



**INSTITUTO SUPERIOR DE ENGENHARIA  
DE LISBOA**

**Eficiência energética e controlo de iluminação em edifícios**

**José Eduardo Cardoso Antão**  
(Licenciado)

Dissertação para obtenção do Grau de Mestre em  
**Engenharia Eletrotécnica – Ramo de Energia**

Presidente: **Professor Doutor José Manuel Prista do Valle Igreja**

Orientador: **Professor Eng.º José Luís Rosa de Almeida**

Co-orientador: **Professor Doutor Pedro Miguens Matutino**

Vogal: **Professor Doutor Acácio João Galhardo Baptista**

**Setembro 2015**



Dissertação realizada sob orientação de  
**Professor Eng.º José Luís Rosa de Almeida**  
e sob co-orientação de  
**Professor Doutor Pedro Miguens Matutino**



## **Resumo**

Na última década, com o aparecimento de lâmpadas com consumos substancialmente mais reduzidos, e mais recentemente, com a evolução da tecnologia LED, o mercado de iluminação tem tido uma evolução muito significativa. Estes novos produtos apresentam um enorme avanço no que concerne, designadamente, ao tempo de vida útil, à eficiência energética e à qualidade de luz.

Com o aparecimento de sistemas de controlo e gestão da iluminação, baseados em eletrónica digital, é possível o controlo dos ambientes de forma mais eficiente, cómoda e económica, tendo estes vindo contribuir para a evolução dos sistemas de iluminação. Paralelamente, a legislação em vigor exige uma redução do consumo de energia por unidade de área, ao mesmo tempo, manter os níveis de iluminação exigidos a cada local, o que releva a importância destes sistemas de comando, controlo e gestão automática da iluminação artificial.

Assim, e dado que a eficiência da iluminação artificial resulta da utilização e conjugação de diversos meios tecnológicos, este trabalho desenvolveu um controlador de iluminação compatível com as tecnologias de iluminação usadas na atualidade, capaz de cumprir cos requisitos de controlo de iluminação exigidos por legislação, promovendo a redução dos consumos e aumentando a eficiência energética em edifícios.

## **Palavras-chave**

*Eficiência Energética, Iluminação, Controlo de Iluminação, Microcontrolador*



## ***Abstract***

In the last decade, the appearance of high efficiency lamps, and more recently, with the development of LED technology, the lighting systems had a significant evolution. These new products achieve better lifetime cycles, also reduces the energy consumption with better light quality.

The novel of control and management lighting systems, based on digital electronics, allows a more efficient, convenient and economical control of the environments, contributing to the evolution of lighting systems.

Futhermore, new legislation points out to a reduction of energy consumption per area unit, maintaining the required illumination requirements for each facility. Consequently, these lighting control and automatic management systems gain importance nowadays.

Therefore, this work focus on developing a lighting controller, to promote the reduction of energy consumption and increase energy efficiency in buildings. The developed prototype is compatible with the state of art lighting technologies, and fulfill the regulamentations.

## **Keywords**

*Energy Efficiency, Lighting, Light Control, Microcontroller*



## **Agradecimentos**

*Aos meus Pais, Francisco Edgar Antão e Maria de Fátima Cardoso Antão, por toda a dedicação e esforço que tiveram ao longo destes anos e por estarem sempre do meu lado. Obrigado Pai e Mãe.*

*Ao Professor Eng.º José Luís Rosa de Almeida e ao Professor Doutor Pedro Miguens Matutino do Instituto Superior de Engenharia de Lisboa, desejo-lhes expressar o meu agradecimento pela oportunidade de desenvolvimento deste trabalho e orientação. A sua disponibilidade incondicional, a sua forma crítica e criativa de arguir as ideias apresentadas, deram rumo a esta dissertação, facilitando o alcance dos meus objetivos; bem hajam, estou-lhes muito grato.*

*A todos aqueles que contribuíram direta ou indiretamente para a elaboração desta dissertação desejo ainda expressar o meu agradecimento.*



## Índice

<b>1. Introdução</b> .....	1
1.1. <b>Motivação</b> .....	3
1.2. <b>Objetivo</b> .....	4
1.3. <b>Estrutura</b> .....	4
<b>2. Estado da arte</b> .....	7
2.1. <b>Conceitos de Luminotecnia</b> .....	9
2.1.1. <b>Grandezas fotométricas</b> .....	9
2.1.2. <b>Grandezas luminotécnicas</b> .....	10
2.2. <b>Equipamento luminotécnico</b> .....	11
2.2.1. <b>Fontes de luz artificial</b> .....	11
2.2.2. <b>Balastros</b> .....	17
2.2.3. <b>Fontes de alimentação de LED</b> .....	18
2.2.4. <b>Luminária</b> .....	19
2.3. <b>Controlo de iluminação</b> .....	20
2.3.1. <b>Controlo manual</b> .....	22
2.3.2. <b>Controlo com Interruptores horários</b> .....	22
2.3.3. <b>Controlo com Interruptores crepusculares</b> .....	22
2.3.4. <b>Controlo por ocupação</b> .....	23
2.3.5. <b>Controlo de fluxo automático</b> .....	24
2.4. <b>Iluminação natural</b> .....	25
2.5. <b>Sistemas de controlo automatizado de Iluminação</b> .....	27
2.6. <b>Normas e Legislação</b> .....	30
2.6.1. <b>Requisitos de iluminância</b> .....	30
2.6.2. <b>Requisitos de controlo, regulação e monitorização</b> .....	31
2.6.3. <b>Requisitos de densidade de potência e fatores de controlo</b> .....	32
<b>3. Protótipo de controlo de iluminação</b> .....	33
3.1. <b>Arquitetura do protótipo</b> .....	35
3.2. <b>Especificação do sistema</b> .....	36
3.2.1. <b>Processador</b> .....	36
3.2.2. <b>Interface com o utilizador</b> .....	38
3.2.3. <b>Controlo de iluminação</b> .....	42
3.2.3.1. <b>Modo on/off Manual</b> .....	42
3.2.3.2. <b>Controlo por Detecção de ocupação</b> .....	42
3.2.3.3. <b>Controlo horário</b> .....	44
3.2.3.4. <b>Regulação de luminosidade</b> .....	46
3.2.3.5. <b>I2C</b> .....	47
3.2.3.6. <b>Armazenamento de configurações</b> .....	49
3.3. <b>Implementação</b> .....	49
3.3.1. <b>Hardware</b> .....	49
3.3.2. <b>Código Arduino</b> .....	50
<b>4. Testes e Resultados experimentais</b> .....	53
4.1. <b>Introdução</b> .....	55
4.2. <b>Caracterização da luminária fluorescente</b> .....	55
4.3. <b>Caracterização da luminária led</b> .....	56
4.4. <b>Caracterização do espaço</b> .....	59
4.5. <b>Custo do Automato</b> .....	62

<b>5. Conclusão</b> .....	65
<b>5.1. Conclusões</b> .....	67
<b>5.2. Trabalho futuro</b> .....	68
Anexo A – PWM.....	73
Anexo B – Source Code .....	75

## Lista de Figuras

Figura 2. 1 – Ilustração das principais propriedades fotométricas .....	9
Figura 2. 2 - Lâmpadas Incandescentes .....	12
Figura 2. 3 - Lâmpadas fluorescentes.....	13
Figura 2. 4 – Exemplo de luminárias LED.....	14
Figura 2. 5 – Eficiência de fontes de luz artificiais [1] .....	15
Figura 2. 6 – Etiqueta energética[7] .....	16
Figura 2. 7 - Balastro ferromagnético .....	17
Figura 2. 8 – Balastro eletrónico .....	18
Figura 2. 9 – Fonte de alimentação de leds .....	19
Figura 2. 10 – Luminárias .....	20
Figura 2. 11 – Interruptor horário digital .....	22
Figura 2. 12 - Sensor de infravermelhos .....	23
Figura 2. 13 - Regulação dos níveis de iluminação artificial a 1% e a 100 % .....	24
Figura 2. 14 – Exemplo de aproveitamento da luz natural.....	26
Figura 2. 15 - Exemplo de sectorização de circuitos.....	26
Figura 2. 16 - Sistema Dali integrado em rede KNX.....	29
Figura 2. 17 – Esquema de ligações para balastros eletrónicos com regulação de fluxo analógica e digital.....	29
Figura 2. 18 – Exemplo de sinal PWM .....	74
Figura 3. 1 – Diagrama de blocos do protótipo.....	36
Figura 3. 2 – Características e placa Arduino Deumilanove.....	37
Figura 3. 3 – Imagem base do Interface .....	38
Figura 3. 4 – Primeiro menu do interface.....	38
Figura 3. 5 – Menu de Controlo .....	39
Figura 3. 6 – Menu Definições.....	39
Figura 3. 7 – Menu para configuração hora ligar/hora desligar .....	40
Figura 3. 8 – Menu Luminosidade .....	40
Figura 3. 9– Características LCD.....	41
Figura 3. 10 – a) rotary encoder ; b) sinais gerados .....	41
Figura 3. 11 - Exemplo do funcionamento do PIR .....	43
Figura 3. 12 - Sensor de infravermelhos PIR .....	44
Figura 3. 13 – Fluxograma da função de controlo horário.....	45
Figura 3. 14 – Resposta espectral do sensor BH1750 e resposta espectral do olho humano ...	46
Figura 3. 15 – Fluxograma da função de controlo de luminosidade .....	47
Figura 3. 16 – Exemplo de linha de comunicação I2C .....	48
Figura 3. 17 – Esquema de ligações do protótipo .....	50
Figura 3. 18 - Fluxograma do código desenvolvido para controlo de iluminação- 1- void Setup( ); 2)- void Loop;.....	51
Figura 4. 1 - Caracterização da luminária Fluorescente.....	57
Figura 4. 2- Caracterização da luminária LED .....	58
Figura 4. 3 – Consumo/ Luminosidade - luminária LED VS luminária fluorescente.....	59
Figura 4. 4 – Planta do espaço em estudo .....	60

Figura 4. 5 - Influência da luz natural na área de trabalho do espaço em estudo.....	61
Figura 4. 6 – Análise da influência da luz natural na área de trabalho .....	61

## **Lista de Tabelas**

Tabela 2. 1 – Requisitos de iluminação para escritórios [9] .....	31
Tabela 2. 2 – Requisitos de controlo de iluminação para edifícios .....	31
Tabela 2. 3 – Requisitos de densidade de potência e fatores de controlo .....	32
Tabela 3. 1 – Características do RTC.....	45
Tabela 4. 1 – Custo do sistema para 1, 100 e 1000 unidades [16] [17] .....	62

## Lista de Siglas e Símbolos

$\alpha$  – Ângulo limite da CDL [°]

$\theta$  – Ângulo entre a superfície iluminada e a área aparente [°]

$\phi$  – Diâmetro [m]

$\Phi$  – Fluxo luminoso [lm]

$\Phi_l$  – Fluxo luminoso irradiado pela fonte de luz [lm]

$\Phi_L$  – Fluxo luminoso irradiado pela luminária [lm]

[cd/m<sup>2</sup>] - Candelas por metro quadrado

$A_p$  – Área aparente [m<sup>2</sup>]

CDL – Curva de distribuição luminosa [cd/klm]

$D$  – Densidade de potência [W/m<sup>2</sup>]

*Dali* – Digital Addressable Lighting Interface

$d$  – Distância entre a fonte luminosa e a superfície iluminada [m]

$E$  – Iluminância [lx]

$E_m$  – Iluminância média [lx]

$E_{min}$  – Iluminância mínima [lx]

*EEPROM* – *Electrically-Erasable Programmable Read-Only Memory*

*FT* – Fluorescente tubular

*FC* – *Fluorescente compacta*

*FCI* - *Fluorescente compacta c/ equipamento auxiliar integrado*

*FCNI - Fluorescente compacta c/ equipamento auxiliar não integrado*

*GND – Ground (massa)*

*hpt – Altura do plano de trabalho [m]*

*PIR - Pyroelectric InfraRed Sensor*

*RTC – Real Time Clock*

*SPI - Serial Peripheral Interface*

*i – Corrente elétrica [A]*

*I – Intensidade luminosa [cd]*

*I2C – Inter-Integrated Circuit*

*IDE – Integrated Development Environment*

*LCD – Liquid Crystal Display*

*Lm- Lúmen*

*L – Luminância [cd/m<sup>2</sup>]*

*LED - (light-emitting diode)*

*ni – Número de equipamentos instalados no sistema de iluminação*

*nl – Número de lâmpadas*

*nL – Número de luminárias*

*nl,L – Número de lâmpadas por luminária*

*$\eta_L$  – Eficiência luminosa da lâmpada [W]*

*$\eta_W$  – Eficácia luminosa da lâmpada [lm/W]*

*P<sub>i</sub> – Potência do equipamento i [W]*

*P<sub>l</sub> – Potência da lâmpada [W]*

*P<sub>le</sub> – Potência da lâmpada com o equipamento auxiliar [W]*

*P<sub>t</sub> – Potência total instalada [W]*

*PMW – Pulse-Width Modulation*

*t – Tempo [s]*

*T<sub>c</sub> – Temperatura de cor [K]*

*USB - Universal Serial Bus*

*V<sub>cc</sub> – Tensão de alimentação [V]*

*W<sub>vis</sub> – Energia luminosa [J]*



---

# **CAPÍTULO 1**

---

## **INTRODUÇÃO**

*Neste capítulo apresentam-se as motivações e os principais objetivos do trabalho. No fim deste capítulo apresenta-se a estrutura do restante documento.*



## 1.1. MOTIVAÇÃO

O consumo energético dos sistemas de iluminação artificial nos edifícios de serviços é um dos mais significativos, representando cerca de 30 % do valor global. Além disso, a baixa eficiência dos sistemas de iluminação levam a um agravamento em termos económicos e ambientais, dado que grande parte da produção de eletricidade resulta na emissão de CO<sub>2</sub>.

É assim, importante que um edifício apresente baixos consumos de energia promovendo, para o efeito, a implementação de sistemas mais eficientes, dando cumprimento às normas europeias em vigor.

No caso dos edifícios de serviços, a atualização dos sistemas de iluminação apresenta um potencial de poupança no consumo global de energia de até 80% [2].

