|  |
| --- |
| 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  **DWR-19**  ***Digital Waiter Robot*** | |
| Projeto integrador  Laboratórios e Práticas Integradas  Trabalho realizado sob a orientação do  **Professor Luís Barros** | |
| 23 junho 2021 | |

**Índice**

[Lista de Figuras 2](#_Toc74931473)

[Lista de Tabelas 2](#_Toc74931474)

[Acrónimos e Siglas 2](#_Toc74931475)

[Capítulo 1 Introdução 2](#_Toc74931476)

[1.1 Introdução 2](#_Toc74931477)

[1.2 Enquadramento 2](#_Toc74931478)

[1.3 Especificações Previstas 2](#_Toc74931479)

[1.3.1 Especificações funcionais 2](#_Toc74931480)

[1.3.2 Especificações técnicas 2](#_Toc74931481)

[1.4 Planeamento 2](#_Toc74931482)

[Capítulo 2 Arquitetura e Módulos Utilizados 2](#_Toc74931483)

[2.1 Introdução 2](#_Toc74931484)

[2.2 Sensores 2](#_Toc74931485)

[2.2.1 *Array* de Sensores de reflexão 2](#_Toc74931486)

[2.2.2 Sensor de obstáculos 2](#_Toc74931487)

[2.3 Driver 2](#_Toc74931488)

[2.4 Módulos 2](#_Toc74931489)

[2.4.1 *Radio Frequency Identification* (RFID) 2](#_Toc74931490)

[2.4.2 Bluetooth 2](#_Toc74931491)

[*2.5* Microcontrolador e *Shield* 2](#_Toc74931492)

[2.6 Circuito de Alimentação 2](#_Toc74931493)

[Capítulo 3 Implementação em Software 2](#_Toc74931494)

[3.1 Introdução 2](#_Toc74931495)

[3.2 Periféricos 2](#_Toc74931496)

[3.2.1 *Direct Memory Acess* (DMA) 2](#_Toc74931497)

[3.2.2 *Analog to Digital Converter* (ADC) 2](#_Toc74931498)

[*3.2.3* *TIMER* 2](#_Toc74931499)

[3.2.4 *Serial Peripheral Interface* (SPI) 2](#_Toc74931500)

[3.2.5 *Universal Synchronous/Asynchronous Receiver/Transmitter* (USART) 2](#_Toc74931501)

[3.3 Descrição de *Software* e Módulos Criados 2](#_Toc74931502)

[3.4 Simulações 2](#_Toc74931503)

[Capítulo 4 Lista de Componentes 2](#_Toc74931504)

[Capítulo 5 Circuito Mecânico Implementado 2](#_Toc74931505)

[Capítulo 6 Resultados Experimentais 2](#_Toc74931506)

[Capítulo 7 Análise do Produto 2](#_Toc74931507)

[7.1 Introdução 2](#_Toc74931508)

[7.2 Fiabilidade 2](#_Toc74931509)

[7.3 Segurança 2](#_Toc74931510)

[7.4 Certificação 2](#_Toc74931511)

[Capítulo 8 Conclusões 2](#_Toc74931512)

[8.1 Conclusão 2](#_Toc74931513)

[8.2 Sugestões de Trabalho Futuro 2](#_Toc74931514)

[8.2.1 Possíveis Opções Alternativas para o Desenho dos Circuitos 2](#_Toc74931515)

[8.2.2 Evolução do Atual Desenho para um Sistema Baseado em Microcomputador 2](#_Toc74931516)

[Referências 2](#_Toc74931517)

Lista de Figuras

[Figura 1.1 - Diagrama de Gantt do planeamento inicial. 2](#_Toc74931602)

[Figura 1.2 - Diagrama de Gantt do planeamento revisto. 2](#_Toc74931603)

[Figura 2.1 - *Array* de sensores QTR-8A. 2](#_Toc74931604)

[Figura 2.2 - Módulo Driver L298N. 2](#_Toc74931605)

[Figura 3.1 - Exemplo de transferência do DMA. 2](#_Toc74931606)

[Figura 3.3 - Configurações possíveis do DMA. 2](#_Toc74931607)

[Figura 3.4 - conversor por aproximação sucessiva. 2](#_Toc74931608)

[Figura 3.5 - Modo de funcionamento independente (a) Single-channel, single conversion mode; (b) Multichannel, single conversion mode; (c) Single-channel, continuous conversion mode; (d) Multichannel, continuous conversion mode. 2](#_Toc74931609)

[Figura 3.6 – Comparação das características do *timer.* 2](#_Toc74931610)

[Figura 5.1 - Desenho (à esquerda) e imagem real (à direita) do DWR (a) vista superior; (b) vista inferior; (c) vista lateral direita; (d) vista dianteira; (e) vista traseira. 2](#_Toc74931611)

[Figura 7.1 - Marcação CE. 2](#_Toc74931612)

[Figura 7.2 - Símbolos de perigo: (a) perigoso para o ambiente; (b) corrosivo; (c) comburente; (d) inflamável; (e) explosivo; (f) tóxico; (g) vários perigos; (h) *Eletrostatic Sensitive Device - ESD*. 2](#_Toc74931613)

[Figura 7.3 - Símbolo WEEE. 2](#_Toc74931614)

Lista de Tabelas

[Tabela 2.1 - Sensores e sua utilização. 2](#_Toc74931615)

[Tabela 2.2 - Tabelas de verdade do *driver* L298N (a) Controlo do Motor A; (b) Controlo do Motor B. 2](#_Toc74931616)

[Tabela 3.1 - Mapeamento dos *timers*. 2](#_Toc74931617)

[Tabela 4.1 - Lista de componentes. 2](#_Toc74931618)

[Tabela 6.1 - *Duty cycle* do sinal PWM de saída em função da tensão de entrada. 2](#_Toc74931619)

[Tabela 7.1 - Tempo de vida dos componentes com maior probabilidade de falha, usados no AWR. 2](#_Toc74931620)

[Tabela 8.1 - Número de horas despendidas por elemento 2](#_Toc74931621)

Acrónimos e Siglas

|  |  |
| --- | --- |
| **Acrónimo/Sigla** | **Significado** |
| DWR | *Digital Waiter Robot*  Robô Empregado de Mesa Digital |
| LED | *Light Emitting Diode*  Díodo Emissor de Luz |
| PWM | *Pulse Width Modulation*  Modulação de Largura de Pulso |
| BMS | *Battery Management System*  Sistema de Manutenção das Baterias |
| FW | *Forward*  Para a Frente |
| BW | *Backwards*  Para Trás |
| FWR | *Forward Right*  Motor Direito para a Frente |
| FWL | *Forward Left*  Motor Esquerdo para a Frente |
|  |  |
| BWR | *Backwards Right*  Motor Direito para Trás |
| BWL | *Backwards Left*  Motor Esquerdo para Trás |
| ESD | *Electrostatic-Sensitive Device*  Equipamento Sensível à Eletricidade Estática |
| PCB | *Printed Circuit Board*  Placa de Circuito Impresso |
| IC | *Integrated Circuit*  Circuito Integrado |
| AmpOp | AmplificadorOperacional |
| WEEE | *Waste Electrical and Electronic Equipment*  Resíduos de Equipamentos Elétricos e Eletrónicos |
| MTBF | *Mean Time Between Failures*  Período Médio Entre Falhas |
|  | Frequência de Oscilação |

# Introdução

## Introdução

Perante o atual panorama pandémico da Covid-19 [1] pretende-se, com a realização do Projeto Integrador da Unidade Curricular de Laboratórios e Práticas Integradas II (LPI II) do curso Mestrado Integrado em Engenharia Eletrónica Industrial e Computadores, a implementação de uma ideia com o objetivo de diminuir os contactos interpessoais que possam surgir no momento da entrega de bens a pessoas hospitalizadas. Esta situação pandémica é uma oportunidade para acelerar a transformação da área saúde.

Tendo em consideração que os hospitais tiveram um aumento considerável do número de internamentos [2], pretende desenvolver-se 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.

A maioria das ideias nesta área tem como foco principal a saúde pública da população em geral, tais como, robôs de desinfeção [3], robôs que repõem o *stock* em hospitais [4] ou que medem a temperatura corporal através de câmaras [5]. Prevê-se que o *Digital Waiter Robot* (DWR) possa ser aplicado em contexto hospitalar. Na China construiu-se um robô (*little peanut*) [6] com a mesma finalidade, que foi utilizado num hotel para entregar comida porta a porta a hóspedes com suspeita de infeção.

A versatilidade do sistema permitirá que o DWR possa auxiliar na distribuição de bens noutros contextos dependendo das funcionalidades requeridas, como, por exemplo, na indústria hoteleira ou em ambiente doméstico.

## Enquadramento

O DWR é um robô seguidor de linha focado na automatização de um hospital. O DWR estará parado num local apropriado à espera do envio de um pedido e de uma rota por parte da unidade de controlo. Assim que esta conclua a comunicação com o robô, um funcionário responsável deverá colocar no seu suporte os pedidos respetivos. O robô fará chegar cada pedido a cada paciente, parando apenas nos quartos solicitados. Depois de atender a todos os pedidos, o DWR voltará ao local de onde partiu.

Considere-se o exemplo de o DWR ser responsável pela distribuição de bens alimentares. O robô deve ser colocado próximo dos locais de confeção dos alimentos. Assim que a unidade de controlo envie a rota a percorrer e as refeições a distribuir, um dos responsáveis pela secção alimentar deverá colocar estes pedidos na base do robô, dar ordem de início de marcha, e o robô tratará de tudo o resto. O DWR pode assumir outras funções, como por exemplo, na distribuição de medicamentos. A unidade de controlo poderia ser uma base de dados de gestão autónoma que tendo os dados dos quartos ocupados e dos pacientes aí hospitalizados gerisse as suas necessidades, como, por exemplo, qual a medicação prescrita, bem como o horário a que deve ser tomada. Neste projeto, implementou-se uma versão mais simples de forma a ser possível validar o conceito do robô. A unidade de controlo é uma aplicação onde, apenas, se escolhe o quarto onde o pedido deve ser entregue, e as rotas para chegar ao destino estão desde já pré-estabelecidas.

## Especificações Previstas

O DWR seguirá uma linha preta previamente colocada no piso do hospital, que define os locais acessíveis pelo robô. Como os hospitais possuem vários quartos, em vários corredores, o robô terá de ser capaz de percorrer um percurso com várias intercessões de corredores. Assim, o sistema a ser implementado será constituído por dois subsistemas: o robô DWR e a unidade de controlo. Apesar de neste projeto ser usado apenas um robô, a unidade de controlo poderá, eventualmente, controlar vários.

Como a alimentação do robô será a baterias, eventualmente, terá de ser ligado à rede elétrica para ser carregado. Assim, existirá uma estação de carregamento que estará presente num local denominado por base.

