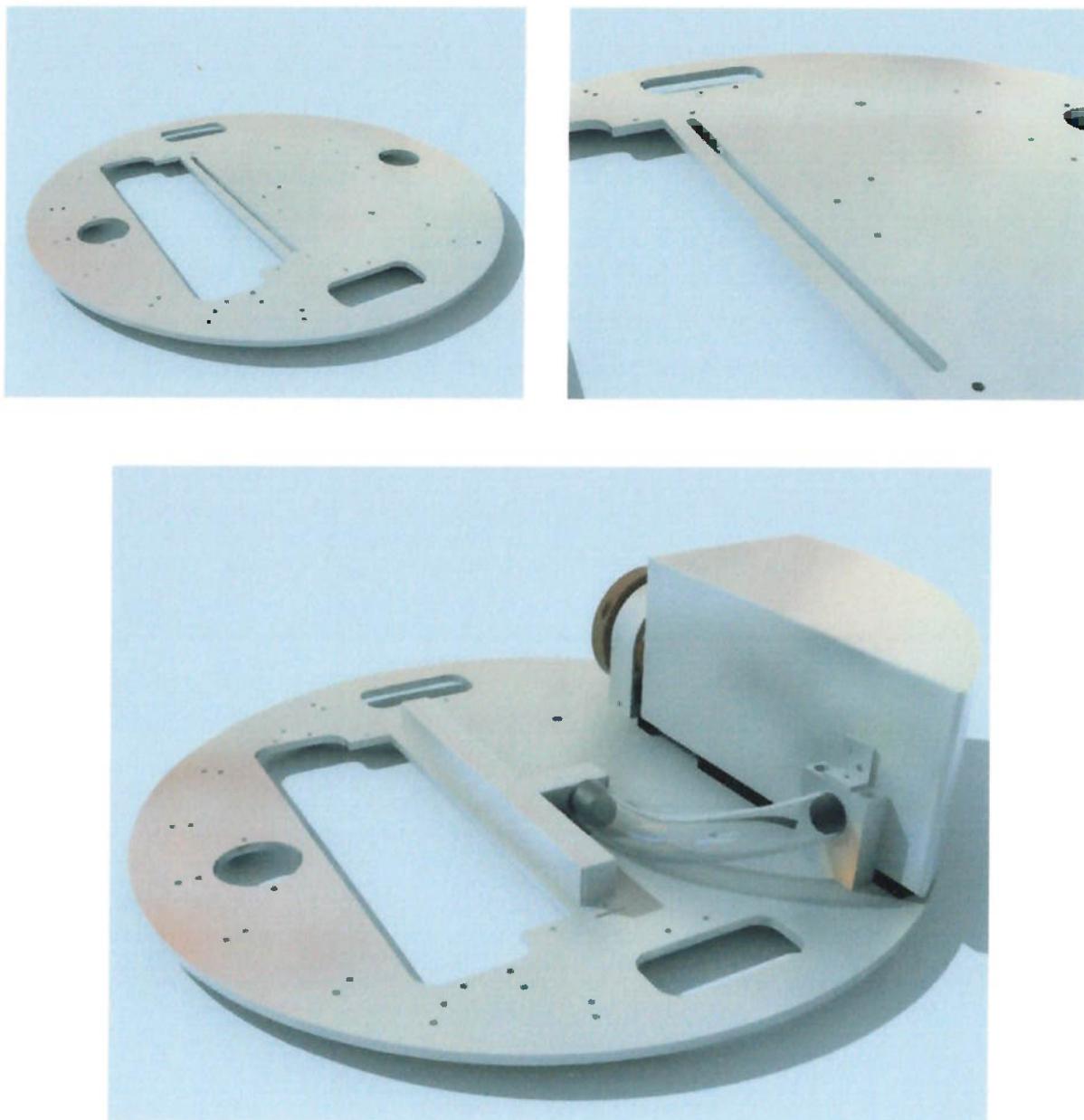


Figura 9.16 – Rasgo de aspiração e localização do sistema de sucção no chassi (Desenho 3D)

### 9.2.2 - Sistema de limpeza com vassoura giratória

Para o sistema de limpeza com vassoura giratória, adquiriu-se uma escova utilizada comercialmente em produtos conhecidos como varredoras de carpete (Feiticeira). A escova é vendida separadamente como produto de reposição, ao custo de R\$ 10,00. Nas varredoras convencionais, as escovas são acionadas por rodas de atrito conforme o usuário movimenta a varredoura pelo piso.

A escova selecionada está apresentada a seguir:



Figura 9.17 – Escova para o sistema de vassoura giratória.

A fixação da escova foi desenvolvida de maneira que o usuário possa retirá-la com facilidade, já que este seria um item de possível troca pelo usuário. Com o tempo a escova começa naturalmente a perder os pelos e a se desgastar, sendo necessária sua troca. Pensando nisso os mancais de sustentação da escova foram projetados da seguinte maneira:

Em um dos lados tem-se um pequeno eixo fixo (já que a escova possui furos em suas extremidades). Do outro lado colocou-se uma pequena esfera móvel, dentro de um encapsulamento com mola. Tal encapsulamento é normalmente utilizado como fecho de pressão em armários e espelhos de banheiro, sendo facilmente encontrado em lojas especializadas.

Dessa maneira, o usuário pode retirar a escova puxando-a pelo lado em que a esfera a sustenta. Para recolocar a escova, o usuário deve encaixá-la no eixo fixo e apertar a outra ponta contra a esfera até que ela se encaixe. Tal operação é feita pela parte de baixo do robô,

já que no produto final o robô seria todo coberto pela carcaça. Os mancais ficaram concebidos da seguinte maneira:

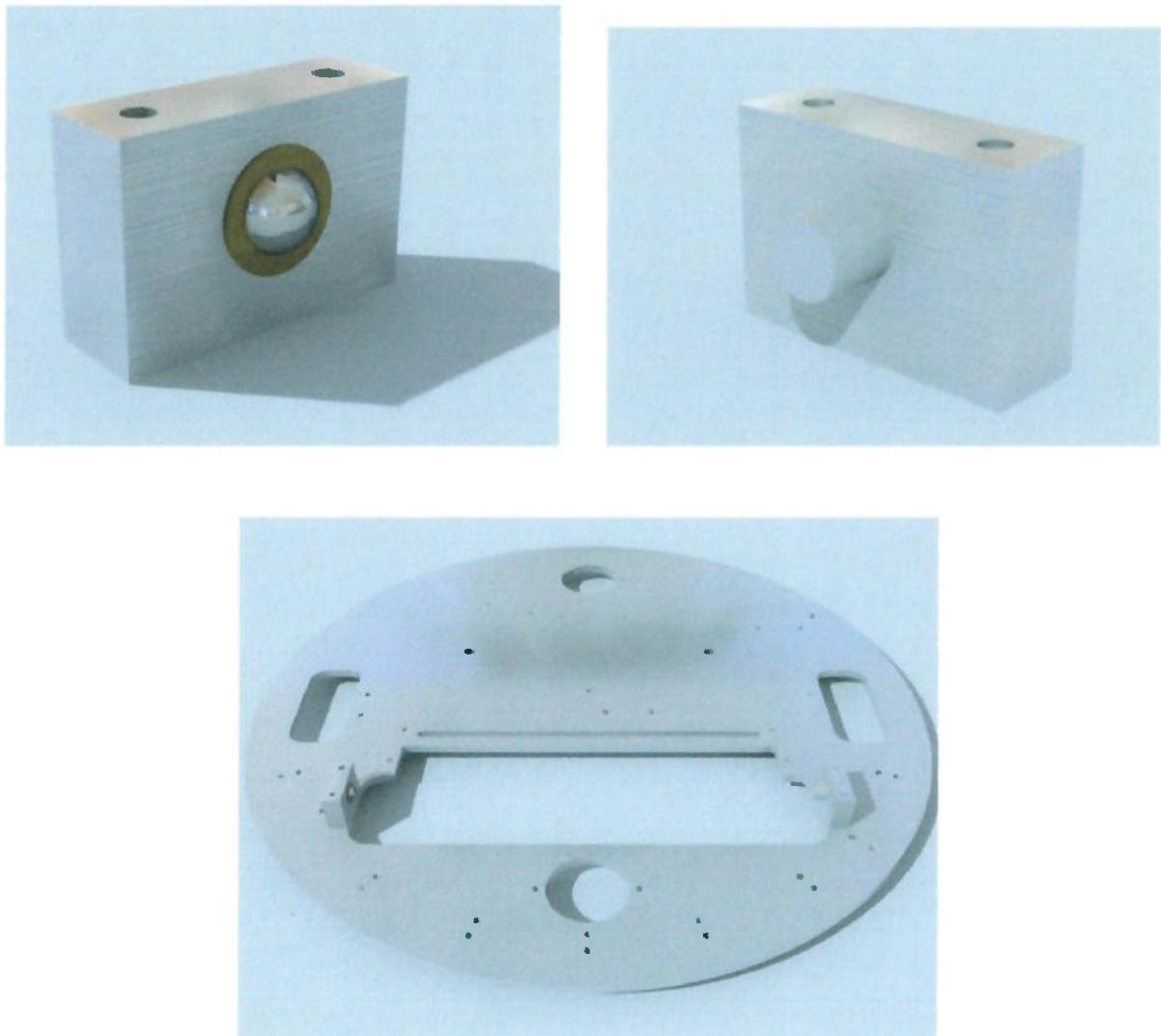


Figura 9.18 – Mancais de sustentação da vassoura giratória. Primeira imagem: Mancal com esfera móvel. Segunda imagem: mancal com eixo fixo. Terceira imagem: localização dos mancais no chassi. (Desenho 3D)

Será utilizado, para a movimentação da escova, um pequeno motor elétrico, transmitindo a rotação através de engrenagens. Tal motor, seu suporte e engrenagens provêm de um sistema de movimentação de um vídeo cassete. Tal escolha se deu pelo tamanho reduzido do motor e da redução presente nas engrenagens. Além disso, o consumo do motor, com uma tensão aplicada de 12V, é de apenas 50mA com torque máximo, o que nos traz uma excelente opção devido à baixa potência consumida.

O suporte deste motor já vem com três furos para fixação, o que facilita tal tarefa. O sistema de engrenagens do motor faz com que seu posicionamento seja vertical. Assim, para sua fixação ao chassi, projetou-se um suporte em “L” em chapa de alumínio com 3 mm de espessura, fixada por parafusos. Uma solução simples, de fácil construção, rígida e barata. Para completar é necessário acoplar uma engrenagem à escova, para que esta possa se acoplar ao sistema de movimentação. Tal engrenagem já acompanharia a escova quando o usuário fosse trocá-la.

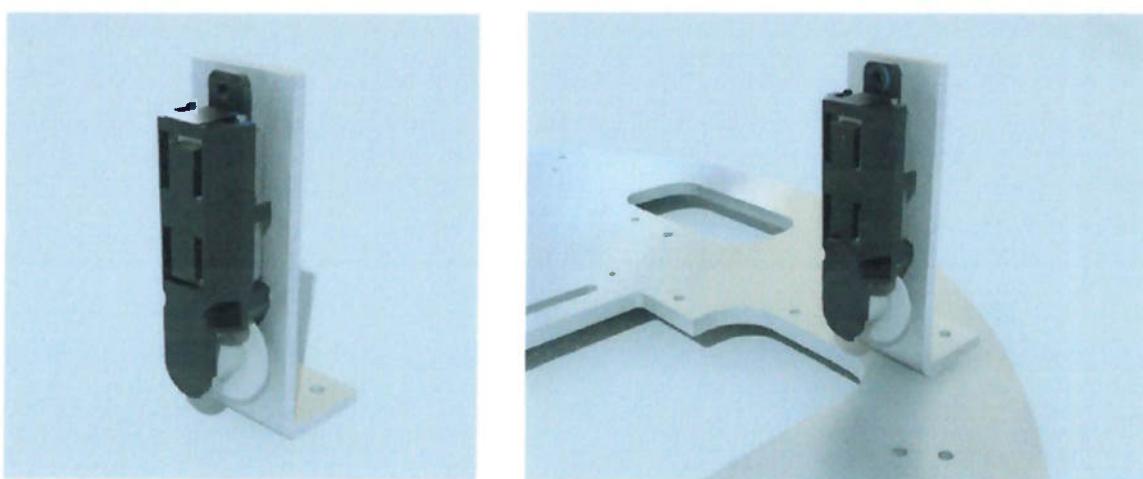


Figura 9.19 – Motor elétrico para movimentação da vassoura giratória. Motor, suporte plástico, engrenagens e suporte para fixação ao chassi. (Desenho 3D)

A sujeira coletada pela vassoura será armazenada em outro compartimento, separado do sistema de aspiração, devido ao seu posicionamento dentro do robô. Tal compartimento feito de chapa de alumínio com 0,5 mm de espessura pode ser retirado com facilidade pela parte de baixo do robô, assim como a escova, e lavado. Assim como o reservatório do sistema de aspiração a fixação é feita por parafusos, o que poderia ser substituído por fechos de pressão em plástico no produto final, para facilitar a retirada do mesmo.

A vassoura giratória é coberta por uma capa feita em chapa de alumínio de 1 mm de espessura que possui duas funções. Uma é evitar que a sujeira coleta pela escova entre no robô, garantindo que vá apenas para o reservatório específico. A outra função é a de dar sustentação para as placas de circuito que são acomodadas logo acima da vassoura. A capa é fixada ao chassi através de dois parafusos.

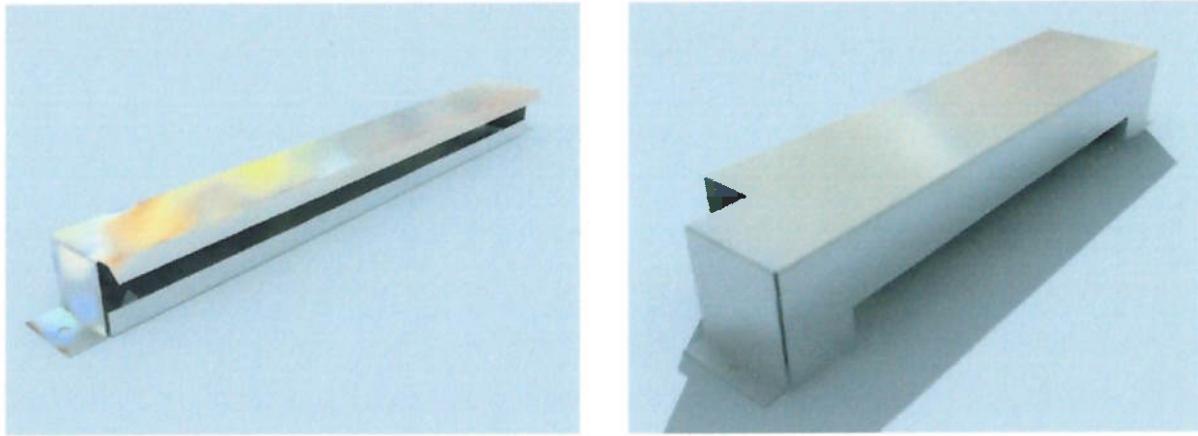


Figura 9.20 – Primeira imagem: reservatório dos sistemas de limpeza com vassoura giratória.