Na última década, o sector dos edifícios, apresentou a mais alta taxa de crescimento do consumo de energia de entre todos os sectores da economia nacional, nomeadamente, no subsector dos serviços, com um crescimento de 12% por ano.

No contexto internacional é consensual a necessidade de reduzir os consumos de energia tendo em vista a redução das emissões de gases que contribuem para o aumento do efeito de estufa. Para se atingir esta redução do consumo energético é necessário desenvolver sistemas de controlo de iluminação mais eficientes, mantendo o conforto dos seus utilizadores.

O aumento da preocupação com os consumos de energia conduziu a que a legislação tenha passado a limitar a potência elétrica utilizada para obter o valor de iluminância adequado a cada local e o sistema de controlo respetivo. A legislação que enquadra estes consumos é a EN 15193, e introduziu o LENI (Lighting Energy Numeric Indicator) para estimar o consumo de energia por unidade de área durante um ano.

A energia consumida pela iluminação artificial já é quantificada como parcela da energia total necessária ao funcionamento dos edifícios.

Ao pretender reduzir-se o consumo de energia por unidade de área durante um ano e ao mesmo tempo manter os níveis de iluminância exigidos torna-se de grande

importância a utilização de sistemas de comando e regulação que adequem a iluminação artificial a cada local e momento, bem como uma criteriosa escolha de luminárias eficientes.

## **1.2. OBJETIVO**

Pretende-se desenvolver um controlador que permita gerir as tecnologias de iluminação artificial mais eficientes que integre as diversas metodologias de controlo de iluminação mais usadas na atualidade e dando cumprimento aos requisitos impostos por legislação para sistemas de controlo de iluminação em edifícios.

## **1.3. ESTRUTURA**

O presente documento da dissertação está organizado em mais quatro capítulos. O segundo Capítulo apresenta o estado da arte, englobando os princípios básicos de luminotecnia, alguns conceitos inerentes ao estudo da aplicação de sistemas de iluminação nos espaços, designadamente, as grandezas fotométricas, equipamento luminotécnico como lâmpadas, luminárias e balastros. É apresentada uma análise comparativa da eficiência dos mesmos equipamentos e são apresentados os diferentes sistemas de controlo de iluminação artificial e natural. Para finalizar o capítulo são apresentados os sistemas de gestão automatizada da iluminação, o protocolo de comunicação mais usados na atualidade e a legislação em vigor que afeta este tipo de sistemas.

O Capítulo 3 descreve o protótipo desenvolvido para controlo da iluminação em edifícios e as metodologias de controlo usadas. É apresentada a caracterização e arquitetura do protótipo de controlo de iluminação desenvolvido, bem como o Hardware/Software, a plataforma de desenvolvimento de *open source* o “Arduino”, protocolos de comunicação e os restantes componentes necessários para o desenvolvimento do sistema de controlo de iluminação.

No Capítulo 4 são apresentados os testes e resultados experimentais. Para o efeito, são testadas as metodologias de controlo implementadas no autómato para o controlo de

iluminação em edifícios, tendo sido contabilizada a energia despendida sem qualquer tipo de controlo e à posteriori com as metodologias implementadas no autómato.

Apresentam-se e comparam-se os resultados destas simulações, com equipamentos e casos reais.

No Capítulo 5 são apresentadas as conclusões e possíveis direcções de trabalho futuro que poderão ser implementadas como continuação deste estudo. É evidenciada uma síntese do trabalho e apresentam-se as principais conclusões que se extraíram do estudo desenvolvido para a resolução do problema de eficiência energética. Apontam-se ainda algumas direcções em que pode ser desenvolvido o protótipo.



---

## CAPÍTULO 2

---

### **ESTADO DA ARTE**

*Neste Capítulo descrevem-se os conceitos inerentes ao estudo da aplicação de sistemas de iluminação em edifícios, designadamente as grandezas fotométricas e luminotécnicas.*

*São apresentados os equipamentos luminotécnicos e é apresentada uma análise comparativa da eficiência destes.*

*Apresentam-se, igualmente, os diferentes tipos de controlo de iluminação, demonstrando-se a importância da iluminação natural em sistemas de iluminação e a eficiência dos sistemas de controlo automatizado.*

*No final deste capítulo é apresentada a legislação e normas em vigor, que enquadram este tipo de sistemas.*



## 2.1. CONCEITOS DE LUMINOTECNIA

Para a compreensão da terminologia do trabalho seguem algumas definições e termos utilizados para descrever as propriedades das fontes luminosas e os efeitos que estas produzem no Ser Humano.

### 2.1.1. GRANDEZAS FOTOMÉTRICAS

- **Fluxo luminoso** - O fluxo luminoso ( $\Phi$ ) ou potência luminosa é a quantidade total de radiação visível emitida por uma fonte de luz por unidade de tempo. A sua unidade é o Lúmen [lm].
- **Intensidade luminosa** - A intensidade luminosa ( $I$ ) é o fluxo luminoso irradiado numa determinada direção. A sua unidade é Candela [cd].
- **Iluminância** - A Iluminância ( $E$ ), é o fluxo luminoso, irradiado por uma fonte luminosa, que incide sobre a área de uma superfície. A sua unidade é o Lux [lx].
- **Luminância** – Luminância ( $L$ ) é a intensidade luminosa produzida ou refletida por uma superfície. A unidade de medida da luminância [cd/m<sup>2</sup>].

Para ajudar a compreensão das grandezas descritas anteriormente, a figura 2.1 ilustra as principais propriedades fotométricas.

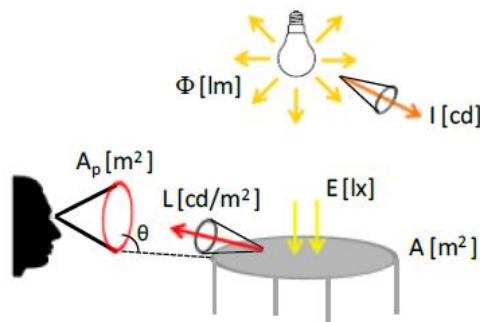


Figura 2. 1 – Ilustração das principais propriedades fotométricas

- **Temperatura de cor** - A temperatura de cor,  $T_c$ , define a aparência de cor da luz branca emitida por uma fonte luminosa. Quanto maior for a temperatura de cor, mais clara é a tonalidade de cor da luz branca.

- **Restituição cromática** - A capacidade com que uma lâmpada reproduz as cores de superfícies iluminadas é designada por índice de reprodução de cor ou índice de restituição cromática ( $IRC$ ), quando comparado com uma fonte luminosa de referência com temperatura de cor idêntica à cor do objeto. Este índice é representado numa escala de 0 a 100, nomeadamente de mau a muito bom. A norma EN 12464-1, referente à iluminação de interiores, recomenda valores de  $IRC$  superiores a 80.

### 2.1.2. GRANDEZAS LUMINOTÉCNICAS

- **Potência total instalada** - A Potência total instalada,  $P_t$ , é a soma das potência de todos os equipamentos que fazem parte de um sistema de iluminação ou seja, as fontes de luz e equipamentos auxiliares. A sua unidade é o Watt [W].

- **Densidade de Potência** - A densidade de potência,  $D$ , indica a potência total instalada por metro quadrado de um espaço. A sua unidade é o Watt por metro quadrado [W/m<sup>2</sup>].

- **Eficiência luminosa** - A eficiência luminosa de uma fonte de luz artificial é representada pela razão entre o fluxo luminoso produzido ( $\Phi_l$ ) e a potência da fonte luminosa ( $P_l$ ). A sua unidade é o lúmen por Watt [lm/W].

## **2.2. EQUIPAMENTO LUMINOTÉCNICO**

Os sistemas de iluminação são compostos por vários tipos de equipamentos, nomeadamente as fontes de luz artificiais, as luminárias, os balastros, os comandos de circuitos de iluminação.

Nos pontos seguintes é feita uma apresentação de alguns dos equipamentos luminotécnicos mais usados na atualidade, bem como uma análise da sua eficiência energética e aplicabilidade.

### **2.2.1. FONTES DE LUZ ARTIFICIAL**

As fontes de luz artificial, são os principais equipamentos de um sistema de iluminação. Existem fontes de diversas tecnologias com diferentes características e eficiência, distinguindo-se as lâmpadas de incandescência, as lâmpadas de descarga e os LEDs.

#### **- Lâmpadas incandescentes**

O princípio de funcionamento das lâmpadas incandescentes é através da passagem de corrente elétrica por um filamento com resistência elevada, aquecendo-o por efeito de Joule até este atingir a incandescência e por sua vez emitir luz. Este tipo de lâmpadas encontram-se em desuso pois apresentam muito baixa eficácia luminosa e com uma vida útil de curta duração (cerca de 1 000 horas), porém, ainda são bastante usadas a nível doméstico.

A União Europeia publicou uma Diretiva (Diretiva 2005/32/CE)[1] e um Regulamento (Regulamento 2009/244/CE)[1] para a eliminação progressiva das lâmpadas incandescentes e de halogéneo.

Atualmente em Portugal a comercialização deste tipo de lâmpadas está muito condicionada e sujeita a taxas elevadas (Decreto-Lei 108/2007, Portaria 54/2008, Portaria 63/2008). A Figura 2.2 ilustra alguns tipos de lâmpadas incandescentes.



**Figura 2. 2 - Lâmpadas Incandescentes**

### **- Lâmpadas fluorescentes**

A lâmpada fluorescente tubular (FT) é constituída por um tubo de descarga comprido com um formato cilíndrico e um elétrodo em cada extremidade. O tubo contém no seu interior uma mistura de gás inerte com uma pequena quantidade de vapor de mercúrio a baixa pressão e o seu interior é revestido por uma substância fluorescente que transforma a radiação UV, em radiação visível. Estas são mais económicas que as incandescentes, pois dissipam menos energia sob forma de calor. Durante o seu funcionamento, necessitam de um equipamento auxiliar designado de balastro para limitar a corrente e evitar uma ionização excessiva e descontrolada, que levaria à destruição da lâmpada. As lâmpadas fluorescentes tubulares (FT) têm uma eficácia luminosa alta (100lm/W) e uma vida útil longa (7500 a 10000 horas), em comparação com as lâmpadas incandescentes. Estas apresentam um índice de restituição cromática de 85 a 95 e uma temperatura de cor de 2700°K a 5000°K.

As lâmpadas fluorescentes tubulares são de aplicabilidade diversa, sendo usadas em ambientes de trabalho como escritórios, laboratórios ou cozinhas, estas também são utilizadas para fins específicos, como lâmpadas ultravioleta, lâmpadas negras e lâmpadas coloridas.

A lâmpada fluorescente compacta (FC) tem um princípio de funcionamento idêntico ao da fluorescente tubular, porém como o nome indica tem uma forma mais compacta, através de um tubo de descarga em forma de U ou em forma de espiral. Estas apresentam uma menor eficiência luminosa, relativamente às tubulares, porém relativamente às de incandescência apresentam eficiência

luminosa superior. As lâmpadas fluorescentes compactas com balastro incorporado designam-se por lâmpada fluorescente compacta integrada (FCI). As lâmpadas fluorescentes compactas com o equipamento auxiliar não integrada (FCNI), permitem efetuar controlo de fluxo se possuírem uma base de 4 pinos, enquanto as FCI não é possível efetuar controlo de fluxo. A Figura 2.3 apresenta alguns tipos de lâmpadas fluorescentes [1].



**Figura 2. 3 - Lâmpadas fluorescentes**

#### **- Díodo emissor de luz - LED**

Um díodo emissor de luz ou mais comumente o LED (Light-Emitting Diode) é uma fonte luminosa que começou recentemente a ser utilizada em diversos sistemas de iluminação, desde iluminação interior, exterior, sinalização, etc.

O princípio de funcionamento deste dispositivo semicondutor baseia-se no fenómeno da eletroluminescência que ocorre em junções p-n constituídas por materiais semicondutores adequados para o efeito. Este fenómeno ocorre em junções p-n quando é aplicada, por um circuito externo, uma tensão entre o ânodo e o cátodo.

Os LED são de todas as fontes de luz artificial as que apresentam a melhor eficácia luminosa, são compactos e leves, permitem um melhor controlo da luz, a sua vida não é afetada pelos curtos ciclos de funcionamento ou frequência de acendimentos, é uma tecnologia resistente, além de constituírem a solução mais amiga do ambiente pelo seu muito baixo consumo e ausência de mercúrio. Têm como grandes limitações o elevado custo inicial e o facto da eficácia luminosa e

vida útil diminuam fortemente com o aumento da temperatura, sendo necessário a ligação a dissipadores de calor e de um equipamento auxiliar designado de driver para limitar a corrente de funcionamento.

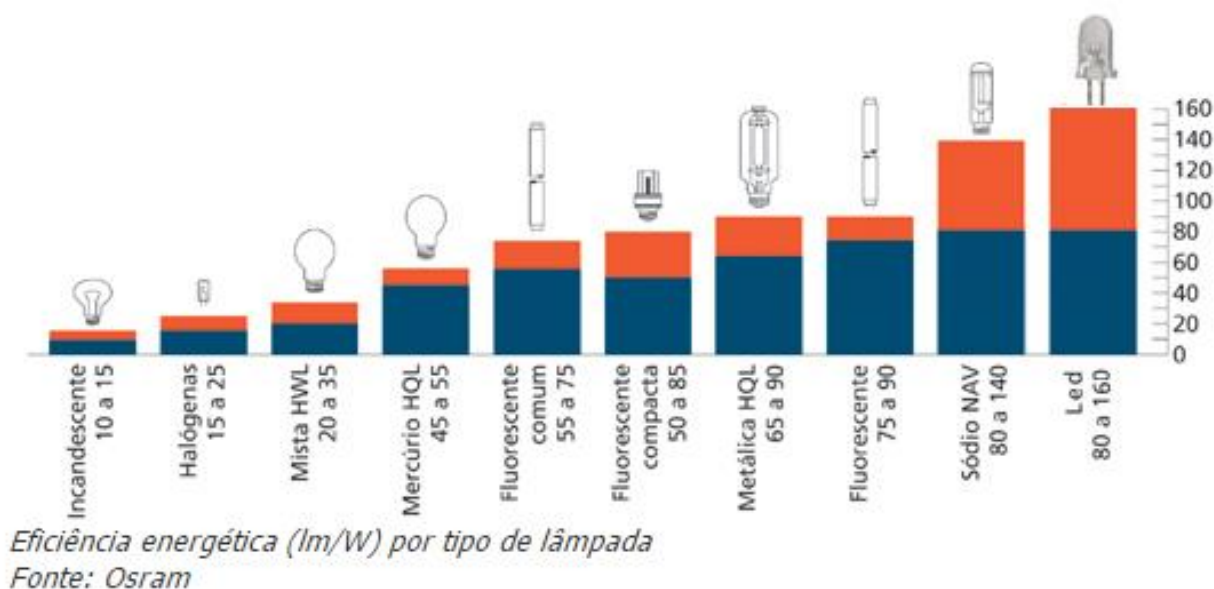
Devido à sua grande versatilidade, existem soluções LED para quase todas as aplicações desde iluminação de interiores a iluminação pública. Como produzem várias cores, também são ideais para a iluminação arquitetónica ou decorativa e por apresentarem uma tecnologia de estado sólido (sem filamento ou bolbo de vidro) são uma solução ideal para locais sujeitos a choques ou vibrações, tais como automóveis, indústria e iluminação pública. As lâmpadas comuns têm a característica de emitir luz em todas as direções. Para muitas aplicações, este facto resulta numa porção de luz a ser desperdiçada, sendo necessário o uso de refletores para tornar a luz mais direcional, não evitando contudo algumas perdas. Como os LED são montados numa superfície plana, eles emitem a luz hemisféricamente, em vez de esféricamente o que leva a uma maior eficiência luminosa. A Figura 2.4 apresenta algumas lâmpadas LED.