O tipo de desenvolvimento deste produto pode ser classificado como “ofensivo”. “O objetivo é colocar no mercado um produto com funcionalidades e características inovadoras ou com preço significativamente mais baixo do que produtos com funcionalidades e características equivalentes, de forma a obter para o produto quota de mercado ou aumento da quota de mercado em relação a produtos antecessores” [7].

### Especificações funcionais

A unidade de controlo é responsável por uma aplicação de interface de gestão de pedidos.

Cada quarto e cruzamento no percurso deverá ter um identificador único e o DWR deve ser capaz de os detetar para que possa seguir a rota previamente estabelecida parando apenas nos quartos previstos. Após o paciente efetuar a recolha dos bens a si destinados, poderá acionar o robô de forma que este reinicie o seguimento da linha. Assim que não existam mais pedidos pendentes, o DWR retorna à base.

Para evitar que o DWR colida, este deve ter um sistema de deteção de obstáculos, que, ao detetar um objeto no seu percurso, faz com que o este pare. Se ao fim de um determinado intervalo de tempo, previamente estabelecido, a via se mantiver obstruída, deve ser emitido um sinal sonoro para alertar as pessoas ao seu redor. Se o problema for resolvido, o DWR continua o seu trajeto. Caso contrário, entrará num estado de erro e envia uma notificação para a aplicação de interface, alertando um funcionário responsável do sucedido

### Especificações técnicas

Para implementação do sistema de controlo do DWR será usado um microcontrolador STM32F767ZI-NUCLEO [8], e o IDE STM32CubeIDE [9], que integra as ferramentas necessárias para a configuração de todos dos periféricos.

Para cumprir o objetivo de seguir de linha, usar-se-á um *array* de oito sensores de reflexão com saídas analógicas. Ao contrário dos sensores digitais que apresentam apenas dois níveis nas suas saídas, alto ou baixo, este tipo de sensores possuem uma maior sensibilidade, permitindo que o sistema de seguidor de linha apresente menos oscilações.

A deteção e identificação de intercessões de corredores será feita por meio da tecnologia RFID [11]. O sistema de deteção de obstáculos será composto por um módulo de sensores de distância infravermelhos adequado para calcular com precisão a distância a objetos que possam aparecer na frente do DWR. O sistema de alerta sonoro fará uso de um *buzzer* ativo. A comunicação entre o robô e a unidade de controlo será implementada recorrendo a tecnologia *Bluetooth.*

## Planeamento

Na Figura 1.1, mostra-se o diagrama de Gantt do planeamento inicial elaborado na etapa 0 deste projeto.

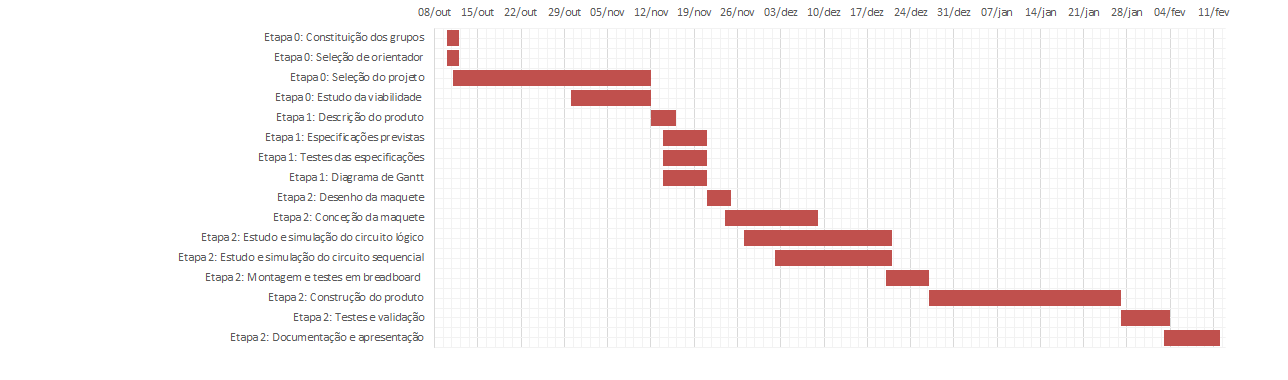


Figura 1.1 - Diagrama de Gantt do planeamento inicial.

Face ao confinamento geral [6], foi necessário reduzir a componente prática do projeto, nomeadamente a realização de ensaios experimentais e validação do produto. Foi feita uma análise mais cuidada do estudo da fiabilidade, segurança e certificação. A Figura 1.2 apresenta o diagrama de Gantt com as alterações efetuadas.

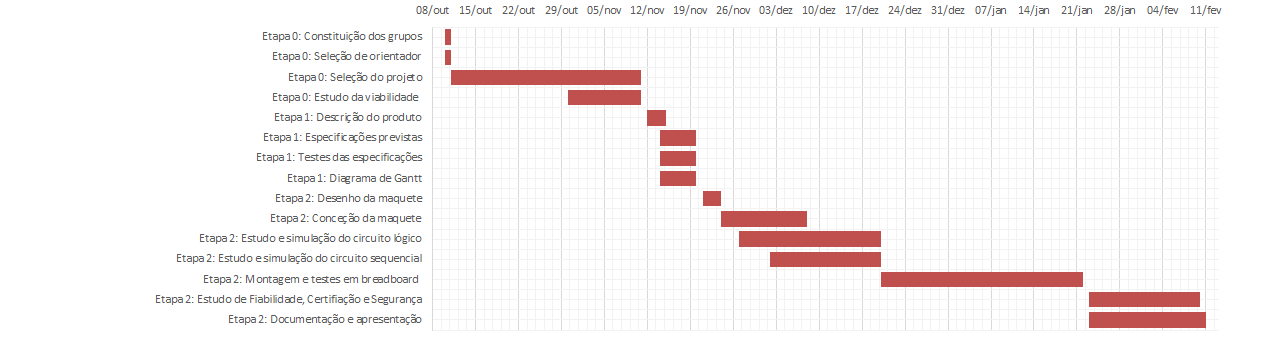


Figura 1.2 - Diagrama de Gantt do planeamento revisto.

# Arquitetura e Módulos Utilizados

## Introdução

O principal objetivo do DWR é auxiliar na distribuição de bens a várias pessoas, sendo controlado por um responsável. Na Figura 2.1, é apresentado o diagrama geral das principais interações do DWR. Considere‑se o operador como o responsável pelo DWR e o utilizador como a pessoa à qual se destinam os bens.

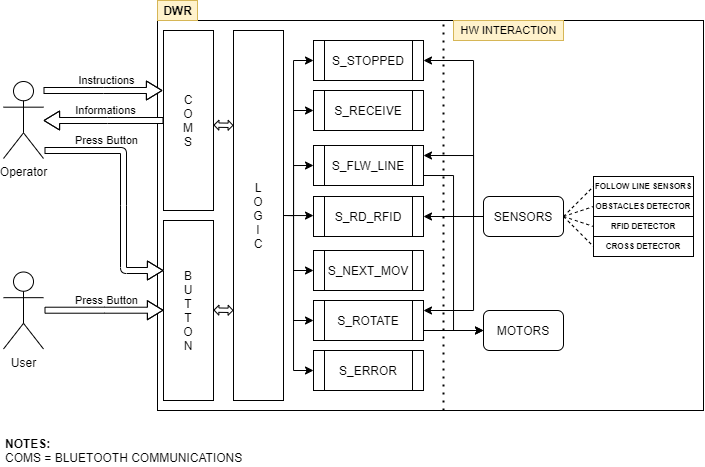


Figura 2.1 - Diagrama geral das principais interações no sistema.

O operador deve enviar instruções para o robô, via *Bluetooth,* (VER INTERAÇÃO COM O OPERADOR – APLICAÇÃO ???) selecionando a rota que pretende que este efetue. Após isto, o operador pode iniciar o movimento do DWR, pressionando o botão de pressão presente na lateral deste. Este botão também é usado pelo utilizador, que, depois de levantar os bens a si relativos, pode premi-lo, permitindo ao robô continuar o seu percurso. Além disso, o DWR envia informações para o operador, de modo que este possa monitorizar o seu estado de funcionamento.

O DWR é estruturado em diferentes blocos. Os blocos de comunicação e botão permitem a interação entre o robô e o operador / utilizador. Estes blocos relacionam-se com a lógica desenvolvida para controlar os estados do robô, tendo isso sido feito através de uma máquina de estados. Os diferentes estados de funcionamento do robô são, tal como apresentado na Figura 2.1, S\_STOPPED, S\_RECEIVE, S\_FLW\_LINE, S\_RD\_RFID, S\_NEXT\_MOV, S\_ROTATE e S\_ERROR. No estado S\_STOPPED, o DWR está parado à espera de algum estímulo. O estado S\_RECEIVE dedica‑se à escolha de novas rotas, através da comunicação entre o operador e o robô. O estado S\_FLW\_LINE implementa o controlo do seguidor de linha. O estado S\_RD\_RFID é responsável pela leitura de um cartão RFID de identificação unívoca para cada quarto e cruzamento. O estado S\_NEXT\_MOV é um estado de decisão, responsável por fazer transitar o robô para um estado que esteja de acordo com o percurso a realizar. O estado S\_ROTATE executa o controlo da mudança de direção do robô. O estado S\_ERROR é o estado para o qual o robô transita aquando da ocorrência de um erro que comprometa o normal funcionamento do sistema, informando o operador do sucedido.

A camada de interação com o *hardware* é composta por sensores e atuadores. As saídas dos sensores de obstáculos, de linha, de RFID e de cruz (sensor de paragem) são utilizadas nos estados S\_STOPPED, S\_FLW\_LINE, S\_RD\_RFID e S\_ROTATE. Os atuadores, os motores, são controlados nos estados S\_FLW\_LINE e S\_ROTATE.

## Sensores

Um sensor é um dispositivo que responde a um estímulo do ambiente, físico ou químico, produzindo um sinal que pode ser transformado noutra grandeza física para fins de medição.

Para o DWR, foram usados sensores infravermelhos, que são muito utilizados em aplicações que envolvem leitura e deteção de proximidade. Estes usam a luz infravermelha, que é uma radiação eletromagnética, de baixa frequência, estando abaixo do espetro eletromagnético da luz visível ao olho humano. Este tipo de sensores fazem a sua leitura através da utilização de um emissor, normalmente, laser ou LED, e um recetor fotoelétrico, que contém um elemento optoelétrico, como, por exemplo um fotodíodo ou um fototransístor. Este recetor fotoelétrico deteta a luz vinda do emissor e converte a intensidade da luz recebida num sinal elétrico em tensão. Perante as características acima enumeradas, os sensores infravermelhos podem ser usados para implementar sensores de obstáculos e sensores leitores de linha.