Segunda imagem: capa da vassoura giratória (Desenho 3D)

Com isso projetou-se o sistema de limpeza com vassoura giratória. O sistema completo é mostrado a seguir:

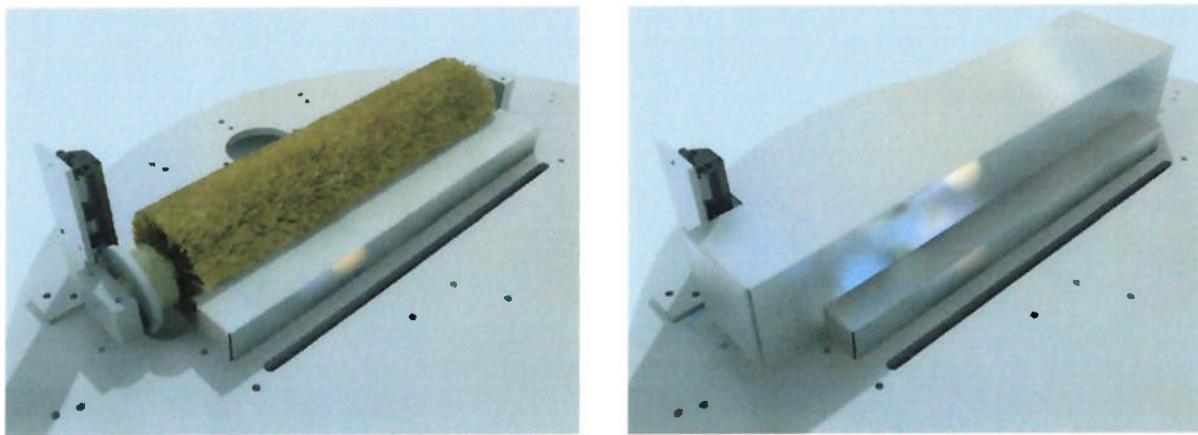


Figura 9.21 – Sistema de limpeza com vassoura giratória completo (Desenho 3D)

Finalizando, o sistema de limpeza do robô, composto pelo sistema de sucção e pelo sistema de limpeza com vassoura giratória, tomou a seguinte concepção:

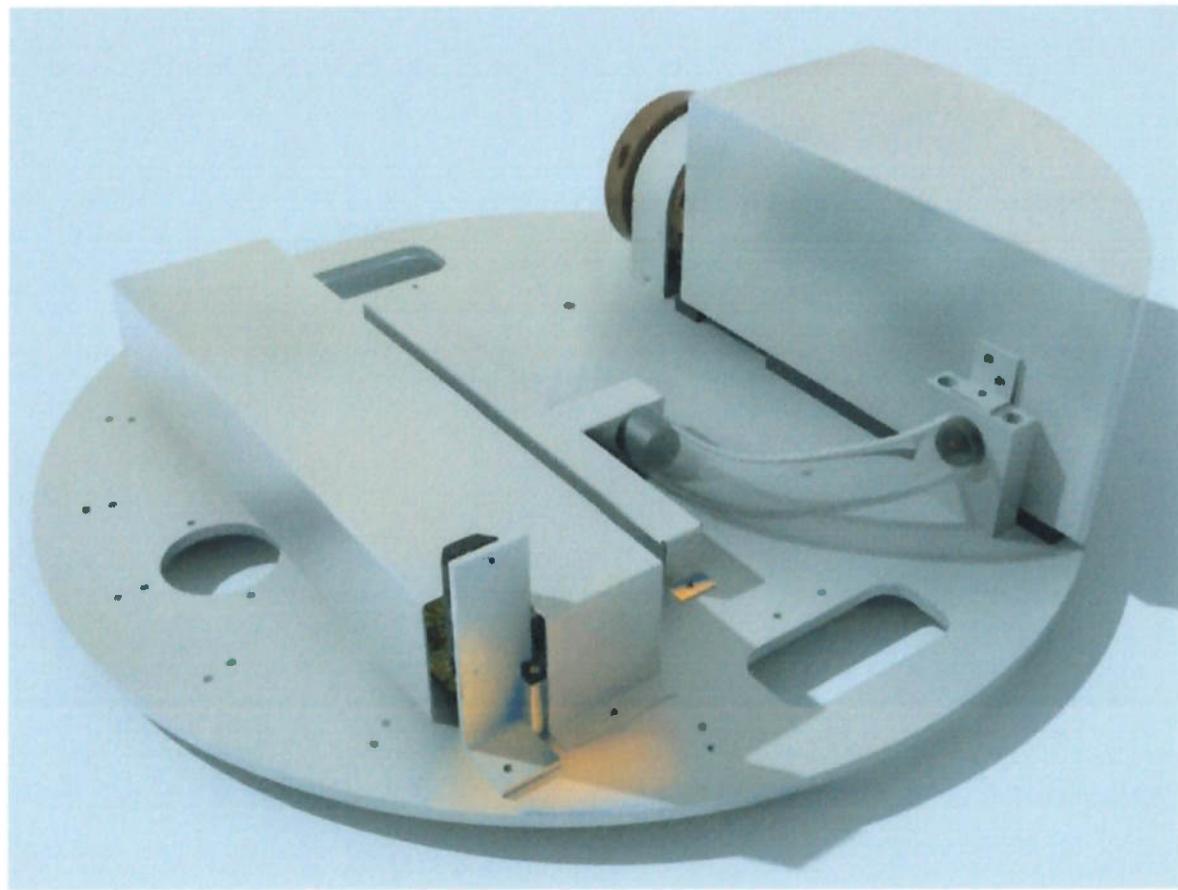


Figura 9.22 – Sistema de limpeza completo do robô (Desenho 3D)

O acionamento do sistema de limpeza será realizado por dois sinais do microcontrolador. Para tanto, será utilizado um relé de comando para ligar ou desligar o motor do aspirador de pó, e apenas um transistor para ligar e desligar o motor da vassoura giratória, já que sua baixa corrente não requer o uso de um relê. O relê de acionamento do aspirador é acionado por um transistor que é protegido por um diodo shotik contra picos de tensão.

### 9.3 - Sensores

O projeto mecânico dos sensores, apesar de já ter sido apresentado, será aqui detalhado.

Como já mencionado, o robô conta com uma série de sete sensores dispostos em sua frente. Cada sensor conta com dois contatos mecânicos e um sensor óptico. O sensor óptico fica disposto no centro dos dois sensores mecânicos. Como esta seção trata apenas do projeto mecânico apresentam-se apenas os sensores mecânicos.

Tais sensores foram projetados de forma que seu tamanho seja igual para todos os sensores para quem olha o robô de frente. Ou seja, as projeções dos sensores têm o mesmo tamanho. Esse fator é importante pois, como o robô se movimenta normalmente para frente, ao encontrar uma barreira, é interessante que se saiba qual parte do robô encostou em tal barreira. Tornando as projeções dos sensores de mesmo tamanho, tem-se uma maior facilidade em decidir qual o desvio deverá ser feito.

A construção do sensor mecânico teve como grande preocupação a rigidez do suporte de apoio dos sensores, já que estes serão os responsáveis por receber a força imposta pela barreira ao ser tocada. A solução encontrada foi a utilização de uma chapa de alumínio de 3mm de espessura dobrada, de forma que duas dobras sejam responsáveis pela sustentação dos contatos e uma terceira dobra central suporte a mola e o sensor óptico. Outra dobra inferior com dois furos fixa o suporte ao chassi.

Os contatos foram feitos com chapas de cobre de 2mm de espessura, conformados de forma que acompanhe a circunferência externa do robô.

A mola utilizada foi conformada com uma chapa de bronze de 0,3mm de espessura, já que não se encontrou um aço mola que pudesse exercer tal função.

Os botões em si foram fabricados a partir de um tarugo de alumínio e, assim como os contatos de cobre, conformados de acordo com a circunferência externa do robô. A esses botões parafusou-se dois parafusos, um cada extremidade, sendo eles os responsáveis pelo contato. Tal artifício foi utilizado pois assim não há necessidade de todo o botão fazer o contato com o cobre.

Os botões foram fixados à mola com uma cola epóxi, assim como os contatos de cobre ao suporte de alumínio. A mola foi fixada ao suporte através de uma abraçadeira e isolada do mesmo através de uma fita isolante colada entre eles.

Dessa maneira, chegou-se a seguinte configuração dos sensores:

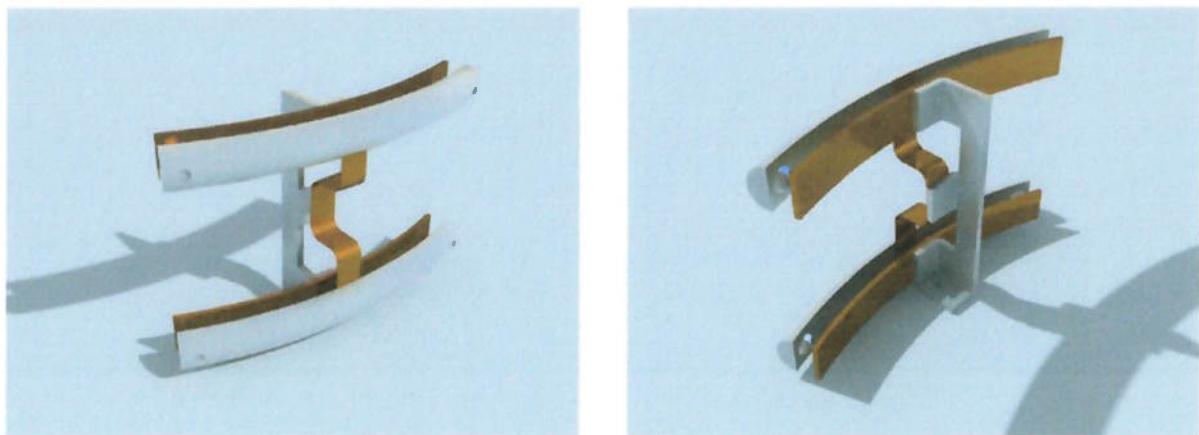


Figura 9.23 – Sensor mecânico (Desenho 3D)

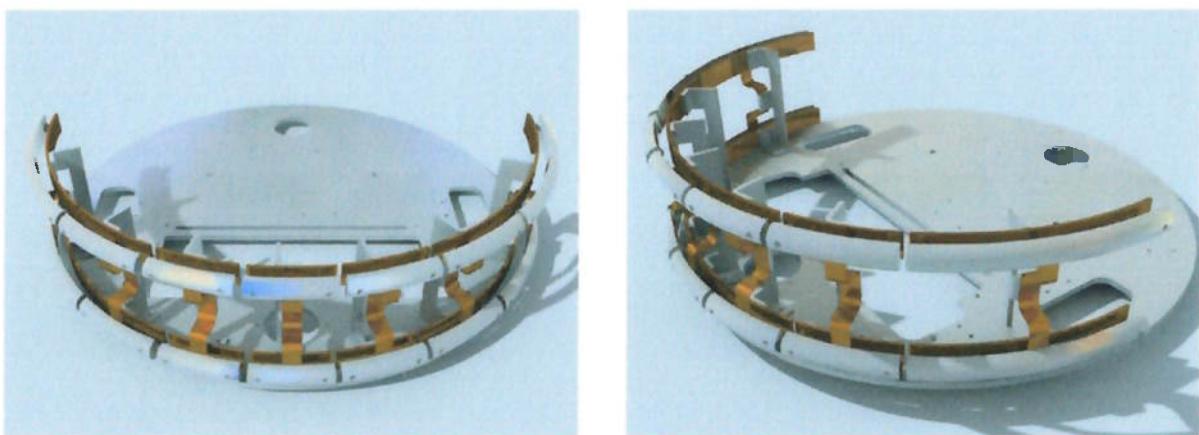


Figura 9.24 – Sensores mecânicos acoplados ao chassi (Desenho 3D)

## 9.4 - Baterias

As três baterias presentes no robô são as maiores responsáveis pelo seu volume do robô e, principalmente, pelo seu peso. A bateria de 12V e 9AH tem dimensões de 151mm x

65 mm x 95 mm e massa igual a 2,66kg. As outras duas baterias de 12V e 2,3AH têm dimensões de 180mm x 60mm x 35 mm e massa igual a 1,0kg.

Como já mencionado anteriormente, deve-se sempre procurar deixar o robô apoiado em três pontos, sendo dois desses pontos as rodas motoras. Em vista disso, a distribuição das baterias no chassi, pelo fato de sua elevada massa, deve ser feita da forma mais simétrica possível. Além disso, como o robô possui a capacidade de detecção de escadas, é interessante que o terceiro ponto seja a roda de apoio traseira, para que a roda frontal possa, em algum momento, ser retirada do piso. Assim, é conveniente que as baterias fiquem para trás do eixo central do robô, deslocando seu centro de massa com elas, e favorecendo o apoio traseiro.

A fixação da bateria maior se dá através de uma chapa de alumínio de 2mm de espessura dobrada e fixada ao chassi através de dois parafusos. Além disso, foram fixados, por pressão, dois pinos que limitam o movimento da bateria lateralmente.