**Figura 2. 4 – Exemplo de luminárias LED**

Para além das lâmpadas apresentadas anteriormente que se diferenciam por serem as mais usadas em iluminação de edifícios, existem ainda outras tecnologias de lâmpadas que se conseguem encontrar em sistemas de iluminação de edifícios.

De modo a ilustrar melhor a eficiência dos diversos tipos de lâmpadas é apresentado na Figura 2.5 uma comparação da eficiência de diversas fontes de luz artificial. Os intervalos a laranja representam a variação da eficiência energética de cada tecnologia dependendo do tipo de lâmpada sendo que a barra azul representa o valor mínimo de eficiência energética para cada tecnologia.



**Figura 2. 5 – Eficiência de fontes de luz artificiais [1]**

De acordo com a diretiva 98/11/CE1 as lâmpadas destinadas a um uso doméstico devem ser munidas de uma etiqueta ‘energia’ (energy label). Esta nova etiqueta energética entrou em vigor a. dia 1 de setembro de 2013 e classifica as lâmpadas de uso doméstico em sete categorias de eficácia energética. Esta classificação vai de A “muito eficaz” para G para “pouco eficaz”.

A classificação de eficiência energética faz-se de acordo com a Diretiva acima mencionada [7], e depende de valores do fluxo luminoso e da potência das lâmpadas.

Nas etiquetas de classificação energética indica-se [7]:

- a categoria de eficiência energética (de A a G);
- o fluxo luminoso da lâmpada em lumen ;
- a potência elétrica absorvida pela lâmpada em Watt ;
- a duração de vida média em horas;



**Figura 2. 6 – Etiqueta energética[7]**

Nesta Diretiva são apenas incluídas as lâmpadas que produzem luz visível (radiação de 400 a 800 nm). São excluídas as lâmpadas que produzam um fluxo luminoso superior a 6500 lumen, as lâmpadas cuja potência absorvida é inferior a 4 W e as lâmpadas refletoras.

Os LEDs, as lâmpadas fluorescentes e as lâmpadas fluorescentes compactas pertencem normalmente às categorias A e B. As lâmpadas de halogéneo pertencem normalmente à categoria D. As lâmpadas de incandescência pertencem normalmente às categorias E e F.



As lâmpadas, a estas frequências, produzem a mesma quantidade de luz, com menos 12% a 25% de energia consumida e aumentam o seu tempo de vida útil.

Relevam-se algumas vantagens dos balastros eletrônicos sobre os ferromagnéticos como a maior economia de energia, maior eficiência, são mais leves, não emitem ruído e não necessitam de compensação do fator de potência [2].

Alguns destes balastros têm uma entrada para tensão de comando que pode ser analógica ou digital permitindo assim fazer uma regulação do fluxo luminoso e por sua vez da potência absorvida pela carga.

Os balastros eletrônicos são classificados segundo as seguintes classes: A1 para balastros eletrônicos com regulação, A2 para balastros eletrônicos com baixas perdas e A3 para balastros eletrônicos standard[2]. A Figura 2.8 apresenta um balastro eletrônico com regulação de fluxo.



**Figura 2. 8 – Balastro eletrônico**

### **2.2.3. FONTES DE ALIMENTAÇÃO DE LED**

As fontes de alimentação de LED, à semelhança dos balastros eletrônicos já referidos, têm como finalidade converter a tensão alternada da rede em tensão contínua e limitar a corrente de funcionamento dos LEDs.

A qualidade de uma luminária LED depende da fonte de luz LED, da eficiência e fiabilidade da fonte de alimentação. Uma fonte de alimentação adequada deve ter no mínimo o mesmo tempo de vida dos LED, que dura geralmente 50.000 horas (a 70% do fluxo luminoso) [2], uma eficiência mínima de 85% e um fator de potência superior a 0,9.

Algumas destas fontes permitem regular o fluxo luminoso dos LED através de um sistema que utiliza dois condutores para comando, onde impondo um intervalo de tensões, é possível ajustar o nível de luminosidade emitido pela luminária. A Figura 2.9 ilustra uma fonte de alimentação para leds.



**Figura 2. 9 – Fonte de alimentação de leds**

#### **2.2.4. LUMINÁRIA**

A luminária é o conjunto de componentes que servem de invólucro para as lâmpadas e que asseguram as funções de proteção, fixação e condicionamento do fluxo luminoso. Os componentes da luminária para condicionamento do fluxo luminoso emitido pelas lâmpadas são o difusor, as alhetas e o refletor. O difusor tem a função de difundir a luz, proteger as lâmpadas e esconder o equipamento de iluminação, podendo ser transparente, opalino ou prismático. As alhetas e o refletor têm a função de controlar corretamente o fluxo luminoso produzido pela lâmpada ou pelas lâmpadas, conduzindo a um aumento da eficiência luminosa. Os refletores podem ser planos ou parabólicos, pintados em alumínio mate ou em alumínio polido, sendo que os últimos apresentam maior eficiência. Outra propriedade das luminárias é o ângulo de *cut-off*, que é o ângulo entre o teto e uma reta imaginária tangente à lâmpada e que passe pelo extremo do refletor

Todos os sistemas de iluminação produzem calor o que diminui a eficiência das lâmpadas e a sua duração. Para minimizar esta situação algumas luminárias vêm equipadas com sistemas de ventilação ou dissipadores de calor.

A eficiência dos sistemas de iluminação está diretamente relacionada com a eficiência das luminárias. A Figura 2.10 apresenta alguns tipos de luminárias.



**Figura 2. 10 – Luminárias**

As luminárias são classificadas em função da sua distribuição fotométrica. Esta classificação é baseada na percentagem de fluxo luminoso que é emitido em cada direção, sendo os ângulos definidos de acordo com planos de referência. As luminárias também são classificadas segundo os seus índices de estanquidade a corpos sólidos e líquidos (EN 60529), de proteção elétrica e de inflamabilidade do invólucro exterior (EN 60598) e de resistência ao impacto (EN 50102).

### **2.3.CONTROLE DE ILUMINAÇÃO**

A importância da iluminação no consumo de energia foi aumentando ao longo dos anos, até se atingir o reconhecimento atual. Um estudo efetuado pela EDF revelou que o custo de eletricidade destinado à iluminação representa entre 10 a 50% do consumo total de um edifício [1] e obteve os seguintes valores por sector do peso médio na fatura global de eletricidade:

*Residencial 11%*

*Industria 15%*

*Comércio 23%*

*Escritórios 30%*

*Saúde 50%*

*Ensino, desporto 39%*

Este facto veio relevar a importância dos sistemas de comando e de gestão de iluminação, bem como do respetivo projeto, para a redução da fatura de energia.

Neste momento já é legalmente obrigatório dotar as instalações de iluminação com soluções técnicas e equipamentos de eficiência elevada e dotados de sistemas de comando e regulação automático [9].

Os sistemas de iluminação a instalar em edifícios de comércio e serviços devem cumprir requisitos gerais e específicos para os parâmetros de iluminação de acordo com as normas europeias aplicáveis, bem como requisitos para a densidade de potência, requisitos de controlo, de regulação de fluxo e de monitorização e gestão, conforme descrito em seguida. Na fase de projeto os equipamentos deverão ser selecionados considerando o indicado na alínea d) do ponto 9.1.2 do Anexo I, da Portaria 349-D, que se transcreve a seguir.

Equipamentos de controlo e regulação de fluxo eficientes, instalados para funcionamento em modo autónomo/individual, ou interligados em rede para efeitos de gestão de iluminação centralizada, que podem ser aplicados com as seguintes funções de controlo:

- i. Comutação por deteção de movimento ou ocupação (deteção de presença);
- ii. Comutação em função do nível de luz natural (comutação por luz natural);
- iii. Regulação do fluxo luminoso em função do potencial de aproveitamento da luz natural (regulação por luz natural);
- iv. Controlo horário;
- v. Comando a partir de um interface, para controlo, parametrização e monitorização de todos os equipamentos de iluminação na rede (comando por interface);

O controlo de iluminação tem um grande impacto no que diz respeito à poupança de energia em sistemas de iluminação sendo que existem diversos métodos que diferem desde a aplicabilidade, ao aumento da eficiência energética, custo equipamentos, etc. De seguida são apresentados diferentes tipos de controlo de iluminação.

### **2.3.1. CONTROLO MANUAL**

O controlo manual foi o primeiro método de controlo de lâmpadas em função da luz natural disponível. A utilização do simples sistema “on /off” permite uma economia de energia ao apagar-se a luz quando esta não é necessária. A utilização de um sistema de controlo manual de fluxo permite regular o nível de iluminação e assim melhorar a adaptação visual. A eficiência deste sistema é relativa pois depende do bom senso dos utilizadores e da luz natural existente no edifício.

### **2.3.2. CONTROLO COM INTERRUPTORES HORÁRIOS**

Os interruptores horários são equipamentos que permitem comandar circuitos de iluminação num horário pré-configurado pelo utilizador. Existem interruptores horários analógicos e digitais. A Figura 2.11 apresenta um interruptor horário digital.



**Figura 2. 11 – Interruptor horário digital**

### **2.3.3. CONTROLO COM INTERRUPTORES CREPUSCULARES**

Os interruptores crepusculares permitem comandar circuitos de iluminação a partir de um dado nível de iluminância medido com uma célula fotoelétrica. Estes dispositivos permitem fazer um aproveitamento da luz natural apenas ativando os circuitos de iluminação quando não existe luz natural disponível.

### **2.3.4. CONTROLO POR OCUPAÇÃO**

Os sensores de ocupação são detetores de movimento que em conjugação com interruptores eletrónicos, ligam e desligam as lâmpadas automaticamente consoante o espaço está ou não ocupado, resultando assim numa economia de energia. O controlo por ocupação que permite reduzir até 40% dos consumos de energia [5]. De seguida serão apresentados alguns dos sensores usados neste tipo de controlo.

#### **- Sensor passivo de infravermelhos**

O sensor passivo de infravermelhos ou mais conhecido pela sigla PIR (pyro-electric) é a tecnologia mais usada em sistemas de deteção de ocupação. Este sensor reage à energia do calor infravermelho emitida pelas pessoas. São considerados passivos porque apenas detetam radiação, não a emitem.

Os sensores de embutir substituem diretamente o interruptor de parede ou o de teto e são os mais indicados para pequenos ambientes.[5]



**Figura 2. 12 - Sensor de infravermelhos**

#### **- Sensores ultra-sónicos**

Os sensores de ocupação ultra-sónicos ativam um cristal de quartzo que emite ondas ultra-sónicas em frequências superiores ao limite da perceção humana (entre 25 e 45Khz), por meio do espaço, para detetar a presença de ocupantes. Este sinal em alta frequência é comparado com a frequência do sinal refletido (efeito Doppler) e qualquer diferença é interpretada como a presença de alguém no espaço de cobertura. Este tipo de sensor é o mais indicado para uso em espaços abertos, espaços com obstáculos de superfície dura e para altura de montagem inferior a 5 metros [5].

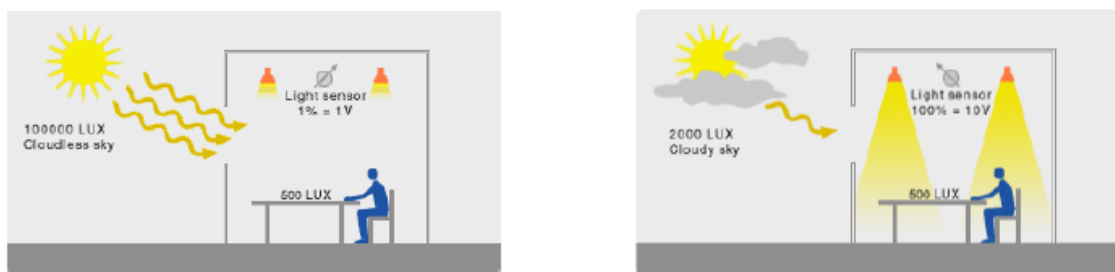
## - Sensores híbridos

Um outro sensor encontrado no mercado é o que utiliza as duas tecnologias, infravermelho e ultra-som. Neste caso, o sistema de iluminação é ativado somente quando ambos detetam a presença de pessoas, o que aumenta a fiabilidade do sistema, evitando que o sistema de iluminação se acenda ou apague desnecessariamente. Por terem um custo maior, são indicados para ambientes em que é necessário um alto grau de deteção, como em salas de aula, salas de conferência, entre outros [5].

### 2.3.5. CONTROLO DE FLUXO AUTOMÁTICO

Os controladores de iluminação usados na atualidade conjugam todos os tipos de controlo apresentados anteriormente. Alguns destes sistemas também permitem fazer um controlo do fluxo luminoso em modo automático.