### *Array* de Sensores de reflexão

Com o objetivo de o DWR seguir a linha escolheu-se o *array* de sensores QTR-8A, apresentado na Figura 2.2. Este pode ser alimentado com 3,3 V ou 5 V e tem oito sensores analógicos que apresentam na sua saída um valor de tensão compreendido entre 0 V e o valor de alimentação. Sobre uma superfície branca, os sensores medem uma tensão de, aproximadamente, 0 V. Já quando se encontram sobre uma superfície preta, os sensores medem uma tensão de, sensivelmente, 3,3 V. Para as restantes cores, os sensores apresentam tensões entre estas duas gamas. No contexto deste projeto, é indispensável a escolha de um *array* que produza na saída uma tensão máxima de 3,3 V, uma vez que este é o limite que os *analog to digital converters* (ADCs) da STM32F767ZI-NUCLEO.

Uma imagem com texto

Descrição gerada automaticamente

Figura 2.2 - *Array* de sensores QTR-8A.

A Tabela 2.1 apresenta a finalidade dos sensores do QTR-8A nos circuitos do DWR.

Tabela 2.1 - Sensores e sua utilização.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Sensor 1** | **Sensor 2** | **Sensor 3** | **Sensor 4** | **Sensor 5** | **Sensor 6** | **Sensor 7** | **Sensor 8** |
| Deteção cruzamento /quarto | - | Seguidor de linha | Deteção linha preta | Deteção linha preta | Seguidor de linha | - | Deteção cruzamento /quarto |

### Sensor de obstáculos

Face à aplicação do DWR, torna-se necessário implementar um sistema de deteção de obstáculos, de forma a evitar a colisão com eventuais objetos que impeçam a sua passagem. Para isso, utilizou-se o sensor de distância infravermelho [ref] da fabricante SHARP. Este sensor permite medir distâncias a objetos entre 10 e 80 cm, tendo uma saída do tipo analógico. Assim, para esta aplicação, este sensor será usado como sensor de proximidade. Este pode ser alimentado com tensões entre 4,5 V e 5,5 V, tendo sido usada uma tensão de alimentação de 5 V.

## Driver

Para controlo dos motores utilizou-se o módulo *driver* apresentado na Figura 2.3. Este *driver* possui seis entradas, três para cada motor: os sinais de *ENA* e *ENB* permitem ativar ou desativar os motores, controlando as suas velocidades de rotação. Por sua vez, os sinais de *IN1*, *IN2*, *IN3* e *IN4* definem o modo de rotação de cada motor. Na Tabela 2.2 encontra-se exemplificado o modo de operação do motor em função da combinação lógica dos sinais de entrada.

Uma imagem com texto, eletrónica

Descrição gerada automaticamente

Figura 2.3 - Módulo Driver L298N.

Tabela 2.2 - Tabelas de verdade do *driver* L298N (a) Controlo do Motor A; (b) Controlo do Motor B.

|  |  |
| --- | --- |
| **(a)** | **(b)** |
| |  |  |  |  | | --- | --- | --- | --- | | ENA | IN1 | IN2 | Motor A | | 0 V | N/A | N/A | OFF | | 5 V | 0 V | 0 V | Trava | | 5 V | 0 V | 5 V | Sentido anti-horário | | 5 V | 5 V | 0 V | Sentido horário | | 5 V | 5 V | 5 V | Trava | | |  |  |  |  | | --- | --- | --- | --- | | ENB | IN3 | IN4 | Motor B | | 0 V | N/A | N/A | OFF | | 5 V | 0 V | 0 V | Trava | | 5 V | 0 V | 5 V | Sentido anti-horário | | 5 V | 5 V | 0 V | Sentido horário | | 5 V | 5 V | 5 V | Trava | |

## Módulos

### *Radio Frequency Identification* (RFID)

### Bluetooth

## Microcontrolador e *Shield*

## Circuito de Alimentação

Para alimentar os circuitos são necessárias tensões de 12 V, 5 V e 3,3 V. Para obter o primeiro valor de tensão, foram utilizadas três baterias recarregáveis LI-ION de 3,7 V [REF] em série. De modo a aumentar a autonomia do DWR, foram colocadas mais três baterias, cada uma em paralelo com as três já existentes. Para proteção das baterias foi usado um dispositivo de BMS [REF]. Este dispositivo controla a descarga das baterias não deixando que a sua tensão desça abaixo de um limite de segurança. Como as baterias necessitam de ser carregadas, adicionou-se ao circuito os terminais de entrada de modo a ser possível ligar um carregador [REF] em paralelo com a carga. Para proteção de todos os componentes, foi usado um fusível entre os terminais do carregador e a carga [REF]. O esquema de ligação é apresentado na Figura 2.4.

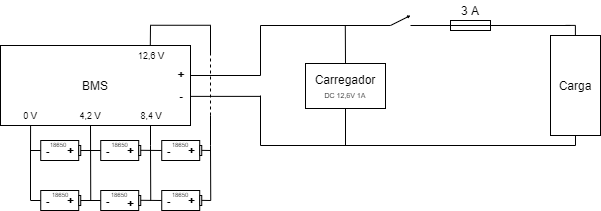


Figura 2.4 - Esquema de ligação das baterias e BMS.

Cada motor usado no DWR consome uma corrente de 580 mA [REF], e como este possui dois motores, a corrente total do par será 1160 mA. Quanto aos sensores usados, o QTR-8A consome, sensivelmente, 100 mA, o RFID 26 mA, o sensor de distância 30 mA e módulo bluetooth é 40 mA. O microcontrolador STM32F767ZI tem um consumo máximo de 258 mA. Somando as correntes consumidas por estes componentes, e assumindo que os restantes componentes têm consumos desprezáveis comparativamente com os restantes, conclui-se que consumo total do DWR será de, aproximadamente, 1614 mA. Visto que cada célula possui 2200 mAh e considerando que esta tem um rendimento de 80 %, então estão disponíveis 1760 mAh. Como cada célula está colocada em paralelo com uma outra, no total, estão disponíveis 3520 mAh para todos os circuitos. Assim, estimou-se uma autonomia de 2 horas e 10 minutos. Caso se pretenda aumentar este valor, poder-se-á adicionar mais células em paralelo com as existentes.

De forma a obter tensões de 3.3V e 5 V, necessárias para alimentação de sensores e módulos do DWR, usou-se uma fonte de alimentação para *breadboard* V2 – 5 V / 3.3 V DC [REF]. Esta fonte é alimentada via USB. Para o efeito usou-se um circuito abaixador para 5V, com quatro portas USB, que tem como entrada os 12V provenientes da BMS [REF]. Uma vez que adquiro este módulo, optou-se por alimentar a STM de igual modo. Atendendo que o *step-down* debita, no máximo 8 A, é suficiente para alimentar todos os componentes que necessitam de 5 V via USB.

# Implementação em Software

## Introdução

## Periféricos

### *Direct Memory Acess* (DMA)

Uma unidade de Acesso à Memória Direta (DMA) é um elemento lógico digital que pode ser usado em conjunto com o microprocessador para executar as operações de transferência de memória. Deste modo reduz-se, significativamente, a carga da unidade central de processamento (CPU). O DMA permite realizar transferências entre memórias, entre o periférico e a memória e entre a memória e o periférico. Os dispositivos compartilham o barramento de memória e os barramentos de periféricos com o CPU, tal como mostrado na Figura 3.1. No diagrama, o dispositivo DMA lê o valor de um periférico a partir do barramento do periférico e grava na memória através do barramento de memória. Na STM o princípio é o mesmo, mas com os vários barramentos de periféricos.

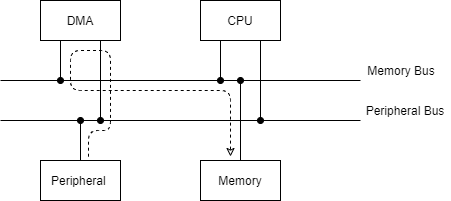


Figura 3.1 - Exemplo de transferência do DMA.

Na STM32F767ZI existem dois controladores DMA (DMA1 e DMA2) que possuem dezasseis *streams* (canais de acesso à memória) no total, oito para cada. Existem quatro níveis de prioridade para o pedido da *stream*. Caso esta tenha igual nível em várias *streams,* a *stream* com o nível mais baixo sobrepõe-se à do nível mais alto, por exemplo, a *stream* 0 tem prioridade em relação à *stream* 1.

O DMA pode ser configurado em dois modos de transferência de dados: modo direto, em que a transferência de dados é feita de forma imediata, ou modo *first in first out* (FIFO)*,* em que os dados são armazenados temporariamente antes de serem transmitidos para a memória. Em relação aos tamanhos dos dados, estes podem variar entre um byte, *half word* (2 *bytes* consecutivos) ou *word* (4 *bytes* consecutivos). Outras potencialidades do DMA surgem, no facto, de os apontadores se auto incrementarem, possibilitando a escrita de várias posições de memória consecutivas e a sua reprogramação ser automática (modo circular). De forma a inicializar o DMA entre um periférico específico e uma memória, é necessário configurar o canal apropriadamente e definir todas as configurações necessárias de acordo com a Figura 3.2.

Uma imagem com mesa

Descrição gerada automaticamente

Figura 3.2 - Configurações possíveis do DMA.

### *Analog to Digital Converter* (ADC)

Um *Analog to Digital Converter* (ADC) é um circuito eletrónico que converte uma entrada analógica num valor digital. Este valor representa o nível de tensão num código binário e permite aos microcontroladores comunicar com o mundo real com base nos valores amostrados. A STM32F767ZI possui ADCs de aproximação sucessiva (composto por um comparador e um *Digital to Analog Converter* (DAC) interno para aproximar, sucessivamente, o valor de saída do ADC ao valor de entrada, Figura 3.3), com até dezanove canais multiplexados, permitindo a leitura dezasseis fontes externas, duas fontes internas e o canal . É possível configurar os canais com uma resolução de 12-bit, 10-bit, 8-bit ou 6‑bit, alinhamento de dados à direita ou à esquerda, e o tempo de amostragem. O ADC apenas realiza leituras corretamente para valores de entrada compreendidos entre 0 V e 3,3 V.

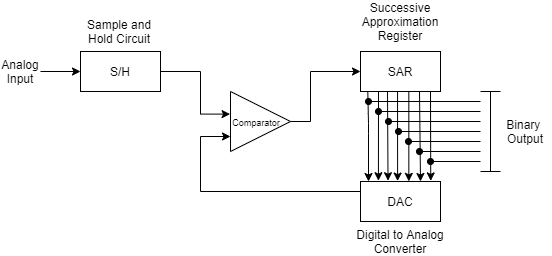


Figura 3.3 - conversor por aproximação sucessiva.