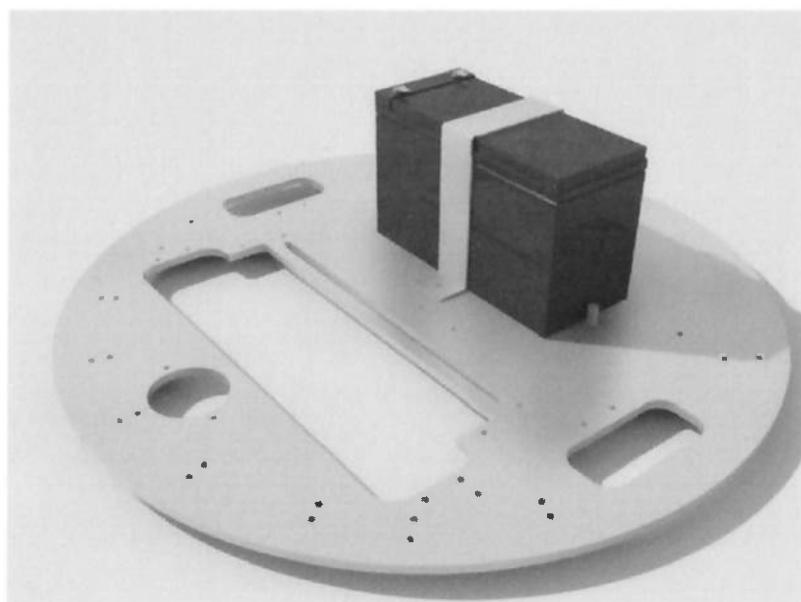


Figura 9.25 – Fixação da bateria maior ao chassi (Desenho 3D)

As baterias menores são fixadas com abraçadeiras plásticas, e ficam dispostas acima dos motores de locomoção.

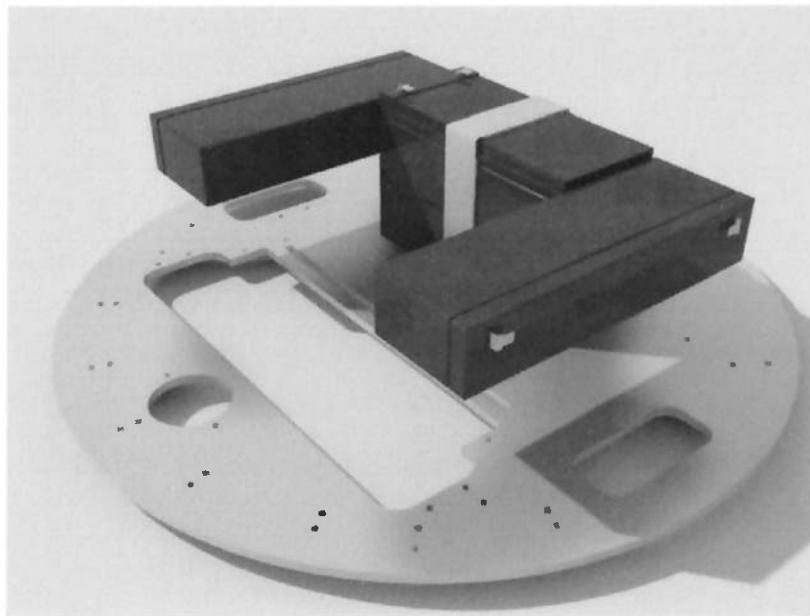


Figura 9.26 – Posicionamento das baterias menores no robô (Desenho 3D)

## 9.5 - Projeto mecânico final

Todas as peças do Robô Aspirador de pó foram desenhadas em CAD, utilizando-se o software Microstation V8. Todas as peças foram modeladas tridimensionalmente já que, assim, pôde-se obter o maior auxílio possível que o software poderia fornecer.

O projeto via computador possibilitou uma melhor visualização de todas as partes do robô, seus sistemas completos, com a possibilidade de se detalhar cada peça, observar a interação entre peças de um mesmo sistema e a interação entre os diversos sistemas do robô. Foram modeladas, não só as peças a serem construídas, mas também as já fabricadas, como motores, rodas, suportes de plástico e outros acessórios. Isso permitiu eliminar a possibilidade de incoerências dimensionais, tais como interpenetração de peças (de difícil visualização apenas com desenhos 2D), e de superdimensionamento ou subdimensionamento.

A modelagem 3D foi baseada em uma precisão dimensional da ordem de milésimos de milímetro, o que garante a fidelidade dos desenhos em relação à construção do protótipo. Assim, teve-se como objetivo, atingir um nível de detalhamento técnico que pudesse minimizar ao máximo o trabalho com reprojeto e ajustes não previstos.

A partir disso, pôde-se gerar todos os desenhos de fabricação das peças, já que todas as dimensões foram definidas nessa fase. Nesse processo de geração das vistas 2D, o software Microstation não se mostrou muito intuitivo. Devido a isso, utilizou-se o software CorelDraw 11, software este com alta capacidade de geração de imagens 2D com altíssima precisão e facilidade de cotagem. Assim, baseando-se nos modelos 3D, gerou-se os desenhos 2D, presentes nos anexos. (Ver Anexo K – Desenhos Técnicos).

Outro software utilizado para facilitar o projeto do robô foi o software de visualização tridimensional 3D Studio MAX 5. Tal software ajudou sobremaneira durante essa fase pois tem grande capacidade de geração de imagens realistas, provendo imagens de conjunto que auxiliaram na tomada de decisões, principalmente, durante a interação dos diversos sistemas do robô.

Percebeu-se aqui, que uma das maiores dificuldades durante o projeto por computador é a visualização completa de todas as partes e sistemas que, ao se unirem, acabam por formar um conjunto que deve ser otimizado ao máximo em dois principais aspectos: a relação entre o volume ocupado e o volume vazio entre peças, e a “harmonia” entre os diversos sistemas. Essa “harmonia” pode ser entendida como a contribuição que cada sistemas dá ao conjunto fazendo com que ele se torne rígido, simples, completo, com a maior facilidade construtiva e de montagem possível. Cada peça isolada não tem a capacidade de fornecer todas essas características, mas o conjunto pode ser feito para que isso ocorra. Em vista disso, todas as facilidades que os softwares de visualização 3D podem fornecer foram utilizadas para melhorar o projeto antes de se iniciar a construção.

Em vista disso, pela facilidade de visualização, optou-se por não realizar desenhos de conjunto 2D do projeto, já que os desenhos 3D proporcionam um melhor entendimento em relação à integração dos sistemas e montagem das peças.

Obteve-se, então, em substituição aos convencionais desenhos de conjunto, uma série de desenhos 3D que demonstram a correta seqüência de montagem dos componentes mecânicos do aspirador robô (Ver Anexo I – Seqüência de montagem)

Toda essa preocupação inicial fez com que o tempo gasto no projeto fosse maior do que o esperado. Mas todo esse tempo gasto reverteu-se em tempo ganho durante a construção do protótipo. As peças projetadas praticamente não tiveram trabalho de reprojeto, a não ser por imprecisões de fabricação, que obviamente não se têm o controle como no projeto por computador.

## 10 Projeto de Software

O programa operacional do robô foi desenvolvido em linguagem de programação C a partir do software Dynamic C, também utilizado para compilar o programa final, *linkar* e carregar o programa no microcontrolador através de porta serial e *debugar* o programa durante sua edição.

O programa final comentado carregado no microcontrolador RabbitCore RCM2000 do robô em linguagem C está no Anexo J – Programa em C.

### 10.1 - Análise do software

O programa implementa uma rotina de varredura da superfície a ser coberta pelo robô com algumas simplificações, que prioriza o reconhecimento do perímetro do ambiente a ser limpo.

Entre algumas das simplificações adotadas, neste primeiro momento, estão a utilização de um ambiente retangular e o posicionamento inicial do robô de modo a ter seu lado direito voltado para uma das paredes para que este percorra o perímetro no sentido anti-horário.

A princípio o software espera o acionamento do botão de *setup*, que pára os motores de locomoção após a chave geral ser ligada, quando estes estão ainda sem velocidade e direção definidos.

Em seguida o robô está em *stand by* à espera do acionamento do botão de início que, quando pressionado, coloca o robô num estado de varredura da superfície, com a ressalva de um tempo de 2 segundo entre o acionamento de tal botão e o início de seu movimento para que o usuário se afaste de seu possível trajeto.

Em seguida o robô aciona os sistemas de aspiração e da vassoura giratória e se move na trajetória de sua direção de ataque (voltado para frente) até que um de seus sensores detecte um obstáculo, reportando ao software qual foi o sensor ativado e realizando um recuo

de 30mm para que seus sensores mecânicos sejam liberados de um possível contato com a barreira física detectada.

A partir da informação de qual foi o sensor ativado, o robô pode identificar qual foi sua região de deslocamento interceptada pelo obstáculo, realizando uma rotação específica para cada sensor de modo a liberar sua trajetória de ataque unicamente da barreira imposta, sem perder superfícies de varredura alcançáveis.

Dessa maneira o robô percorre, preferencialmente no sentido anti-horário o perímetro do ambiente de trabalho, guardando em coordenadas cartesianas sua posição inicial definida como (0,0), em (x,y) e sua posição atual, juntamente com seu ângulo atual de rotação em relação ao original.

Quando o robô retorna à sua posição inicial o programa é interrompido e aguarda um reinício dado pelo botão de *reset*.

A cada detecção de barreiras físicas em sua trajetória de deslocamento, o robô pára seu movimento por aproximadamente 1 segundo e aguarda o comando de interrupção de sua operação por parte do usuário. Isso se dá devido à aplicação ainda experimental do software, de modo a evitar um comportamento indesejado do robô. Caso esse comando não venha dentro deste limite de tempo estabelecido, o robô continua seu deslocamento normalmente. Esse princípio tem como objetivo meramente proteger o robô de choques violentos que possam danificar seus componentes, dando ao usuário, ainda tratado como desenvolvedor do produto, a possibilidade de forçar o desligamento do sistema.

Após a finalização desta rotina, o aspirador é desativado juntamente com a vassoura giratória, enquanto os motores permanecem com tensão, porém travados com velocidade nula. Os leds piscam alternadamente indicando fim do processo e possível reinicialização por parte do usuário.

## DISCUSSÕES

Por atrasos no cronograma inicialmente sugerido não foi possível implementar alguns dos subsistemas que foram objetivados na proposição do projeto.

O robô não monitora o nível de carga das baterias, impossibilitando a detecção da necessidade de uma recarga durante seu funcionamento. O circuito de nível das baterias foi proposto e testado, porém não foi consolidado no protótipo. Assim, também foi descartada utilização de uma base fixa plugada à rede elétrica na qual o robô se conectararia para recarregar suas baterias de chumbo-ácido.

Durante a fase de testes dos sistemas de hardware, uma porta inteira de I/O do microcontrolador foi queimada, possivelmente devido a um curto elétrico, porém não observado. Assim como esta porta, que era extremamente importante devido à sua flexibilidade de implementação tanto de entradas quanto de saídas em seus bits, mais 2 bits de outras portas foram perdidos.

Sem tempo hábil para substituição do microcontrolador por outro semelhante ou, ainda, troca por um microcontrolador alternativo, que exigiria adaptações de hardware, foram realocados alguns dos pinos nas portas de modo a permanecer com o mesmo produto. Dessa forma, não havia pinos suficientes para todas as entradas e saídas que desejava-se utilizar, fazendo com que fossem priorizados alguns dos sistemas de comunicação e interface. Dos sete pares de sensores mecânicos/ópticos instalados e disponíveis no robô, apenas cinco foram de fato utilizados, descartando-se os dois imediatamente ao lado do sensor frontal (sensor 4).

Este procedimento não afetou de forma grave o funcionamento do protótipo em ambientes pouco acidentados e em acordo com as simplificações propostas ao seu funcionamento, porém em ambientes de trabalho com uma grande quantidade de obstáculos ou paredes dispostas em ângulos não retos, sua operação fica prejudicada e há risco de choques indesejáveis.

Os sensores ópticos que seriam direcionados para baixo de modo a detectar uma escada e, assim, evitar a queda do robô, foram removidos do protótipo devido à escassez de pinos de entrada de sinais.

Ainda em relação ao possível curto-circuito que teria queimado parte das portas de comunicação com periféricos e, por receio de que tal fato voltasse a ocorrer, os sensores mecânicos que, a princípio, têm papel de proteção contra uma não detecção de um obstáculo pelos sensores ópticos tiveram que ser desabilitados. Isso ocorreu porque tinham seus pólos negativos (terra) interconectados e ligados ao chassi do robô por construção mecânica, pois o chassi é de alumínio e os botões são de alumínio, cobre e bronze. Temeu-se, então, um risco maior de curto elétrico com o terra distribuído e demasiadamente exposto por todo o robô. Essa alternativa possibilitou que as tensões elétricas ficassem restritas às placas.

A placa de encoder possui um sistema de detecção de quadratura simplificado, porém desativado devido à falta de componentes que, a princípio, foram trazidos dos Estados Unidos. Isso em nada prejudicou o funcionamento tanto da placa quanto dos sistemas que utilizavam os sinais nela gerados. Essa adaptação foi introduzida apenas interconectando-se quatro terminais da placa dois a dois, de modo que o retorno à disposição original da placa pode ser facilmente alcançado.

Observando o microcontrolador utilizado ao longo dos testes de hardware e software, chegou-se a conclusão de que este foi superdimensionado.

Dado que não havia como prever exatamente as necessidades de memória, processamento e comunicabilidade de portas e interface, foi escolhido um microcontrolador com alta performance e o maior número de portas e bits de I/O.

Por se tratar de um protótipo, tal fato não tem grande relevância pois, em possíveis reprojeto e produção futura, este microcontrolador poderia ser facilmente substituído por outro da família 8051 ou similar, mais acessível e de fácil manipulação.

Não foram feitos carcaça ou qualquer outro tipo de cobertura estética ou ainda, proteção mecânica, para facilitar o acesso aos sistemas mecânicos e eletrônicos que exigiam constante manutenção e verificação de problemas.

O programa implementado no microcontrolador e responsável pela operação do robô está simplificado e atende apenas a requisitos mínimos de operação. Este fato é, também, consequência do atraso em relação ao cronograma inicial e, principalmente, ao mau funcionamento (possível queima por curto-circuito) de portas e pinos de I/O.

Todavia, os projetos mecânico e eletrônico foram bem executados, com destaques para a excelente utilização do espaço físico disponível no robô e sua otimização para a aplicação desejada, ao ótimo resultado conseguido no projeto e fabricação das peças em alumínio e à flexibilidade para futura programação aprofundada da operação do robô, que pode ser ainda bastante explorada.

## CONCLUSÕES

A autonomia e eficiência do robô de pequeno porte ficam limitadas ao atendimento das simplificações propostas e considerações feitas em relação às superfícies a serem limpas. O uso de baterias de chumbo-ácido também limita a eficiência da aspiração devido à pequena corrente elétrica fornecida e sua baixa relação potência/peso.