Num sistema de controlo de fluxo automático os níveis de iluminância provenientes da iluminação natural são detetados por um sensor, que em função dos valores lidos, ajusta e controla o fluxo da iluminação artificial de forma a obter o nível de iluminância desejado. Neste caso os níveis de iluminação artificial são regulados de forma contínua à medida que os níveis de iluminação natural variam, como se pode visualizar na figura 2.16. Com este procedimento evitam-se os problemas dos sistemas liga/desliga e reduz-se o consumo de energia.



**Figura 2. 13 - Regulação dos níveis de iluminação artificial a 1% e a 100 %**

Os controladores automáticos em conjugação com sensores de luminosidade ambiente regulam a intensidade luminosa em função das configurações feitas pelo utilizador.

Estes controladores são microprocessadores programados para através da variação de um sinal analógico, ou seja um valor tensão analógico, ou um sinal de tensão digital, designado por PWM (ver anexo A), conseguem variar a intensidade da luminária de 0% a 100%. De importância mencionar que este tipo de controlo é apenas aplicável a luminárias com balastos eletrónicos ou drivers que permitam fazer regulação de fluxo.

## **2.4. ILUMINAÇÃO NATURAL**

A iluminação natural como fonte de luz constitui um dos principais fatores de qualidade ambiental em espaços internos para além de apresentar um elevado potencial de economia de energia. A economia de energia advém da redução do uso das fontes de luz artificial existentes no edifício. A iluminação natural em conjugação com sistemas de controlo de iluminação artificial pode contribuir para a obtenção de 30 a 70% de economia de energia [2].

Na conceção do projeto deve ser considerado o carácter dinâmico da iluminação natural associado a variações diárias, sazonais e a condições de nebulosidade para se prever possíveis cargas térmicas ou níveis de ofuscamento que possam ocorrer em determinadas horas do dia [2]. Os sistemas de fenestração devem ser dimensionados de modo proporcionarem os níveis de iluminação pretendidos nos espaços interiores durante o maior período de tempo possível [2].

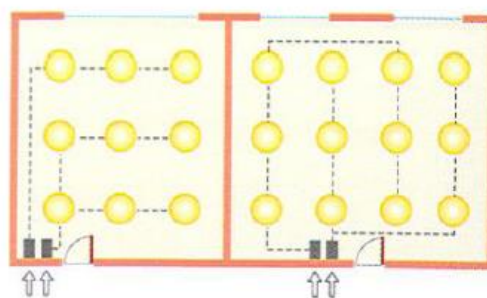
O aproveitamento da luz natural num edifício depende fortemente de fatores arquitetónicos, porém é o projetista do sistema de iluminação que assegura a eficiência energética do mesmo em função da luz natural disponível sem prejuízo das condições do espaço. Um exemplo de gestão da luz artificial em função da luz natural disponível vem exemplificado na Figura 2.15 em que a luminária que está perto da janela apresenta um nível de regulação diferente da luminária mais afastada da janela.



**Figura 2. 14 – Exemplo de aproveitamento da luz natural**

Este aproveitamento depende do sistema de controlo escolhido e de como estão divididos os circuitos de iluminação.

Dividir a iluminação artificial em circuitos comandados separadamente, como ilustra a Figura 2.16, permite que estes possam ser ligados seletivamente em função da iluminação natural disponível em cada zona. Na maioria dos edifícios o aproveitamento da iluminação natural é feito através de janelas situadas na lateral do edifício, logo a distribuição da luz é pouco uniforme. Nesta situação é recomendado que as armaduras sejam colocadas paralelamente à fachada do edifício. A sectorização de circuitos permite que as lâmpadas de cada sector sejam comandadas de tal forma que possam fornecer um fluxo luminoso diferente das localizadas noutros sectores, permitindo assim desligar ou regular a luz artificial em zonas bem iluminadas pela luz natural e ainda desligar a iluminação em zonas desocupadas [2].



**Figura 2. 15 - Exemplo de sectorização de circuitos**

## **2.5. SISTEMAS DE CONTROLO AUTOMATIZADO DE ILUMINAÇÃO**

Estes sistemas eletrônicos permitem o controlo total da iluminação combinando todos os métodos de controlo apresentados anteriormente desde o ajuste da iluminação pela quantidade de luz natural existente, o controlo por sensores de ocupação, o controlo por programação horária, até ao simples interruptor on/off. Apresentam ainda novas funcionalidades como a possibilidade de criação e memorização de cenários, isto é, o utilizador tem a possibilidade de adaptar a iluminação de um local à atividade a realizar no momento.

Este tipo de tecnologias pode ser usado em edifícios industriais ou de serviços como bem como em edifícios residenciais. Estes são caracterizados pelo elevado custo dos equipamentos e da instalação como consequência da separação dos circuitos de potência e de comando.

Este tipo de sistemas são bastante flexíveis, pois possibilitam inúmeras configurações permitindo o controlo de iluminação automático e centralizado possibilitando deste modo uma poupança de energia até 80% [8].

Com o crescente desenvolvimento dos sistemas de controlo de iluminação foi necessário normalizar interfaces digitais de balastros eletrônicos e fontes de alimentação de LED, para tal foram criados protocolos de comunicação que permitiram aos projetistas e fabricantes de componentes de iluminação implementar, criarem soluções de iluminação complexas de forma fácil e conveniente. De seguida é apresentado o protocolo de comunicação mais usado pelos principais fabricantes de sistemas de gestão de iluminação.

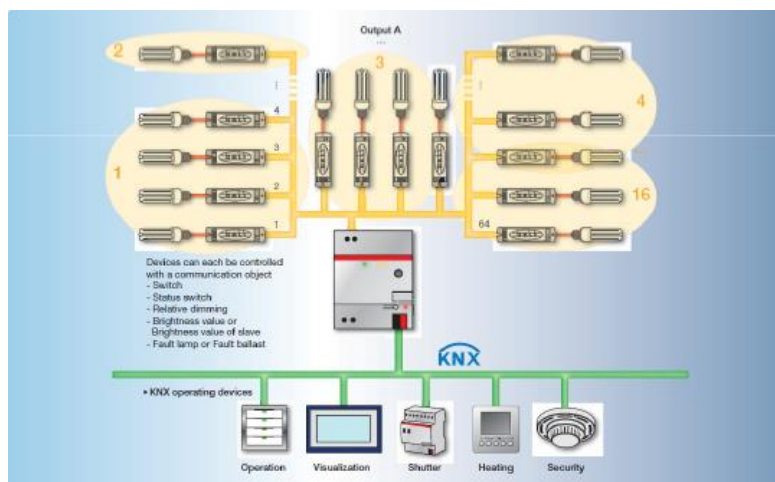
## **- DALI - Digital Addressable Lighting Interface**

Um dos sistemas de gestão automatizada da iluminação mais implementado na actualidade é o DALI (Digital Addressable Lighting Interface). O DALI é um protocolo de comunicação especificado pela norma IEC 60929 que independentemente dos fabricantes dos produtos, assegura compatibilidade e operabilidade de dispositivos com regulação de fluxo luminoso de vários fabricantes. O DALI é a maneira ideal e simplificada de comunicação digital ajustada às necessidades tecnológicas do projeto de iluminação contemporâneo e foi adotada como um novo padrão por importantes fabricantes de equipamentos como Helvar, Hüco, Philips, Osram, Tridonic, Trilux e Vossloh-Schwabe. Ao contrário de outras tecnologias para automação de edifícios, o DALI é especialmente concebido para gestão de iluminação e é orientado para as necessidades do mercado, principalmente para edifícios grandes de serviços e comerciais. Ao longo do tempo foram investidos esforços para que num sistema de iluminação, todas as cargas possam ser controladas separadamente e utilizar apenas um cabo de controlo para todos os dispositivos no sistema. Com uma tecnologia totalmente digital, o protocolo DALI proporciona inúmeros recursos de controlo e gestão dos sistemas de iluminação, sendo o seu funcionamento extremamente simples. A OSRAM é uma das empresas criadoras desta moderna tecnologia que apresenta sistemas completos de gestão da iluminação e componentes que possuem a tecnologia DALI incorporada com objetivo de oferecer o que há de mais moderno em iluminação, onde, além da economia e eficiência, o conforto das pessoas ocupa um papel preponderante.

Estes sistemas têm simples procedimentos de comissionamento podendo integrar outros sistemas de automação (KNX, LON, etc) existentes num edifício. [8]

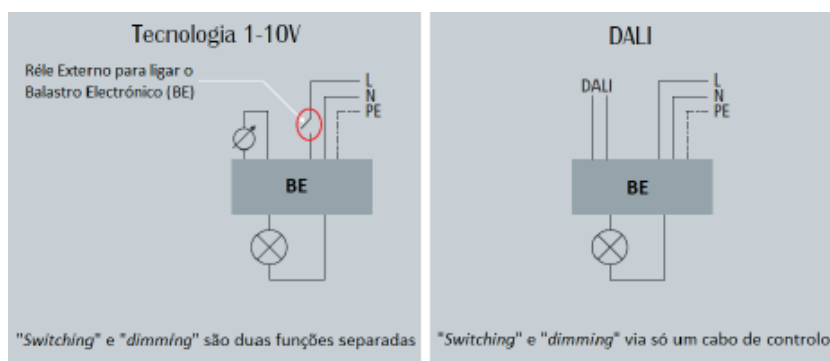
O DALI é um protocolo *standard*, onde podemos ter até 64 dispositivos de campo por controlador DALI, podendo os dispositivos serem ligados por um cabo não blindado de 2 condutores. É uma tecnologia de fácil instalação que possibilita uma ligação totalmente digital de luminárias e outros dispositivos semelhantes em aplicações de

automação de edifícios. Cada controlador DALI pode endereçar individualmente até 64 dispositivos, que podem ser divididos em 16 grupos e 16 cenários de iluminação.



**Figura 2. 16 - Sistema Dali integrado em rede KNX**

O DALI é um sistema mais económico e mais flexível em termos de infraestrutura de cabos em comparação com a tecnologia analógica 1-10V. Ao contrário da tecnologia analógica convencional 1-10V, a energia liga diretamente os dispositivos DALI, não sendo necessário um relé externo como vem representado na Figura 2.18. Por cada rede DALI é necessário um controlador DALI e respetiva alimentação 230 VAC. Na construção de linhas DALI podem ser utilizadas topologias em estrela, árvore ou mista. [8].



**Figura 2. 17 – Esquema de ligações para balastros eletrónicos com regulação de fluxo analógica e digital**

## **2.6. NORMAS E LEGISLAÇÃO**

A EN 12464-1 determina o valor da iluminância adequado a cada atividade ou local de trabalho, e reflete a maior ou menor exigência que se prevê venha a ser solicitada à visão humana. A Diretiva n.º 2002/91/CE, do Parlamento Europeu e do Conselho, de 16 de Dezembro de 2002, relativa ao desempenho energético dos edifícios, conduziu à publicação da EN 15193. Esta limita as potências elétricas a utilizar para obter os valores exigidos pela EN 12464-1, e introduziu o LENI e alguns métodos de cálculo para estimar e depois limitar o consumo de energia utilizado pela iluminação artificial. Em 2013 foram publicados dois documentos legais com muita importância em termos da iluminação, o Dec. Lei n.º 118/2013, de 2013-08-20, e a Portaria n.º 349-D, de 2013-11-29.

Os sistemas de iluminação a instalar em edifícios de comércio e serviços devem cumprir requisitos gerais e específicos para os parâmetros de iluminação, de acordo com as normas europeias referidas anteriormente, bem como requisitos para a densidade de potência, requisitos de controlo, de regulação de fluxo e de monitorização e gestão, conforme descrito de seguida.

### **2.6.1. REQUISITOS DE ILUMINÂNCIA**

Os valores de iluminância recomendados para áreas interiores são indicados no ponto 5.3 “Requisitos de iluminação para espaços interiores, tarefas e atividades” da EN 12464-1. Esta norma apresenta em anexo tabelas com os requisitos de iluminação para a maioria das atividades realizadas em edifícios, especificando três critérios para cada atividade:

- A média de iluminância mínima requerida Em (lux);
- Índice de encadeamento  $UGR_L$ ;
- Reprodução mínima exigida das cores  $R_A$ ;

Para efeitos de aplicação do regulamento de eficiência energética, os valores máximos admissíveis de iluminância não poderão exceder em mais de 30% os valores presentes no ponto 5.3 “Requisitos de iluminação para espaços interiores, tarefas e atividades”

da EN 12464-1. Como exemplo é apresentada a Tabela 2.1 com os requisitos de iluminação apenas para escritórios.

Ref. No.	Atividade	Em(lux)	UGRL	RA
3.1	Fotocópias	300	19	80
3.2	Escrita, leitura, utilização PC, processamento de dados	500	19	80
3.3	Desenho técnico	750	16	80
3.4	Desenho em CAD	500	19	80
3.5	Sala reuniões/ conferências	500	19	80
3.6	Recepção	300	22	80
3.7	Arquivos	200	25	80

**Tabela 2. 1 – Requisitos de iluminação para escritórios [9]**

## 2.6.2. REQUISITOS DE CONTROLO, REGULAÇÃO E MONITORIZAÇÃO

A escolha das soluções de controlo e regulação para qualquer edifício de comércio e serviços, deverá ser feita em função das exigências operacionais das instalações e do potencial de aumento sustentável de eficiência energética que resulte da sua implementação.

Os tipos de controlo obrigatórios para edifícios novos e edifícios sujeitos a grande intervenção vêm apresentados na Tabela 2.2, segundo a Portaria nº 349-D.

	Edifícios novos		Edifícios sujeitos a grande intervenção	
	GES	PES	GES	PES
Deteção de presença(1)	x	x	x	x
Comutação por luz natural(2)	--	--		--
Regulação por luz natural(2)	x	--	--	--
Controlo horário	x	x	x	--
Comando por interface	x	--	x	--
Gestão operacional	x	--	x	--

(1) Deve ser complementada com temporização ajustável, para evitar a desativação da iluminação na ausência de movimento nesse período;

(2) Os valores de referência definidos para o plano de trabalho serão fixados mecanicamente ou com recurso a software;

**Tabela 2. 2 – Requisitos de controlo de iluminação para edifícios**

### 2.6.3. REQUISITOS DE DENSIDADE DE POTÊNCIA E FATORES DE CONTROLO

As potências dos sistemas de iluminação a instalar nos edifícios de comércio e serviços não podem exceder os valores indicados pela Tabela 3.3, extraída da portaria nº 349-D já referida e em vigor na qual são também indicados os valores dos fatores de correção a utilizar pela existência de sistemas de controlo.

