|  |
| --- |
| C:\Users\lbarros.DEI\AppData\Local\Microsoft\Windows\INetCache\Content.Word\EE-C.PNG |
| Bruno Miguel Vasconcelos da Silva, a88289  Diogo Miguel Cunha Fernandes, a88262  Duarte Miguel Novo Rodrigues, a88259  Francisco Lopes Salgado, a88231  João Pedro Dias Miranda, a88237  José Tomás Lima de Abreu, a88218  **AWR-19**  ***Analog Waiter Robot*** |
| Projeto integrador  Laboratórios e Práticas Integradas  Trabalho realizado sob a orientação do  **Professor Luís Barros** |
| dezembro de 2020 |

**Índice**

[Lista de Figuras v](#_Toc63157255)

[Lista de Tabelas vii](#_Toc63157256)

[Acrónimos e Siglas ix](#_Toc63157257)

[Capítulo 1 Introdução 11](#_Toc63157258)

[1.1 Introdução 11](#_Toc63157259)

[1.2 Enquadramento 11](#_Toc63157260)

[1.3 Especificações previstas 11](#_Toc63157261)

[1.4 Testes previstos 12](#_Toc63157262)

[1.5 Estudo da fiabilidade 13](#_Toc63157263)

[1.6 Planeamento inicial 13](#_Toc63157264)

[Capítulo 2 Desenho dos circuitos eletrónicos 15](#_Toc63157265)

[2.1 Introdução 15](#_Toc63157266)

[2.2 Controlo dos motores 16](#_Toc63157267)

[2.3 Isolamento e tratamento dos valores dos sensores 19](#_Toc63157268)

[2.4 Sistema eletrónico para o controlo dos motores 21](#_Toc63157269)

[Capítulo 3 Simulação dos circuitos eletrónicos 30](#_Toc63157270)

[3.1 Controlo dos motores 30](#_Toc63157271)

[3.2 Sistema eletrónico para o controlo dos motores 33](#_Toc63157272)

[Capítulo 4 Desenho da implementação dos circuitos eletrónicos 37](#_Toc63157273)

[4.1 Introdução 37](#_Toc63157274)

[4.2 Controlo dos motores 37](#_Toc63157275)

[4.3 Sensores 40](#_Toc63157276)

[4.4 Máquina de estados 42](#_Toc63157277)

[4.5 Lista de componentes 45](#_Toc63157278)

[Capítulo 5 Desenho do encapsulamento 52](#_Toc63157279)

[Capítulo 6 Possíveis opções alternativas para o desenho dos circuitos 54](#_Toc63157280)

[6.1 Introdução 54](#_Toc63157281)

[Capítulo 7 Evolução do atual desenho para um sistema baseado em microcomputador 55](#_Toc63157282)

[7.1 Introdução 55](#_Toc63157283)

[Referências 56](#_Toc63157284)

Lista de Figuras

[Figura 1 - Diagrama de Gantt 14](#_Toc63171681)

[Figura 2 - Módulo Driver L298N 15](#_Toc63171682)

[Figura 3 - Array de sensores QTR-8A 15](#_Toc63171683)

[Figura 4 - Sensores usados para seguir a linha: S3 e S6 16](#_Toc63171684)

[Figura 5 - Esquemático do gerador de onda PWM 16](#_Toc63171685)

[Figura 6 - Esquemático do circuito estabelece a tensão de comparação 18](#_Toc63171686)

[Figura 7 - Esquemático do circuito para os sensores a ser usados como componentes analógicos 20](#_Toc63171687)

[Figura 8 - Esquemático do circuito para os sensores a ser usados como componentes digitais 20](#_Toc63171688)

[Figura 9 - Máquina de Moore 21](#_Toc63171689)

[Figura 10 - Linha preta usada para os testes 21](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171690)

[Figura 11 – Sensores usados na máquina de estados: S1, S5 e S8 22](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171691)

[Figura 12 - Botão de pressão 22](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171692)

[Figura 13 - Diagrama da máquina de estados 23](#_Toc63171693)

[Figura 14 - Lógica do próximo estado 25](#_Toc63171694)

[Figura 15 - Lógica de saída 25](#_Toc63171695)

[Figura 16 - Desenho da máquina de estados 26](#_Toc63171696)

[Figura 17 - Esquema do Flip-Flop J-k 26](#_Toc63171697)

[Figura 18 - Implementação do Oscilador 27](#_Toc63171698)

[Figura 19 - Resposta do Oscilador de 1 kHz 28](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171699)

[Figura 20 - Circuito RC de reset e a sua resposta no tempo. 28](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171700)

[Figura 21 - Circuito de debounce do botão de pressão (quando premido) 29](#_Toc63171701)

[Figura 22 - Circuito gerador PWM com entrada de 0 V 30](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171702)

[Figura 23 - Circuito gerador PWM com entrada de 2,1 V 30](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171703)

[Figura 24 - Circuito gerador PWM com entrada de 3,6 V 31](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171704)

[Figura 25 - Circuito gerador da tensão de comparação com ambas as entradas a 0 V 31](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171705)

[Figura 26 - Circuito gerador da tensão de comparação com entrada do sensor 1 a 5 V e entrada do sensor 2 a 0 V 32](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171706)

[Figura 27 - Circuito gerador da tensão de comparação com entrada do sensor 1 a 2,5 V e entrada do sensor 2 a 0 V 33](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171707)

[Figura 28 - Estado 0 (Parado) 34](#_Toc63171708)

[Figura 29 - Estado 1 (Anda para a frente) – Botão foi pressionado 34](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171709)

[Figura 30 - Estado 2 (Dá a volta para a esquerda) – S1 e S8 a nível lógico alto 35](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171710)

[Figura 31 - Estado 3 (Dá a volta) - S1 e S8 a nível lógico baixo 35](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171711)

[Figura 32 - Estado 0 (Parado novamente) - S5 = 1 36](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171712)

[Figura 33 - Esquemático PCB controlo dos motores 37](#_Toc63171713)

[Figura 34 - Layout PCB controlo dos motores 38](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171714)

[Figura 35 - Vista bottom PCB controlo dos motores 38](#_Toc63171715)

[Figura 36 - Vista top PCB controlo dos motores 39](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171716)

[Figura 37 - Vista 3D PCB controlo dos motores 39](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171717)

[Figura 38 - Esquemático PCB sensores 40](#_Toc63171718)

[Figura 39 - Layout PCB sensores 40](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171719)

[Figura 40 - Vista bottom PCB sensores 41](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171720)

[Figura 41 - Vista 3D PCB sensores 41](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171721)

[Figura 42 - Esquemático PCB Máquina de Estados 42](#_Toc63171722)

[Figura 43 - Layout PCB Máquina de Estados 43](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171723)

[Figura 44 - Vista bottom PCB máquina de estados 43](#_Toc63171724)

[Figura 45 - Vista top PCB máquina de estados 44](#_Toc63171725)

[Figura 46 - Vista 3D PCB máquina de estados 44](#_Toc63171726)

[Figura 47 - Vista superior do robô 52](#_Toc63171727)

[Figura 48 - Vista inferior do robô 52](#_Toc63171728)

[Figura 49 - Vista lateral do robô 53](#_Toc63171729)

[Figura 50 - Vista traseira do robô 53](#_Toc63171730)

[Figura 51 - Vista frontal do robô 53](#_Toc63171731)

Lista de Tabelas

[Tabela 1 - Planeamento do Projeto 14](#_Toc63171900)

[Tabela 2 - Tabela completa para determinação da lógica completa da máquina de estados 24](#_Toc63171901)

[Tabela 3 - Atribuição de estados 25](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171902)

[Tabela 4 - Tabela de excitação do Flip-Flop JK 25](file:///C:\1Sem\LPI\PI\01Relatorios\Relatorio.docx#_Toc63171903)

[Tabela 5 - Lista de Componentes 45](#_Toc63171904)

Acrónimos e Siglas

|  |  |
| --- | --- |
| **Acrónimo/Sigla** | **Significado** |
| AWR | *Analog Waiter Robot* |
| LED | *Light emitting diode* |
| PWM | *Pulse with Modulation* |
| BMS | *Battery Management System* |
|  |  |

Introdução

Introdução

Perante o atual panorama pandémico da Covid-19 [1], pretende-se implementar uma ideia que terá um impacto positivo na vida das pessoas infetadas pela doença bem como dos que as rodeiam.

A maioria das ideias nesta área tem como foco principal a saúde pública da população em geral, tais como, robôs de desinfeção, por exemplo. Há menos projetos com foco individual em pessoas que tenham contraído a doença.

Na China existe um robô (*little peanut*) [2] com a mesma finalidade que foi utilizada num hotel para entregar comida porta a porta a hóspedes com suspeita de infeção.

Tendo em consideração que uma pessoa em isolamento deve evitar o contacto com o mundo exterior, pretende-se desenvolver um produto que permita a entrega e recolha de bens essenciais de forma segura. De forma a facilitar a sua desinfeção e o seu manuseamento, o robô deverá ter superfícies lisas e uma interface simples.

## Enquadramento

O *Analog Waiter Robot* (AWR) é um robô seguidor de linha focado na assistência a pessoas em isolamento que não podem contactar com o mundo exterior. Basta colocar no seu suporte o que pretende fazer chegar ao paciente, colocá-lo sobre a linha, ligá-lo e este seguirá o percurso até ao destino.

## Especificações previstas

O AWR terá de deslocar-se entre dois pontos, previamente definidos, com base numa linha instalada nos meios de acesso à divisão em que o paciente em questão se situa.

O AWR terá de ser colocado sobre a linha com os bens essenciais (comida e/ou alimentos) num tabuleiro. Após estar alinhado, deverá ser ligado por um funcionário responsável, para que inicie a sua trajetória. Ao chegar ao destino, demarcado com uma linha perpendicular, o robô terá de parar de forma a que o paciente recolha os bens a si destinados. Quando o paciente desejar, poderá acionar o robô de forma a que este reinicie o seguimento da linha de volta ao ponto de partida. Para tal, o AWR fará uma rotação de 180 ° sobre a linha e iniciar a marcha até ao ponto de início. Como a alimentação do robô será a baterias, eventualmente, terá de ser ligado à rede elétrica para ser carregado.

O tipo de desenvolvimento do nosso produto pode ser classificado como “misto”, uma vez que não tem necessariamente características inovadoras, mas é concebido para servir as necessidades particulares de um comprador. O produto será composto na sua totalidade por componentes ou subsistemas que são comprados “*off-the-shelf*”. Isto é, que só precisam de ser instalados e configurados para entrarem em funcionamento.

## Testes previstos

De forma a testar as especificações acima previstas, deverão ser realizados testes.

Para que o AWR siga a linha, será implementado um seguidor de linha através de um *array* de sensores de reflexão de luz. Irão ser realizados ensaios experimentais simulando o ambiente no qual este irá operar. Estes permitirão determinar os valores de referência que servirão para o dimensionamento e calibração dos circuitos de direção do robô. Para isso, este deverá ser colocado em diferentes posições sobre a linha.