O uso de diversas placas eletrônicas próximas e diferentes tensões de alimentação pode causar interferência nos circuitos e necessitaria um melhor isolamento e blindagem dos sistemas. Ainda em relação às tensões utilizadas, 5V, 12V e 24V, há um excesso de transformações e dissipação de energia. Uma alternativa seria o uso apenas de 5V e 24V, o primeiro para alimentar os componentes eletrônicos e o segundo para os acionamentos dos motores de locomoção, aspiração e vassoura giratória.

O microcontrolador utilizado está superdimensionado e seria, possivelmente, substituído por um microntrolador da família 8051. Seu uso se justifica, nesse momento, pelo desconhecimento inicial da capacidade de processamento e memória, além de outros recursos físicos que o projeto requeria.

Para garantir o uso do protótipo em qualquer superfície, livre de simplificações de projeto, a realimentação da posição angular dos eixos dos motores ao longo do tempo deveria ser utilizada também para corrigir desvios de percurso em relação ao trajeto ideal. Até o momento, essa realimentação tem por finalidade definir e acionar os motores objetivando o trajeto ideal, porém, não corrige em tempo real possíveis desvios devido a irregularidades nas superfícies de trabalho.

## REFERÊNCIAS

TEXAS INSTRUMENTS. Apresenta descrições de componentes eletrônicos.  
Diponível em: <<http://www.ti.com>>. Acesso em: 05/11/2004.

FAIRCHILD SEMICONDUCTOR. Apresenta descrições de componentes eletrônicos.  
Diponível em: <<http://www.fairchildsemiconductor.com>>. Acesso em: 05/11/2004.

NATIONAL SEMICONDUCTOR. Apresenta descrições de componentes eletrônicos.  
Diponível em: <<http://www.national.com>>. Acesso em: 05/11/2004.

MICROCHIP. Apresenta descrições de componentes eletrônicos.  
Diponível em: <<http://www.microchip.com>>. Acesso em: 05/11/2004.

DISCOVER CIRCUITS. Apresenta descrições de circuitos eletrônicos.  
Diponível em: <<http://www.discovercircuits.com>>. Acesso em: 10/10/2004.

CXI SCHEMATICS. Apresenta descrições de circuitos eletrônicos.  
Diponível em: <<http://www.mitedu.freeserve.co.uk/schematics.htm>>. Acesso em: 08/10/2004.

ROBOT ROOM. Apresenta descrições de circuitos eletrônicos – Ponte H.  
Diponível em: <<http://www.robotroom.com/HBridge.html>>. Acesso em: 16/06/2004.

ELECTRONIC CIRCUITS. Apresenta descrições de circuitos eletrônicos.  
Diponível em: <<http://www.ee.oulu.fi/~arik/schematics.html>>. Acesso em: 19/09/2004.

ELECTRONIC CIRCUITS FOR THE ROBBYIST. Apresenta descrições de circuitos eletrônicos. Diponível em: <<http://www.uoguelph.ca/~antoon/circ/circuits.htm>>. Acesso em: 20/08/2004.

ELECTRONICS ZONE. Apresenta descrições de circuitos eletrônicos. Diponível em: <[http://www.electronic-circuits-diagrams.com/electronic\\_circuits\\_index.shtml](http://www.electronic-circuits-diagrams.com/electronic_circuits_index.shtml)>. Acesso em: 14/07/2004.

ROBOTICS PROJECTS. Apresenta descrições de circuitos eletrônicos. Diponível em: <<http://users.frii.com/dlc/robotics/projects/botproj.htm>>. Acesso em: 12/11/2004.

ELETRONICA.ORG. Apresenta descrições de circuitos eletrônicos - PWM.  
Diponível em: <<http://www.eletronica.org/modules.php?name=News&file=article&sid=59>>. Acesso em: 16/05/2004.

ELETRONICA.ORG. Apresenta descrições de circuitos eletrônicos – Motores elétricos.  
Diponível em: <<http://www.eletronica.org/modules.php?name=News&file=article&sid=15>>. Acesso em: 16/05/2004.

FACULDADE DE ENGENHARIA ELÉTRICA E COMPUTAÇÃO - UNICAMP. Apresenta descrições de circuitos eletrônicos – PWM.  
Diponível em: <<http://www.demic.fee.unicamp.br/~elnatan/ee832/exp/exp5-2002.pdf>>. Acesso em: 20/05/2004.

KIT MICROCONTROLADOR 8051. Apresenta descrições do microcontrolador 8051.  
Diponível em: <<http://www.kit80c31.hpg.ig.com.br/#1>>. Acesso em: 03/07/2004.

K-TEAM. Apresenta descrições de softwares de controle de robôs.  
Diponível em: <<http://www.k-team.com/software/webots.html>>. Acesso em: 05/08/2004.

MICROCONTROLADOR.COM.BR. Apresenta descrições do microcontrolador 8051.  
Diponível em: <<http://www.microcontrolador.com.br/verprodutos.php?prod=8051&id=3>>. Acesso em: 04/07/2004.

## **Anexo A - Concepção do Robô Aspirador de Pó**

**Anexo A -Concepção do Robô Aspirador de Pó**

## **Anexo B - Opções de motores encontrados**

## Anexo B - Opções de motores encontrados

### 1 - INLINE 90 VDC GEARMOTOR

New DAYTON. Inline permanent magnet DC gearmotors.  
Ideal for variable speed applications.



#### SPECIFICATIONS

13 RPM  
90 volt DC  
1 amp  
1/12 hp  
125 in-lbs torque  
128:1 ratio  
Reversible  
Cont duty  
Shaft 5/8" x 1 1/2" w/keyway  
Foot/Face mounts  
Size 8 1/2" x 4 3/4" x 4 3/4"  
Shpg 11 lbs

### 2 - 18 RPM, 24 VDC

New BUEHLER PRODUCTS.



#### SPECIFICATIONS

RPM 18  
Voltage 24 DC  
Amps 36 mA no load  
Torque 3 in. lbs.  
Rotation Reversible  
Duty intermittent  
Shaft 0.2" diam. x 3/4" w/flat  
Mount 1 bolt in face  
Size 1 7/16" diam. x 2 3/4"  
Shpg. 3 lbs.

### 3 - 24 VDC 20 RPM GEARMOTOR

New AUTOTROL CORP. gearmotor.  
Microswitch with spade terminals mounted on motor case.  
Opens during roughly 1/4 of motors rotation.



#### SPECIFICATIONS

20 RPM  
24 VDC  
0.3 Amps  
7 Watts  
12 in-lb torque  
Reversible  
Intermittent duty  
5/16" dia. x 7/8" long plastic shaft  
4 mounting holes thru case on 2-1/8"  
x 2-15/16" centers  
Size 2-1/8" x 2-1/2" x 3-1/2"  
Shpg 0.75 lb

**4 - 24 RPM 24 VDC INLINE GEARMOTOR**

New ITT AUTOMOTIVE model 402.757  
inline gearmotor.

**SPECIFICATIONS**

24 RPM  
24 VDC  
50 mA @ no load  
30 in-oz torque (calculated)  
Reversible rotation  
Totally enclosed  
Continuous duty  
6 mm (0.235") x 18 mm (0.71") long shaft w/ flat  
Four tapped mounting holes in face on 11/16"  
x 1-1/8" centers  
Overall size 3-1/4" x 1-1/2" x 1-3/8"  
Shpg 3/4 lb

**5 - 24 RPM, 24 VDC**

BARBER-COLMAN. Inline gearmotor.  
Operates on 12 volt DC at reduced speed.

**SPECIFICATIONS**

RPM 24  
Voltage 24 DC  
Amps 1.0  
Torque 30 in. lbs.  
Ratio 376.9:1  
Rotation reversible  
Duty continuous  
Shaft 1/4" diam. x 5/8" w/flat  
Mount face  
Size 2" diam. x 4 1/8"  
Shpg. 3 lbs.

**6 - 30.8 RPM, 24 DC**

New BARBER-COLMAN.  
Permanent magnet inline gearmotor.  
High torque in a compact size.  
Ideal for displays, powering toys, rotary actuators, etc.

**SPECIFICATIONS**

RPM 30.8  
Voltage 24  
Amps 0.255  
Torque 80.5 in. oz.  
Ratio 103.8:1  
Reversible  
Duty continuous  
Shaft 3/16" diam. x 3/4" w/flat  
Mount 3 hole end mount  
Size 1 1/4" diam. x 2 5/16"  
Shpg. 1 lb.

**7 - 24 VDC 68 RPM GEARMOTOR**

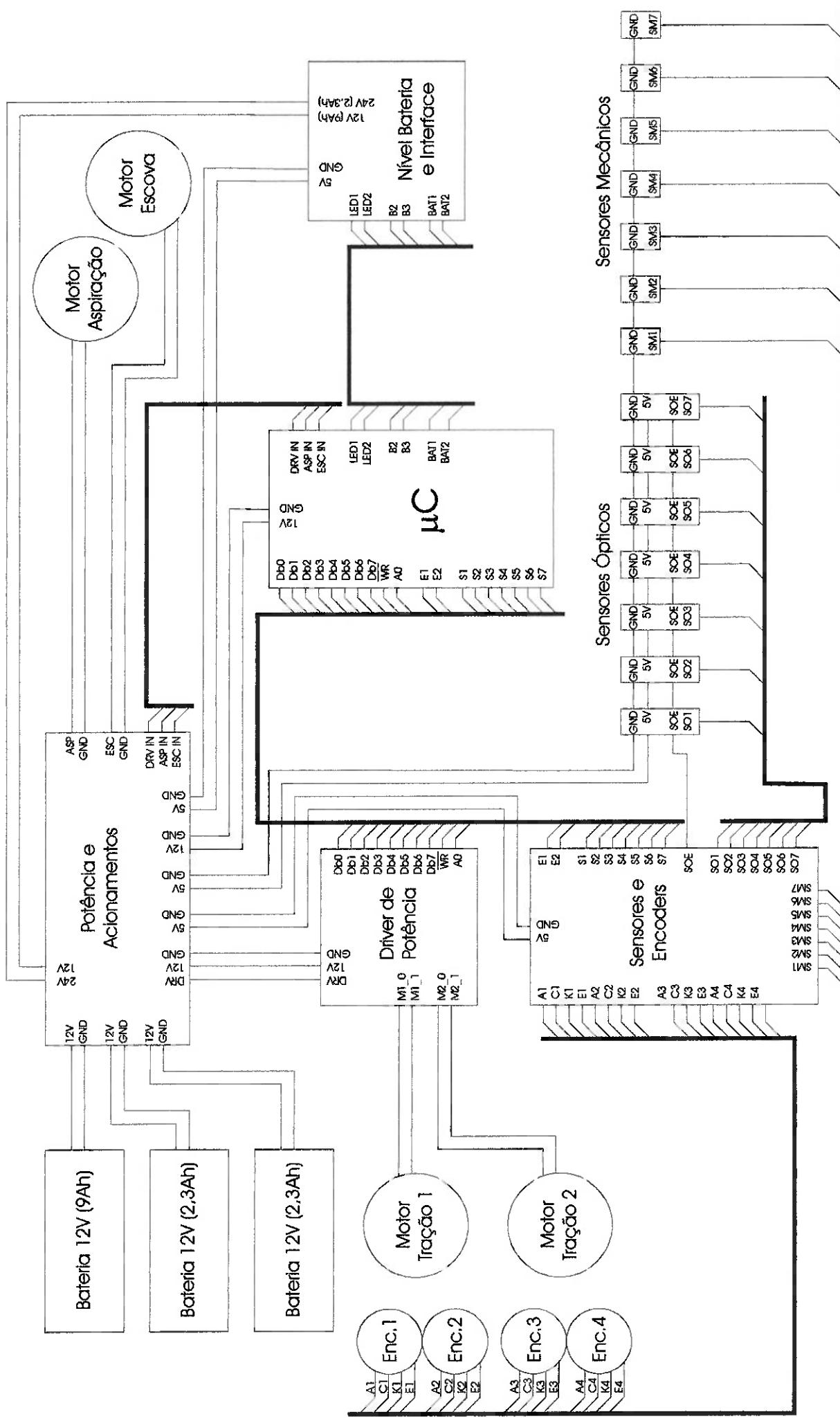
New BUEHLER gearmotor.

**SPECIFICATIONS**

68 RPM  
24 VDC  
50 mA  
1.2 Watts  
14 in-oz torque  
Reversible  
Intermittent duty  
0.2" dia. x 3/8" long shaft w/ 0.16" wide flat  
3 tapped mounting holes in motor face  
Size 1-3/8" dia. x 2-5/8" long  
Shpg 3/4 lb