Tipo de espaço	DPI		Fator de controlo	
	[(w/m <sup>2</sup> ) / 100 lux]		Ocupação FO	Disponibilidade de luz natural FD
	Entrada em vigor	31/dez/15		
Escritórios com mais de 6 pessoas, salas de desenho;	2,5	2,1	0,9	0,9
Escritório individual 1-6 pessoas;	2,8	2,4	0,9	0,9
Show room, salas de exposição e museus;	2,8	2,4	1	1
Salas de aula, salas de leitura, bibliotecas, salas de trabalho de apoio, salas de reuniões/conferências/auditórios	2,8	2,4	0,9	0,8
Laboratórios, salas de exames/tratamentos, blocos operatórios;	2,8	2,4	1	1
Salas de pré e pós-operatório;	4	3,4	0,8	0,8
Cozinhas, armazéns, arquivos, polidesportivos/ginásios, salas técnicas, parques de estacionamento interiores	4	3,4	0,9	1
Plataformas de transportes e similares	4	3,4	1	1
Lojas de comércio e serviços, retalhistas em geral - zona de público, espaços fabris em geral	4	3,4	1	1
Entradas, corredores, escadas, salas de espera, instalações sanitárias, enfermarias e quartos individuais de clínicas e hospitais, salas de refeições	4,5	3,8	0,8	0,9

**Tabela 2.3 – Requisitos de densidade de potência e fatores de controlo**

---

## CAPÍTULO 3

---

### PROTÓTIPO DE CONTROLO DE ILUMINAÇÃO

*Na sequência do estudo apresentado no capítulo anterior e nomeadamente nos requisitos impostos por legislação relativamente aos sistemas de iluminação de edifícios, foi desenvolvido um autómato para controlar sistemas de iluminação em edifícios. Neste capítulo é feita a caracterização da arquitetura proposta do protótipo de controlo de iluminação sendo apresentado o Hardware e o Software.*



### **3.1.ARQUITETURA DO PROTÓTIPO**

Pretende-se desenvolver um autómato para controlar luminárias, permitindo reduzir o consumo de energia e melhorar o conforto dos utilizadores do espaço, com vertente de utilização genérica, para instalações novas ou existentes de dimensão pequena a média. Este sistema permitirá fazer uma gestão centralizada das luminárias através de uma interface local e será compatível com qualquer balastro que permita regulação de fluxo.

Características do sistema:

- Capacidade de controlo, configuração e visualização do estado das luminárias através de uma Interface.
- Integração de deteção de ocupação, programação horária e regulação de luminosidade ambiente.
- Redução o impacto do consumo de energia da iluminação em edifícios e melhorar a conforto dos utilizadores.
- Portabilidade e adaptação a variados tipos de edifícios desde empresariais a domésticos.

O sistema de controlo será concebido com sensores, atuadores e um relógio de tempo real implementados num microcontrolador. O sistema permitirá controlar balastros eletrónicos com regulação de fluxo digital ou analógica e ativar ou desativar qualquer outro tipo de luminária. As luminárias com controlo de fluxo serão controladas através de sinais PWM gerados nas saídas do microcontrolador. As luminárias sem regulação de fluxo serão controladas com o auxílio de reles de potência. A interface gráfica digital vai permitir uma monitorização local da data, hora, temperatura, luminosidade e configuração do sistema. O protótipo permite a ativação ou desativação de cada luminária através de um sinal digital ou então colocá-las em modo automático que através das configurações efetuadas na interface do utilizador.

Os sensores são os responsáveis por fazer a conversão de uma grandeza física para sinais elétricos, que podem ser lidos e interpretados pelo microcontrolador. Estes

sinais são posteriormente enviados, através das ligações existentes, para os pinos de entrada, onde podem ser lidos no microcontrolador que por sua vez através das funções definidas pelo programador, este irá controlar os atuadores do sistema. O software consiste no código desenvolvido com a finalidade de controlar luminárias. A implementação de todos os elementos que constituem o protótipo, bem como os seus princípios de funcionamento, formas de comunicação e metodologias de controlo são descritos nos pontos que se seguem deste capítulo. Na Figura 3.1 é apresentado um diagrama de blocos do protótipo desenvolvido.

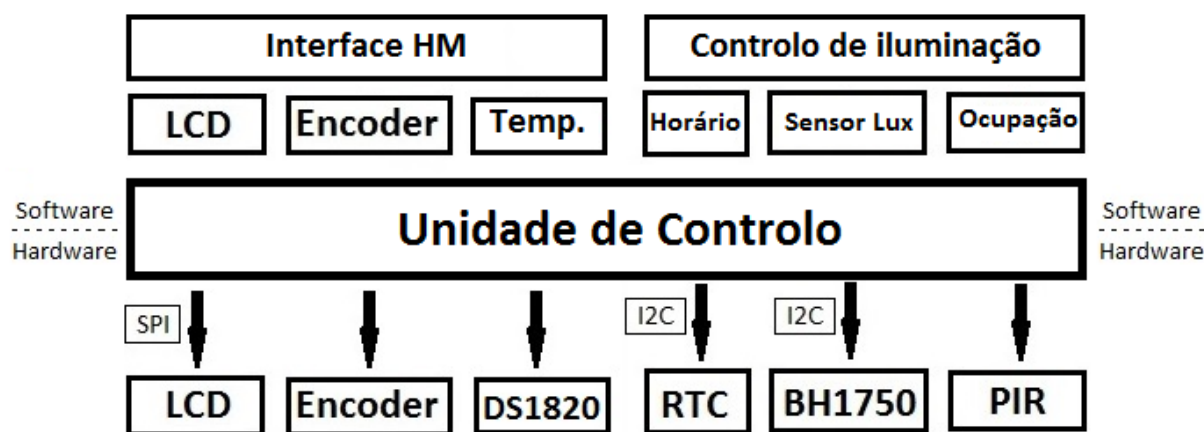


Figura 3. 1 – Diagrama de blocos do protótipo

## 3.2.ESPECIFICAÇÃO DO SISTEMA

### 3.2.1. PROCESSADOR

Para desenvolver o autómato foi selecionado o Arduino, visto que é uma plataforma de Hardware, projetada com um microprocessador Atmel AVR, num circuito de entradas e saídas e com uma linguagem de programação padrão que é essencialmente C/C+. Uma típica placa Arduino é composta por um controlador, linhas de entrada e saída digital e analógica, saídas de digitais PWM e uma entrada serial ou USB, para a conexão com o Arduino IDE, que é usado para desenvolver, descarregar o código e bibliotecas necessárias para o funcionamento do sistema.

O Arduino possibilita a comunicação com outros dispositivos através de interface serie USB, I2C ou SPI. A programação do microcontrolador pode ser feita através do *software* Arduino IDE, cujo código será descarregado via conexão USB. O Arduino pode ser alimentado através da entrada USB, de uma fonte alimentação AC/DC ou de uma bateria.

O Arduino permite fazer com versatilidade a interligação entre interfaces, sensores e atuadores, resultando assim numa excelente opção para desenvolver sistemas de controlo versáteis a preço apelativo e com uma enorme gama de aplicações. Deste modo e após análise das soluções existentes no mercado optou-se por usar o Arduino neste projeto.

Existem diversas versões desta placa, o que as diferencia é o microprocessador usado e o número de entradas e saídas, sendo que para este trabalho foi utilizado, o Arduino Deumilanove que é apresentado na Figura 3.2 [13].

Microprocessador	Atmega 328
Tensão operacional	5V
Tensão de entrada	7-12V
Pinos digitais de entrada/saída	14(6 PWM)
Pinos analógicos de entrada/saída	6 (10bits)
Corrente CC por pino entrada/saída	40mA
Corrente CC por pino 3,3V	50mA
Memória Flash	32KB
SRAM	2KB
EEPROM	1KB
Interface de Hardware	I2C,SPI, UART(5V)
Velocidade de relógio	16MHz
Dimensões	75x53x15 mm

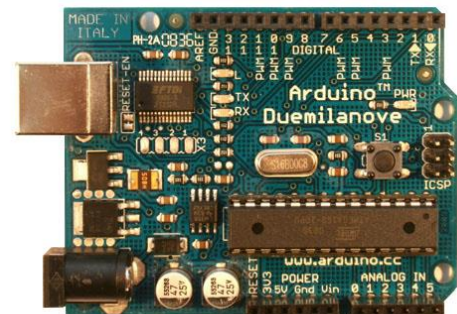


Figura 3. 2 – Características e placa Arduino Deumilanove

### 3.2.2. INTERFACE COM O UTILIZADOR

Esta interface foi desenvolvida através da implementação de um LCD para visualização gráfica e de um *rotary encoder* para navegação e seleção dos menus. De modo a permitir a interação com o utilizador, permite realizar as configurações do sistema, como regular luminosidade ou programar as horas de Ligar/Desligar.

A interface foi desenvolvida considerando o controlo à zona, num sistema com controlo de iluminação de duas zonas distintas de trabalho, uma zona de passagem definida como um corredor e uma zona exterior. Nas figuras seguintes será apresentada a Interface detalhadamente.

2	0	:	4	5															
1	2	/	0	5	/	1	5			T	=	2	0	°	C				

**Figura 3. 3 – Imagem base do Interface**

Em modo de repouso o sistema torna visível no LCD as horas, a data e a temperatura como é apresentado na Figura 3.3. Quando o utilizador prime o botão de menu, o sistema apresenta as zonas controladas. A Figura 3.4 representa este menu em relação ao exemplo descrito.

>	1	.	Z	o	n	a	1												
	2	.	Z	o	n	a	2												
	3	.	S	a	i	r													

**Figura 3. 4 – Primeiro menu do interface**

No primeiro menu podemos aceder aos menus das zonas definidas para controlo e temos uma opção de sair, ou seja voltar ao menu inicial. A Figura 3.5 representa o menu de controlo.

>	1	.	O	n													*	
	2	.	O	f	f													
	3	.	C	e	n	á	r	i	o		1							
	4	.	C	e	n	á	r	i	o		2							
	5	.	C	e	n	á	r	i	o		3							
	6	.	A	u	t	o												
	7	.	D	e	t	e	c	c	a	o								
	8	.	C	r	e	p	u	s	c	u	l	a	r					
	9	.	D	e	f	i	n	i	c	o	e	s						
	10	.	S	a	i	r												

**Figura 3.5 – Menu de Controlo**

No menu de controlo as duas primeiras opções “On” e “Off” funcionam como um simples interruptor ligando e desligando a iluminação da respetiva zona. As três opções seguintes “Cenário 1/ Cenário 2/ Cenário 3” permitem acionar as luminárias dos espaços com um dos três níveis de luminosidade pré-definidos. A opção “Auto” permite colocar o sistema em modo automático em que as luminárias são ativadas e desativadas por controlo horário, com controlo do nível da luminosidade. Este controlo é realizado automaticamente em função do nível de luminosidade pré definido pelo utilizador e do nível de luminosidade natural no espaço. O estado em que o sistema se encontra a funcionar é representado por um asterisco como podemos verificar no modo “On” da Figura 3.5.

As opções “Deteção” e “Crepuscular” colocam a luminária em modo deteção de presença ou controlo crepuscular respetivamente. De seguida temos a opção “Definições” cujo menu vem representado na Figura 3.6.

>	1	.	H	o	r	a	r	i	o										
	2	.	L	u	m	i	n	o	s	i	d	a	d	e					
	3	.	S	a	i	r													

**Figura 3.6 – Menu Definições**

No menu definições a 1ª opção permite configurar a hora de ligar e de desligar as luminárias da respetiva zona. A segunda opção como o nome indica permite regular a luminosidade pretendida para essa zona.

>	1	.	H	o	r	a	L	i	g	a	r			
	2	.	H	o	r	a	D	e	s	l	i	g	a	r
	3	.	S	a	i	r								

**Figura 3. 7 – Menu para configuração hora ligar/hora desligar**

No menu luminosidade representado na Figura 3.8 podemos regular o nível de luminosidade pretendido, desde que a saída desta função esteja ligada a um balastro ou driver com regulação de fluxo. Ao rodarmos o *rotary encoder* aumentamos ou diminuimos o valor do PWM e por sua vez o nível de luminosidade da luminária. No mesmo menu é apresentado do lado direito a luminosidade dessa zona em lux para facilitar a configuração do sistema.

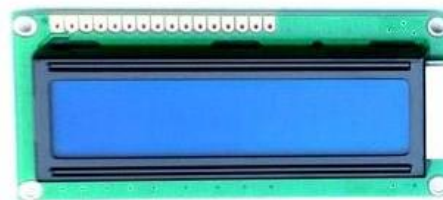
P	W	M	1	0	0	3	2	3	l	x

**Figura 3. 8 – Menu Luminosidade**

O LCD escolhido para implementação deste interface foi o módulo refª 016M002B da marca Vishai. Este exibe duas linhas de texto até 16 caracteres o que permite exibir mensagens de texto para o utilizador como dados de sensores ou o estado dos atuadores. O LCD utiliza uma interface paralelo de comunicação. No entanto para permitir maior flexibilidade na interligação com o Arduino foi utilizada a comunicação SPI através da implementação do circuito da Figura 3.17. Neste protocolo a ligação de entrada (MOSI), de saída (MISO) e de sinal de relógio (SCLK) estão separadas onde as ligações MOSI, MISO e de SCLK são feitas pelos pinos digitais 10, 11 e 13, respetivamente. O código desenvolvido para efetuar o interface encontra-se descrito no anexo B.10.

O LCD usado no protótipo e as suas características vêm apresentados na Figura 3.9.