Cada ADC pode ser configurado, no modo independente, como *Single-channel* ou *Multichannel*. No primeiro, apenas é lido um canal, ao passo que, no segundo, são lidos vários canais sucessivamente. Além disso, pode ser configurado em *single conversion mode* e *continuous conversion mode*. No primeiro, o ADC realiza uma única conversão. No segundo, o ADC inicializa uma nova conversão logo que a conversão em curso termine. As possíveis combinações destes modos de operação estão apresentadas na Figura 3.4. É possível definir a *flag* de final de conversão para fim de cada conversão ou para o fim de todas as conversões. O ADC possui um único registo de dados, por isso, quando se lê múltiplas entradas analógicas, é essencial que os dados sejam lidos entre duas amostras. Isto pode ser realizado através de *polling,* interrupção ou DMA. Interrupções e DMA podem ser disparados no final de cada conversão. Em dispositivos com 2 ADCs ou mais, como é o caso de versão da STM32 em uso, é possível configurar os ADCs em modo duplo/triplo e o DMA para guardar os dados. Este periférico permite mais modos de funcionamento que não serão abordados, uma vez que o seu grau de complexidade não serve o propósito do trabalho desenvolvido.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| (a) | (b) | (c) | (d) |

Figura 3.4 - Modo de funcionamento independente (a) Single-channel, single conversion mode; (b) Multichannel, single conversion mode; (c) Single-channel, continuous conversion mode; (d) Multichannel, continuous conversion mode.

Neste projeto, foram utilizados dois ADCs com resolução de 12-bit e alinhamento dos dados à direita. O ADC2 foi usado para a leitura do sensor de obstáculos e o ADC3 para a leitura dos sensores utilizados do QTR-8A. Assim, foi usado um canal do ADC2 (*Single-channel*) e seis canais do ADC3 (*Multichannel*). Como se pretende realizar várias conversões, foi usado o modo de conversão contínua, com *flag* de final de conversão para fim de cada conversão, em ambos os ADCs. Para a transferência dos dados deste periférico para a memória utilizou-se DMA.

### *TIMER*

Um *timer* é um dispositivo de *hardware* capaz de medir uma base de tempo. Pode ser usado, por exemplo, para ativar eventos a frequências ou atrasos conhecidos, gerar sinais a várias frequências, sinais de saída modulados por largura de pulso (PWM) e medir pulsos de entrada.

A STM32F767ZI tem dois *timers* básicos (TIM6 e TIM7), dez *timers* do tipo *general-purpose* (TIM2 até ao TIM5 e TIM9 até ao TIM14) e dois *timers* do tipo *advanced-control* (TIM1 e TIM8). Os *timers* básicos são os mais simples. Podem ser usados para gerar bases de tempo, circuitos de sincronização para despoletar *digital to analog converters* (DAC) e gerar interrupções/DMA. Os *General purpose* podem ser configurados como *output compare, one-pulse mode output, input capture* e PWM *generation* *output compare, one-pulse mode output, input capture* e PWM *generation*.Estas informações podem ser observadas Figura 3.5. Na Tabela 3.1, pode‑se ver os barramentos a que cada timer pertence, bem como a frequência máxima do barramento.

Uma imagem com mesa

Descrição gerada automaticamente

Figura 3.5 – Comparação das características do *timer.*

Tabela 3.1 - Mapeamento dos *timers*.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Timer | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| APB1(108 MHz) |  | x | x | x | x | x | x |  |  |  |  | x | x | x |
| APB2(216 MHz) | x |  |  |  |  |  |  | x | x | x | x |  |  |  |

Para configurar o *timer* com a frequência desejada é necessário configurar os valores de *prescaler* de *preload*. O *prescaler* divide a frequência de oscilação do barramento, ao passo que o *preload* define o número de contagens do *timer*. Por exemplo, se o barramento tiver uma frequência de 108 MHz e o *prescaler* um valor de 108 ­‑ 1 (a contagem do registo começa em zero), a frequência do timer será de 108 MHz / 108 = 1 MHz, o que equivale a uma base de tempo de 1 us. Se o valor de *preload* for 10 ‑ 1, o timer terá uma frequência de 1 MHz / 10= 100 KHz, ou seja, um período de 1 / 100 KHz =10 us. Tendo em conta estes parâmetros e o valor da frequência do timer em questão, é possível calcular a frequência do *timer* usando a equação (3.1).

|  |  |  |
| --- | --- | --- |
|  |  | (3.1) |

Neste projeto, forem usados quatro *timers* para aplicações distintas. Para controlo dos *timeouts* que podem ocorrer durante o tempo de operação do DWR, do período de amostragem do algoritmo PID, sensor de obstáculos e do *debounce* apenas a necessário um *timer* que despolete uma interrupção ao fim de um determinado período de tempo, ou seja, um *timer* básico. Para o período de amostragem do algoritmo PID e sensor de obstáculos utilizou-se o mesmo *timer*, TIM6, com uma base de tempo 10 ms. Para o *debounce* do botão o TIM7 com uma base de tempo de 50 ms. Como o microcontrolador apenas possui dois *timers* básicos, utilizou-se um *timer* genérico, TIM3, para gerir os *timeouts* com uma base de tempo de 1 segundo. Para gerar os sinais de PWM à saída do controlador PID é necessário um *timer* no modo *PWM Generation*. Sendo assim, selecionou-se um *timer* genérico, TIM4,e utilizaram-se dois canais (*channel* 3e *channel* 4).

### *Serial Peripheral Interface* (SPI)

A *Serial Peripheral Interface* (SPI) é um protocolo de comunicação de curto alcance criado para a troca de dados entre microcontroladores ou entre microcontroladores e sensores. O SPI é síncrono e necessita que o transmissor e o recetor estejam sincronizados através de uma linha de *clock* partilhada. Este suporta transferências de dados em três modos: *full-duplex* (transferência de dados bidirecional simultânea), *half-duplex* (transferência de dados bidirecional não simultânea) ou *simplex* (transferência de dados unidirecional).

Os dispositivos conectados por SPI encontram-se numa relação *master-slave.* O *master* é, tipicamente, um microcontrolador que envia instruções ao *slave*, que, na maioria dos casos, se trata de um sensor, um *chip* de memória ou um display. De modo a proceder à comunicação *full‑duplex* entre um dispositivo *master* e um dispositivo *slave,* as 4 linhas lógicas para a transferência de dados do protocolo SPI, Tabela 3.2, do *master* e do *slave* devem ser conectados segundo a configuração apresentada na Figura 3.6.

Tabela 3.2 - Linhas lógicas para a transferência de dados do protocolo SPI.

|  |  |
| --- | --- |
| Serial Clock (SCLK) | Saída proveniente do master para sincronizar as transferências de dados nas linhas MISO e MOSI. |
| Master Out Slave In (MOSI) | Saída de dados do master com destino ao slave. |
| Master In Slave Out (MISO) | Saída de dados do slave com destino ao master. |
| Slave Select, (SS) | Linha que permite selecionar o dispositivo com o qual se pretende comunicar. |

Uma imagem com texto

Descrição gerada automaticamente

Figura 3.6 - Esquema de ligação entre Master e Slave.

Neste projeto, utilizou-se o protocolo SPI para comunicação com o chip MFRC522 [ref]. Optou‑se por utilizar a SPI3, dado que os pinos mapeados para esta interface não entram em conflito com pinos usados por outros periféricos. Foi configurada para funcionar com dados no formato Motorola e tamanho de 8-bit, em que o MSB é enviado primeiro. Configurou-se, ainda, o *clock* da interface para a transferência de dados ocorrer a 3,375 Mbits/s (sabendo que o *clock* do barramento APB1 é 54 Mhz, o *prescaler* terá valor 16), a polaridade do *clock* (CPOL) a *low* e a sua fase (CPHA) a zero. Deste modo, executa-se a captura dos dados provenientes do MFRC522 durante a transição ascendente.

### *Universal Synchronous/Asynchronous Receiver/Transmitter* (USART)

A *universal asynchronous receiver transmitter* (UART) é uma comunicação série largamente usada para comunicar entre dois dispositivos, como sistemas embebidos, microcontroladores e computadores. Na comunicação UART, a comunicação direta entre dois dispositivos faz-se através de duas linhas de dados, a de transmissão (Tx) e a de recessão (Rx), como representado na Figura 3.7.

Uma imagem com texto, relógio

Descrição gerada automaticamente

Figura 3.7 - Esquema de ligação entre 2 UARTs.

Este periférico transmite dados de forma assíncrona, ou seja, não necessita de um sinal de *clock* para se manter sincronizado. Em vez disso, o emissor acrescenta *bits* ao pacote a ser transmitido, sinalizando o início e o fim dos dados a ser transferidos, tal como mostrado na Figura 3.8.

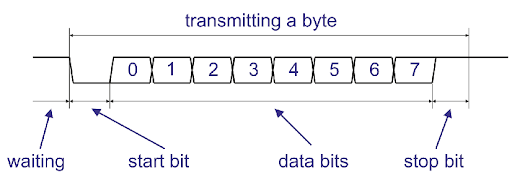


Figura 3.8 - Estrutura de um pacote de dados enviado por UART

Quanto ao recetor, após detetar o *start* bit, começará a ler os bits que se seguem a uma determinada frequência, denominado *baud rate*, expresso em bits por segundo (bps). Para que a comunicação entre dois dispositivos possa ocorrer sem erros, ambas as UART têm de operar ao mesmo *baud rate*.

Na STM32F767ZI encontram-se disponíveis quatro módulos UART (UART4, UART5, UART7 e UART8) e quatro USART (USART1, USART2, USART3 e USART6). As últimas diferem das primeiras apenas no facto de poderem ser configuradas no modo síncrono. Todos podem ser configurados em *full-duplex* (transferência de dados bidirecional simultânea) ou *half-duplex* (transferência de dados bidirecional não simultânea). A estrutura dos pacotes de dados pode ser modificadas, podendo escolher ter 7,8 ou 9 bits de dados, 1 ou 2 stop bits e uma definição da ordem da transmissão de dados.

Neste projeto, a UART possibilita a comunicação entre a STM e o módulo Bluetooth [ref]. O formato da comunicação série deste módulo é de oito *bits* de dados, um stop *bit* e nenhum de paridade e opera a um *baud rate* de 9600 bits/s. Deste modo, configurou-se a USART1 com as características acima descritas e ativaram-se as interrupções para processamento dos dados

## Descrição de *Software* e Módulos Criados

### MódulosCriados

### Circuito de Controlo

Para a implementação do módulo do seguidor de linha é necessário o desenvolvimento e implementação de um controlador.

#### O que é um Controlador?

Um controlador é responsável pelo controlo de processos através de algoritmos específicos. O seu principal objetivo consiste na monitorização, identificação e interpretação de processos, via modelos matemáticos, de forma a produzir uma ação de controlo conveniente. Existem três ações de controlo distintas: proporcional, integral e derivativa, que podem ser conjugadas entre si. A primeira tem uma ação imediata, proporcional ao valor atual do erro, acelera a resposta de um processo controlado, reduz o tempo de subida e o erro máximo. No entanto, aumenta o *overshoot*, o tempo de estabilização e produz um *offset* inversamente proporcional ao ganho. A ação integral produz uma ação de controlo gradual proporcional à integral do erro, respondendo, assim, ao passado do erro enquanto este for diferente de zero, elimina o *offset* e reduz o tempo de subida. Porém, aumenta o *overshoot*, o período de oscilação e tempo de estabilização, produzindo respostas lentas e oscilatórias. A ação derivativa produz uma ação antecipatória e proporcional à derivada do erro. É usada para acelerar e estabilizar a malha, reduzir o *overshoot*, o erro máximo e o período de oscilação. Contudo, não é indicada para processos com ruído.