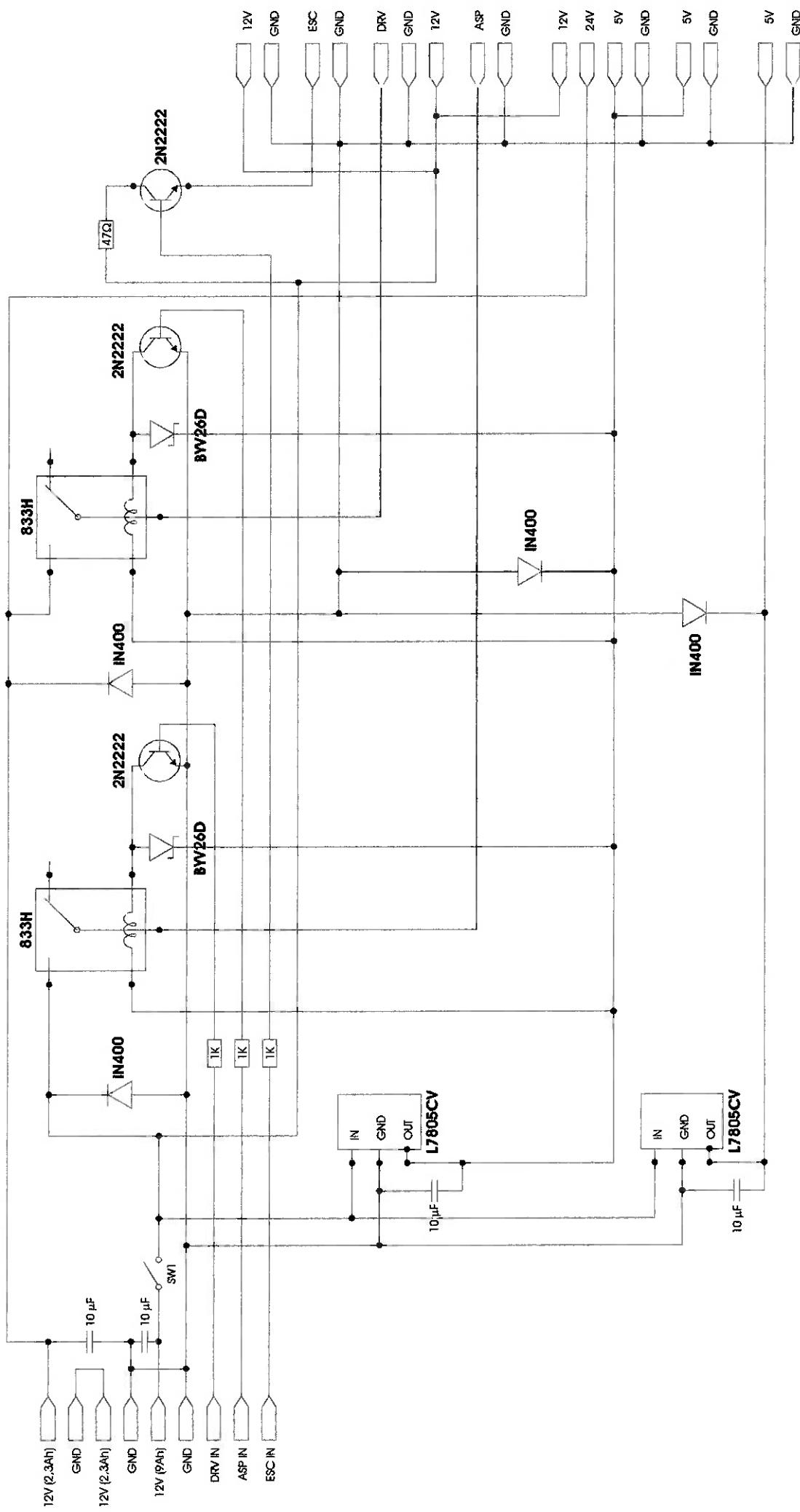
**Anexo C - Diagrama esquemático dos circuitos**

## Anexo C - Diagrama esquemático dos circuitos



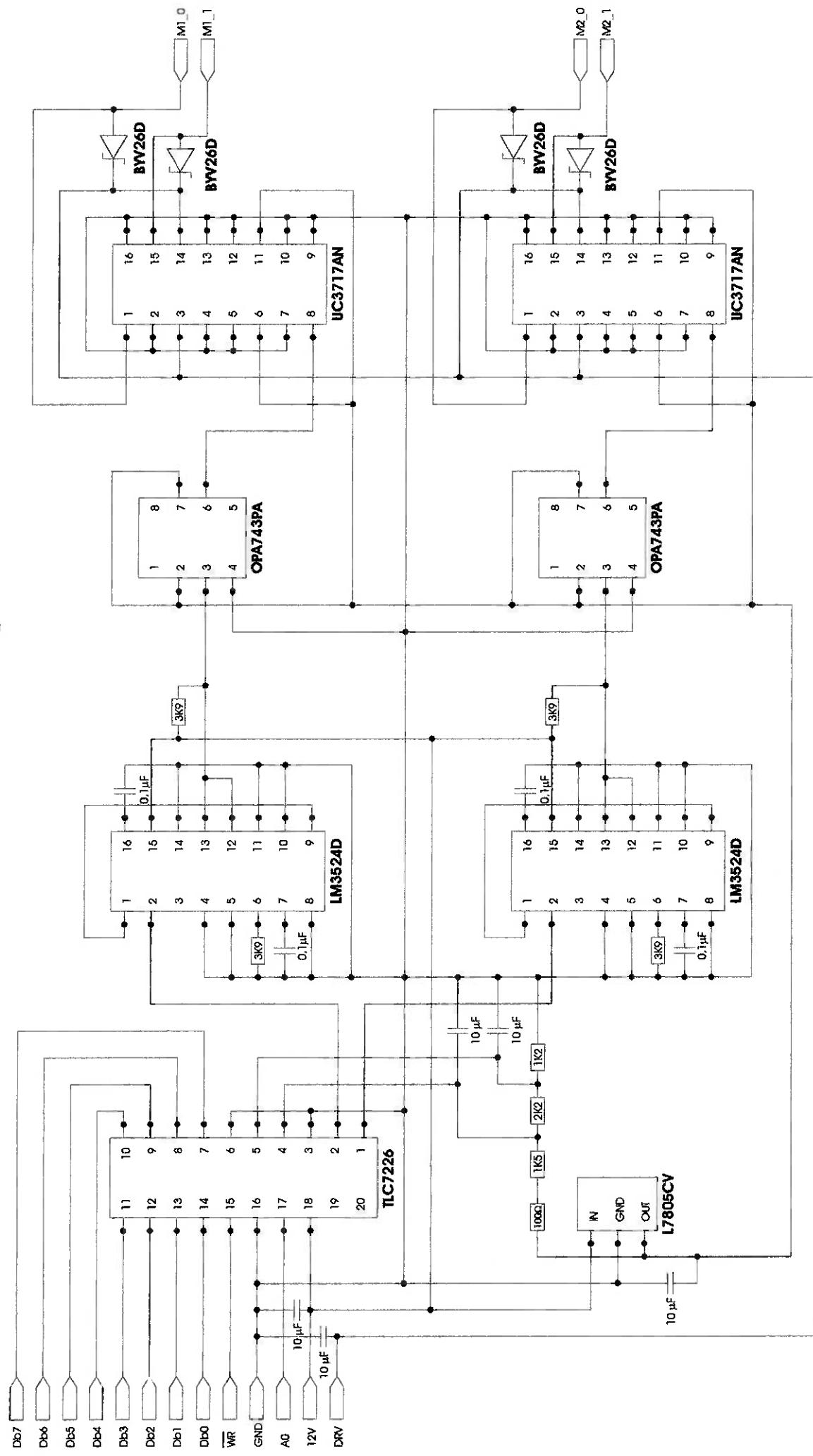
**Anexo D - Circuito de potência e acionamentos do sistema  
de limpeza e do driver de potência**

## Anexo D - Circuito de potência e acionamentos do sistema de limpeza e do driver de potência



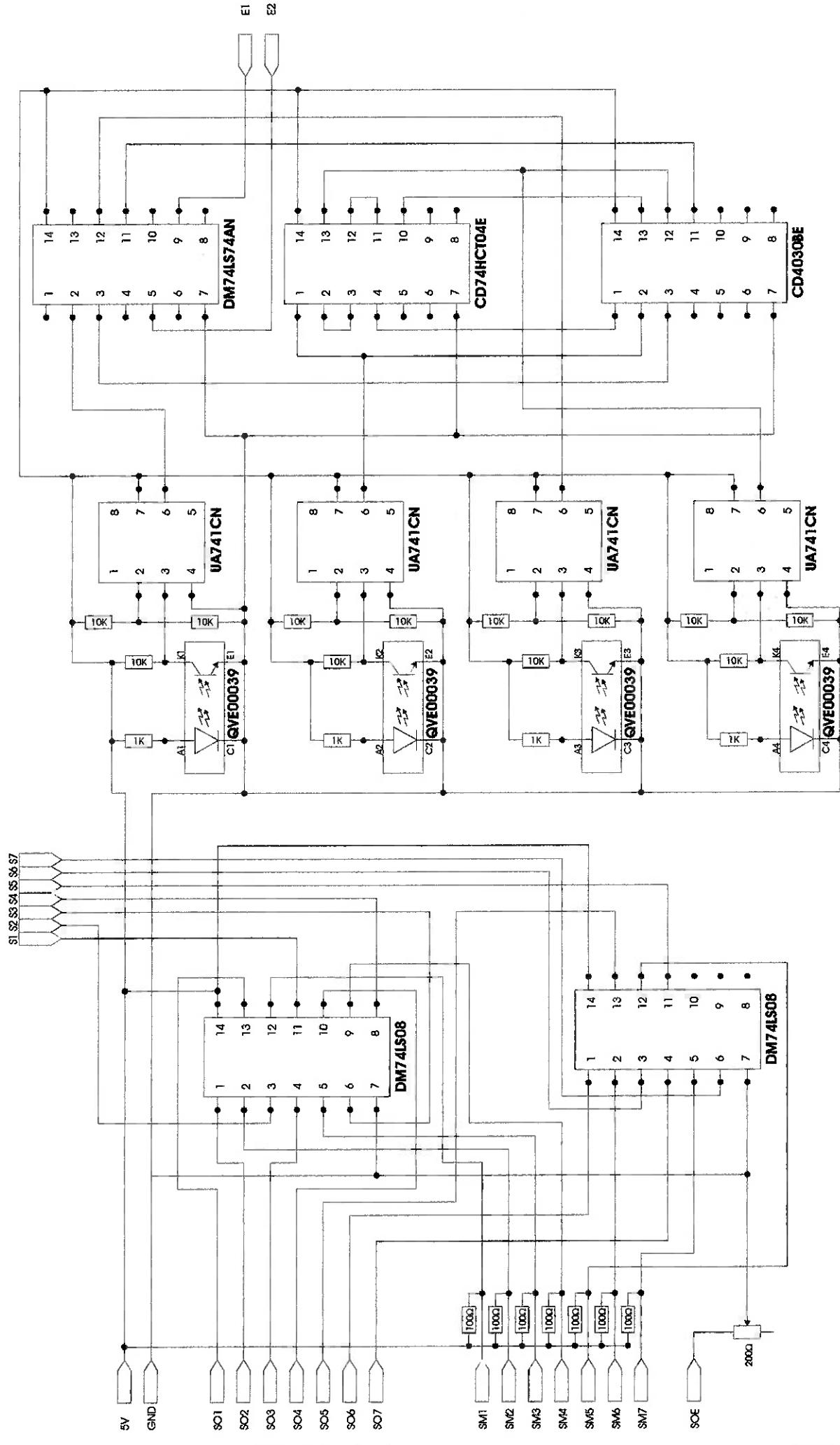
## **Anexo E - Circuito do driver de potência dos motores CC**

## Anexo E - Circuito do driver de potência dos motores CC



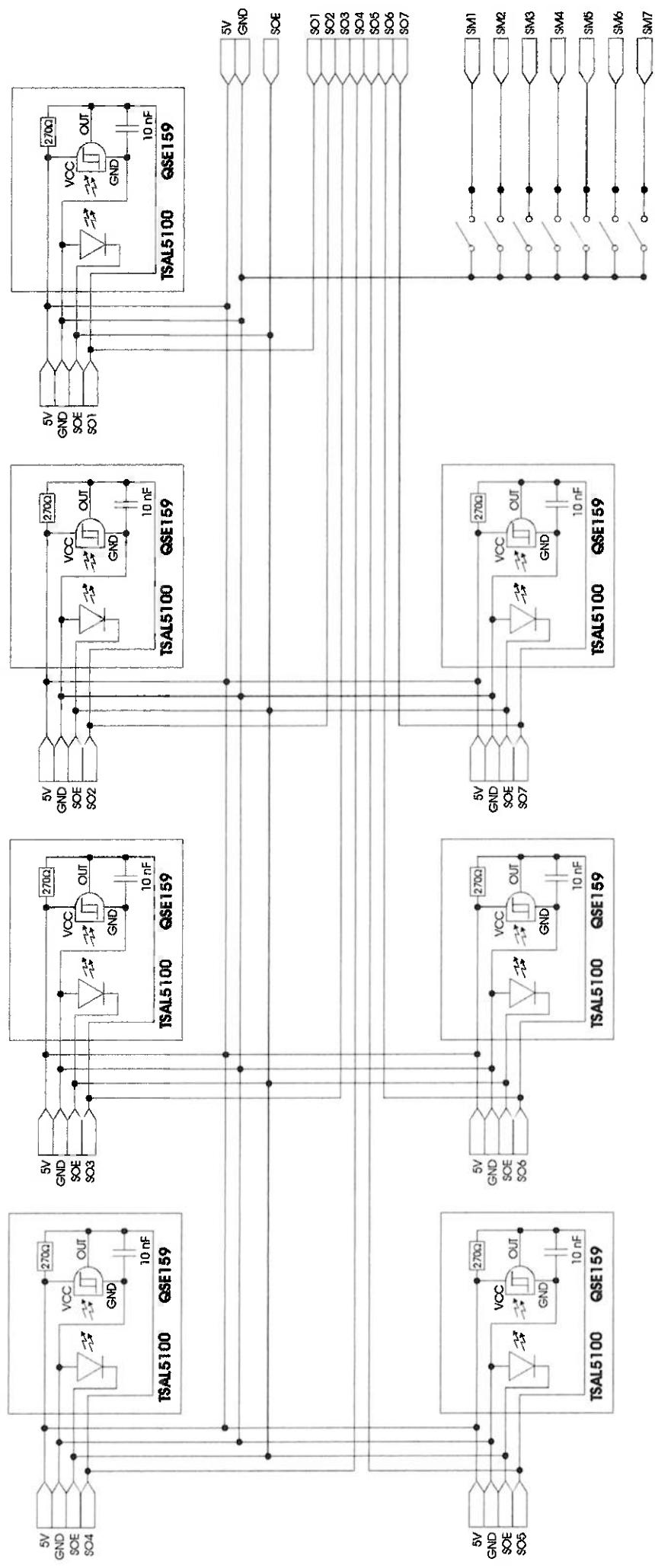
## **Anexo F - Circuito dos encoders e sensores**