Tensão operacional	5V ou 3,3V
Controlador	Compatível com HD44780
Contraste da tela	Ajuste com Potenciômetro
Interface de Hardware	I2C ou SPI



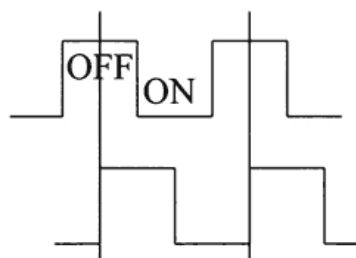
**Figura 3. 9– Características LCD**

Para se efetuar a navegação no interface foi escolhido um *rotary encoder* do tipo botão rotativo que gera sinais eletrônicos nas suas saídas conforme o seu eixo for rodado para a direita ou para a esquerda. Este modelo ainda possui um botão do tipo “Push-Boton” que é acionado ao ser pressionado.

No protótipo desenvolvido o *rotary encoder* foi usado como sistema de três botões independentes, rotação para a direita (mover cursor para baixo), rotação para a esquerda (mover cursor para cima) e push-boton( botão de seleção). Para a integração do *rotary encoder* no protótipo foi usada a biblioteca AdaEncoder[12]. A Figura 3.10 apresenta o encoder usado no protótipo bem como os sinais gerados nas suas saídas.



a)



b)

**Figura 3. 10 – a) rotary encoder ; b) sinais gerados**

O sensor de temperatura escolhido para adicionar uma visualização da temperatura no protótipo foi o DS1820. Este é um sensor digital que permite adquirir medições de temperatura de 9-bit a 12-bit em graus Celsius. O sensor efetua medidas de -10°C a 80°C, com uma resolução de 0,5°C. Uma das vantagens deste tipo de sensor é não necessitar de alimentação externa pois este é alimentado através do cabo de dados.

### **3.2.3. CONTROLO DE ILUMINAÇÃO**

Após o desenvolvimento da interface foi necessário criar as funções de controlo de iluminação, que são a chave para o funcionamento correto do sistema. Estas permitem efetuar um controlo total das luminárias de um espaço.

Nos pontos seguintes são explicadas com mais detalhe as funções implementadas para o controlo de luminárias.

#### **3.2.3.1. MODO ON/OFF MANUAL**

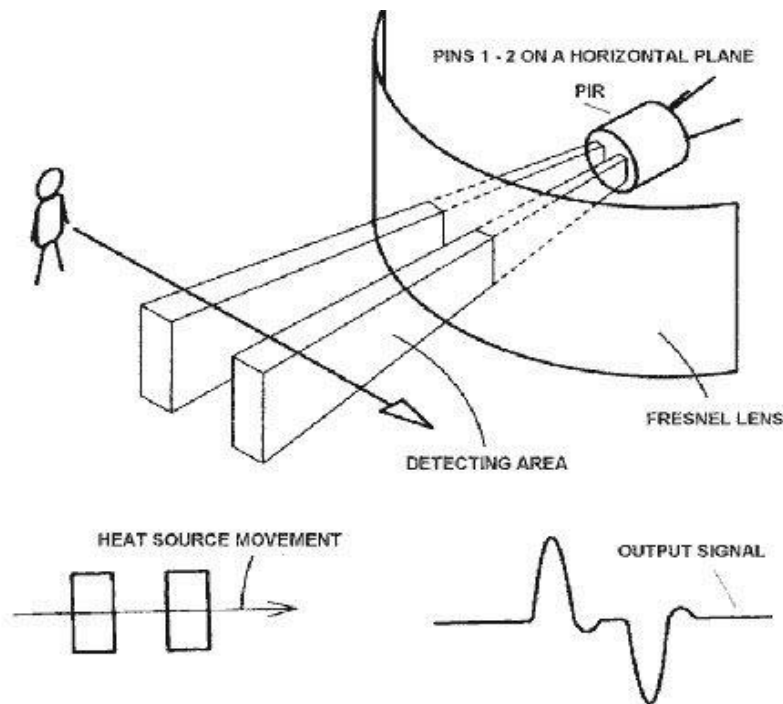
A primeira função de controlo implementada foi o controlo ON/OFF manual que permite ao utilizador ligar ou desligar uma luminária. Este controlo foi desenvolvido de modo a não ser necessário recorrer ao nenhum interruptor auxiliar. Para ativar ou desativar luminárias equipadas com balastros ou fontes de alimentação digitais com regulação de fluxo, é usada a modelação PWM em que um *duty cycle* de 0 %, corresponde ao estado *off* e um *duty cycle* 100% corresponde ao estado *on*. Para os restantes tipos de luminárias é necessário usar um relé de potência auxiliar para ativar ou desativar as mesmas.

#### **3.2.3.2. CONTROLO POR DETEÇÃO DE OCUPAÇÃO**

O controlo por deteção de ocupação consiste em ativar uma luminária apenas quando há ocupação em uma determinada zona e desativar a mesma assim que não houver deteção de presença. Os componentes principais para implementar esta função foram os sensores de deteção de ocupação. Este tipo de controlo é ideal para zonas pouco utilizadas ou zonas de passagem.

Os sensores de deteção de ocupação escolhidos para o protótipo são do tipo PIR (Pyroelectric InfraRed Sensor). Este sensor permite monitorizar os movimentos dentro de um ambiente, sendo usado habitualmente para detetar a presença de humanos ou animais em movimento num determinado espaço, e assim indicar a presença ou intrusão. São dispositivos de baixo consumo e alimentados a baixa tensão. A ideia básica do funcionamento destes sensores é bastante perceptível através da Figura 3.11,

onde observamos que o sensor de PIR tem duas ranhuras idênticas que são feitas de um material sensível ao infravermelho.



**Figura 3. 11 - Exemplo do funcionamento do PIR**

Quando o sensor está inativo, ambas as ranhuras detectam a mesma quantidade de IR, a quantidade irradiada a partir do ambiente, da sala, paredes ou ao ar livre. Quando um corpo quente, como um humano ou animal passa na zona de detecção, ele intercepta sempre primeiro um dos feixes do sensor PIR, o que provoca uma diferença de potencial entre os dois feixes. Quando o corpo quente deixa a zona de detecção ocorre o inverso, são esses pulsos de mudança que são detetados.

O sensor PIR vem equipado com um circuito eletrônico capaz de amplificar os sinais e assim poder modular um sinal de saída em nível digital. A integração do sensor PIR no projeto foi bastante simples visto que o sensor não efetua medições, nem devolve valores numéricos, tendo apenas dois estados, ativo ou desativo (3V ou 0V).

A função de controlo por detecção de ocupação foi implementada com o auxílio de uma variável de estado. Esta tem três estados possíveis, o estado IDLE que corresponde ao estado estacionário, sempre que o sensor deteta uma ocupação, o sistema liga a

luminária e a variável passa para o estado ACTIVE, em que a luminária está ativa e automaticamente a variável passa para o terceiro estado, o TIMEOUT em que se não houver deteção durante cinco segundos a luz é desligada e a função volta ao estado IDLE. A Figura 3.12 apresenta o sensor PIR e as suas principais características e o código implementado para o funcionamento correto desta funcionalidade encontra-se disponível no anexo B.7.

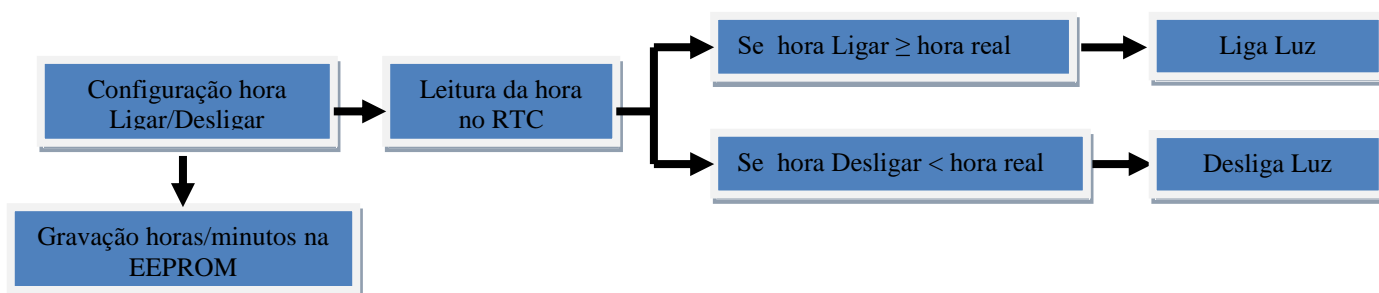
Saída	Sinal digital (3V com deteção)
Alcance	6m
Alimentação	5V- 9V



**Figura 3. 12 - Sensor de infravermelhos PIR**

### **3.2.3.3. CONTROLO HORÁRIO**

O controlo horário consiste em ligar e desligar uma luminária a determinadas horas pré-definidas nas “definições” da interface LCD e posteriormente gravadas na memória do sistema. Para a contagem da hora e data foi usado um RTC, relógio de tempo real. A biblioteca Time [14] usada para este controlo vem por definição com funções que devolvem as horas e a data. Em cada ciclo o programa, insere o valor real da hora e dos minutos em dois vetores [hora], [minuto]. Os valores da hora de ligar e desligar pré-definidos no LCD também são inseridos em quatro vetores [hora-ligar], [minuto-ligar], [hora-desligar], [minuto-desligar]. Os valores hora e minuto reais, hora ligar e hora de desligar são convertidos num só valor inteiro multiplicando a hora por 60 e somando os minutos. Deste modo por comparação de um único valor inteiro foi possível efetuar o controlo horário. A Figura 3.13 apresenta um fluxograma que elucida sobre como a função de controlo horário foi implementada. A função de controlo horário vem apresentada no código apresentado no anexo B.9.



**Figura 3. 13 – Fluxograma da função de controlo horário**

Para a contagem e visualização da hora e data o relógio de tempo real escolhido o modelo DS1307 que consiste no chip DS1307 com respetivas ligações. O relógio de tempo real vem equipado com uma bateria que lhe permite uma contagem contínua do tempo, por um período de pelo menos nove anos, mesmo que a alimentação do sistema seja desligada.

Este possibilita a transferência de dados através do protocolo I2C e disponibiliza contagem de segundos, minutos, horas, dias, dias da semana, mês e ano contabilizando os anos bissextos. A sua flutuação de tempo pode ir até 1 minuto por mês devido a desvios provocados por alterações de temperatura ou influência de um campo magnético. A integração do relógio de tempo real no presente projeto foi feita através da biblioteca DS1307RTC [14] e a Tabela 3.1 apresenta as principais características do relógio de tempo real.

Tensão operacional	5V
Ligações	Vcc, GND, pino A4(SDA), pino A5 (SCL)
Interface de Hardware	I2C
Calendário	até 2100
Memória NV SRAM	56 Bytes
Relógio/Calendário	Código binário decimal (BCD)
Consumo	< 500nA
Dimensões	20x20mm

**Tabela 3. 1 – Características do RTC**

### 3.2.3.4. REGULAÇÃO DE LUMINOSIDADE

Esta é a principal funcionalidade do autômato, que permite através das saídas PWM do microcontrolador regular o nível de luminosidade de lâmpadas e LEDs. O principal componente para implementar esta função foram os sensores de luminosidade digitais. No menu definições configura-se a luminosidade pretendida para cada zona ou grupo de luminárias, e posteriormente ao colocarmos o sistema em modo automático este regula a luminosidade para o valor definido em função da luminosidade medida pelo sensor. O sistema ajusta o fluxo luminoso das luminárias em função da luminosidade natural disponível, estando as luminárias apenas a debitar o fluxo necessário ou até desligadas no caso de a iluminação natural suprir os requisitos luminosos do espaço.

Os sensores de intensidade luminosa escolhidos para o sistema foram do tipo BH1750 [21] que são constituídos por um fotodíodo de banda larga para medir a luminosidade dentro do espectro visível, um conversor analógico-digital (ADC) que transforma a corrente do diodo num sinal digitais de 16 bits, podendo este ser lido por um microprocessador através de comunicação série ou I2C. Este sensor realiza medições de valores de iluminância de 0 a 65535 lux e com uma resposta espectral muito semelhante ao olho humano como podemos verificar pela Figura 3.14. As restantes características deste sensor vêm detalhadas no *datasheet* do equipamento [21].

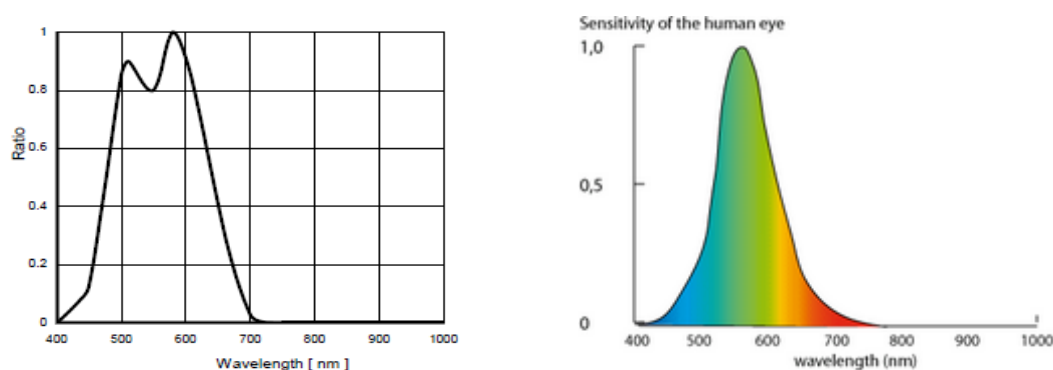
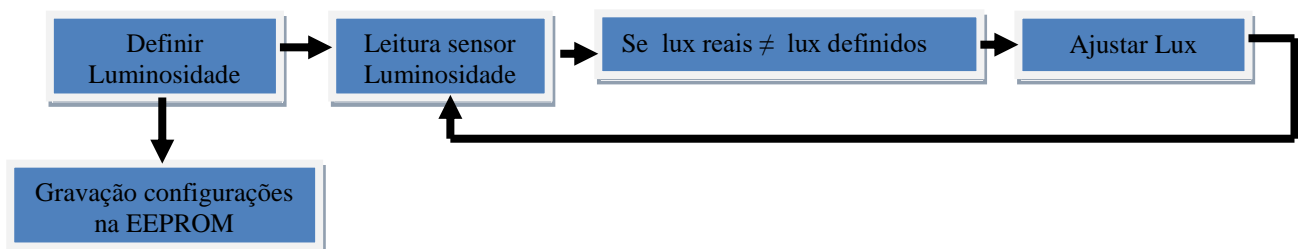


Figura 3. 14 – Resposta espectral do sensor BH1750 e resposta espectral do olho humano

Esta função complementada com a função de controlo horário, permite colocar o sistema em modo automático em que as luminárias são totalmente controladas pelo autómato. No modo “Auto” o autómato liga e desliga as luminárias e regula o fluxo luminoso das mesmas ao longo do dia. Como as saídas PWM do Arduino apenas variam entre 0V e 5V, foi usado o chip ULN2803 para converter o sinal PWM para o intervalo de 0V a 10V pois estes são os valores de tensão que a maioria dos equipamentos existentes no mercado utilizam para regulação de fluxo. O código desenvolvido para efetuar o controlo de luminosidade encontra-se descrito no anexo B.4. A Figura 3.15 apresenta o fluxograma de controlo automático da luminosidade.