Depois de determinados os valores de referência, deverão ser testados os circuitos de direção do robô, que usarão controladores para manter os valores dos sensores na gama de referência. Para isso, o AWR deverá ser colocado em diferentes posições sobre a linha para verificar se o controlador cumpre o seu propósito. Isto é, se o robô segue a linha até ao destino final.

Para controlar os diferentes estados de funcionamento do robô, irá ser desenvolvida uma máquina de estados baseada em circuitos sequenciais. De forma a testar o seu funcionamento, esta deverá ser sujeita a diferentes inputs de forma a verificar a comutação entre estados com a utilização de LEDs nas suas saídas. As saídas da máquina de estados irão controlar o estado de funcionamento dos motores.

Consoante as saídas da máquina de estados, cada motor poderá ter três modos de funcionamento: sentido horário, sentido anti-horário ou parado. Nesse momento poderão ser testadas todas as funcionalidades do robô, verificando se segue a linha ou não.

Estudo da fiabilidade

## Planeamento inicial

De forma a testar as especificações acima previstas, deverão ser realizados testes.

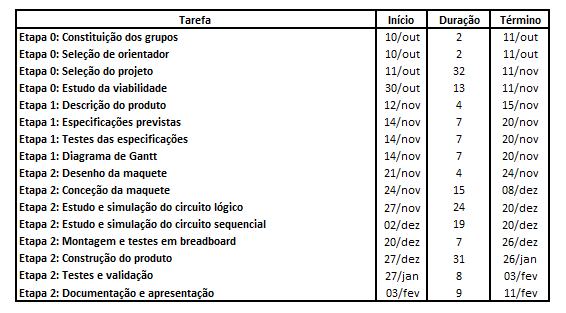
Para que o AWR siga a linha, será implementado um seguidor de linha através de um *array* de sensores de reflexão de luz. Irão ser realizados ensaios experimentais simulando o ambiente no qual este irá operar. Estes permitirão determinar os valores de referência que servirão para o dimensionamento e calibração dos circuitos de direção do robô. Para isso, este deverá ser colocado em diferentes posições sobre a linha.

Depois de determinados os valores de referência, deverão ser testados os circuitos de direção do robô, que usarão controladores para manter os valores dos sensores na gama de referência. Para isso, o AWR deverá ser colocado em diferentes posições sobre a linha para verificar se o controlador cumpre o seu propósito. Isto é, se o robô segue a linha até ao destino final.

Para controlar os diferentes estados de funcionamento do robô, irá ser desenvolvida uma máquina de estados baseada em circuitos sequenciais. De forma a testar o seu funcionamento, esta deverá ser sujeita a diferentes inputs de forma a verificar a comutação entre estados com a utilização de LEDs nas suas saídas. As saídas da máquina de estados irão controlar o estado de funcionamento dos motores.

Consoante as saídas da máquina de estados, cada motor poderá ter três modos de funcionamento: sentido horário, sentido anti-horário ou parado. Nesse momento poderão ser testadas todas as funcionalidades do robô, verificando se segue a linha ou não.

Tabela 1 - Planeamento do Projeto



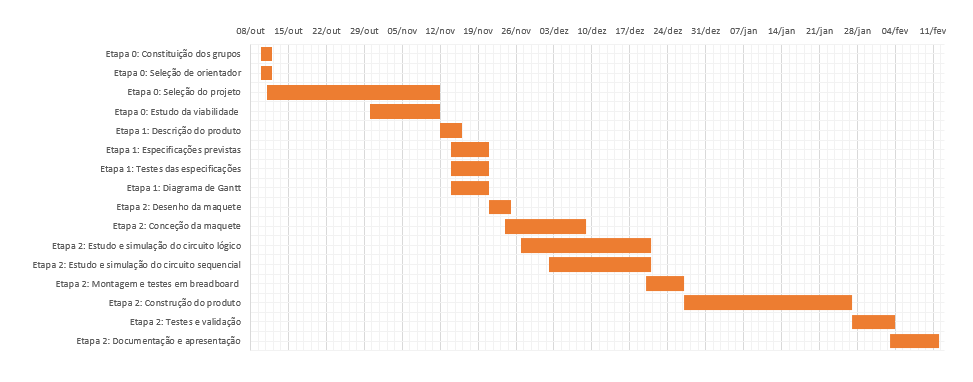


Figura 1 - Diagrama de Gantt

Desenho dos circuitos eletrónicos

Introdução

O desenho dos circuitos eletrónicos tem por base um módulo - placa driver de motores com dupla ponte H e base no L298N [3] - que faz a ligação entre os diferentes circuitos.

Este módulo permite controlar dois motores fazendo uso de seis entradas de controlo. A velocidade de rotação pode ser ajustada através de um sinal de entrada PWM, ligado a ENA para controlar o motor A e ENB para o motor B As restantes quatro entradas (IN1, IN2, IN3 e IN4) controlam, duas a duas, o estado e sentido de rotação dos motores.

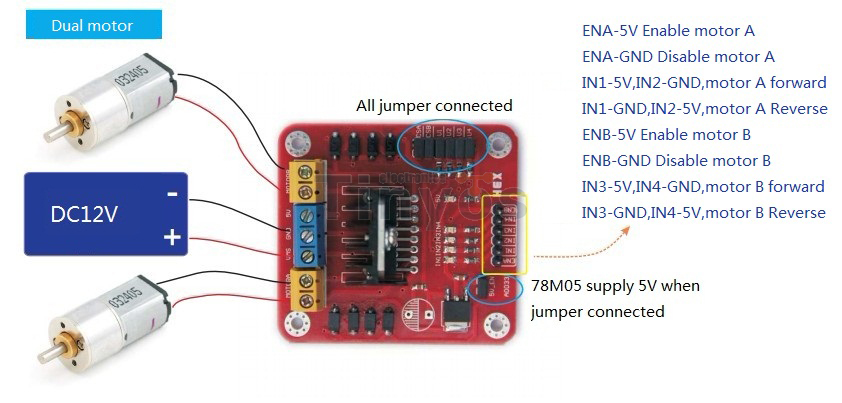


Figura 2 - Módulo Driver L298N

Um sensor é um dispositivo que responde a um estímulo físico com um sinal elétrico. Neste caso, o estímulo físico é a cor da linha que deverá ser traduzida numa tensão. De forma a ser possível cumprir os vários objetivos – seguir uma linha, dar a volta e parar – utilizou-se um *array* de sensores.



Figura 3 - Array de sensores QTR-8A

Controlo dos motores

O controlo da velocidade de rotação dos motores está diretamente relacionado com leitura dos valores de tensão obtidos pelo sensor. Quando ambos os sensores estão fora da linha têm de ter velocidade quase máxima. Os sensores utilizados para o propósito estão ilustrados na imagem abaixo.



Figura 4 - Sensores usados para seguir a linha: S3 e S6

Quando um dos sensores se aproxima da linha preta, o motor do lado deste sensor deverá ter a sua velocidade reduzida, provocando um desvio de direção, que é compensado por um aumento de velocidade do motor do lado oposto.

Como referido anteriormente, o controlo da velocidade de rotação dos motores é efetuado através de uma entrada *PWM*. Para gerar este sinal recorre-se a dois circuitos. O primeiro estabelece uma tensão que será usada na entrada do segundo circuito de forma a gerar na saída o sinal *PWM* desejado.

Para uma melhor compreensão, inicia-se a explicação pelo segundo circuito. Este tem como objetivo gerar uma onda PWM consoante uma tensão de entrada. Quanto maior o valor deste sinal maior será o *duty cycle* da onda PWM à saída. Para realizar este circuito utilizou-*se TL494 Pulse-Width-Modulation Control Circuits* [4]. De realçar que este circuito gera uma onda triangular interna entre 0 V e 3,3 V (valor teórico - na prática e em simulações verificou-se que este valor se aproxima mais de 3,6 V). Sendo assim, para ser possível variar a velocidade do motor, a tensão de entrada deste circuito terá de estar compreendida entre estes dois valores. O circuito tem o seguinte esquemático.

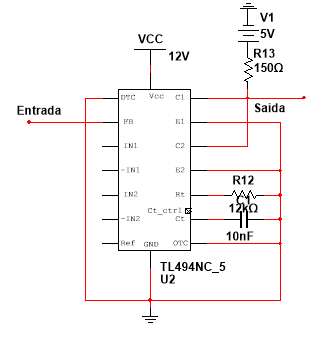
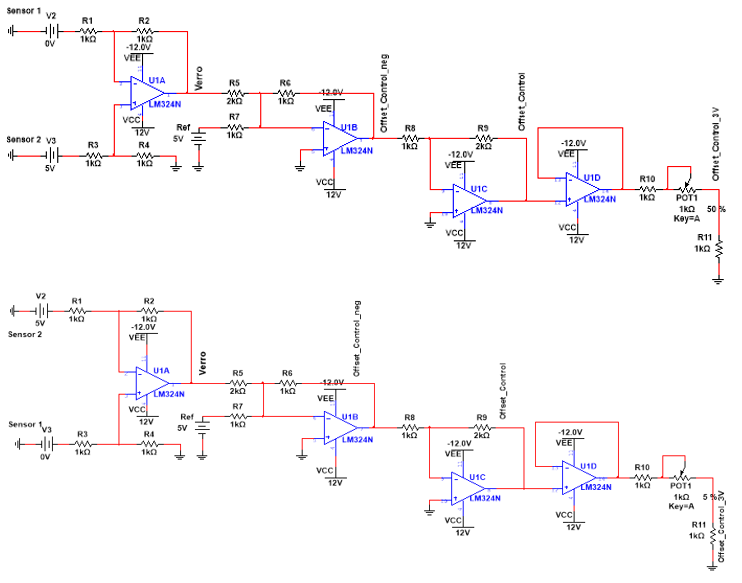


Figura 5 - Esquemático do gerador de onda PWM

A entrada deste circuito é na porta de *feedback.* Às entradas Rt e Ct ligou-se uma resistência de 12 kΩ e um condensador de 10 nF, respetivamente. Estes valores definem a frequência da onda triangular gerada. A saída tem uma resistência *pull-up* ligada a 5 V para que a amplitude da onda PWM gerada tenha este valor.

Para estabelecer a tensão de referência é necessário ter em conta alguns aspetos. O primeiro prende-se com os valores de leitura do sensor. Quando o robô se encontra sobre uma superfície branca os sensores medem, aproximadamente, 0 V. Já quando se encontra sobre uma superfície preta os sensores medem, sensivelmente, 5 V. Como o *array* de sensores usado é analógico, para as restantes cores os sensores apresentam valores entre estas duas gamas.

Outro aspeto importante está relacionado com a zona de funcionamento do motor. Segundo testes realizados, verificou-se que os motores arrancam quando o *duty cycle* do sinal PWM ronda os 50%. Já depois do robô se encontrar em andamento, o *duty cycle* necessário para manter o motor em funcionamento é ligeiramente inferior ao inicial. De salientar que estes valores forem obtidos com o motor em vazio, isto é, sem que a massa deste tivesse influência nas medições. Isto leva a que, na prática, os valores de *duty cycle* não sejam exatamente os previstos. Posto isto, a tensão de entrada deste circuito terá de ter, no mínimo, metade da amplitude da onda triangular que é gerada internamente pelo circuito integrado, para que o motor não pare por completo, evitando oscilações. O esquemático do circuito é o seguinte.