## Anexo F - Circuito dos encoders e sensores



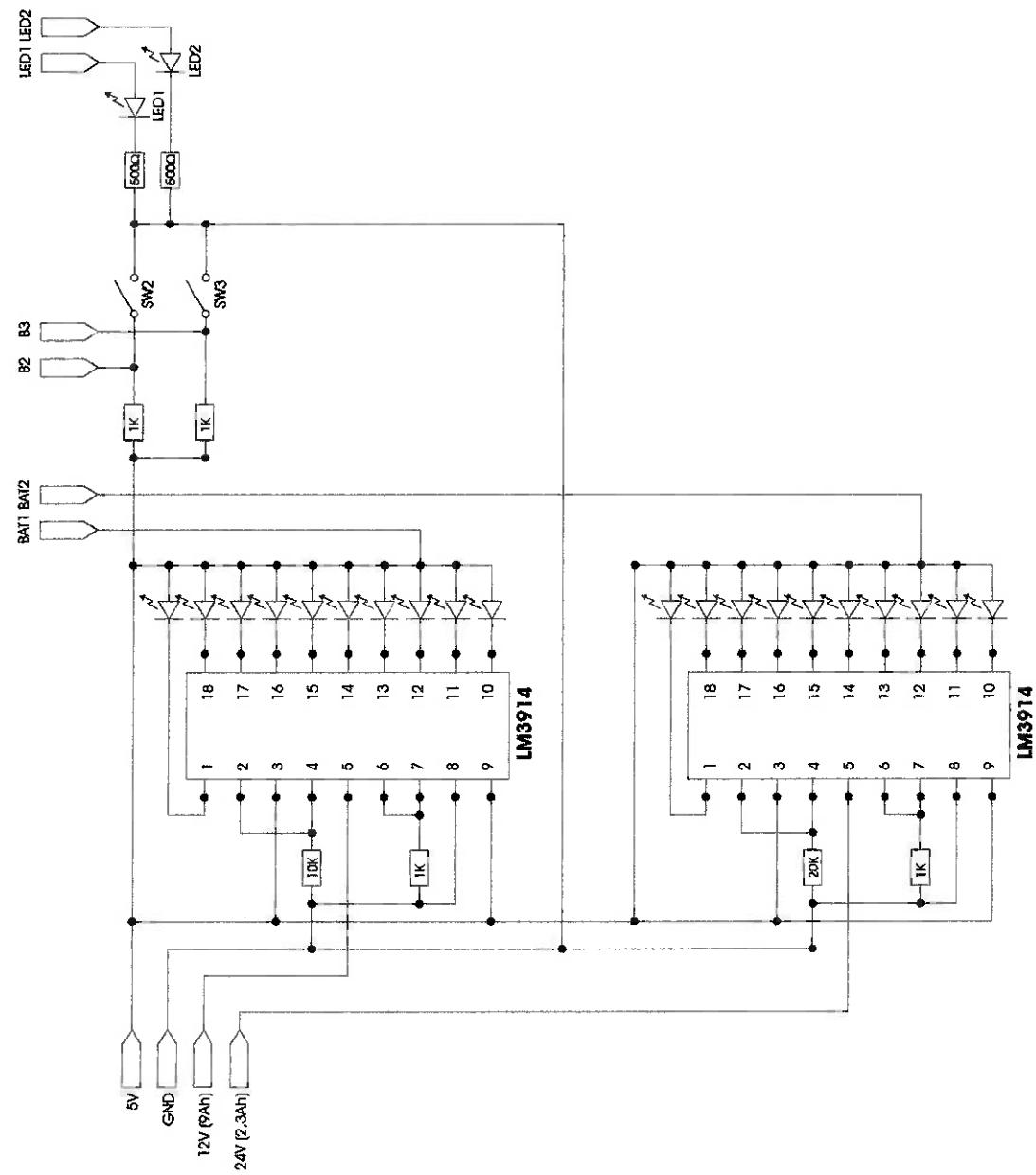
## **Anexo G - Circuito do chicote de sensores**

## Anexo G - Circuito do chicote de sensores



## **Anexo H - Circuito de nível de bateria e interface**

## Anexo H - Circuito de nível de bateria e interface

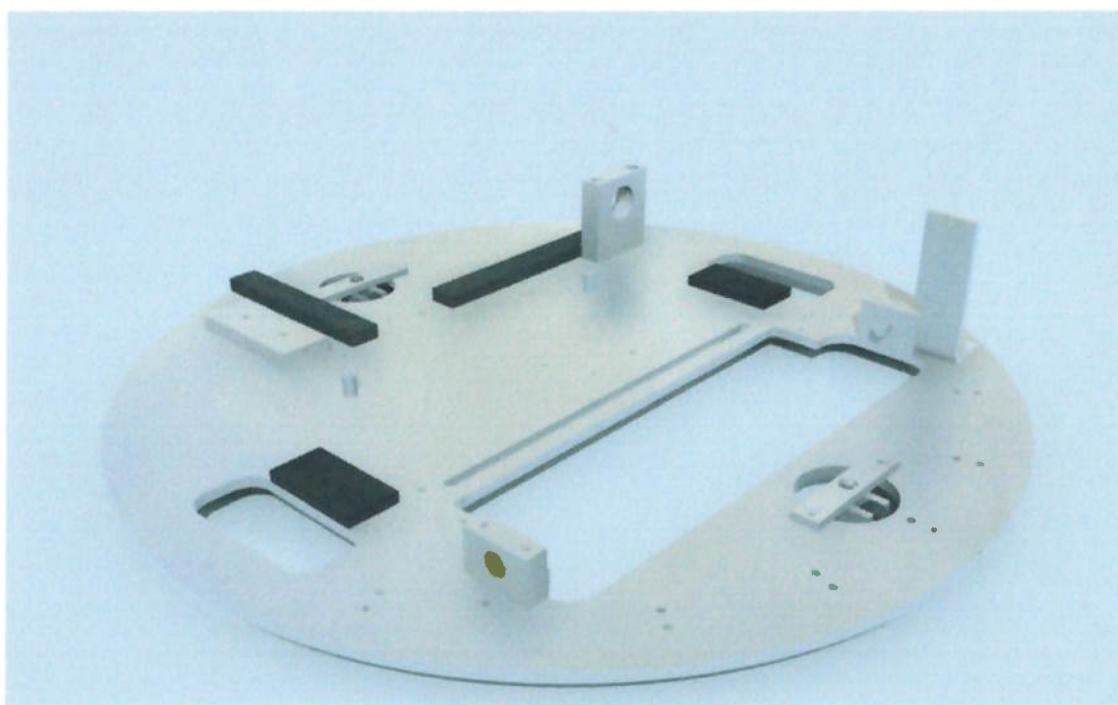


## **Anexo I - Seqüência de montagem**

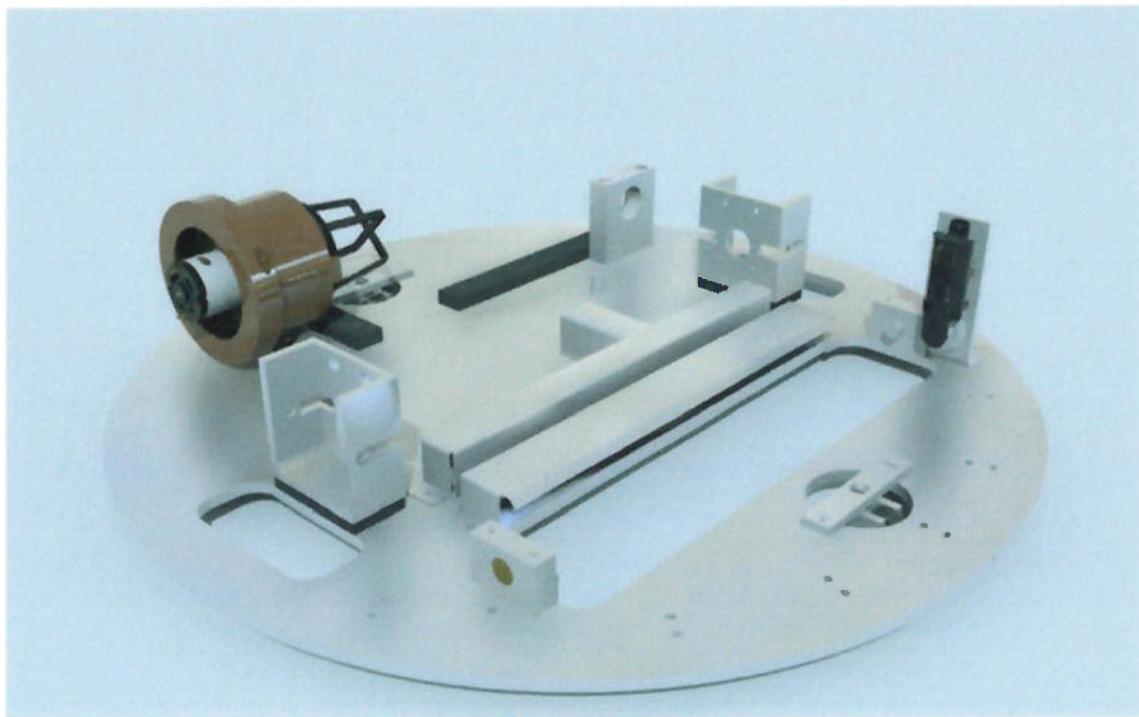
## Anexo I - Seqüência de montagem



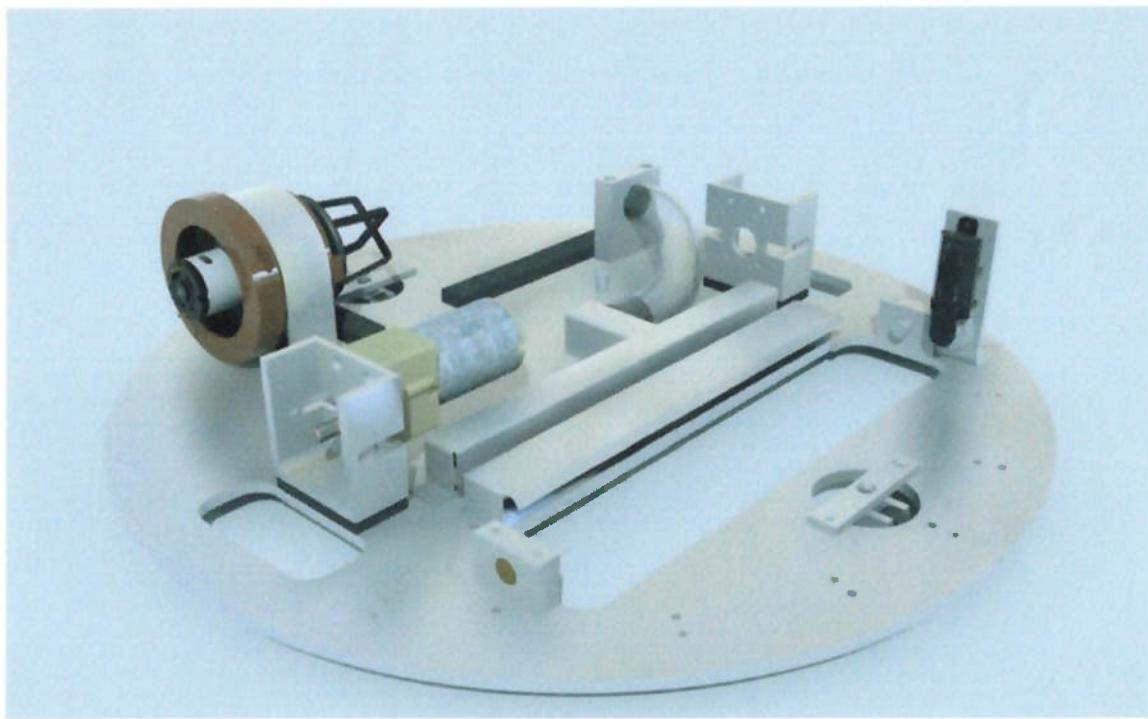
Seqüência 01 –Chassi.



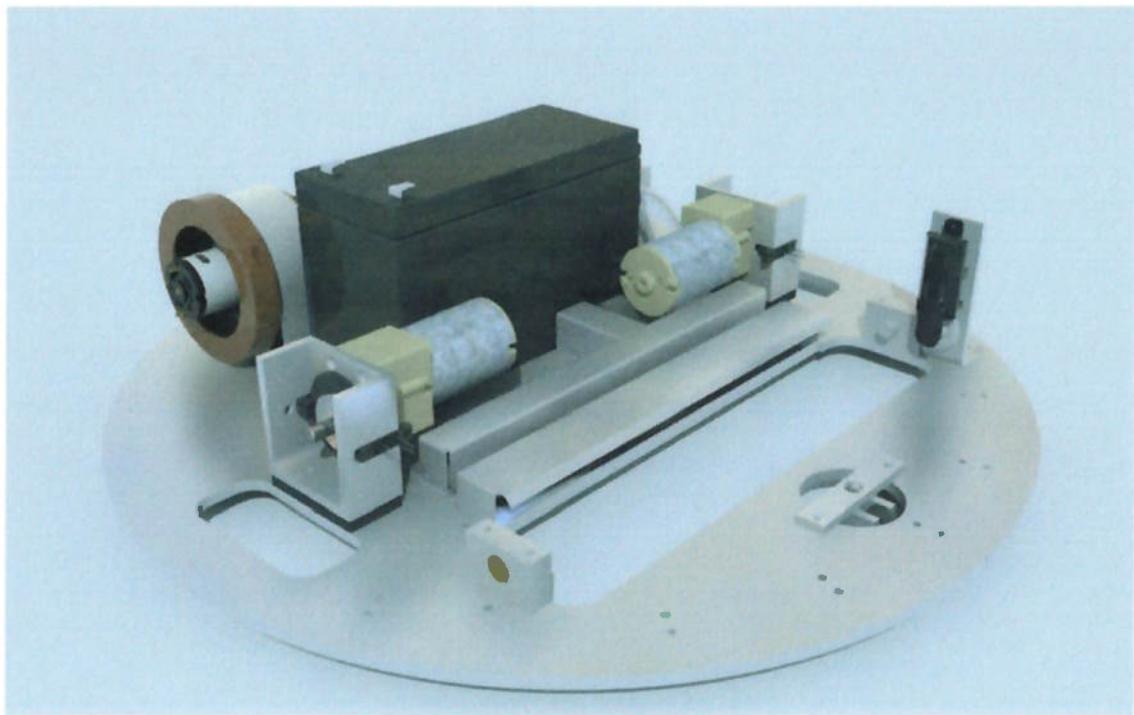
Seqüência 02 –Suportes da escova, suporte do motor da escova, rodas de apoio, calços de borracha do motor de locomoção, calços de borracha do reservatório, suporte do motor de aspiração, pinos limitadores da bateria, suporte da tubulação.