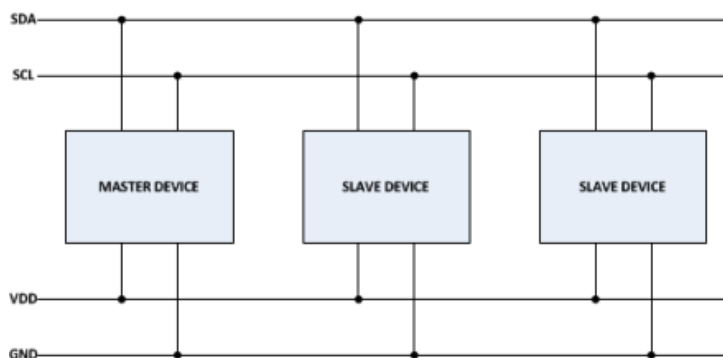


**Figura 3.15 – Fluxograma da função de controlo de luminosidade**

### **3.2.3.5. I2C**

O protocolo de comunicação I2C (*Inter-Integrated Circuit*) foi desenvolvido para facilitar a troca de informação entre sensores, atuadores e microcontroladores, como é o exemplo do Atmega328. O seu princípio de funcionamento baseia-se num barramento com duas ligações: o SDA (*Serial Data Line*) que é usado para a transferência de dados, e o SCL (*Serial Clock Line*) que funciona como relógio, fazendo o compasso de espera entre cada transferência de informação, assegurando que o barramento só processa dados de um equipamento de cada vez. A este barramento podem ser conectados até 127 equipamentos, denominados de *slaves*, que partilham os fios SCL,SDA, estes por sua vez estão sempre conectados a um dispositivo que coordena a comunicação, denominado de *master* que no presente projeto é o Arduino Deumilanove.

O Arduino Deumilnove utiliza o pino analógico 5 para o SCL, que fornece um sinal de relógio, e o pino analógico 4 para o SDA. Na Figura 3.16 vem apresentado, através de um esquema, a exemplificação de uma ligação de barramento I2C.



**Figura 3. 16 – Exemplo de linha de comunicação I2C**

Cada sensor tem que ter um endereço diferente, que o identifica durante a comunicação. Dependendo do dispositivo, em relação ao endereço estes podem apresentar as seguintes características:

- ter um endereço fixo;
- permitir que se configure o endereço através da ligação, ou não, de um fio específico;
- necessitar de acionamento de comandos de iniciação;

Para se integrar a comunicação de equipamentos no protótipo através do protocolo I2C foi necessário analisar o *datasheet* dos equipamentos, onde podemos encontrar, os endereços I2C ou opções de escolha do mesmo, as tensões de operação e ligações elétricas dos equipamentos.

No presente trabalho, a ligação dos sensores BH1750 e do RTC é feita através da utilização deste protocolo, sendo que o Arduino desempenha a função de *master*.

### **3.2.3.6. ARMAZENAMENTO DE CONFIGURAÇÕES**

O armazenamento de configurações foi feito através do uso da EEPROM interna do Arduino.

Uma EEPROM (sigla do inglês "*erasable programmable read-only memory*") é um chip de memória que permite registrar dados e mantê-los na ausência de energia. Esta é programada através de um dispositivo eletrônico que gera tensões maiores do que as usadas normalmente em circuitos elétricos. O microcontrolador disponível na placa de desenvolvimento Arduino Duemilanove (ATMega328) possui 1024 bytes de memória EEPROM. Neste trabalho a EEPROM é usada para gravar o código desenvolvido e para gravar as diferentes configurações de controlo de iluminação selecionadas através do interface do utilizador.

## **3.3. IMPLEMENTAÇÃO**

De seguida será abordado todo o desenvolvimento do sistema, incluindo a parte do hardware e do software. Em relação ao hardware cujos componentes foram apresentados nos pontos anteriores será apresentado um esquema representativo das ligações dos mesmos com o Arduino. No que se refere ao software será feita uma explicação do mesmo, sob a forma de fluxograma de código, de modo a elucidar sobre o código desenvolvido para o funcionamento do sistema, que vem apresentado no anexo 1.

### **3.3.1. HARDWARE**

No esquema apresentado na Figura 3.17 pode observar-se, de uma forma global, como foram feitas estas ligações entre os componentes escolhidos e o Arduino Duemilanove.

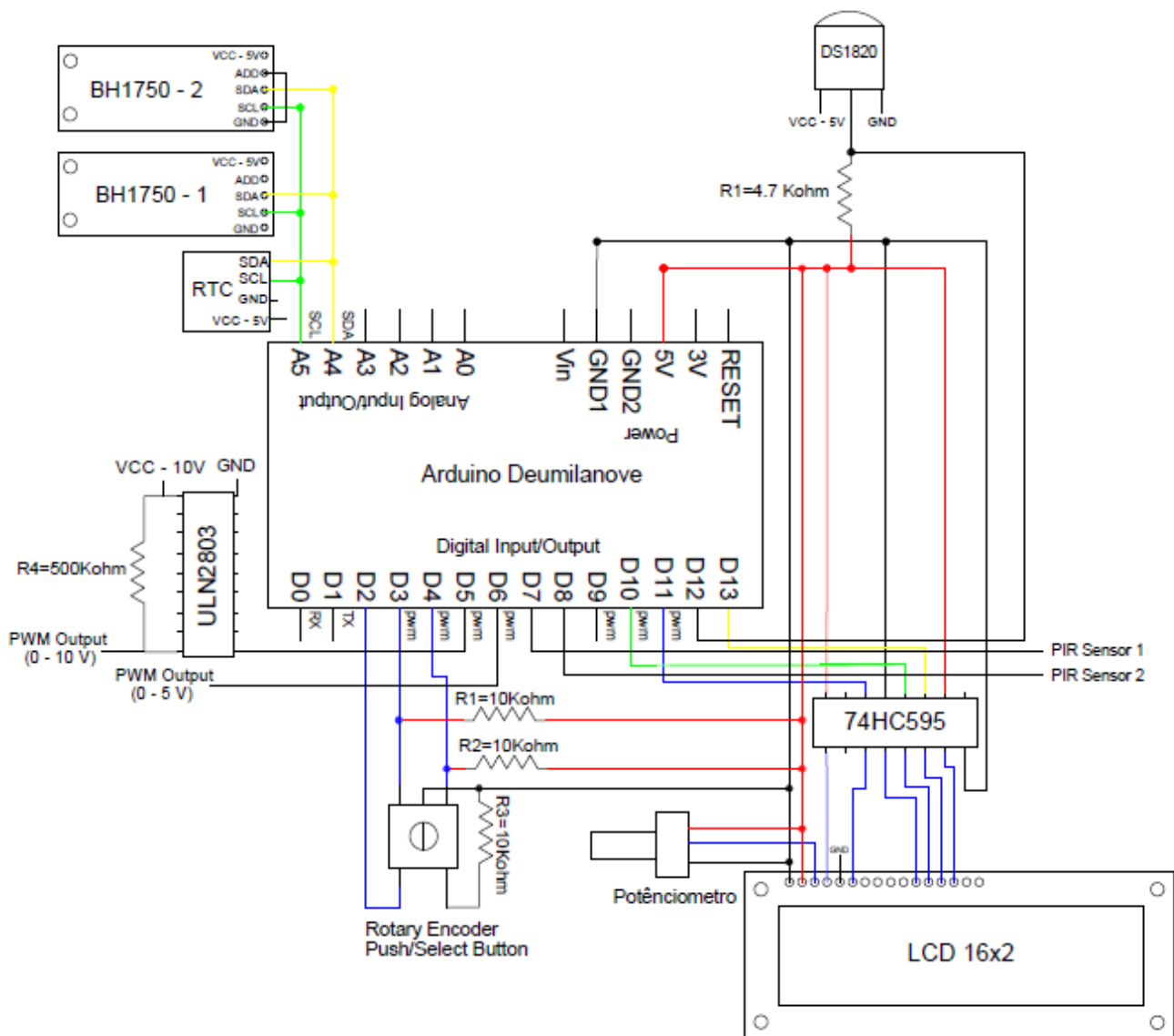


Figura 3. 17 – Esquema de ligações do protótipo

### 3.3.2. CÓDIGO ARDUÍNO

O programa elaborado (ver Anexo C), cujo fluxograma de código se apresenta na Figura. 3.18, é constituído pelas duas funções principais, o *setup* e o *loop*, que caracterizam um *sketch* do Arduino IDE. Todas as funcionalidades do programa foram desenvolvidas em código genérico e este executa o autómato em função de variáveis, que para além de reduzir substancialmente o número de linhas do código, possibilita a

expansão do mesmo em função do número de zonas que o utilizador pretender controlar estando apenas limitado pelas características do microprocessador. Deste modo cada função desenvolvida pode controlar diferentes luminárias com diferentes configurações.

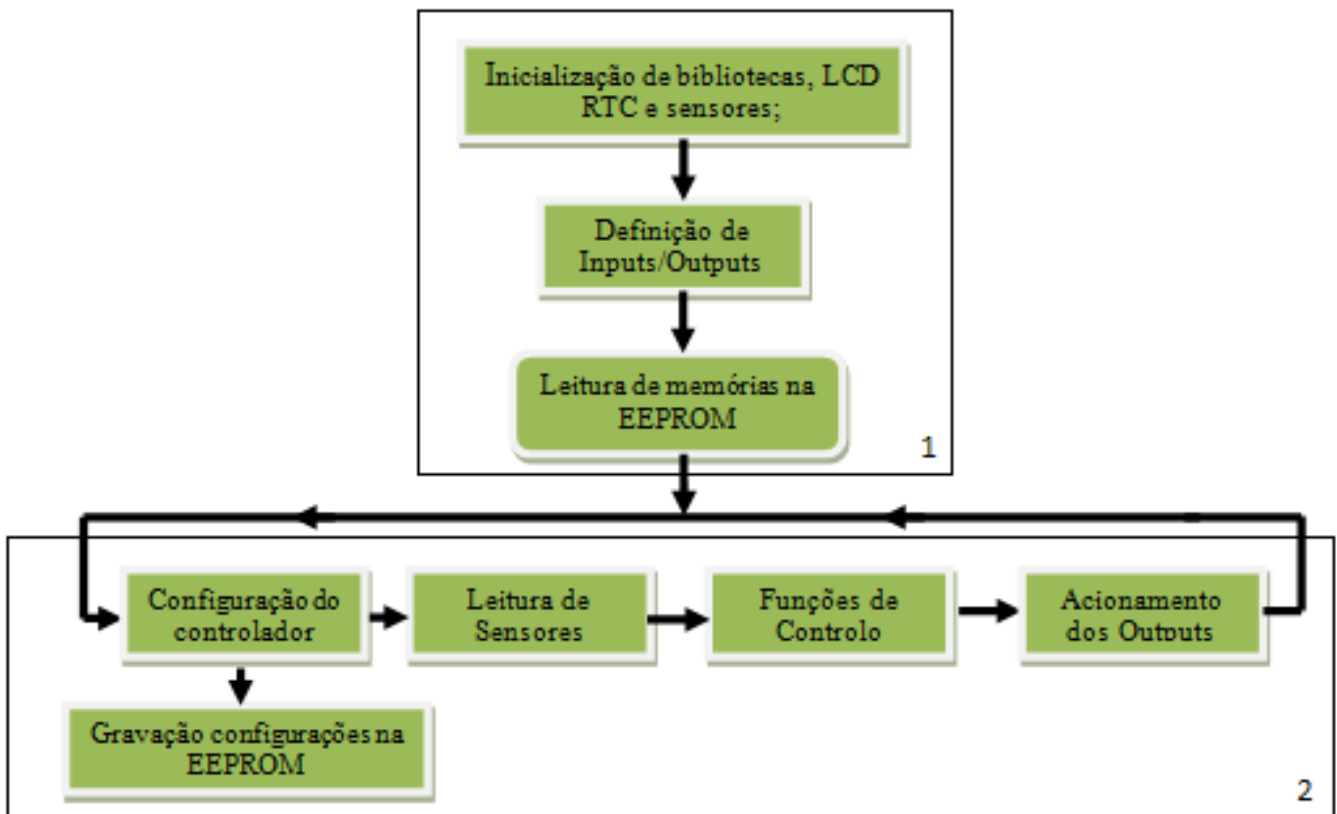


Figura 3. 18 - Fluxograma do código desenvolvido para controlo de iluminação- 1- void Setup( );  
2)- void Loop;



---

## CAPÍTULO

# 4

---

### **TESTES E RESULTADOS EXPERIMENTAIS**

*Neste capítulo são apresentados todos os ensaios feitos no presente trabalho, são testadas as metodologias de controlo implementadas no sistema desenvolvido para melhoria da eficiência energética em sistemas de iluminação em edifícios e para finalizar é feita uma análise económica do custo do sistema desenvolvido*



## **4.1. INTRODUÇÃO**

Neste capítulo são apresentadas as metodologias de testes e os resultados experimentais obtidos, avaliando o desempenho do protótipo desenvolvido, demonstrando a otimização da eficiência energética em sistemas de iluminação em edifícios. Para se concluir sobre os ensaios efetuados, foi necessário caracterizar os componentes do sistema. Os resultados numéricos obtidos dizem respeito aos consumos de energia para dois tipos de lâmpadas, nomeadamente as lâmpadas fluorescentes compactas e leds, testados com e sem controlo automatizado para se concluir sobre o desempenho do autómato quando aplicado o controlo de iluminação.