Na Figura 3.9, estão presentes as três ações de controlo descritas. A Figura 3.9 (a) mostra a resposta de um sistema a uma ação proporcional. A saída deste corresponde à variável de erro multiplicada por uma dada constante. A Figura 3.9 (b) mostra a resposta de um sistema a uma ação integral, onde a saída deste corresponde à integral da variável de erro. Como a integral de uma constante é uma reta, quando a entrada (variável de erro) é do tipo degrau, a resposta do sistema vai corresponder a uma rampa de declive igual à amplitude da variável de entrada multiplicada por uma constante. A Figura 3.9 (c) mostra a resposta de um sistema a uma ação derivativa, em que a saída deste corresponde à derivada da variável de erro. Como a derivada de uma reta é uma constante, quando a entrada é do tipo rampa, a saída do sistema corresponde a um degrau de amplitude igual à amplitude da rampa multiplicada por uma constante. Um controlador que conjugue as três ações é denominado por controlador proporcional integral derivativo (PID).

|  |  |  |
| --- | --- | --- |
|  |  |  |
| (a) | (b) | (c) |

Figura 3.9 - Ações de controlo (a) ação proporcional; (b) ação integral; (c) ação derivativa

#### Análise do sistema de controlo

O primeiro passo num projeto de controlo está relacionado com a compreensão qualitativa do sistema. É necessário compreender como o sistema a controlar funciona fisicamente, quais as variáveis medidas, a controlar, de atuação, distúrbio e comando, bem como o funcionamento dos atuadores, sensores e controlador.

O sistema a ser controlado assemelha-se a um paralelepípedo com uma roda posicionada na parte central de cada lateral do robô. Sendo o objetivo do sistema seguir uma linha, que pode conter trajetórias retas ou curvilíneas, a velocidade de rotação dos motores terá de variar de modo a ser possível ajustar as velocidades de translação e de rotação do centro de massa do robô. Se se pretender que o robô siga uma trajetória retilínea, os motores terão de rodar à mesma velocidade. Se se pretender que o robô efetue uma trajetória curvilínea, o motor do lado oposto ao que se pretende efetuar a trajetória terá de ter uma velocidade de rotação superior. Ou seja, quando a trajetória é uma curva à esquerda, o motor do lado direito terá de ter uma velocidade de rotação superior comparativamente com a do motor do lado direito. Quando a trajetória é uma curva à direita, passa-se o oposto.

As variáveis medidas são as leituras efetuadas pelos dois sensores do seguidor de linha, o sensor 3 e o sensor 6 (ref imagem do qtr). As variáveis a controlar são a velocidade de translação e velocidade de rotação do centro de massa do robô. As variáveis de atuação são os binários dos motores. As variáveis de comando são a fração de modulação do amplificador PWM de cada motor. A variável de perturbação será o atrito provocado pela superfície (binário de perturbação).

O atuador é composto pelo motor DC (ref), a ponte H(ref) e pelo microcontrolador(ref). O binário produzido pelo motor DC pode ser alterado através da variação da tensão de alimentação do motor (produzida por um amplificador PWM).

O sistema de controlo é responsável pela variação deste parâmetro, de maneira a produzir o binário adequado à velocidade requerida para que o DWR siga a linha corretamente. Em conjunto, o microcontrolador e a ponte H implementam o amplificador PWM. O algoritmo de controlo implementado no microcontrolador produzirá na saída a variável de comando, a fração de modulação do amplificador de PWM, sendo o ganho deste amplificador correspondente a tensão aplicada ao motor DC quando a fração de modulação é um.

O controlador é efetuado através do microcontrolador. A regra de controlo executada por este controlador está implementada numa rotina de serviço à interrupção (ISR), que será despoletada por um *timer* com período igual ao período de amostragem escolhido pelo sistema de controlo. Esta ISR terá como parâmetros de entrada os valores das leituras dos dois sensores a utilizar e como saída a fração de modulação de PWM, que servirá de entrada aos amplificadores PWM.

Na Figura 3.10, está ilustrado o sistema de controlo implementado. Sendo o objetivo principal do DWR o seguimento de uma linha, é necessário manter os sensores na parte exterior da mesma. Quando um dos sensores se aproximar da linha, o motor do lado oposto terá de compensar o desvio da trajetória. Pretende‑se que a diferença entre as leituras dos dois sensores, ou seja, a variável de erro, seja nula, portanto, conclui-se que a variável de referência tem o valor zero.

Se o valor de erro for positivo, significa que o sensor esquerdo se encontra mais próximo da linha do que o sensor direito, implicando que o motor direito tenha uma velocidade de rotação superior. À variável de saída do bloco controlador PID soma-se um valor de *offset* que servirá de variável de comando para o atuador direito (*driver* motor e motor). Ao simétrico da variável de saída do bloco PID soma‑se o mesmo valor de offset, que servirá de variável de comando para o atuador esquerdo (*driver* motor e motor). Deste modo, o valor da variável de comando do atuador direito é superior ao valor da variável de comando do atuador esquerdo.

Se o valor do erro for negativo, passa-se o oposto. Se a variável de saída do controlador PID for nula não é necessário fazer ajustes de direção e o DWR seguirá o percurso com velocidade de rotação constante em ambos os motores, com valor igual ao valor de *offset.*

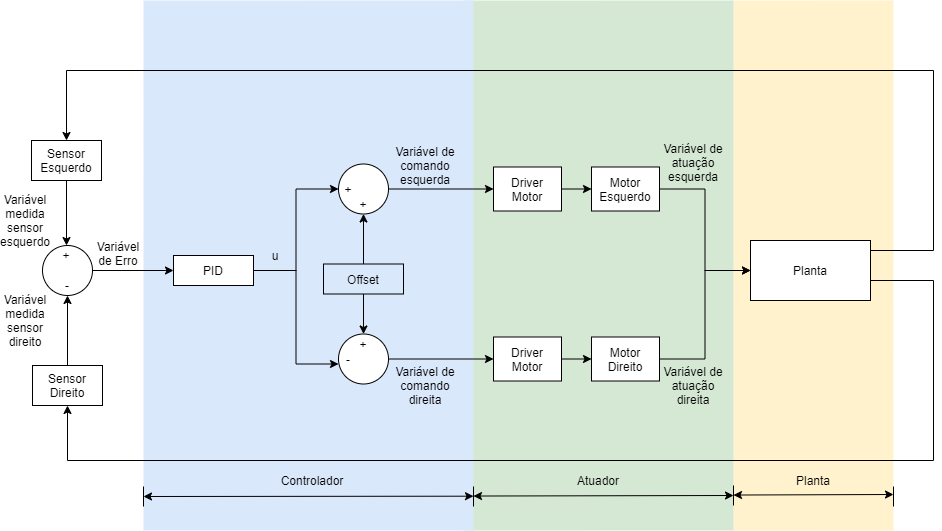


Figura 3.10 - Diagrama de blocos do sistema de controlo

#### Controlador implementado

Um controlador PID contínuo é definido pela equação (3.2), em que , e são referidos, respetivamente, como os ganhos proporcional, integral e derivativo, e como a variável de erro ao longo do tempo. A variável de comando, , corresponde ao somatório das ações proporcional, integral e derivativa.

|  |  |  |
| --- | --- | --- |
|  |  | (3.2) |

Para uma implementação em microcontrolador é necessária uma versão digital do controlador. De entre as diversas famílias das regras PID implementou-se o algoritmo de posição. Esta versão consiste em substituir os termos integral e derivativo, da equação (3.2), pelos seus equivalentes discretos: soma e a diferença divida de 1ª ordem, respetivamente. Assim, obtém-se a equação (3.3), em que e são, respetivamente, a ação de controlo e o erro no instante . Em cada instante é calculado o valor real (de posição) do sinal de saída do controlador. No contexto do problema, a variável de erro corresponde à diferença entre as leituras dos dois sensores, aqui representados, simbolicamente, por e , equação (3.4).

|  |  |  |
| --- | --- | --- |
|  |  | (3.3) |
|  |  | (3.4) |

Para efeitos de cálculo, o valor de pode ser escrito de forma mais adequada, calculando o erro apenas uma vez e colocando em evidência as constantes na aproximação à integral e à derivada (equação (3.5)). Pode-se, agora, redefinir os ganhos proporcional, integral e derivativo, equações (3.6), (3.7) e (3.8), respetivamente, obtendo‑se, por substituição, a equação (3.9).

|  |  |  |
| --- | --- | --- |
|  |  | (3.5) |
|  |  | (3.6) |
|  |  | (3.7) |
|  |  | (3.8) |
|  |  | (3.9) |

Para calcular o somatório dos erros, presente na equação (3.9), define-se a variável , equação (3.10), que pode ser escrita de forma recursiva, equação(3.11). Juntando todas as transformações efetuadas, obtém-se, a equação (3.12), uma nova versão da expressão inicial, equação (3.3).

|  |  |  |
| --- | --- | --- |
|  |  | (3.10) |
|  |  | (3.11) |
|  |  | (3.12) |

Uma forma de reduzir os problemas com erros de medida prende-se com a utilização de um filtro passa-baixo na ação derivativa. Na equação (3.13), apresenta-se a expressão desta ação. Calculando a sua transformada em obtém-se a equação (3.14). A aproximação à derivada introduz um zero em e um polo em , no plano. Pode-se deslocar o polo para a direita, no plano , com um filtro passa‑baixo resultando as equações (3.15) a (3.17). A função de transferência tem um zero em , como no caso anterior, mas o polo situa-se em , e não em . A implementação desta função de transferência está representada na equação (3.18). Pode-se então redefinir como apresentado na equação (3.19). Adicionando a componente do filtro passa-baixo à equação (3.12), chega-se à equação (3.20). A segunda reduz-se à primeira se *a* for zero, ou seja, a ação derivativa sem filtro de passa-baixo é obtida com , o que implica que .

|  |  |  |
| --- | --- | --- |
|  |  | (3.13) |
|  |  | (3.14) |
|  |  | (3.15) |
|  |  | (3.16) |
|  |  | (3.17) |
|  |  | (3.18) |
|  |  | (3.19) |
|  |  | (3.20) |

Um dos problemas de algoritmo de posição está relacionado com a possível saturação do valor de Definem-se, então, limites de saturação superior () e inferior (). Se o resultado do cálculo do valor de for superior ao valor do limite de saturação superior, ou inferior ao valor do limite de comparação inferior, é necessário limitar o mesmo aos valores de saturação impostos. Assim, modifica-se a interrupção para detetar a ultrapassagem dos valores de saturação e, caso tal aconteça, fixa-se o valor de nos valores de saturação. Além disso, a saturação do atuador pode dar origem a um aumento brusco dos valores dos somatórios dos erros levando a oscilações prejudiciais no valor da variável controlada.