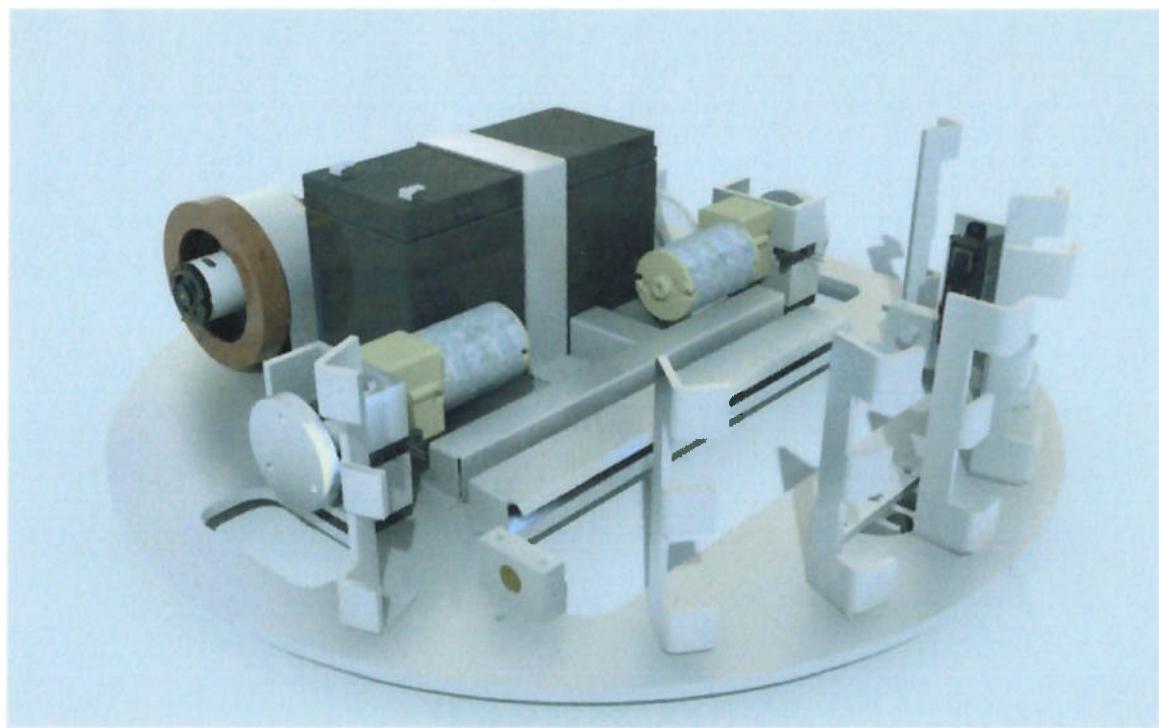
Seqüência 03 – Suportes dos motores de locomoção, motor da escova, reservatório da escova, ponta da tubulação, motor de aspiração.



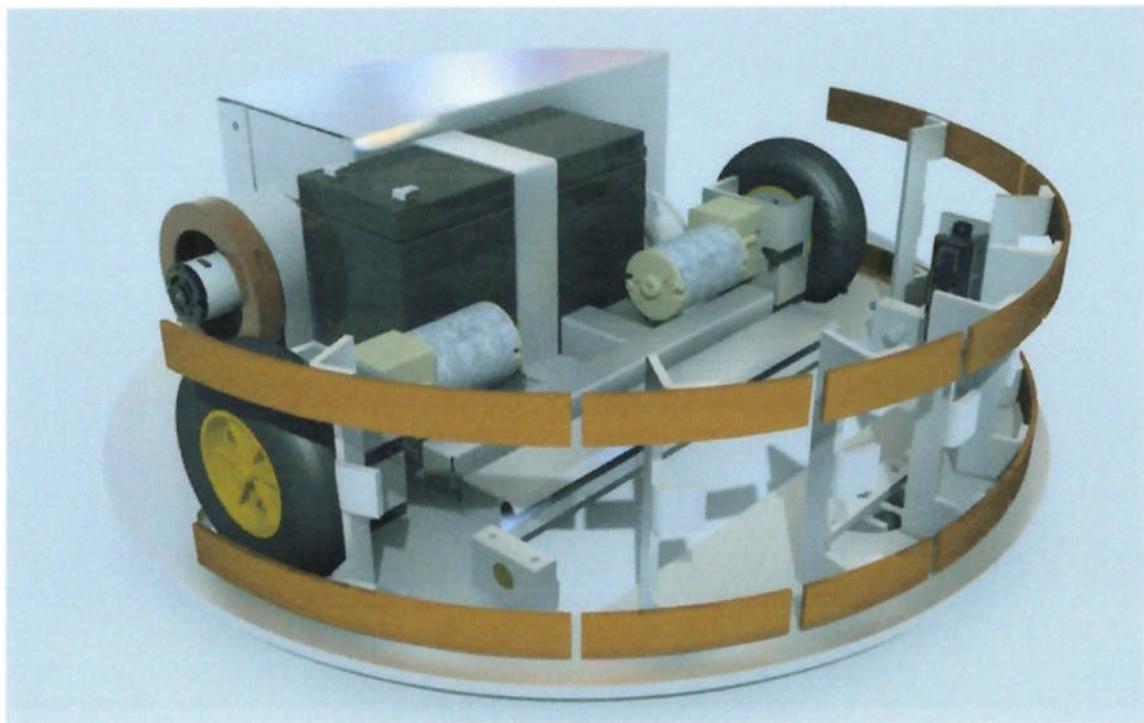
Seqüência 04 – Motores de locomoção(apenas um está representado), mangueira, abraçadeira do motor de aspiração.



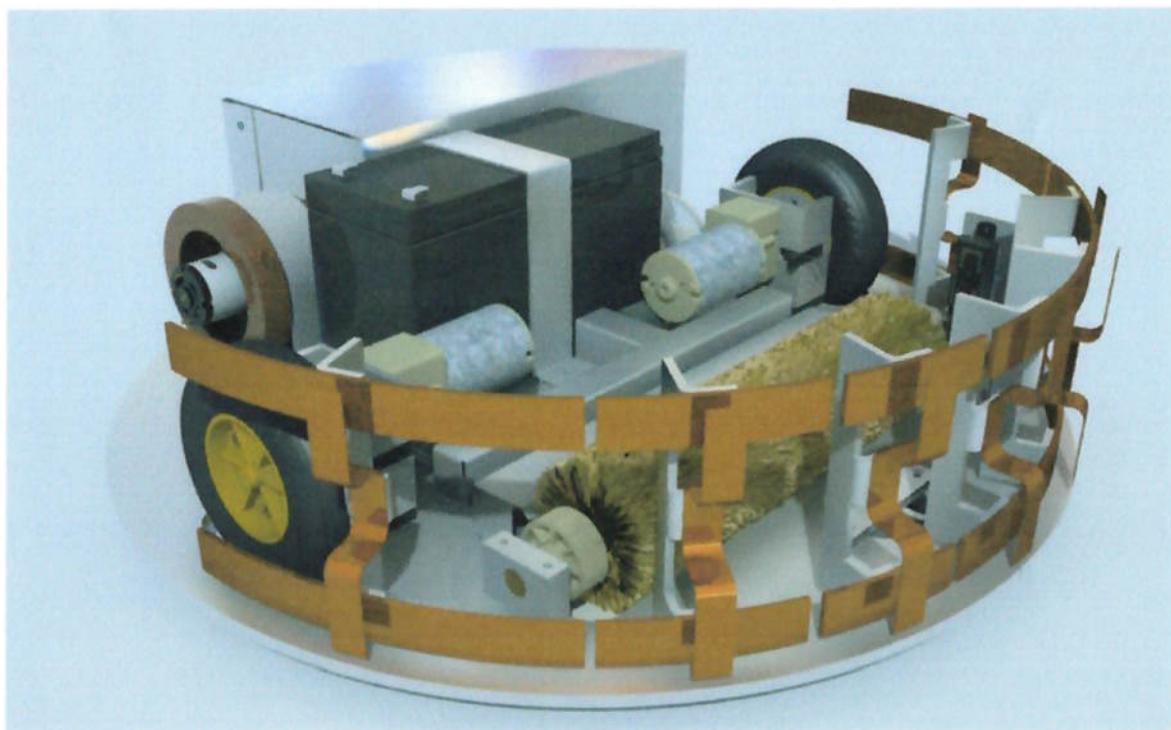
Seqüência 05 – Bateria maior, discos de encoder, placa de fixação dos discos, sensores do encoder.



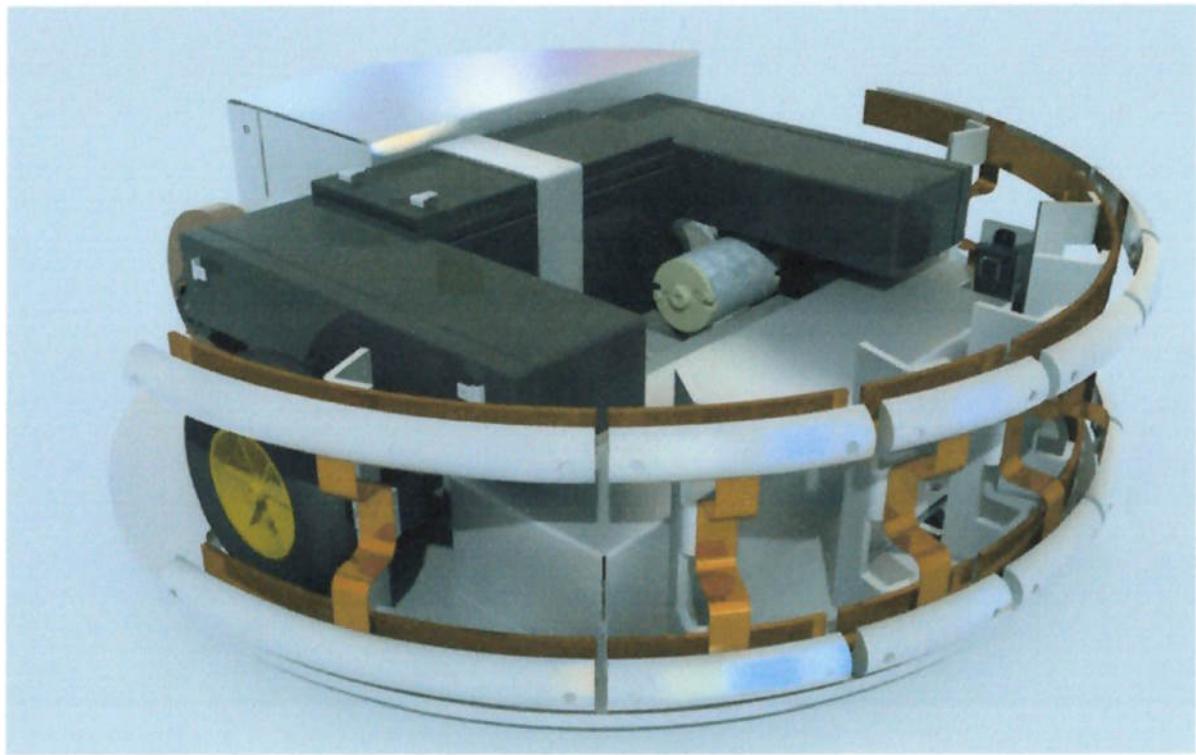
Seqüência 06 – Acoplamentos das rodas, suportes dos sensores mecânicos, abraçadeira da bateria.



Seqüência 07 – Contatos de cobre, reservatório de aspiração.



Seqüência 08 – Molas de bronze, escova.



Seqüência 09 – Baterias menores, encostos dos sensores mecânicos, capa da escova.

## **Anexo J - Programa em C**

## Anexo J - Programa em C

```
#class auto

void HabilitaMotorDireito(){
    auto int i;
    auto int espera;

    espera = 500;

    BitWrPortI(PBDR, &PBDRShadow, 0, 7); //escreve "0" em A0 (Seleciona
motor direito)
    for(i=0; i<espera; i++);

    BitWrPortI(PBDR, &PBDRShadow, 0, 6); //escreve "0" em /WR
    for(i=0; i<espera; i++);
    BitWrPortI(PBDR, &PBDRShadow, 1, 6); //escreve "1" em /WR
    for(i=0; i<espera; i++);
}

void HabilitaMotorEsquerdo(){
    auto int i;
    auto int espera;

    espera = 500;

    BitWrPortI(PBDR, &PBDRShadow, 1, 7); //escreve "1" em A0 (Seleciona
motor esquerdo)
    for(i=0; i<espera; i++);

    BitWrPortI(PBDR, &PBDRShadow, 0, 6); //escreve "0" em /WR
    for(i=0; i<espera; i++);
    BitWrPortI(PBDR, &PBDRShadow, 1, 6); //escreve "1" em /WR
    for(i=0; i<espera; i++);
}

void AcionaMotores(int velocidade){ //aciona os dois motores com tensões
    auto long i; //distintas porém que resultam em
    auto int espera; //velocidades idênticas
    auto int espera2;

    espera = 1000;
    espera2 = 300;

    if(velocidade==1){
//        WrPortI(PDDR, &PDDRShadow, 0x08); //escreve velocidade maxima
para tras
        WrPortI(PDDR, &PDDRShadow, 0x50); //escreve velocidade baixa
para tras
        HabilitaMotorEsquerdo();

//        WrPortI(PDDR, &PDDRShadow, 0x00); //escreve velocidade maxima
para tras
        WrPortI(PDDR, &PDDRShadow, 0x50); //escreve velocidade baixa
para tras
        HabilitaMotorDireito();
    }
    if(velocidade==0){
        WrPortI(PDDR, &PDDRShadow, 0x80); //escreve velocidade zero
    }
}
```

```

    HabilitaMotorEsquerdo();           //parada sem desaceleração
dos motores
    HabilitaMotorDireito();
}
if(velocidade==1){
    //Perfil de velocidades - aceleracao

    WrPortI(PDDR, &PDDRShadow, 0x96);      //escreve velocidade 25% para
frente
    HabilitaMotorEsquerdo();
    WrPortI(PDDR, &PDDRShadow, 0xA0);      //escreve velocidade 25% para
frente
    HabilitaMotorDireito();

    for(i=0; i<espera; i++);

    WrPortI(PDDR, &PDDRShadow, 0xB6);      //escreve velocidade 50% para
frente
    HabilitaMotorEsquerdo();
    WrPortI(PDDR, &PDDRShadow, 0xC0);      //escreve velocidade 50% para
frente
    HabilitaMotorDireito();

    for(i=0; i<espera; i++);

    WrPortI(PDDR, &PDDRShadow, 0xD6);      //escreve velocidade 75% para
frente
    HabilitaMotorEsquerdo();
    WrPortI(PDDR, &PDDRShadow, 0xE0);      //escreve velocidade 75% para
frente
    HabilitaMotorDireito();

    for(i=0; i<espera; i++);

    WrPortI(PDDR, &PDDRShadow, 0xF0);      //escreve velocidade 100% para
frente
    HabilitaMotorEsquerdo();
    WrPortI(PDDR, &PDDRShadow, 0xFE);      //escreve velocidade 100% para
frente
    HabilitaMotorDireito();

}

if(velocidade==10){
    //Perfil de velocidades - desaceleracao

    WrPortI(PDDR, &PDDRShadow, 0xD6);      //escreve velocidade 75% para
frente
    HabilitaMotorEsquerdo();
    WrPortI(PDDR, &PDDRShadow, 0xE0);      //escreve velocidade 75% para
frente
    HabilitaMotorDireito();

    for(i=0; i<espera2; i++);

    WrPortI(PDDR, &PDDRShadow, 0xB6);      //escreve velocidade 50% para
frente
    HabilitaMotorEsquerdo();
    WrPortI(PDDR, &PDDRShadow, 0xC0);      //escreve velocidade 50% para
frente
    HabilitaMotorDireito();
}