1º estágio

2º estágio

3º estágio

4º estágio

5º estágio

Figura 6 - Esquemático do circuito estabelece a tensão de comparação

Uma vez que cada motor tem de ser controlado individualmente, é necessário implementar dois circuitos tal como apresentado na figura acima. Estes circuitos apenas diferem no facto de as entradas de um estarem trocadas em relação às entradas do outro.

Quando nenhum sensor se encontra sobre a linha, as saídas dos circuitos serão iguais. Quando um sensor se aproxima da linha, a saída do circuito correspondente a este sensor diminui o valor de tensão e, a saída do outro aumenta, permitindo ao robô fazer a compensação do desvio da direção, causado pela diferença das saídas dos dois circuitos.

O primeiro estágio deste circuito é um subtrator e estabelece a diferença entre a tensão à entrada não inversora com a tensão à entrada inversora, ou seja, a diferença entre os dois sensores. Como cada um destes valores pode variar entre 0 V e 5 V, o valor da tensão de erro terá um valor máximo de 5 V e um valor mínimo de -5 V. Tendo em consideração as características de funcionamento do motor, acima descritas, é necessário que o valor mínimo à saída deste circuito seja aproximadamente metade da amplitude da onda triangular de comparação. Assim, é necessário somar uma tensão de *offset* a estes valores.

O segundo e terceiro estágios são, respetivamente, um somador inversor e um amplificador inversor. Em conjunto, estes dois estágios somam uma tensão de *offset* de 10 V à saída do primeiro estágio, fazendo com a que a tensão à saída do terceiro estágio tenha um valor compreendido entre 5 V e 15 V.

O quarto estágio é um seguidor de tensão e tem como objetivo isolar os dois circuitos.

O último estágio é um divisor de tensão. Permite estabelecer uma tensão de saída do circuito dentro dos valores pretendidos. Colocou-se um potenciómetro com o objetivo de permitir corrigir e compensar diferenças entre os resultados esperados e os resultados reais devido às tolerâncias dos componentes, bem como, conseguir um ajuste mais fino da velocidade.

Isolamento e tratamento dos valores dos sensores

Este circuito tem dois objetivos principais. O primeiro objetivo é isolar os valores lidos dos sensores do restante circuito, obtendo-se assim, valores mais fiáveis. O segundo objetivo é fazer um tratamento dos valores a ser usados pelo circuito digital. Num circuito digital apenas interessam dois tipos de valores: nível lógico alto e nível lógico baixo. Atendendo às características dos circuitos integrados utilizados é considerado nível lógico alto quando os valores de tensão atingem 2 V. Uma tensão deste valor corresponde a uma cor muito mais próxima do branco do que do preto, impossibilitando o robô de fazer o pretendido. Por este motivo, é necessário fazer um tratamento dos valores dos sensores que vão ser usados no circuito digital. Após passar o valor do sensor por um seguidor de tensão, introduz-se este valor num comparador não inversor fazendo com que o valor à saída deste comparador seja 0 V até um valor de entrada mais próximo do valor de referência da cor preta, e que seja 5 V a partir deste valor.

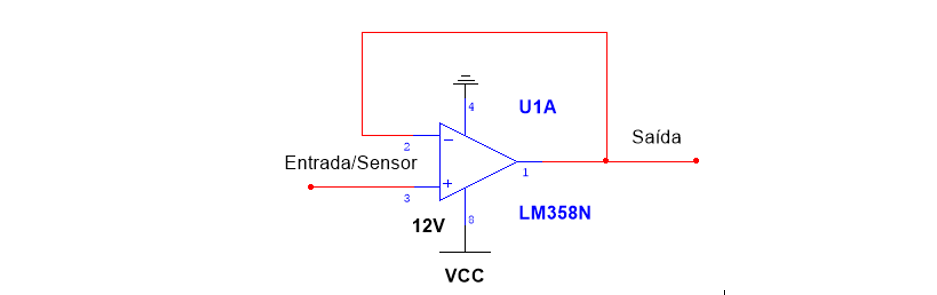


Figura 7 - Esquemático do circuito para os sensores a ser usados como componentes analógicos

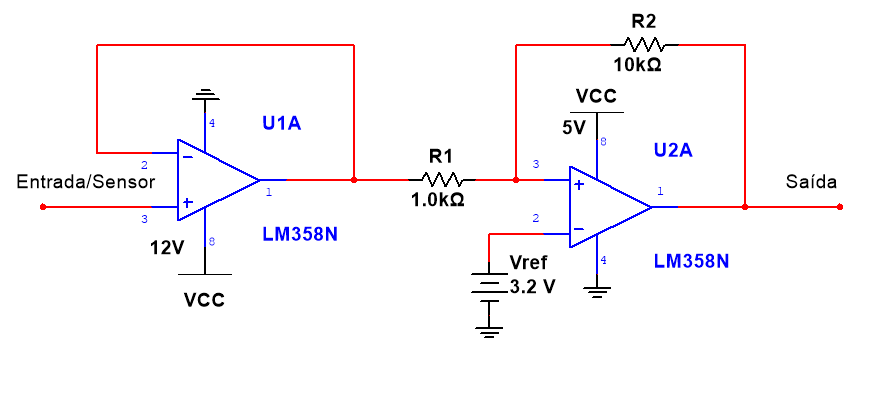


Figura 8 - Esquemático do circuito para os sensores a ser usados como componentes digitais

Fazendo e considerando o valor teórico de . Na prática, este valor será ligeiramente inferior, no entanto, não é um valor crítico para o funcionamente do circuito uma vez que, tal como referido anteriormente, o nível lógico alto dos circuitos integrados é atingido a partir de um valor de tensão de 2 V. Além do mais, as medidas dos sensores nunca são ideais. Os cálculos apenas são usados como referência. Após determinadas as resistências a usar e de estimado um valor para a tensão de referência, esta pode ser ajustada para um valor que se melhor adapta aos resultados práticos.

Fazendo tem-se que.

Fazendo bem que e , ou seja, a saída do comparador só fica a nível lógico alto quando a leitura do sensor passa os 4 V e apenas volta a nível lógico baixo quando a leitura do sensor desce dos 3 V.

Sistema eletrónico para o controlo dos motores

De forma a ser possível controlar o estado de funcionamento do robô, ou seja, fazê-lo seguir a linha, parar ou rodar 180 °, desenvolveu-se uma máquina de estados sequencial síncrona, mais especificamente uma máquina de Moore, uma vez que as suas saídas dependem apenas do estado atual.



Figura 9 - Máquina de Moore

Numa máquina de estados, o objetivo é determinar a “Lógica do Próximo Estado” e a “Lógica de Saída” do circuito. Como referido, a “Lógica de saída” depende apenas do estado atual. Já a “Lógica do Próximo Estado” depende do estado atual e das entradas do circuito.

Como referido, o objetivo do robô é seguir uma linha, devendo parar sobre uma linha horizontal colocada no final do percurso (Figura 10) e, nesse momento, o robô deve dar uma volta de 180 ° sobre o seu eixo, voltando a seguir a linha em direção ao ponto de partida, após o acionamento de um botão.



Figura 10 - Linha preta usada para os testes

Denominou-se a linha de final de curso como a linha horizontal e a linha determinante do percurso como linha vertical.

De modo a detetar a linha no final do percurso, usaram-se os dois sensores das pontas do *array* de sensores (S1 e S8, rodeados a amarelo na Figura 11). Usou-se também o sensor do meio (S5, rodeado a verde na Figura 11) de forma a interromper a rotação quando este encontrar a linha preta.



Figura 11 – Sensores usados na máquina de estados: S1, S5 e S8

Nesse momento, o robô poderá ser acionado através de um botão de pressão (B) (Figura 12) para que este volte a seguir a linha.



Figura 12 - Botão de pressão

Os motores são controlados por meio de um *driver* [3], podendo estar parados, a rodar no sentido horário ou no sentido anti-horário.

Para isso, na máquina de estados existem duas saídas por cada motor, sendo que uma controla o movimento no sentido horário e outra no sentido anti-horário.

Como o robô usa dois motores, existem quatro saídas, que quando estão ativas (nível lógico alto) despoletam as seguintes respostas:

• FWR – motor da direita roda no sentido horário (*Forward* *Right*);

• FWL – motor da esquerda roda no sentido horário (*Forward* *Left*);

• BWR – motor da direita roda no sentido anti-horário (*Backwards* *Right*);

• BWL – motor da esquerda roda no sentido anti-horário (*Backwards* *Left*).



Figura 13 - Diagrama da máquina de estados

Na figura seguinte, mostra-se o diagrama da máquina de estados, usada para controlar o modo de funcionamento do robô. FW representa os valores de FWR e FWL e BW representa os valores de BWR e BWL.

Inicialmente, no estado “A”, o robô está parado (FW = 0 e BW = 0), e espera pelo acionamento do botão para iniciar a marcha. Após o acionamento do botão, B a nível lógico alto, no estado “B”, o robô inicia a sua trajetória, seguindo a linha com os dois motores a rodar no sentido horário (FW = 1 e BW = 0). Quando o *array* de sensores estiver posicionado sobre a linha horizontal preta (fim do trajeto), os sensores S1 e S8 estarão a nível lógico alto, e ocorrerá a transição para o estado “C”. Neste estado, o robô inicia a rotação para a esquerda (FWR = 1, FWL = 0, BWR = 0 e BWL = 1) e não transita de estado enquanto os sensores S1 e S8 estiverem a nível lógico alto, isto é, enquanto o *array* de sensores estiver sobre a linha horizontal. Quando o *array* de sensores sair desta, S1 e S8 estarão a nível lógico baixo, avançando para o estado “D”, no qual o robô continua a rotação, mantendo as suas saídas, até que o sensor S5 esteja sobre a linha vertical, S5 a nível lógico alto. Assim posicionado, o robô volta ao estado inicial.

De forma a implementar a máquina de estados, de acordo com o diagrama de estados, determinou-se toda a lógica necessária, construindo a Tabela 2.

Tabela 2 - Tabela completa para determinação da lógica completa da máquina de estados



Adicionaram-se duas colunas que indicam os bits do estado atual (B1 e B0) e também as entradas da máquina de estados (S1, S5, S8 e B). Como existem quatro entradas, existem dezasseis (24) combinações possíveis para cada estado, o que resulta em sessenta e quatro possibilidades no total (24+2 = 26 = 64) que será o número de linhas da Tabela 2

A seguir, “E” representa o estado atual e “E\*” o estado seguinte. “B1\*” e “B0\*” representam os bits do estado seguinte. Estas colunas devem ser preenchidas de acordo com a tabela ao lado (Tabela 3).