Tendo em conta os aspetos mencionados acima e fazendo uso da equação (3.20), é possível desenhar o fluxograma que permite a implementação do algoritmo de controlo no microcontrolador, Figura 3.11. Como se pode verificar, a atualização do somatório dos erros só é efetuada quando a variável se encontra dentro dos limites de saturação. Caso contrário, a atualização é anulada, o que equivale, em tempo contínuo, a parar a integração do erro, fixando‑se o valor de no valor de saturação respetivo.

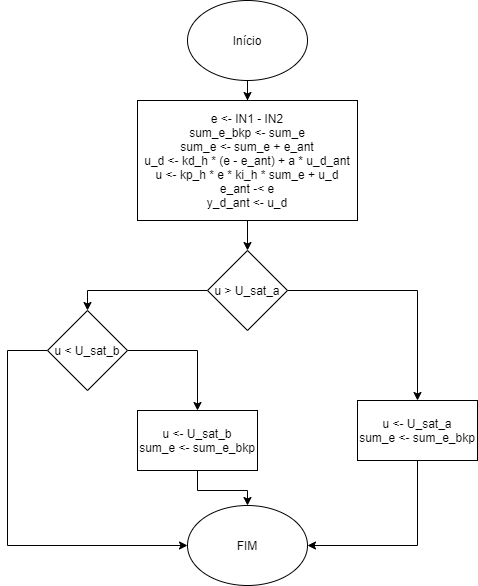


Figura 3.11 - Serviço de Rotina à Interrupção do algoritmo PID

Tendo em conta que, para um valor da variável de comando baixo, os motores estão parados, torna‑se necessária a soma de um *offset* a esta variável, de forma que, quando o erro for nulo, ou seja, quando , a variável de comando não seja nula. Por exemplo, para um valor de *offset* igual a 0,7 e , a variável de comando será igual a e a variável de comando será igual a . Desta forma, garante-se que a variável de comando nunca será nula.

## Simulações

# Lista de Componentes

Na Tabela 4.1, listam-se todos os componentes usados para o DWR, bem como a quantidade e o preço associado. É de realçar que nem todos os componentes foram adquiridos pelo preço indicado. Alguns foram emprestados, oferecidos ou adquiridos em condições mais favoráveis. No entanto, optou-se por especificar o preço de mercado.

Tabela 4.1 - Lista de componentes.

|  | Material | | Loja | Quantidade | Preço unitário | Preço total |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Motor Bot'n Roll ONE 300 rpm |  | Bot'n Roll | 2 | 17,50 € | 35,00 € |
| 2 | Par de rodas Bot'n Roll ONE A (D67 mm) |  | Bot'n Roll | 1 | 11,50 € | 11,50 € |
| 3 | Par de hubs Bot'n Roll ONE A |  | Bot'n Roll | 1 | 4,50 € | 4,50 € |
| 4 | Rodas livres  (D25 mm) |  | Leroy Merlin | 2 | 1,69 € | 3,38 € |
| 5 | QTR-8A |  | Bot'n Roll | 1 | 19,90 € | 19,90 € |
| 6 | Sensor Sharp 2Y0A21YK |  | Bot'n Roll | 1 | 11,95 € | 11,50 € |
| 7 | Módulo RFID RC522 | Uma imagem com texto, eletrónica  Descrição gerada automaticamente | Bot'n Roll | 1 | 4,90 € | 4,90 € |
| 8 | Módulo Bluetooth HC-05 | Uma imagem com eletrónica, circuito  Descrição gerada automaticamente | Bot'n Roll | 1 | 6,80 € | 6,80 € |
| 9 | Módulo *driver* de motores L298N |  | Bot'n Roll | 1 | 12,90 € | 12,90 € |
| 10 | Fonte de alimentação 5 V / 3,3 V | Uma imagem com eletrónica  Descrição gerada automaticamente | Bot'n Roll | 1 | 6,5 € | 6,5 € |
| 11 | Step-Down para UBS 5 V | Uma imagem com eletrónica  Descrição gerada automaticamente | Bot'n Roll | 1 | 9,70€ | 9,70€ |
| 12 | Cabo USB A p/ micro USB B |  | Bot'n Roll | 2 | 1,60 € | 3,20 € |
| 13 | Suporte para uma pilha 18650 c/fios |  | Bot'n Roll | 6 | 0,85 € | 5,10 € |
| 14 | BMS para proteção baterias 18650 3S  12,6 V 20 A |  | Bot'n Roll | 2 | 5,50 € | 11,00 € |
| 15 | Pilha LI-ION 18650 3,7 V 2200 mAh 18X65 mm – 22 A MAX |  | Bot'n Roll | 6 | 3,90 € | 23,40 € |
| 16 | Carregador baterias 18650 | Uma imagem com assobio, adaptador, chave  Descrição gerada automaticamente | Bot'n Roll | 1 | 8,90 € | 8,90 € |
| 17 | Cabo DC com conector reto 9,5 mm | Uma imagem com seta  Descrição gerada automaticamente | Bot'n Roll | 1 | 1,70 € | 1,70 € |
| 18 | Suporte Fusível Auto |  | AlfaElektor | 1 | 1,22 € | 1,22 € |
| 19 | Fusível Auto 3A | Uma imagem com isqueiro, caso  Descrição gerada automaticamente | AlfaElektor | 1 | 0,10 € | 0,10 € |
| 20 | STM32F767ZI | Uma imagem com eletrónica, circuito  Descrição gerada automaticamente | Bot'n Roll | 1 | 30,90 € | 30,90 € |
| 21 | PCB Shield STM32F767ZI | Uma imagem com texto, eletrónica  Descrição gerada automaticamente | Guimo circuito | 1 | 63,96 € | 63,96 € |
| 22 | Conectores Buchanan 284517-4 | 4P TERMIBLOK HDR DIR POL-284517-4 | TE Connectivity | 12 | 1,19 € | 14,26 € |
| 23 | Conectores Buchanan 284507-4 | Uma imagem com brinquedo  Descrição gerada automaticamente | TE Connectivity | 12 | 2,77 € | 33,24 € |
| 24 | Barra (2x8) fêmea 2,54 mm PCB | Uma imagem com texto  Descrição gerada automaticamente | Mauser | 1 | 0,30 € | 0,30 € |
| 25 | Barra (2x10) fêmea 2,54 mm PCB |  | Mauser | 1 | 0,40 € | 0,40 € |
| 26 | Barra (2x16) fêmea 2,54 mm PCB | Uma imagem com música  Descrição gerada automaticamente | Mauser | 1 | 0,62 € | 0,62 € |
| 27 | Barra (2x17) fêmea 2,54 mm PCB | Uma imagem com texto  Descrição gerada automaticamente | Mauser | 1 | 0,69 € | 0,69 € |
| 28 | Barra (2x3) fêmea 2,54 mm PCB | Uma imagem com contentor, caixa  Descrição gerada automaticamente | Mauser | 2 | 0,16 € | 0,32 € |
| 29 | Barra (1x6) fêmea 2,54 mm PCB |  | Mauser | 1 | 0,09 € | 0,09 € |
| 30 | Bloco terminal 12 conetores |  | Gmlux | 1 | 0,40 € | 0,40 € |
| 31 | Condensador poliéster |  | AlfaElektor | 4 | 0,10 € | 0,40 € |
| 32 | Botão de painel 12 mm | Uma imagem com luz  Descrição gerada automaticamente | Bot'n Roll | 1 | 0,60 € | 0,60 € |
| 33 | Interruptor de painel redondo | Uma imagem com eletrónica, preto  Descrição gerada automaticamente | Bot'n Roll | 1 | 1,20 € | 1,20 € |
| 34 | Alumínio 2 mm | Alumínio: Chapa de ALUMÍNIO em fms.pt | **-** | **-** | **-** | 35,00 € |
| 35 | Placa madeira | Placa de contraplacado 2500x1220x10mm - Leroy Merlin Portugal | **-** | 1 | **-** | 5,00 € |
| 36 | Placa de acrílico policarbonato |  | **-** | 1 | - | 10,00 € |
| 37 | Parafuso 4 mm |  | Ferritrofa | 50 | 0,05 € | 2,50 € |
| 38 | Parafuso 3 mm cabeça chata | Uma imagem com utensílios de metal, parafuso  Descrição gerada automaticamente | Ferritrofa | 13 | 0,05 € | 0,65 € |
| 39 | Parafuso 6 mm | Uma imagem com utensílios de metal, parafuso  Descrição gerada automaticamente | Ferritrofa | 6 | 0,05 € | 0,30 € |
| 40 | Parafuso 3 mm auto-roscante | Uma imagem com parafuso, utensílios de metal  Descrição gerada automaticamente | Ferritrofa | 6 | 0,04 € | 0,24 € |
| 41 | Porca com asas 4 mm |  | Ferritrofa | 8 | 0,10 € | 0,80 € |
| 42 | Porca 4 mm |  | Ferritrofa | 32 | 0,05 € | 1,60 € |
| 43 | Porca 3 mm | Uma imagem com projetor  Descrição gerada automaticamente | Ferritrofa | 13 | 0,05 € | 1,15 € |
| 44 | Rebites |  | Ferritrofa | 18 | 0,10 € | 1,80 € |
| 45 | Anilhas | Resultado de imagem para anilha | Ferritrofa | 12 | 0,01 € | 0,12 € |
| 46 | Kit Conectores 2,54 mm | Uma imagem com eletrónica  Descrição gerada automaticamente | Bot'n Roll | *pack* | 10,95 € | 10,95 € |
| 47 | Conjunto condutores flexíveis (1 m) |  | Bot'n Roll | 4 | 1,00 € | 4,00 € |
| Total: | | | | | | 227,85 € |

# Circuito Mecânico Implementado

|  |  |
| --- | --- |
|  |  |
| **(a)** | |
|  |  |
| **(b)** | |
|  |  |
| **(c)** | |
|  |  |
| **(d)** | |
|  |  |
| **(e)** | |

Figura 5.1 - Desenho (à esquerda) e imagem real (à direita) do DWR (a) vista superior; (b) vista inferior; (c) vista lateral direita; (d) vista dianteira; (e) vista traseira.

# Resultados Experimentais

De forma a testar as especificações acima previstas, foram realizados ensaios experimentais simulando o ambiente no qual o robô irá operar.

Começou-se por testar o sensor e verificou-se que as medidas dos sensores não atingem nem 0 V nem + 5 V.

Em seguida, montou-se o circuito de atuação dos sinais PWM de modo a testar o seu funcionamento. Para isso, variou-se o sinal na entrada de *feedback* do TL494, através do ajuste de um potenciómetro, obtendo-se os valores de *duty cycle* à saída em função da tensão de entrada, como apresentado na Tabela 6.1.