```

```

for(i=0; i<espera2; i++);

WrPortI(PDDR, &PDDRShadow, 0x96);      //escreve velocidade 25% para
frente
    HabilitaMotorEsquerdo();
    WrPortI(PDDR, &PDDRShadow, 0xA0);      //escreve velocidade 25% para
frente
    HabilitaMotorDireito();

for(i=0; i<espera2; i++);

WrPortI(PDDR, &PDDRShadow, 0x80);      //escreve velocidade zero
HabilitaMotorEsquerdo();
HabilitaMotorDireito();

}

}

void Avanco(int distancia, int *encoderEsquerdo, int *encoderDireito, int
*sensor){
    float pulsos;
    int aborta;                  //aciona os dois motores com velocidades
    int enc1;                    //idênticas para frete e retorna a leitura
    int enc2;                    //dos encoders e o sensor ativo quando ativado
    int sens;

    aborta = 0;
    pulsos = distancia / 5.25; //cada pulso de encoder representa um
deslocamento
    enc1 = 0;                   //linear de 5.25mm
    enc2 = 0;
    sens = 0;

    AcionaMotores(1);

    while(aborta==0){
        costate {           //conta os pulsos do encoder esquerdo
            if (BitRdPortI(PCDR, 3)){
                waitfor(BitRdPortI(PCDR, 3)==0); //incrementa o encoder
quando
                enc1 = enc1 + 1;          //bit 3 da porta C vai a 1
            }
            if(aborta==1)           //saída do costate
                abort;
        }

        costate {           //conta os pulsos do encoder direito
            if (BitRdPortI(PCDR, 1)){
                waitfor(BitRdPortI(PCDR, 1)==0); //incrementa o encoder
quando
                enc2 = enc2 + 1;          //bit 1 da porta C vai a 1
            }
            if(aborta==1)           //saída do costate
                abort;
        }

        costate{
            if( (enc1 >= pulsos) || (enc2 >= pulsos) ){
                AcionaMotores(10); //faz a parada com desaceleração dos
motores
            }
        }
    }
}

```

```

        aborta=1;           //quando o robô percorreu a distância
        abort;             //especificada pelo número de pulsos * 5.25
    }
    if(aborta==1)          //saída do costate
        abort;
}

costate{
    if( BitRdPortI(PBDR, 0)==0 ){
        sens=1;
        AcionaMotores(10);      //faz a parada com desaceleração dos
motores
        aborta=1;           //quando o sensor 1 é ativado
        abort;             //saída do costate
    }
    if( BitRdPortI(PBDR, 1)==0 ){
        sens=2;
        AcionaMotores(10);      //faz a parada com desaceleração dos
motores
        aborta=1;           //quando o sensor 2 é ativado
        abort;             //saída do costate
    }
    if( BitRdPortI(PBDR, 4)==0 ){
        sens=4;
        AcionaMotores(10);      //faz a parada com desaceleração dos
motores
        aborta=1;           //quando o sensor 4 é ativado
        abort;             //saída do costate
    }
    if( BitRdPortI(PBDR, 5)==0 ){
        sens=6;
        AcionaMotores(10);      //faz a parada com desaceleração dos
motores
        aborta=1;           //quando o sensor 6 é ativado
        abort;             //saída do costate
    }
    if( BitRdPortI(PCDR, 5)==0 ){
        sens=7;
        AcionaMotores(10);      //faz a parada com desaceleração dos
motores
        aborta=1;           //quando o sensor 7 é ativado
        abort;             //saída do costate
    }
    if(aborta==1)
        abort;             //saída do costate
}
}

if(aborta==1){           //altera as variáveis fora da função
    *encoderEsquerdo = enc1;       //utilizadas no escopo main
    *encoderDireito = enc2;
    *sensor = sens;              //altera a variável para o número
}                         //do sensor detectando obstáculo
}

void Retrocesso(int distancia, int *encoderEsquerdo, int *encoderDireito){
    float pulsos;
    int aborta;                  //aciona os dois motores com velocidades
    int enc1;                    //idênticas para trás e retorna a leitura
    int enc2;                    //dos encoders e o sensor ativo quando ativado
}

```

```

aborta = 0;
pulosos = distancia / 5.25; //distância linear percorrida é 5.25mm por
pulso
enc1 = 0;
enc2 = 0;

AcionaMotores(-1);           //chama a função que aciona motores para trás

while(aborta==0){
    costate {               //lê encoder esquerdo (conta os pulsos)
        if (BitRdPortI(PCDR, 3)){
            waitfor(BitRdPortI(PCDR, 3)==0);
            enc1 = enc1 + 1;
        }
        if(aborta==1)
            abort;
    }

    costate {               //lê encoder direito (conta os pulsos)
        if (BitRdPortI(PCDR, 1)){
            waitfor(BitRdPortI(PCDR, 1)==0);
            enc2 = enc2 + 1;
        }
        if(aborta==1)
            abort;
    }

    costate{
        if( (enc1 >= pullosos) || (enc2 >= pullosos) ){
            AcionaMotores(0);      //põe os motores sem desaceleração
            aborta=1;              //não é necessário desacelerar
            abort;                 //pois velocidade é baixa
        }
        if(aborta==1)
            abort;
    }
}

if(aborta==1){                //quando os costates são abortados
    *encoderEsquerdo = enc1;   //altera as variáveis utilizadas
    *encoderDireito = enc2;    //no escopo main
}
}

void Rotacao(int graus){      //rotação do robô ao redor de seu centro
    float pullos;
    int executando;          //variável em 1 quando os motores estão acionados
    int aborta;
    int enc1;
    int enc2;

    enc1 = 0;
    enc2 = 0;
    aborta=0;
    executando=0;
    pullos = abs(graus) * 0.53; //cada 1 grau de rotação equivale a 0.53
                                //pulosos de cada encoder
    while(aborta==0){
        costate {           //conta os pulsos do encoder esquerdo
            if (BitRdPortI(PCDR, 3)){
                waitfor(BitRdPortI(PCDR, 3)==0);

```

```

        enc1 = enc1 + 1;
    }
    if(aborta==1)
        abort;
}

costate {          //conta os pulsos do encoder direito
    if (BitRdPortI(PCDR, 1)){
        waitfor(BitRdPortI(PCDR, 1)==0);
        enc2 = enc2 + 1;
    }
    if(aborta==1)
        abort;
}

costate{
    if( (enc1 >= pulsos) || (enc2 >= pulsos) ){
        AcionaMotores(0);
        aborta=1;
        abort;           //aborta todos os costates quando
                        //o número de pulsos programado
    if(aborta==1)      //foi contado em um dos encoders
        abort;
    }

    if( (graus>0) && (executando==0) ){
//        WrPortI(PDDR, &PDDRShadow, 0xFE);      //escreve velocidade maxima
para frente
        WrPortI(PDDR, &PDDRShadow, 0xB0);      //escreve velocidade baixa
para frente
        HabilitaMotorDireito();

//        WrPortI(PDDR, &PDDRShadow, 0x08);      //escreve velocidade maxima
para tras
        WrPortI(PDDR, &PDDRShadow, 0x50);      //escreve velocidade baixa
para tras
        HabilitaMotorEsquerdo();

        executando=1;
    }

    if( (graus<0) && (executando==0) ){
//        WrPortI(PDDR, &PDDRShadow, 0x00);      //escreve velocidade maxima
para tras
        WrPortI(PDDR, &PDDRShadow, 0x50);      //escreve velocidade baixa
para tras
        HabilitaMotorDireito();

//        WrPortI(PDDR, &PDDRShadow, 0xF0);      //escreve velocidade maxima
para frente
        WrPortI(PDDR, &PDDRShadow, 0xA8);      //escreve velocidade baixa
para frente
        HabilitaMotorEsquerdo();

        executando=1;
    }
}
}

void Parada(int tempo){          //espera um tempo determinado pela
    int aborta;                  //variável tempo (em ms) sem que mais

```

```

aborta = 0;                                //nada seja executado pelo programa

while(aborta==0){
    costate{
        AcionaMotores(0);
        waitfor(DelayMs(tempo));
        aborta=1;
        abort;
    }
}

void main(){

    auto int i;                      //variáveis utilizadas
    auto int j;
    auto int botaoDS2;
    auto int botaoDS3;
    auto int velocidade;
    auto int encoderEsquerdo;
    auto int encoderDireito;
    auto int sensor;
    auto float x;
    auto float y;
    auto int rot;
    auto float rad;
    auto float distancia;
    auto int origem;
    auto long h;

    origem=0;                         //inicialização das variáveis
    rad=0;
    distancia = 0;
    x=0;
    y=0;
    rot=0;
    sensor=0;
    encoderEsquerdo=0;
    encoderDireito=0;
    botaoDS2=0;
    botaoDS3=0;

    WrPortI(PDFR, &PDFRShadow, 0x00);      // bits funcionam normalmente
    WrPortI(PDCR, &PDCRShadow, 0x00);      //clock da porta é clk/2
    WrPortI(PDDCR, &PDDCRShadow, 0x00);    //portas com nível high e low
    WrPortI(PDDDR, &PDDDRShadow, 0xFE);    //define os bits D1-D7 como
outputs

    WrPortI(SPCR, &SPCRShadow, 0x84);      // porta A como output

    BitWrPortI(PBDR, &PBDRShadow, 1, 6);   //escreve "1" em /WR

    BitWrPortI(PADR, &PADRShadow, 1, 5);   //habilita o relê do driver
    for(i=0; i<6000; i++);

    while(botaoDS2==0){                    //espera o botão DS2 ser precionado
        if (!BitRdPortI(PBDR, 2))          //para fazer o setup dos motores
e
            botaoDS2=1;                   //pará-los com velocidade 0x80
    }
}

```

```

BitWrPortI(PADR, &PADRShadow, 1, 0);
BitWrPortI(PADR, &PADRShadow, 0, 6);
BitWrPortI(PADR, &PADRShadow, 0, 7);
AcionaMotores(0);                                //parada dos motores com
                                                //velocidade 0x80
while(botaoDS3==0){                               //espera o botão DS3
    if (!BitRdPortI(PBDR, 3))                  //ser precionado para
        botaoDS3=1;                            //iniciar o programa de
}                                              //varredura da superficie
botaoDS3=0;

BitWrPortI(PADR, &PADRShadow, 1, 1);
Parada(2000);          //espera 2 segundos após o
                      //acionamento do botão
while( (botaoDS3==0) && (origem==0) ){
    sensor=0;                           //loop principal

    BitWrPortI(PADR, &PADRShadow, 1, 6); //aciona aspirador
    BitWrPortI(PADR, &PADRShadow, 1, 7); //aciona vassoura

    Avanco(10000, &encoderEsquerdo, &encoderDireito, &sensor);
    distancia = ((encoderEsquerdo + encoderDireito)/2)*5.25;
    Parada(500);           //espera de 0.5 segundo

    rad=rot*3.1415/180;

    if(sensor==1){
        Retrcesso(30, &encoderEsquerdo, &encoderDireito);
        Parada(500);           //se o sensor 1 for ativado
        Rotacao(-20);         //retrocede 3cm gira -20 graus
        Parada(500);

        rot = rot - 20;
    }
    if(sensor==2){
        Retrcesso(30, &encoderEsquerdo, &encoderDireito);
        Parada(500);           //se o sensor 2 for ativado
        Rotacao(-45);         //retrocede 3cm gira -45 graus
        Parada(500);

        rot = rot - 45;
    }
    if(sensor==4){
        Retrcesso(30, &encoderEsquerdo, &encoderDireito);
        Parada(500);           //se o sensor 4 for ativado
        Rotacao(90);           //retrocede 3cm gira 90 graus
        Parada(500);

        rot = rot + 90;
    }
    if(sensor==6){
        Retrcesso(30, &encoderEsquerdo, &encoderDireito);
        Parada(500);           //se o sensor 6 for ativado
        Rotacao(45);           //retrocede 3cm gira 45 graus
        Parada(500);

        rot = rot + 45;
    }
    if(sensor==7){
        Retrcesso(30, &encoderEsquerdo, &encoderDireito);
        Parada(500);           //se o sensor 7 for ativado
    }
}

```

```

Rotacao(20);           //retrocede 3cm gira 20 graus
Parada(500);

    rot = rot + 20;      //armazena e atualiza a atual
}                           //orientação angular do robô
if(sensor==0){            //em relação ao estado inicial
    distancia = distancia +30; //armazena a distancia
}                           //percorrida

distancia = distancia - 30; //deslocamento cartesiano
y = y + cos(rad) * distancia; //em relação ao ponto inicial
x = x - sin(rad) * distancia; //definido como (0,0)

if((x>-200) && (x<200) && (y>-200) && (y<200)){
    origem=1;           //verifica se o robô retornou ao ponto
    botaoDS3=1;         //inicial com erro de 20cm para mais
}                           //ou para menos em cada direção

BitWrPortI(PADR, &PADRShadow, 0,1); //acende o led

for(h=0; h<30000; h++) {          //espera o programa ser finalizado
    if (!BitRdPortI(PBDR, 3)) //pelo botão DS3 ou inicia outro
        botaoDS3=1;           //ciclo de avanço
}

}

BitWrPortI(PADR, &PADRShadow, 0, 6);
BitWrPortI(PADR, &PADRShadow, 0, 7);
while(1){                      //pisca os leds quando o programa
    Parada(150);              //foi finalizado
    BitWrPortI(PADR, &PADRShadow, 0,0);
    BitWrPortI(PADR, &PADRShadow, 1,1);
    Parada(150);
    BitWrPortI(PADR, &PADRShadow, 1,0);
    BitWrPortI(PADR, &PADRShadow, 0,1);
}
}

```

## **Anexo K - Desenhos Técnicos**