Tabela 3 - Atribuição de estados



Sabendo que cada *flip-flop* permite a existência de dois estados e que existem quatro estados (A – D), selecionaram-se dois *flip-flops* do tipo J-K, que apresentam a tabela de excitação ao lado (Tabela 4). De acordo com esta tabela, foram preenchidas as colunas “J” e “K” para cada *flip-flop* (Ja e Ka, relativo a B0; Jb e Kb, relativo a B1). Para este efeito, determinou-se a lógica do próximo estado, em função das quatro entradas da máquina de estados, simplificada através de mapas de Karnaugh. Resolveram-se quatro mapas de Karnaugh (um para cada entrada J e K dos *flip-flops*) e obteve-se os resultados apresentados na Figura 14



Tabela 4 - Tabela de excitação do Flip-Flop JK



Figura 14 - Lógica do próximo estado

Adicionaram-se as colunas das saídas (FWR, FWL, BWR e BWL), as quais foram preenchidas de acordo com o estado atual. Para cada uma destas determinou-se a lógica de saída, que também foi simplificada através de mapas de Karnaugh. Os resultados obtidos foram os da Figura 15



Figura 15 - Lógica de saída

Para implementar as expressões lógicas calculadas foram usados integrados da família HCT [5]. Esta família de integrados apresenta elevada impedância de entrada, e dessa forma, não afetam a parte do circuito onde estão conectadas.

De acordo com a lógica determinada, desenhou-se o circuito que implementa a máquina de estados, o qual foi testado usando a ferramenta *MultiSim*, verificando-se a sua funcionalidade.

Lógica de próximo estado

Entradas



Lógica de saída

Saídas

Flip-Flops

Estado atual

Lógica de Entrada

Figura 16 - Desenho da máquina de estados

Tal como mostra a Figura 17, cada *flip-flop* J-K necessita de um sinal *clock* como entrada. O *clock* é um sinal em forma de onda quadrada com um *duty* *cycle* de, idealmente, 50 %. Dá-se uma transição entre estados sempre que, a cada pulso de *clock*, uma das entradas (J ou K) estiver a nível lógico alto.



Figura 17 - Esquema do Flip-Flop J-k

Para este efeito, implementou-se um oscilador usando o integrado NE555 numa montagem como multivibrador astável. O condensador C1 carrega até 2\*Vcc/3 através das resistências RA e RB (saída = 1). Quando C1 atinge este valor, irá descarregar por RB (saída=0), até uma tensão de Vcc/3 e o ciclo repete-se.



Figura 18 - Implementação do Oscilador

Através dos seguintes cálculos, dimensionaram-se os componentes para obter uma onda quadrada de frequência 1 kHz, *duty cycle* de 50 %, aproximadamente.

Para e :

Foram selecionados os valores standard das resistências (Série E24) e do condensador (Série E12) mais próximos dos obtidos:

A Figura 19 mostra a saída do circuito obtida em simulação com a ferramenta *TINA*.

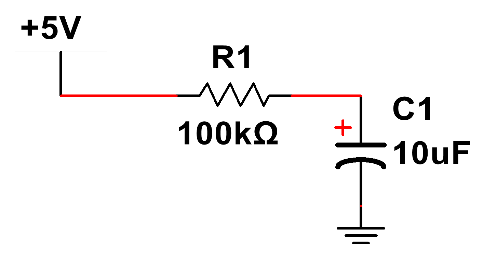
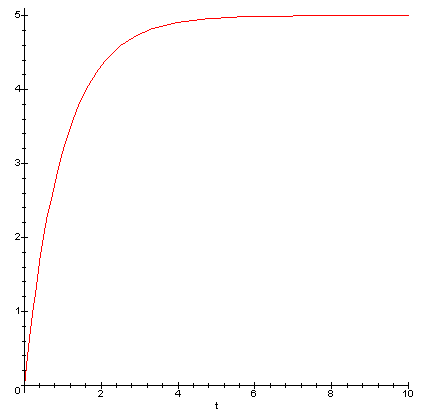


Figura 19 - Resposta do Oscilador de 1 kHz

As entradas Clear dos Flip-Flops J-K utilizados são ativas a nível lógico baixo, isto é, se ligarmos esta entrada ao *ground*, ocorrerá um *reset* no *flip-flop* e sua a saída Q virá a nível lógico baixo, independentemente das suas entradas J e K. Assim, se na máquina de estados as entradas *Clear* forem ligadas ao *GND*, B0 e B1 ficarão a nível lógico baixo.

Experimentalmente, verificou-se que, quando alimentados, os flip-flops mantêm as saídas do estado anterior. Para que o robô inicie no primeiro estado de funcionamento (estado “A”), utilizou-se um circuito para efetuar um *reset* aos *flip-flops*, tal como mostra a figura 10.

Figura 20 - Circuito RC de reset e a sua resposta no tempo.



Sabendo que a saída deste circuito estará a nível lógico baixo enquanto o condensador não carregar, a entrada *reset* dos *flip-flops* estará também a nível lógico baixo, o que resulta na ação de *reset*. Assim, logo que se alimentar a máquina de estados, os *flip-flops* apresentarão nas saídas “Q” o nível lógico baixo que representa o primeiro estado de funcionamento do robô. Após o condensador carregar, a entrada *reset* estará a nível lógico alto durante o resto do tempo de funcionamento, não provocando um *reset* indesejado.

Foi usado um botão de pressão para que o robô inicie a marcha. No entanto, sabe-se que, quando uma saída provém diretamente de um botão de pressão, esta é afetada por problemas de vibração mecânica (*bounce*) quando o contacto se fecha. De forma a garantir que a mudança de nível lógico se faça sem oscilações, torna-se necessário implementar um circuito de *debounce* como, por exemplo, o apresentado na Figura 21

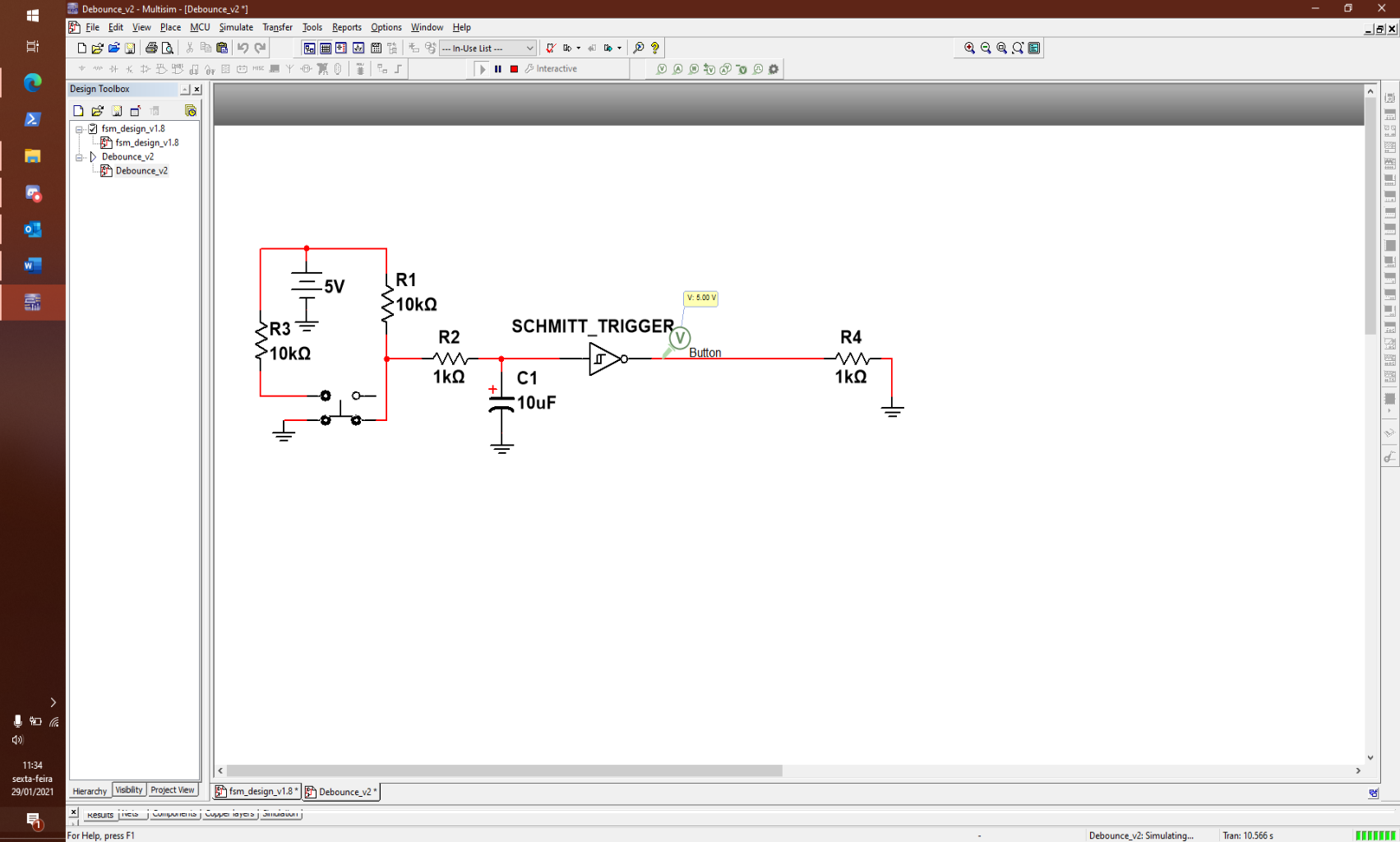


Figura 21 - Circuito de debounce do botão de pressão (quando premido)

O condensador em conjunto com a resistência funciona como filtro passa-baixo, que ajuda a reduzir o ruído elétrico provocado pelas vibrações mecânicas do botão. Assim, quando o botão não está pressionado, os terminais do condensador apresentarão 5 V e, quando pressionado, apresentarão 0 V. Uma vez que, num circuito sequencial, o nível lógico alto corresponde ao acionamento de uma entrada (botão, neste caso), será necessário utilizar uma lógica inversora na saída do condensado. Para isso, usou-se uma porta lógica Schmitt-Trigger (integrado 74HCT14 [5]) que funciona como um circuito comparador inversor com histerese centrado em 0 V. Assim, aumenta a imunidade ao ruído e, como inverte, a saída do circuito estará a nível lógico alto quando o botão for pressionado.

Simulação dos circuitos eletrónicos

Controlo dos motores

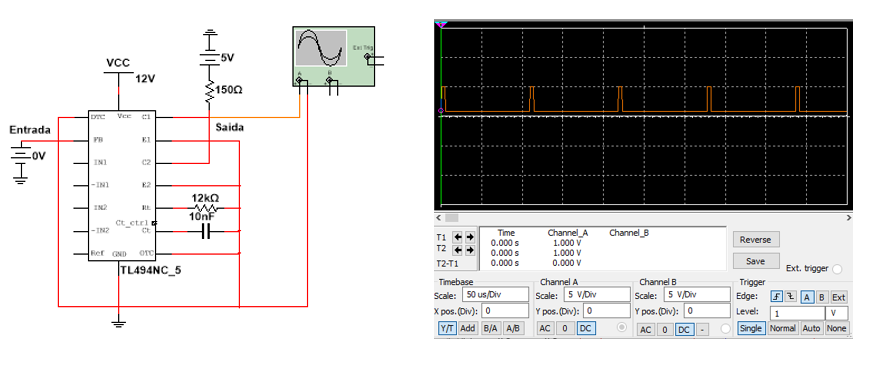
Tal como no desenho e explicação do circuito no capítulo anterior ir-se-á começar a simulação dos circuitos eletrónicos pela segunda parte do circuito, ou seja, pelo circuito que gera a onda PWM quando lhe é fornecida uma tensão de entrada.