Tabela 6.1 - *Duty cycle* do sinal PWM de saída em função da tensão de entrada.

|  |  |
| --- | --- |
| Vcomando [V] | *Duty cycle* [%] |
| 1,15 | 0 |
| 1,39 | 10 |
| 1,9 | 30 |
| 2,3 | 50 |
| 2,9 | 70 |
| 3,35 | 90 |
| 3,6 | 97 |

Depois de validado este circuito, ligou-se a saída deste ao *driver* [10] para testar a resposta do motor. Concluiu-se que o motor apenas possui binário de arranque suficiente quando o *duty cycle* do sinal PWM ronda os 50 %. Já com o motor em funcionamento, o *duty cycle* pode atingir um mínimo de 30 % sem que os motores parem. Estes dados foram retirados com os motores em vazio.

# Análise do Produto

## Introdução

Em todos os projetos práticos é necessária uma análise do produto, em áreas como a fiabilidade, segurança e certificação do sistema. Esta análise permite identificar pontos de falha do equipamento e os potenciais perigos a estes associados, de forma a informar o utilizador.

## Fiabilidade

Um sistema ou equipamento diz-se “*fiável quando está livre de erros catastróficos, é capaz de recuperar de erros e apresenta resultados previsíveis (determinismo)*” [15]. Apesar de, neste caso, não ser possível fazer uma avaliação do tipo quantitativa, realizou-se uma avaliação do tipo qualitativa por meio do estudo dos modos de falha e as suas consequências para o sistema e, também, influência do ambiente e do tempo.

Relativamente aos modos de falha, o desenho do sistema deve ter em conta as condições em que o robô se desvia da linha preta. Assim, este deverá ter uma velocidade adequada, uma vez que o sistema pode não responder suficientemente rápido a uma variação. Para diminuir a probabilidade de erros no seguimento da linha, deve ser favorecido o uso de linhas retas em todo o percurso e, quando necessário mudar de direção, devem ser feitas curvas ligeiras. Em acrescento, a grossura da linha deve ser ligeiramente inferior à distância entre os dois sensores usados para seguir a linha (3 cm, aproximadamente).

Do ponto de vista do ambiente que envolve o robô, devem ter-se em conta as condições climatéricas, interações com o utilizador, condições do equipamento e obstrução da via de passagem do robô. O percurso não deverá ser obstruído com objetos, sob pena de danificar tanto o robô como o objeto que o obstrui ou pessoas que se encontrem próximas, visto que o robô não possui um sistema de deteção de obstáculos. Além disso, sendo o robô um equipamento eletrónico, este não deverá estar sujeito a um ambiente húmido, tal como água ou neve, sujidade e temperaturas extremas. As condições anteriores podem provocar um mau funcionamento dos circuitos constituintes do robô, podendo, em situações extremas, danificar permanentemente o equipamento.

Em relação ao tempo de operação, sabe-se que os componentes têm um tempo médio de vida previsto pelos fabricantes. Visto que não existe informação suficiente para calcular o consumo dos circuitos do AWR, torna-se difícil fazer uma previsão do tempo de vida dos seus componentes com exatidão. Num circuito eletrónico, os componentes com maior probabilidade de falha são os condensadores, semicondutores, baterias e motores. Para quantificar o número de horas previstos de funcionamento médio de um componente, existe um parâmetro denominado *Mean Time Between Failures* –MTBF. Os componentes enunciados têm os MTBFs apresentados na Tabela 7.1.

Tabela 7.1 - Tempo de vida dos componentes com maior probabilidade de falha, usados no AWR.

|  |  |
| --- | --- |
| Componente | MTBF |
| Baterias 18650 | 300 – 500 ciclos |
| Semicondutores | 10 anos |
| Condensadores Eletrolíticos | 11,4 anos |
| Condensadores Cerâmicos | 142,69 anos |
| Motores | 1000 a 3000 horas |

Anormalidades nos circuitos poderão diminuir estes tempos consideravelmente, exigindo uma manutenção ou reparação precoce do robô.

Como calculado no subcapítulo 2.6, a autonomia prevista do robô é de uma hora e dez minutos. Caso o robô seja destinado para uso individual, desprezando o consumo dos circuitos durante o tempo de espera, e considerando, por exemplo, dez minutos como tempo de deslocação, o robô terá autonomia suficiente para transportar seis refeições. Assumindo que a cada dia existem quatro refeições (pequeno almoço, almoço, lanche e jantar), a bateria deverá ser recarregada a cada um dia e meio. Assim sendo, as baterias terão um tempo de vida útil compreendido entre um ano e três meses e dois anos e um mês, aproximadamente. Por outro lado, se o AWR for usado num âmbito coletivo, assumindo uma hora de funcionamento por refeição, a bateria deverá ser recarregada quatro vezes por dia. Calcula-se, assim, que as baterias terão de ser substituídas ao fim de dois meses e meio, podendo durar, no máximo, até quatro meses.

Conclui-se que as baterias deverão ser os elementos que requerem maior atenção, pois, além das deficiências acima apresentadas, com o aumento do tempo de utilização, estas descarregam, conduzindo a alterações dos resultados práticos.

## Segurança

A interação do robô com o utilizador deve ser realizada em segurança.

Os sistemas elétricos e fios condutores não devem estar em contacto com a armadura do robô que, sendo metálica, é condutora, evitando curto-circuitos ou choques elétricos que podem ser fatais ao sistema e prejudiciais à saúde do utilizador. Para garantir o isolamento da armadura do robô, usou-se, por questões económicas e para validação de conceitos, uma placa de madeira para suportar todos os circuitos do robô. Os motores e o *array* de sensores, que ficam no exterior da armadura, estão expostos ao utilizador. De forma a isolá-los eletricamente, usou-se fita isoladora envolvendo todos os terminais destes. Portanto, todos os componentes do robô estão encapsulados e isolados do utilizador. Todavia, sabe-se que este equipamento é um dispositivo sensível à eletricidade estática (frequentemente abreviado como ESD – *Eletrostatic-Sensitive Device*), o que significa que possui componentes que podem ser danificados por cargas elétricas estáticas que se acumulam em pessoas, ferramentas e outros materiais não condutores [16]. Se o utilizador violar o encapsulamento do produto, poderá representar perigos para este ou uma avaria no produto (Figura 7.2 (h)).

Deverá ter-se em conta que o robô possui elementos que apresentam perigos para o utilizador. A madeira usada para isolar os circuitos eletrónicos da armadura do robô é considerada um material comburente (Figura 7.2 (c)) e inflamável (Figura 7.2 (d)). Os condensadores eletrolíticos, usados nos circuitos do AWR, são componentes inflamáveis (Figura 7.2 (d)), explosivos (Figura 7.2 (e)), tóxicos (Figura 7.2 (f)) e perigosos para o ambiente (Figura 7.2 (a)). Além dos perigos associados aos condensadores, as baterias podem ser, ainda, elementos corrosivos (Figura 7.2 (b)) e comburentes (Figura 7.2 (c)). Assim, o robô não deverá estar exposto a qualquer condição ambiental extrema, tal como referido no subcapítulo anteriorrelacionado com a fiabilidade.

O percurso não deverá ser bloqueado por objetos, uma vez que o embate do robô poderá provocar situações imprevisíveis, com possibilidade de danificação de bens materiais ou ferimentos pessoais.

Devido aos perigos acima descritos, deve ser vigiado o manuseamento do robô por crianças.

## Certificação

Nenhum produto poderá ser exposto ao comércio antes de ser certificado. A certificação é o modo pelo qual uma entidade competente dá uma garantia escrita de que um produto está em conformidade com os requisitos especificados.

A marcação CE, apresentada na Figura 7.1, é um indicativo de conformidade obrigatória para diversos produtos comercializados no Espaço Económico Europeu. Esta marca indica que um produto respeita a legislação da União Europeia em requisitos como segurança, higiene e proteção ambiental, estando, desta forma, credenciado a circular por todo Espaço Económico Europeu [17].

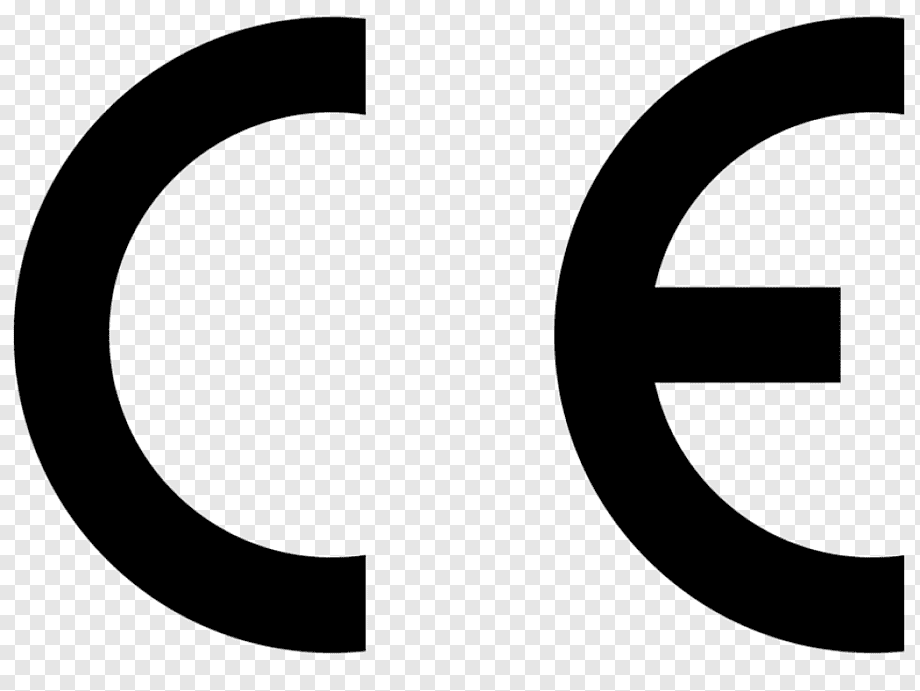


Figura 7.1 - Marcação CE.

De acordo com DIRETIVA 2014/35/UE [18] ANEXO I, que informa os principais elementos dos objetivos de segurança para o material elétrico usado no equipamento, devem ser previstas medidas de ordem técnica a fim de que:

1. As pessoas e os animais domésticos fiquem protegidos de forma adequada contra os riscos de ferimentos ou de outros acidentes resultantes de contactos diretos ou indiretos;
2. Não se produzam temperaturas, descargas ou radiações que possam provocar perigo;
3. As pessoas, os animais domésticos e os bens sejam protegidos de forma adequada contra os riscos de natureza não elétrica provenientes do material elétrico que a experiência venha a revelar;
4. O isolamento seja adequado aos condicionamentos previstos.

Como já foi apresentado, o robô cumpre com os pontos a), b) e d). Uma vez que este projeto serve fins académicos, ou seja, tem o propósito de validar conceitos, não foi implementado um sistema de deteção de obstáculos. Assim, o ponto c) não é cumprido, pelo que o certificado CE não poderá ser atribuído ao AWR.