Figura 22 - Circuito gerador PWM com entrada de 0 V

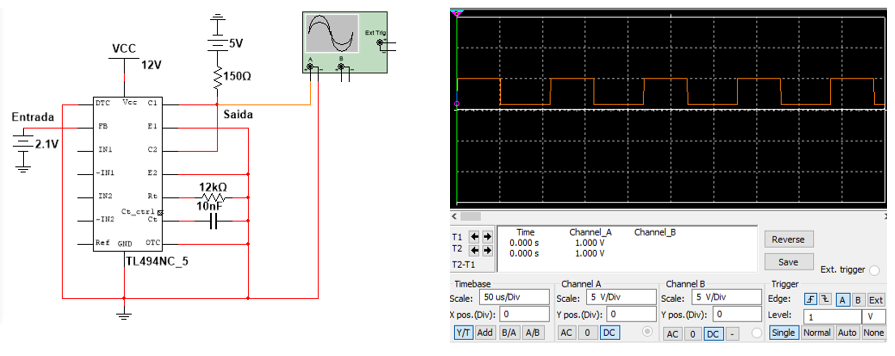


Figura 23 - Circuito gerador PWM com entrada de 2,1 V

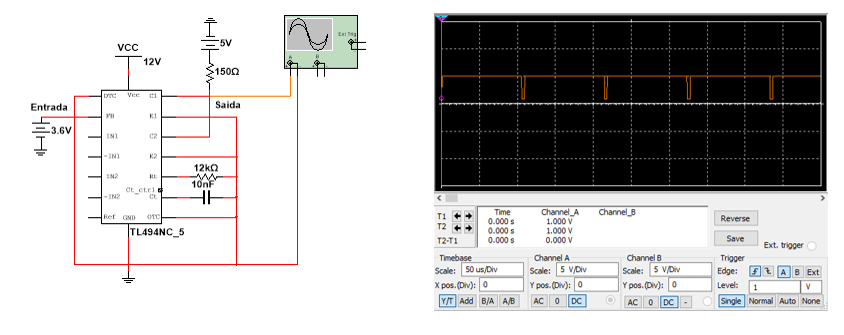
 Através das simulações acima pode-se concluir que quanto maior a tensão de entrada maior o *duty cycle.* Quando a entrada é zero o *duty cycle* é aproximadamente zero. Quando é próxima da amplitude da onda triangular interna do circuito integrado TL494 o *duty cycle* é praticamente 100 %.

Figura 24 - Circuito gerador PWM com entrada de 3,6 V

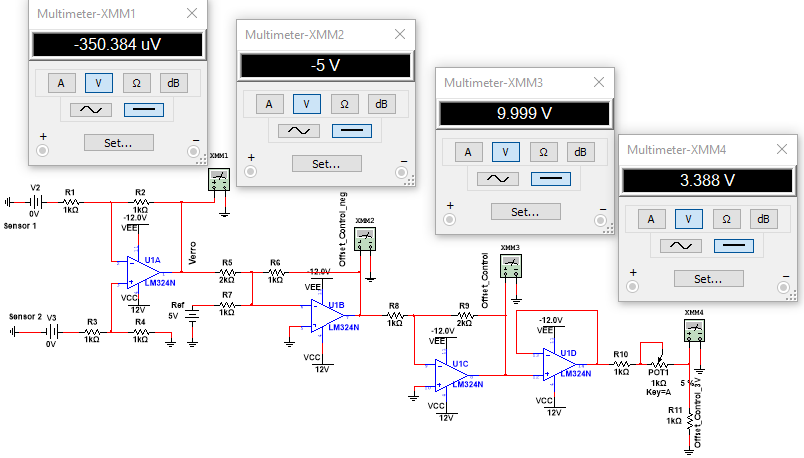
Apresenta-se de seguida as simulações do circuito que geram a tensão de entrada a ser usada no circuito acima.

Figura 25 - Circuito gerador da tensão de comparação com ambas as entradas a 0 V

Como podemos ver na figura acima, quando nenhum dos sensores está sobre a linha preta a tensão de saída tem um valor próximo do máximo o que implicará um *duty cycle* próximo dos 100% e o robô seguirá a mesma trajetória.

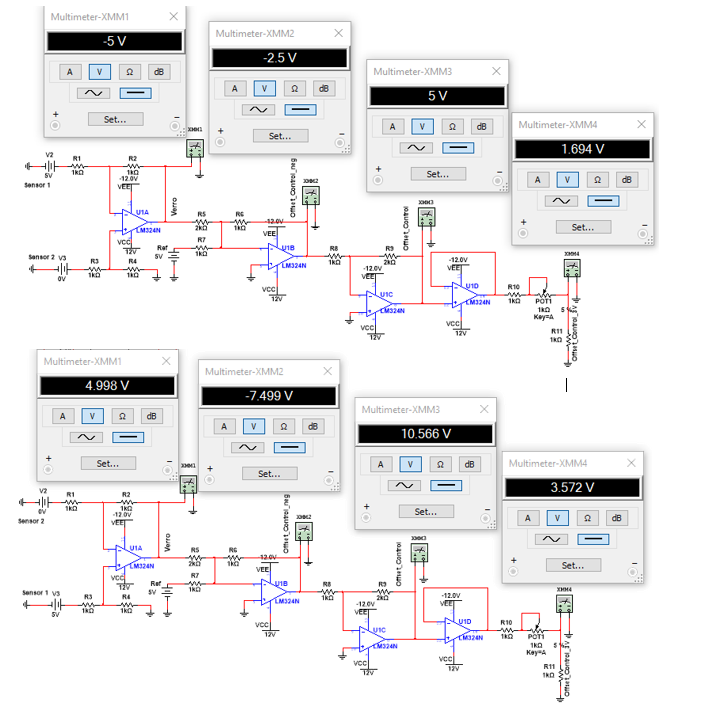


Figura 26 - Circuito gerador da tensão de comparação com entrada do sensor 1 a 5 V e entrada do sensor 2 a 0 V

Quando um dos sensores se aproxima da linha preta – sensor 1 neste caso – o motor que se encontra do lado deste sensor terá de abrandar para que o robô se desvie da linha. Como se pode verificar, o valor da tensão de saída do primeiro circuito desce e o motor tende a abrandar. No lado oposto, como as entradas dos circuitos estão trocadas, verifica-se o contrário. O motor terá de acelerar (ou manter a velocidade) para que se aproxime da linha. É de referir que a saída *offset control* tem um valor de 10,566 V uma vez que que se encontra limitado pela saturação do **Amp**. **Op**. O valor esperado seriam 15 V, no entanto em nada interfere com o normal funcionamento.

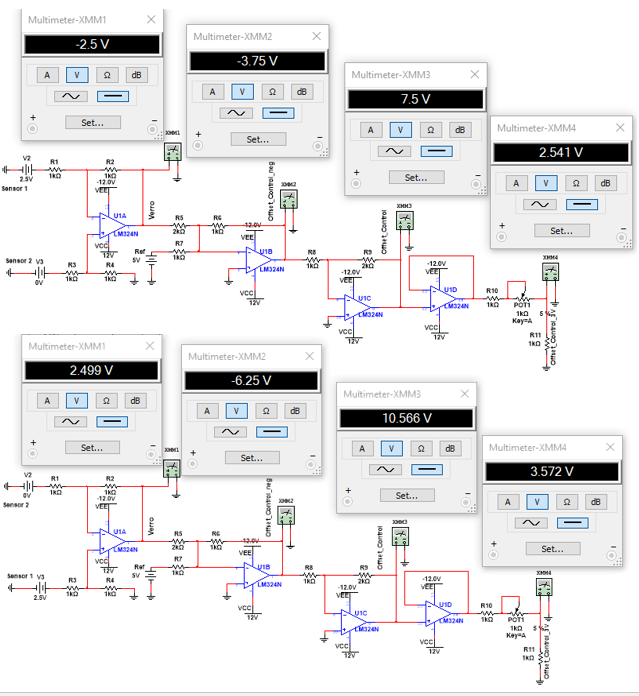


Figura 27 - Circuito gerador da tensão de comparação com entrada do sensor 1 a 2,5 V e entrada do sensor 2 a 0 V

Este caso demonstra uma situação mais real, visto que o circuito corrige antes de chegar a situações de valores extremos, e pode ser analisado como no último caso.

Sistema eletrónico para o controlo dos motores

Foi feito o teste em simulação do circuito da máquina de estados através da ferramenta *MultiSim*. Os resultados obtidos apresentam-se nas figuras abaixo. Verificou-se, portanto, a boa funcionalidade do circuito dimensionado.



Figura 28 - Estado 0 (Parado)

Figura 29 - Estado 1 (Anda para a frente) – Botão foi pressionado



Figura 30 - Estado 2 (Dá a volta para a esquerda) – S1 e S8 a nível lógico alto



Figura 31 - Estado 3 (Dá a volta) - S1 e S8 a nível lógico baixo



Figura 32 - Estado 0 (Parado novamente) - S5 = 1



Desenho da implementação dos circuitos eletrónicos

Introdução

Todos os circuitos acima descritos serão implementados em placas de circuito impresso. Para o efeito fez-se uso de duas ferramentas: o *PADS LOGIC* e *PADS LAYOUT*. Para simular o design da PCB em 3D utilizou-se a ferramenta KiCad.

Controlo dos motores

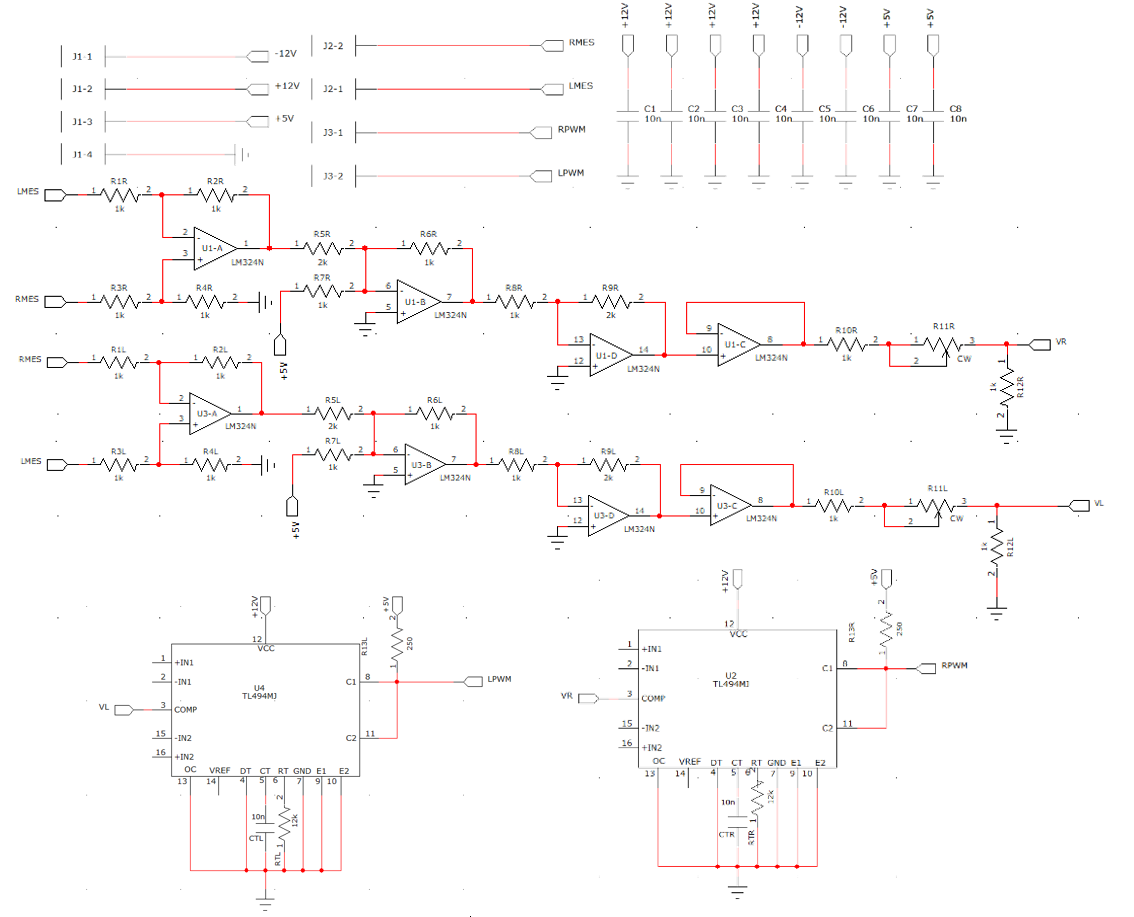


Figura 33 - Esquemático PCB controlo dos motores

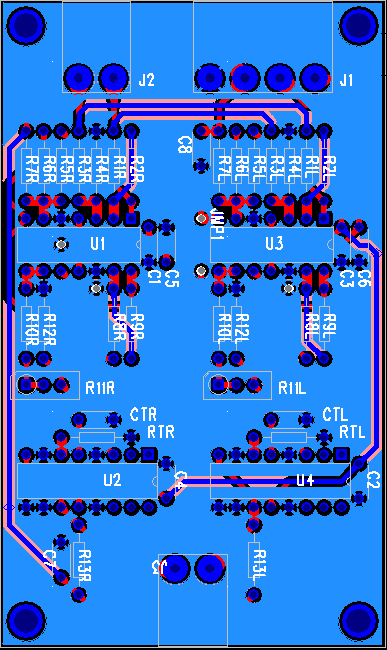


Figura 34 - Layout PCB controlo dos motores

Figura 35 - Vista bottom PCB controlo dos motores

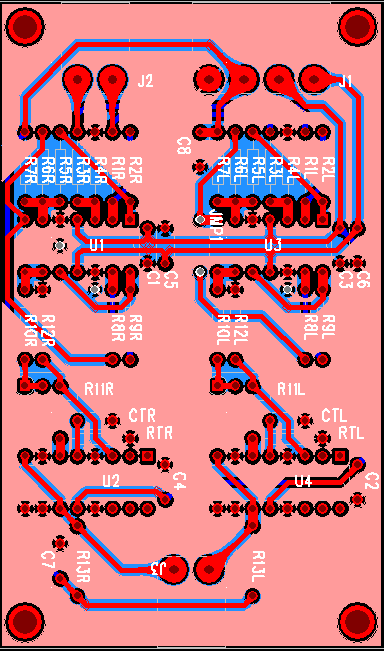


Figura 36 - Vista top PCB controlo dos motores



Figura 37 - Vista 3D PCB controlo dos motores

Sensores

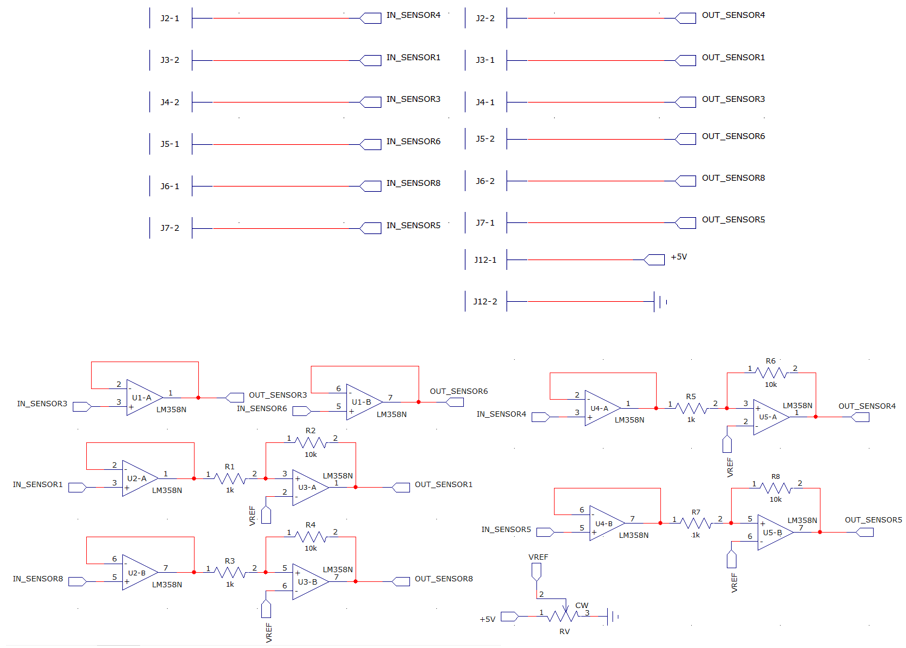


Figura 38 - Esquemático PCB sensores



Figura 39 - Layout PCB sensores

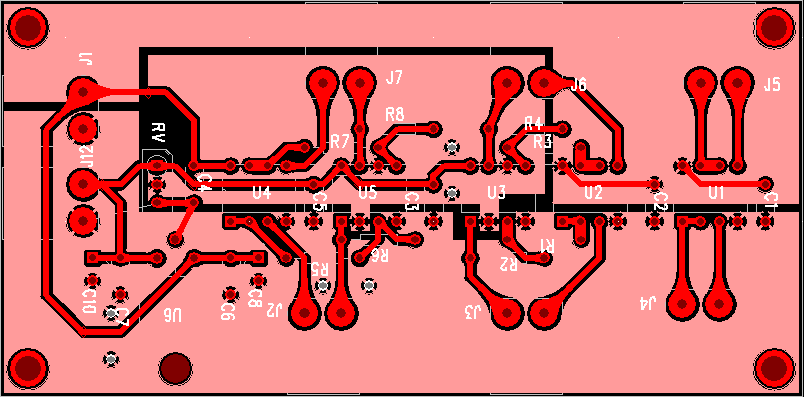


Figura 40 - Vista bottom PCB sensores



Figura 41 - Vista 3D PCB sensores

Máquina de estados

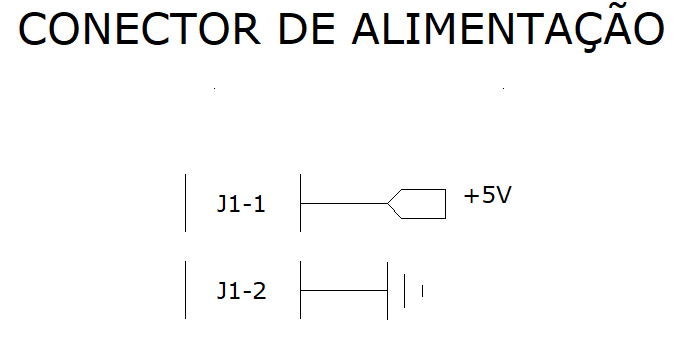
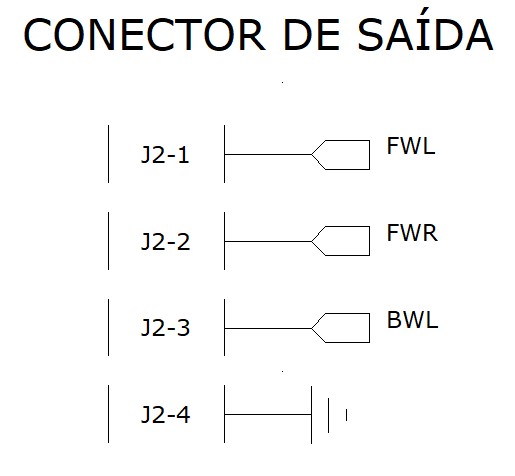
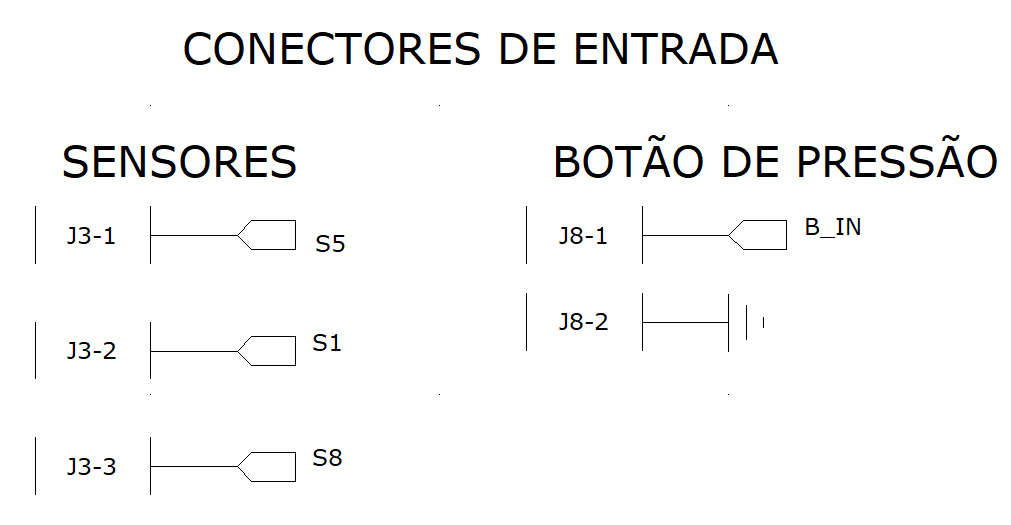