Devido aos perigos enunciados no subcapítulo anterior, relacionado com a segurança, deverão ser apresentados vários símbolos ao utilizador.

|  |  |  |  |
| --- | --- | --- | --- |
| Uma imagem com texto, símbolo, gráficos de vetor  Descrição gerada automaticamente | Uma imagem com texto, símbolo, exterior, volta  Descrição gerada automaticamente |  |  |
| (a) | (b) | (c) | (d) |
| Uma imagem com texto, símbolo, relógio  Descrição gerada automaticamente | Uma imagem com texto, símbolo  Descrição gerada automaticamente |  |  |
| (e) | (f) | (g) | (h) |

Figura 7.2 - Símbolos de perigo: (a) perigoso para o ambiente; (b) corrosivo; (c) comburente; (d) inflamável; (e) explosivo; (f) tóxico; (g) vários perigos; (h) *Eletrostatic Sensitive Device - ESD*.

O robô será constituído por circuitos e componentes eletrónicos que não podem ser enviados para o lixo doméstico. Estes devem ser encaminhados para locais próprios de reciclagem (Figura 7.3).

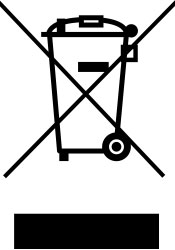


Figura 7.3 - Símbolo WEEE.

# Conclusões

## Conclusão

O projeto integrador da unidade curricular de LPI I pôs à prova algumas competências adquiridas ao longo do curso. Aplicaram-se conhecimentos das unidades curriculares de Sistemas Digitais, Eletrónica I e II, Instrumentação e Sensores, Controlo Automático, Máquinas Elétricas, entre outras. Foi um projeto ímpar comparativamente a projetos já realizados porque envolveu um maior conhecimento de várias áreas e requereu muito trabalho autónomo.

Ao longo do *design* e testes do robô foram encontrados vários problemas, que provocaram alterações na forma como se abordou o desenvolvimento do projeto. Inicialmente usou-se o sensor de linha no centro do robô de forma a facilitar a inversão do sentido de marcha no final do seu trajeto, evitando assim que o robô necessitasse de efetuar uma volta de 180 ° sobre o seu eixo. Assim, os motores teriam de rodar num sentido quando o robô fosse na direção do paciente, e no sentido contrário quando tivesse de voltar ao ponto de partida, simplificando a lógica que controla o estado de funcionamento do robô. Foram implementadas e testadas diversas soluções que não cumpriram o seu propósito, pois, durante a realização de testes, verificou-se que esta abordagem revela problemas mecânicos. Desta forma, o ponto central do robô (posição do sensor) é menos sensível a alterações de direção, aumentando o seu tempo de resposta, impedindo-o de seguir a linha corretamente. Após a alteração da posição do sensor para a dianteira do robô, estes problemas resolveram-se.

Relativamente ao desenho e implementação da máquina de estados, também se encontraram dificuldades. Inicialmente, não houve a preocupação de selecionar integrados da mesma família lógica, para implementar a máquina de estados, o que originou resultados indefinidos. Verificou-se que todos os integrados devem ser da mesma família lógica, ou compatíveis com a lógica TTL, sendo possível obter posteriormente resultados mais próximo dos desejados quanto utilizada a família HCT [10].

Infelizmente, não foi possível aprofundar nem testar todas as funcionalidade previstas para o projeto, devido ao confinamento geral [6]. O planeamento inicial vinha a ser cumprido até que as circunstâncias mudaram e alguns dos objetivos propostos não puderam ser realizados.

Relativamente ao tempo despendido no desenvolvimento do projeto, visto que todo o grupo se juntou para trabalhar em horários definidos, todos os elementos do grupo trabalharam o mesmo número de horas. Após o desenho e conceção que foi desenvolvido em conjunto, o grupo foi dividido em duas equipas de trabalho, sendo que uma se dedicou à implementação do controlo dos motores e outra ao desenvolvimento da máquina de estados que controla o estado de funcionamento do robô. Assim, foram contabilizadas, individualmente, um total de 250 horas, como mostrado na Tabela 8.1.

Tabela 8.1 - Número de horas despendidas por elemento

|  |  |
| --- | --- |
| Nome | Número de Horas |
| Bruno Silva | 250 |
| Diogo Fernandes | 250 |
| Duarte Rodrigues | 250 |
| Francisco Salgado | 250 |
| João Miranda | 250 |
| José Abreu | 250 |

Em suma, este projeto foi desafiante, revelando a sua importância na formação enquanto futuros engenheiros.

## Sugestões de Trabalho Futuro

Durante a realização deste projeto, foram feitas opções que ditaram um rumo. Poder-se-iam ter tomado outras opções que modificariam os resultados obtidos. Este projeto não representa o fim de uma ideia, é, apenas, uma implementação de um conceito. Assim, neste subcapítulo são feitas algumas sugestões de, não só, melhorias à implementação desenvolvida, mas também de novas abordagens para esta ideia.

### 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, não existe nenhum mecanismo de deteção caso o robô saia fora da rota. 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 encontrasse sobre a linha e um objeto obstruísse a sua passagem, este teria de parar de forma a evitar a colisão e possíveis danos materiais de ambas as partes. Além disso, o robô deveria 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 sobre o robô. Assim que o utilizador ou operador colocassem o tabuleiro sobre o robô, após o término de um temporizador, 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 deveria ser acionado após finalizar o seu percurso (ida e volta).

Como o robô funciona à base de baterias, estas poderiam ser recarregadas numa estação de carregamento. Para isso, poderia ser desenvolvida uma estação de carregamento onde se colocaria o robô, sempre que não estivesse em utilização.

Tal como foi visto, apontaram-se apenas algumas formas de melhorar o trabalho desenvolvido. Possivelmente, existirão outras melhorias a serem 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 será visto na unidade curricular de Laboratórios e Práticas Integradas II (LPI II).

Como o propósito do robô é fazer chegar bens a alguém que se encontre numa situação de isolamento, o seu aspeto físico seria idêntico ao implementado.

Através do uso de um microcomputador, existirão múltiplas possibilidades de implementação. Uma delas seria o uso do controlo remoto, através do qual uma pessoa responsável controlaria o percurso do robô até ao local pretendido. Teria de ser adicionada uma câmara ao robô, sendo a imagem transmitida para o comando, ou algum dispositivo eletrónico com ecrã, 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. Além disso, o robô poderia estimar se teria carga suficiente nas baterias de modo a conseguir completar a trajetória pretendida. Caso não se verificasse, dirigir-se-ia automaticamente à sua estação de carregamento.

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 [19] e implementar as funcionalidades referidas no capítulo anterior.

Neste subcapítulo 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] “Conheça Jaci: o robô de desinfecção que auxilia no combate a Covid-19,” Tecnopuc, 29 abril 2020. [Online]. Available: https://www.pucrs.br/tecnopuc/2020/04/29/conheca-jaci-o-robo-de-desinfeccao-que-auxilia-no-combate-covid-19/. [Acedido em 8 fevereiro 2021].

[3] E. L. Brand Talk, “Pandemic and the Smarter World: A Future of Robots?” 5 maio 2020. [Online]. Available: https://www8.gsb.columbia.edu/articles/brand-talk/pandemic-and-smarter-world-future-robots. [Acedido em 10 fevereiro 2021].

[4] R. K. Erico Guizzo, “How Robots Became Essential Workers in the COVID-19 Response,” IEEE SPECTRUM, 30 setembro 2020. [Online]. Available: https://spectrum.ieee.org/robotics/medical-robots/how-robots-became-essential-workers-in-the-covid19-response. [Acedido em 10 fevereiro 2021].

[5] 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].

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

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

[8] STMicroelectronics, “DUAL FULL-BRIDGE DRIVER”. L298 datasheet, janeiro 2000.

[9] 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].

[10] T. Corporation, “TENERGY 18650 2200 mAh Li-Ion Cell”. Tenergy 18650 datasheet.

[11] “BMS PARA PROTECÇÃO BATERIAS 18650 3S 12,6V 20A,” [Online]. Available: https://www.botnroll.com/pt/acessorios/2558-bms-para-protec-o-baterias-18650-3s-12-6v-20a.html. [Acedido em 10 fevereiro 2021].

[12] Z. Electromotor. ZGB37RG datasheet.

[13] T. Instruments, “LM340, LM340A and LM78xx Wide VIN 1.5-A Fixed Voltage Regulators”. LM7805 datasheet, setembro 2000 [revisto a julho de 2016].

[14] P. Carvalhal, “Fiabilidade e boas práticas de projeto,” 2014. [Online]. Available: https://elearning.uminho.pt/bbcswebdav/pid-1045855-dt-content-rid-3987823\_1/courses/2021.9305O4\_1/FiabilidadeBoasPraticasProjeto\_PCarvalhal.pdf. [Acedido em 3 fevereiro 2021].

[15] “Electrostatic-sensitive device,” Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Electrostatic-sensitive\_device. [Acedido em 8 fevereiro 2021].

[16] TUR CERT, “O que é o certificado CE?” [Online]. Available: https://www.ceisaret.com/pt/ce-sertifikasi-nedir/. [Acedido em 2 fevereiro 2021].

[17] Jornal Oficial da União Europeia, “DIRETIVA 2014/35/UE DO PARLAMENTO EUROPEU E DO CONSELHO,” 29 março 2014. [Online]. Available: https://eur-lex.europa.eu/legal-content/PT/TXT/PDF/?uri=CELEX:32014L0035&from=EN. [Acedido em 8 fevereiro 2021].

[18] A. G. e. I. Cardoso, “Fecha tudo. Escolas e universidades em casa a partir de sexta-feira.,” Jornal de Notícias, 21 janeiro 2021. [Online]. Available: https://www.jn.pt/nacional/fecha-tudo-escolas-e-universidade-em-casa-a-partir-de-sexta-feira-13256762.html. [Acedido em 3 fevereiro 2021].

[19] P. Corporation, “pololu.com,” 2001-2014. [Online]. Available: https://www.pololu.com/docs/pdf/0J12/QTR-8x.pdf. [Acedido em 3 fevereiro 2021].

[20] G. Bocock, “Electrolytic Capacitors Determine The Lifetime Of A Power Supply,” XP Power, [Online]. Available: https://www.xppower.com/resources/blog/electrolytic-capacitor-lifetime-in-power-supplies. [Acedido em 11 fevereiro 2021].

[21] “How can I estimate the life of ceramic capacitors?” Taiyo Yuden, [Online]. Available: https://www.yuden.co.jp/eu/product/support/faq/q020.html. [Acedido em 11 fevereiro 2021].

[22] “How to Prolong the Life of an 18650 Battery,” instructables ciruits, [Online]. Available: https://www.instructables.com/How-to-Prolong-the-Life-of-an-18650-Battery/. [Acedido em 11 fevereiro 2021].

[23] A. Perzan, “Brushed vs. brushless DC motors,” drive.tech, [Online]. Available: https://drive.tech/en/stream-content/brushed-vs-brushless-dc-motors. [Acedido em 11 fevereiro 2021].

[24] T. Instruments, “Calculating Useful Lifetimes of Embedded Processors,” 2014. [Online]. Available: https://www.ti.com/lit/an/sprabx4b/sprabx4b.pdf?ts=1612984192026. [Acedido em 11 fevereiro 2021].