Figura 42 - Esquemático PCB Máquina de Estados



Figura 43 - Layout PCB Máquina de Estados

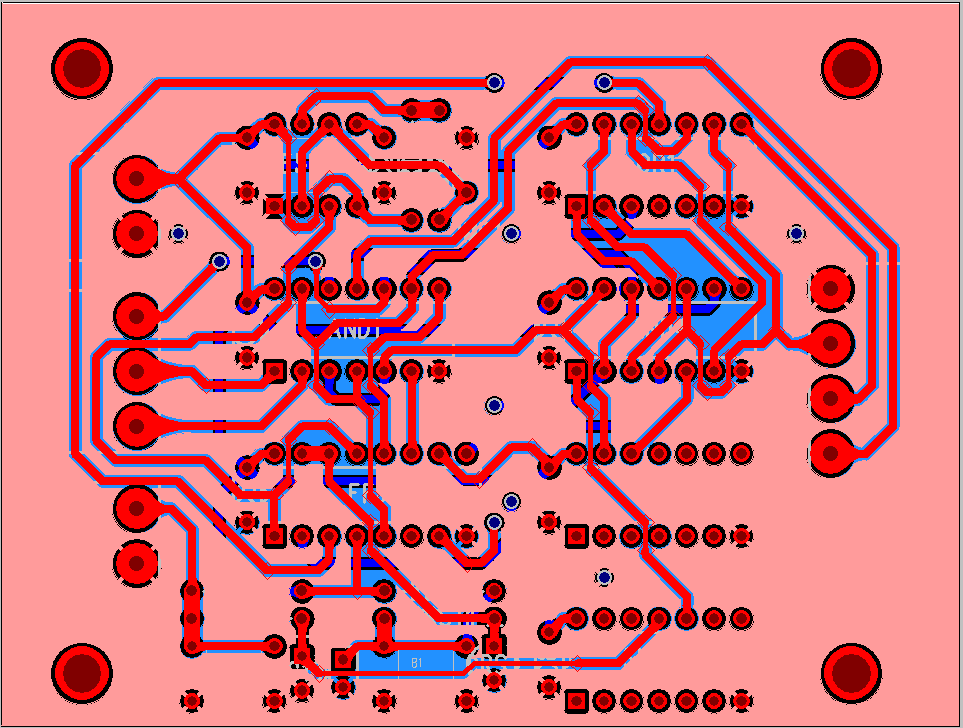


Figura 44 - Vista bottom PCB máquina de estados



Figura 45 - Vista top PCB máquina de estados



Figura 46 - Vista 3D PCB máquina de estados

Lista de componentes

Tabela 5 - Lista de Componentes

| Material | | Loja | Quantidade | Preço unitário | Preço total |
| --- | --- | --- | --- | --- | --- |
| Motor Bot'n Roll ONE 300rpm |  | Bot'n Roll | 2 | 17,50 € | 35,00 € |
| Par de rodas Bot'n Roll ONE A |  | Bot'n Roll | 1 | 11,50 € | 11,50 € |
| Par de hubs Bot'n Roll ONE A |  | Bot'n Roll | 1 | 4,50 € | 4,50 € |
| Rodas Livres  (D25MM) |  | Leroy Merlin | 2 | 1,69 € | 3,38 € |
| Placa *driver* de motores L298N |  | Bot'n Roll | 1 | 12,90 € | 12,90 € |
| QTR-8A |  | Bot'n Roll | 1 | 19,90 € | 19,90 € |
| Suporte para uma pilha 18650 c/fios |  | Bot'n Roll | 6 | 0,85 € | 5,10 € |
| BMS para proteção baterias 18650 3S 12,6V 20A |  | Bot'n Roll | 2 | 5,50 € | 11,00 € |
| Pilha LI-ION 18650 3,7V 2200mAh 18X65mm - 22A MAX |  | Bot'n Roll | 6 | 3,90 € | 23,40 € |
| LM7805 |  | Bot'n Roll | 1 | 0,50 € | 0,50 € |
| TL494 |  | Aquário | 2 | 1,70 € | 3,40 € |
| LM324 | Texas Instruments LM324 LM324N DIP14 Quadruple Operational Amplifier DIP14  1 Pack: Amazon.com: Industrial & Scientific | Aquário | 2 | 1,95 € | 3,90 € |
| LM358 | TEXAS INSTRUMENTS LM358P Lm358 Dual Operational Amplifier, For Electronics,  8 Pin Dip, Rs 5 /piece | ID: 21935084862 | Aquário | 5 | 0,74 € | 3,70 € |
| 74HCT04 | DIP IC LM324L / NE555L / LM358L / LM339L / MC1458L / MC4558L / MC34063AL at  Rs 3.90/piece | Integrated Circuits | ID: 21268048588 | Mauser | 1 | 0,92 € | 0,92 € |
| 74HCT08 | DIP IC LM324L / NE555L / LM358L / LM339L / MC1458L / MC4558L / MC34063AL at  Rs 3.90/piece | Integrated Circuits | ID: 21268048588 | Mauser | 2 | 0,90 € | 1,80 € |
| 74HCT14 | DIP IC LM324L / NE555L / LM358L / LM339L / MC1458L / MC4558L / MC34063AL at  Rs 3.90/piece | Integrated Circuits | ID: 21268048588 | Mauser | 1 | 0,61 € | 0,61 € |
| 74HCT32 | DIP IC LM324L / NE555L / LM358L / LM339L / MC1458L / MC4558L / MC34063AL at  Rs 3.90/piece | Integrated Circuits | ID: 21268048588 | Mauser | 1 | 0,48 € | 0,48 € |
| 74HCT112 |  | Mauser | 1 | 1,32 € | 1,32 € |
| NE555P |  | Aquário | 1 | 0,32 € | 0,32 € |
| Bloco Terminal 2 Pinos |  | Aquário | 17 | 0,80 € | 13,60 € |
| Bloco Terminal 3 Pinos | Terminal bloco conector comprar on-line da China Factory | Aquário | 1 | 1,00 € | 1,00 € |
| Potenciómetro *trimmer* multivolta |  | Aquário | 2 | 2,20 € | 4,40 € |
| Resistência |  | Bot'n Roll | 34 | 0,05 € | 1,70 € |
| Condensador Cerâmico |  | Aquário | 28 | 0,10 € | 2,80 € |
| Condensador eletrolítico |  | Aquário | 4 | 0,25 € | 1,00 € |
| Botão de pressão |  | Bot'n Roll | 2 | 0,15 € | 0,30 € |
| Botão de painel |  | Bot'n Roll | 1 | 0,60 € | 0,60 € |
| Bloco terminal 12 conetores |  | Gmlux | 3 | 0,40 € | 1,20 € |
| Alumínio 2 mm | Alumínio: Chapa de ALUMÍNIO em fms.pt | **-** | **-** | **-** | 35,00 € |
| Placa madeira | Placa de contraplacado 2500x1220x10mm - Leroy Merlin Portugal | **-** | 1 | **-** | 5,00 € |
| Placa de acrílico policarbonato |  | **-** | 1 | - | 10,00 € |
| Parafuso 5 mm |  | Ferritrofa | 28 | 0,05 € | 1,40 € |
| Parafuso 3 mm |  | Ferritrofa | 12 | 0,05 € | 0,60 € |
| Porca com asas |  | Ferritrofa | 6 | 0,10 € | 0,60 € |
| Porca 5 mm |  | Ferritrofa | 22 | 0,05 € | 1,10 € |
| Rebites |  | Ferritrofa | 18 | 0,10 € | 1,80 € |
| *Jumpers* |  | Bot'n Roll | *pack* | 2,00 € | 2,00 € |

Desenho do encapsulamento

Mecanicamente o robô é composto por:

* Duas rodas motoras na sua parte central;
* Duas rodas livres com altura ajustável nas partes frontal e traseira;
* Suporte com altura ajustável para o *array* de sensores na parte frontal;
* Botão de painel de interface com utilizador na parte lateral;
* Placa de acrílico policarbonato removível na parte superior;
* Placa de madeira isoladora elétrica para suporte dos circuitos.

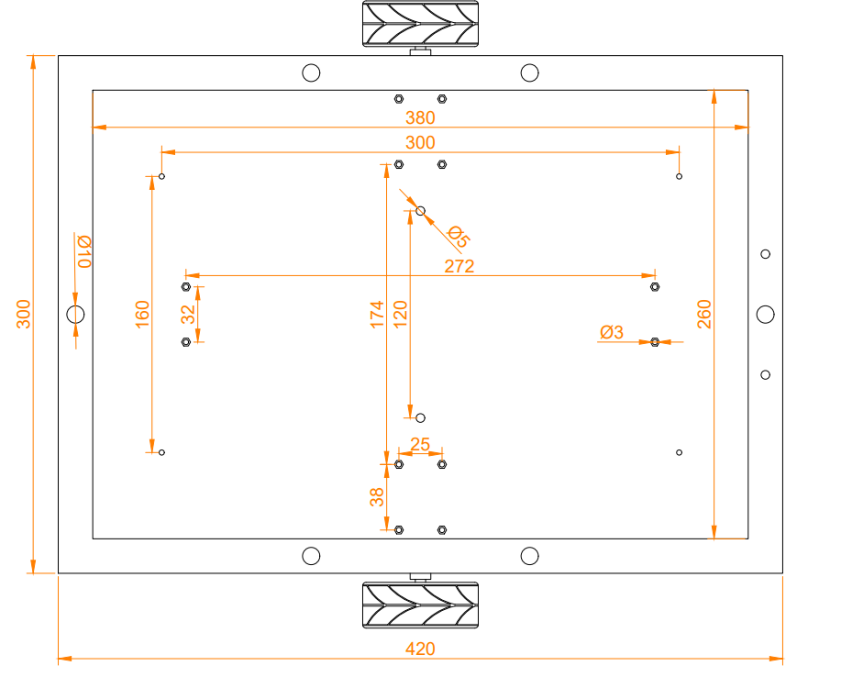


Figura 47 - Vista superior do robô

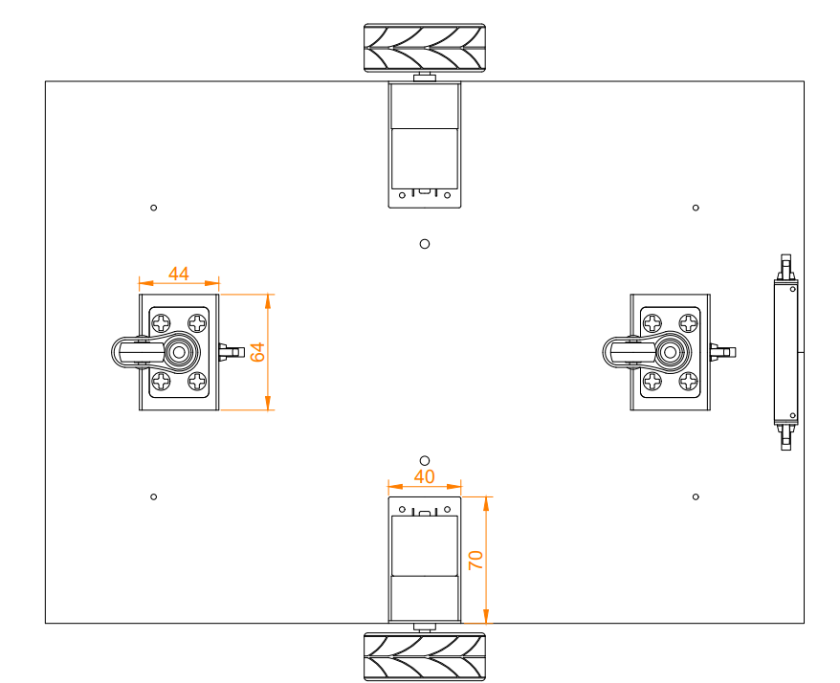


Figura 48 - Vista inferior do robô

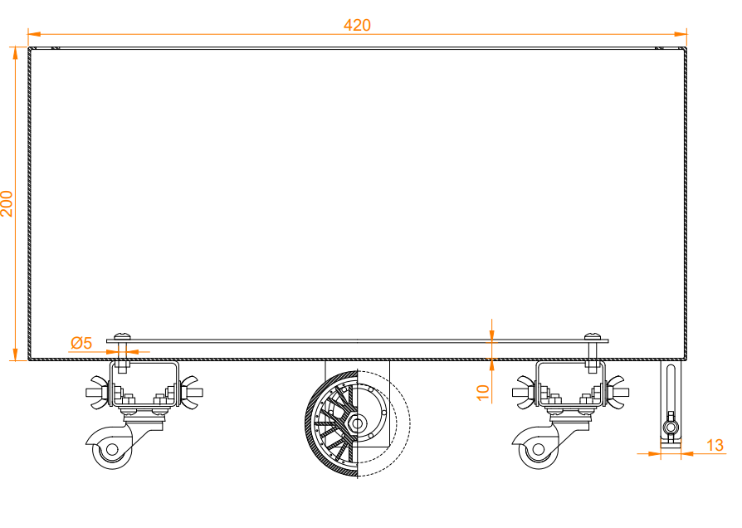


Figura 49 - Vista lateral do robô

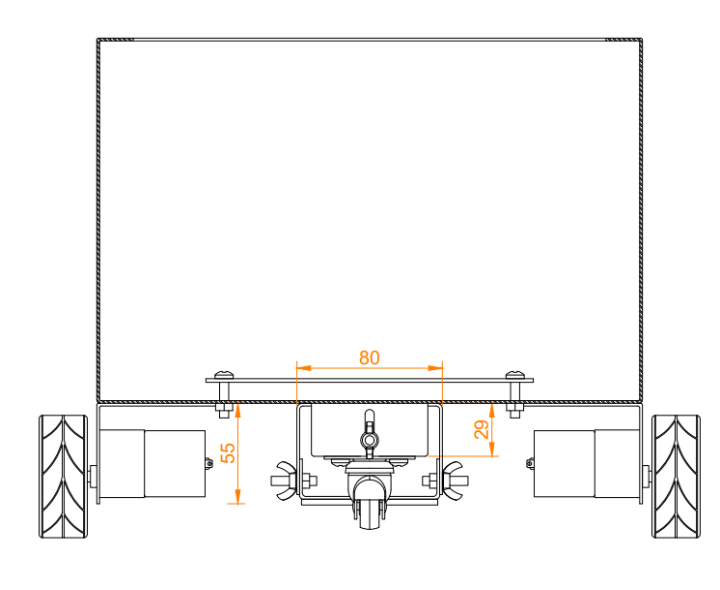


Figura 50 - Vista traseira do robô

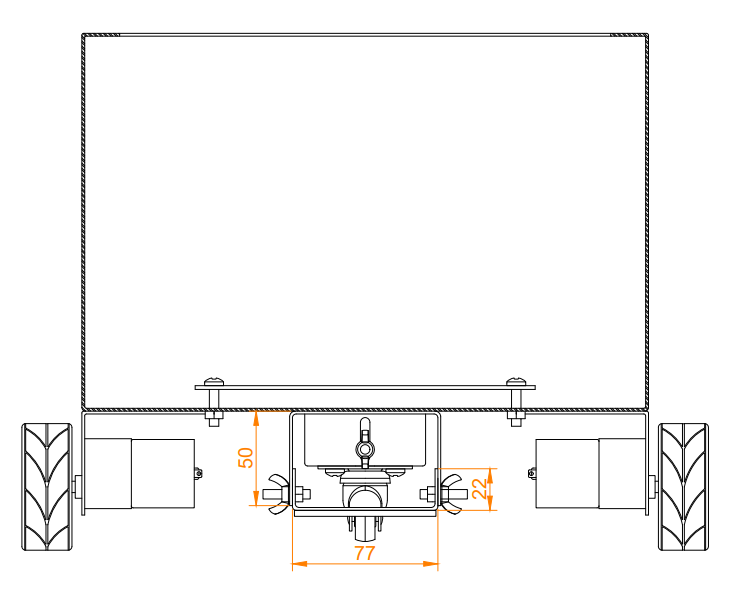


Figura 51 - Vista frontal do robô

Fotos Reais

Possíveis opções alternativas para o desenho dos circuitos

Todos os projetos são passíveis de melhorias. Sendo um projeto totalmente analógico, algumas melhorias não terão uma implementação tão simples e barata quando comparadas com uma implementação digital.

Tal como implementado, caso o robô saia fora da rota, não existe nenhum mecanismo de deteção do sucedido. Uma possível melhoria para a resolução deste problema seria usar um ou dois dos sensores centrais do *array* de sensores para deteção da linha. Caso esta não fosse detetada, o robô teria de parar, evitando que este ande indefinidamente.

Outro aspeto a melhorar seria a implementação de um mecanismo de deteção de obstáculos. Caso o robô se encontre sobre a linha e um objeto obstrua a sua passagem, este deve parar de forma a evitar a colisão e possíveis danos materiais de ambas as partes. Além disso, o robô deve emitir um som intermitente de forma a alertar o responsável pelo sucedido. Este mecanismo poderia ser implementado através de um sensor de proximidade colocado na dianteira do robô. Sem este sistema, o robô embaterá no objeto, provocando um comportamento indefinido.

Tendo em conta que o objetivo deste robô é minimizar os contactos possivelmente infeciosos com outras pessoas, seria relevante implementar uma interface diferente com o utilizador. Ao invés de fazer uso de um botão de pressão para iniciar a marcha, seriam usadas células de carga que detetam a colocação de um tabuleiro com alimentos sobre o robô. Assim que o utilizador ou operador colocassem o tabuleiro sobre o robô, este iria iniciar a marcha, minimizando os contactos com a armadura do robô. Além disso, poderia ser desenvolvido um sistema automático de desinfeção das superfícies do robô, que deverá ser acionado após finalizar o seu percurso (ida e volta).

Como o robô funciona à base de baterias, estas deveriam ser carregadas numa estação de carregamento. Deste modo, colocar-se-ia o robô na sua estação de carregamento sempre que não estivesse em utilização.

Tal como foi visto, apontaram-se apenas algumas formas de melhorar o trabalho desenvolvido. Possivelmente, existirão outras melhorias que poderiam ser aplicadas e que não foram abordadas neste capítulo. No entanto, algumas seriam descartadas por não justificarem o trabalho, terem uma execução complexa para o efeito ou não serem economicamente viáveis.

Evolução do atual desenho para um sistema baseado em microcomputador

O atual desenho do robô poderia ser implementado num sistema baseado em microcomputador, tal como irá ser visto na unidade curricular de Laboratórios e Práticas Integradas II (LPI II).

Como o objetivo do robô é fazer chegar alimentos a alguém que se encontre numa situação de isolamento, o seu aspeto físico será idêntico ao implementado. Deverão diferir apenas no posicionamento e suporte para o microcomputador e novos sensores.

Através do uso de um microcomputador, existirão múltiplas possibilidades de implementação. Uma delas seria o uso do controlo remoto, em que uma pessoa responsável controla o percurso do robô até ao local pretendido. Ao robô teria de ser adicionada uma câmara e a sua imagem deveria ser transmitida para o comando que o utilizador estivesse a utilizar. Este sistema poderia ser implementado através da tecnologia *Bluetooth*, em que o robô seria controlado por um *smartphone*. Outra possibilidade, seria a criação de rotas pré-definidas para o robô seguir. Fazendo uso de sensores de distância, o robô poderia desviar-se de obstáculos que se encontrassem na sua rota. A solução que mais se assemelha à implementação atual, seria fazer o uso do microcomputador para seguir uma linha preta, através do *array* de sensores (QTR Citação) e implementar as funcionalidades referidas no capítulo anterior.

Neste capítulo apenas foram apresentadas algumas possibilidades, sem idealizar a sua implementação ou grau de complexidade. Assim, no próximo semestre, na unidade curricular de LPI II, será avaliada a viabilidade de todas as ideias acima referidas.

# Referências

|  |  |
| --- | --- |
| [1] | SNS - Serviço Nacional de Saúde, “Covid-19 | Pandemia,” 11 março 2020. [Online]. Available: https://www.sns.gov.pt/noticias/2020/03/11/covid-19-pandemia/. [Acedido em 16 dezembro 2020]. |
| [2] | J. D'Onfro, “Robots To The Rescue: How High-Tech Machines Are Being Used To Contain The Wuhan Coronavirus,” 2 fevereiro 2020. [Online]. Available: https://www.forbes.com/sites/jilliandonfro/2020/02/02/robots-to-the-rescue-how-high-tech-machines-are-being-used-to-contain-the-wuhan-coronavirus/?sh=73364f201779. [Acedido em 16 dezembro 2020]. |
| [3] | STMicroelectronics, “DUAL FULL-BRIDGE DRIVER”. Patente L298 datasheet, janeiro 2000. |
| [4] | T. Instruments, “TL494 Pulse-Width-Modulation Control Circui”. Patente TL494 datasheet, janeiro 1983. |
| [5] | Texas Instruments, “SN54/74HCT CMOS Logic Family Applications and Restrictions,” maio 1996. [Online]. Available: https://www.ti.com/lit/an/scla011/scla011.pdf?ts=1612201599681&ref\_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FCD74HCT251. [Acedido em 2 fevereiro 2021]. |
| [6] | P. Semiconductors, “74HCT14 Hex inverting Schmitt trigger”. Patente 74HCT14, 30 outubro 2003. |

[1] SNS - Serviço Nacional de Saúde, “Covid-19 | Pandemia,” 11 março 2020. [Online]. Available: https://www.sns.gov.pt/noticias/2020/03/11/covid-19-pandemia/. [Acedido em 16 dezembro 2020].

[2] J. D'Onfro, “Robots To The Rescue: How High-Tech Machines Are Being Used To Contain The Wuhan Coronavirus,” 2 fevereiro 2020. [Online]. Available: https://www.forbes.com/sites/jilliandonfro/2020/02/02/robots-to-the-rescue-how-high-tech-machines-are-being-used-to-contain-the-wuhan-coronavirus/?sh=73364f201779. [Acedido em 16 dezembro 2020].

[3] STMicroelectronics, “DUAL FULL-BRIDGE DRIVER”. L298 datasheet, Jan. 2000.

[4] T. Instruments, “TL494 Pulse-Width-Modulation Control Circui”. TL494 datasheet, janeiro 1983 [revisto março 2017].

[5] Texas Instruments, “SN54/74HCT CMOS Logic Family Applications and Restrictions,” maio 1996. [Online]. Available: https://www.ti.com/lit/an/scla011/scla011.pdf?ts=1612201599681&ref\_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FCD74HCT251. [Acedido em 2 fevereiro 2021].

[6] P. Semiconductors, “74HCT14 Hex inverting Schmitt trigger”. 74HCT14 datasheet, 26 agosto 1997[revisto 30 outubro 2003].