Os ensaios efetuados servem apenas como confirmação do correto funcionamento do autómato e foram efetuados numa escala reduzida contemplando apenas uma luminária em cada tipo de controlo implementado, devido a questões de custos de instalação e de equipamentos auxiliares para o mesmo efeito. De salientar que numa aplicação real, cada tipo de controlo seria aplicado a diversas luminárias.

Cada ensaio foi efetuado independentemente dos restantes apesar de todos se basearem no consumo de energia de diversas luminárias a diversos níveis de luminosidade. Para o registo e leitura de valores dos sensores foi usado o serial monitor do Arduino IDE e para a medição da corrente consumida pelos equipamentos foi usado um multímetro digital. Nos pontos seguintes deste capítulo os testes e resultados são explicados mais detalhadamente.

## **4.2. CARACTERIZAÇÃO DA LUMINÁRIA FLUORESCENTE**

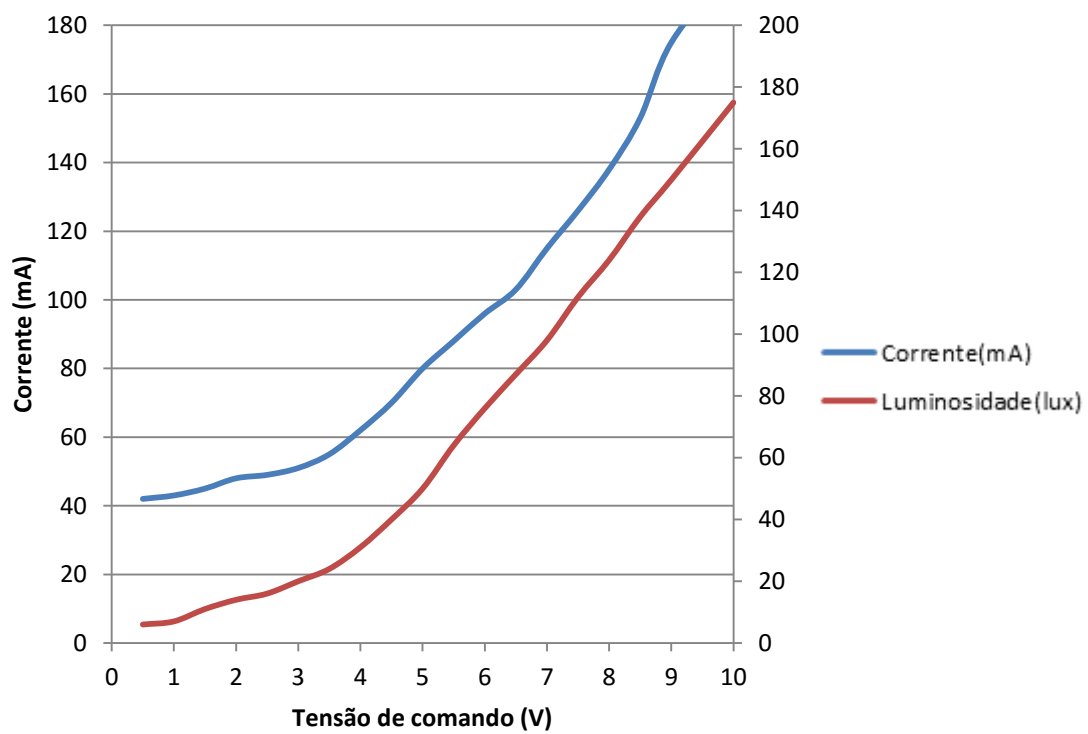
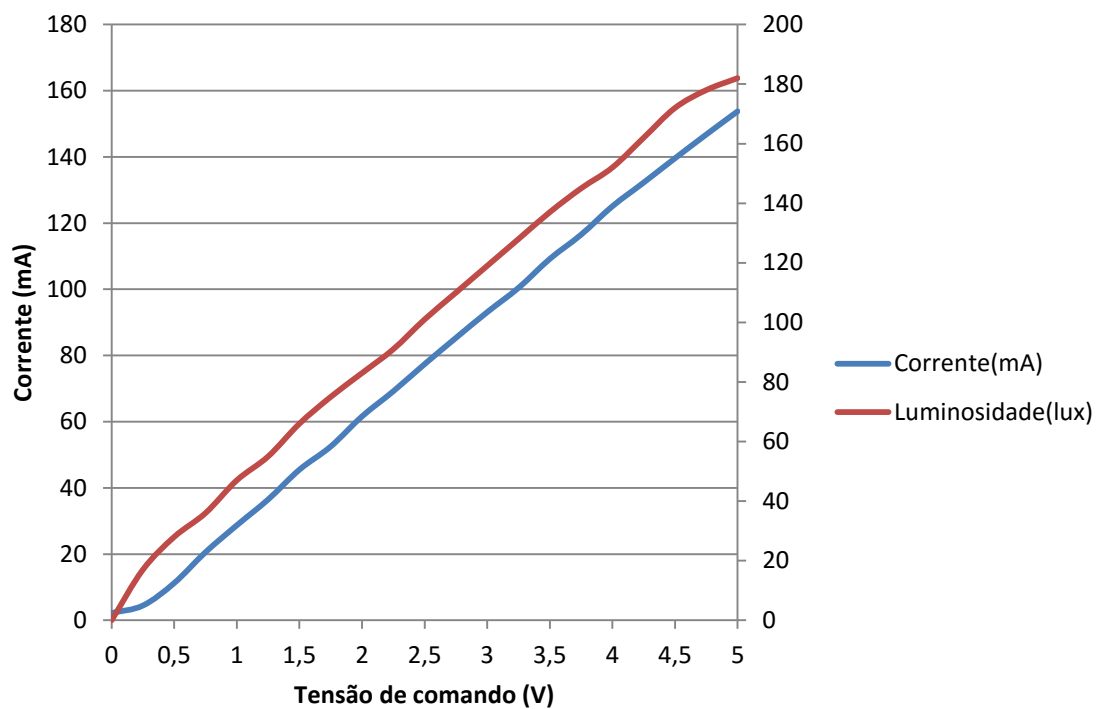
Para testar o autómato foi usada uma luminária equipada com duas lâmpadas fluorescentes do tipo PL 18W FCNI [19] e um balastro eletrónico da marca VOSSLOH SCHWABE, modelo 188219 [20], que vem equipado com entrada de tensão de comando de 1V a 10V analógica para controlo de fluxo luminoso, que correspondem ao fluxo mínimo e máximo da lâmpada respetivamente. Para realizar a

caracterização da luminária usada foram realizados dois ensaios. O primeiro ensaio consistiu em medir a corrente consumida pelo balastro para os diferentes níveis de regulação de fluxo. Foram impostos valores de tensão entre 1V e 10 V em intervalos de 0,5 V na entrada de comando do balastro e com o auxílio de um amperímetro foram registados os diferentes valores de corrente consumida pela luminária para os diferentes níveis de regulação.

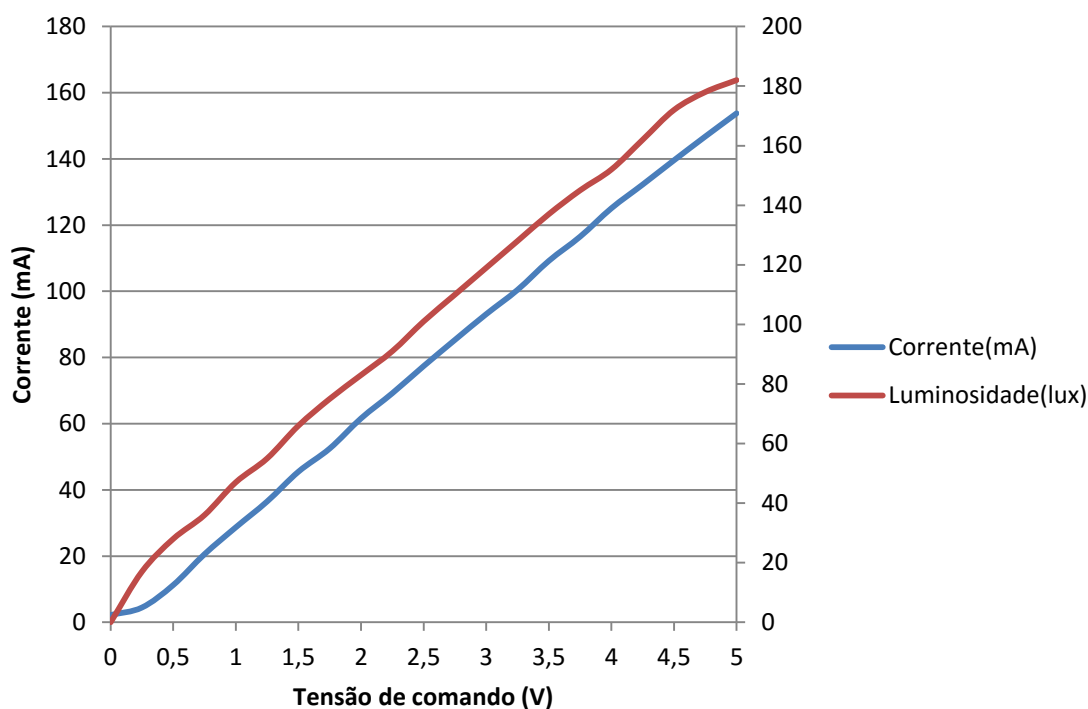
O segundo ensaio consistiu em medir a luminosidade emitida pela luminária nos mesmos intervalos de tensão de comando impostos no primeiro ensaio. Para realizar este ensaio foi instalado o sensor de luminosidade em cima de uma área de trabalho do mesmo espaço, a 198cm da luminária na vertical e variando os valores de tensão de comando registaram-se os diferentes níveis de luminosidade lidos pelo sensor. Os resultados numéricos obtidos vêm apresentados na Figura 4.1.

### **4.3. CARACTERIZAÇÃO DA LUMINÁRIA LED**

Para além da luminária fluorescente o autómato também foi testado com uma luminária led de 10 W equipada com uma driver digital com uma entrada de tensão de comando digital de 0 V a 5 V. A grande diferença entre a entrada de comando digital e analógica é que na entrada digital, o valor 0 V corresponde ao estado desligado e os valores 1V e 5V correspondem ao fluxo mínimo e máximo da luminária. Para realizar a caracterização desta luminária foram efetuados ensaios idênticos aos realizados na luminária fluorescente, porem apenas se variaram os valores de tensão de comando de 1 V a 5 V. Os resultados numéricos obtidos vêm apresentados na Figura 4.2.



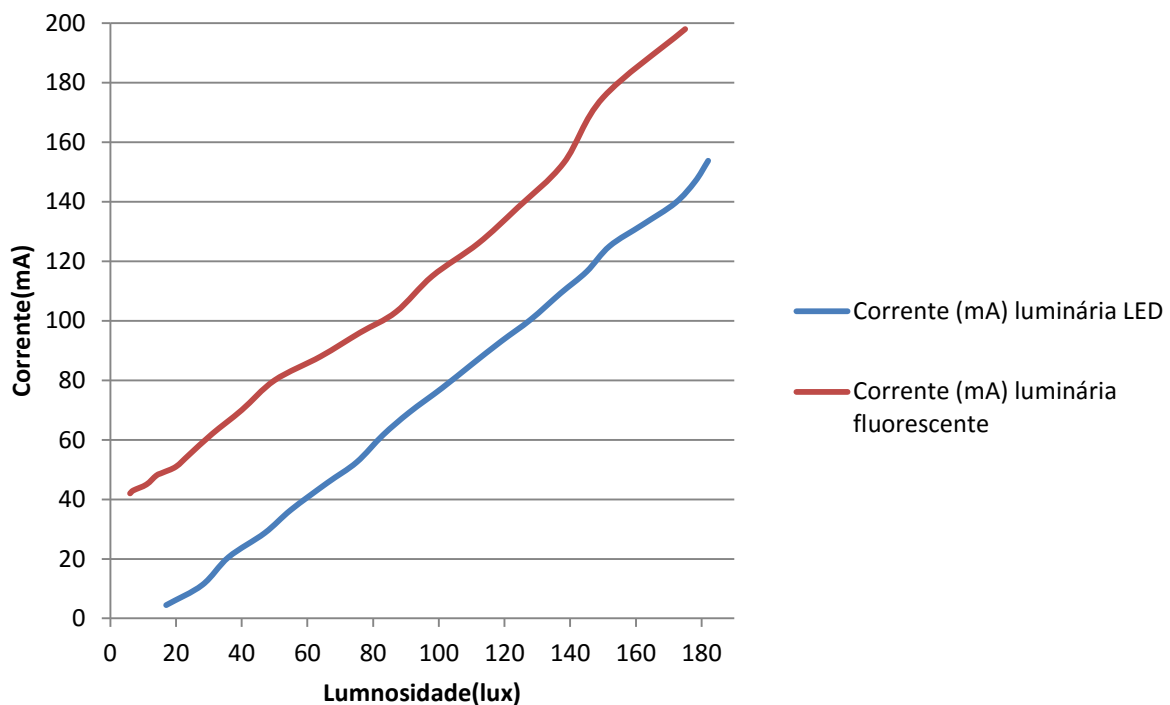
**Figura 4. 1 - Caracterização da luminária Fluorescente**



**Figura 4. 2- Caracterização da luminária LED**

Como se verifica na Figura 4.1, o consumo da luminária fluorescente é proporcional à luminosidade emitida pela mesma e que estes valores apresentam maiores variações no intervalo de 5V a 10 V, logo uma variação dos níveis de luminosidade não linear. Com este tipo de controlo analógico não é possível desligar a luminária através do sinal de comando, ficando esta sempre a debitar um fluxo mínimo como verificamos na Figura 4.1, necessitando de um relé auxiliar para desligar a mesma. Também se verifica a importância do controlo de iluminação visto que o consumo da luminária à potência máxima é quatro vezes superior ao consumo à potência mínima. Na figura 4.2 verificamos que a variação da energia consumida e a luminosidade emitida pela luminária LED são lineares ao longo do intervalo de regulação da fonte de alimentação. Isto acontece devido ao controlo de fluxo desta luminária ser digital ao invés da luminária fluorescente ser analógico. As principais vantagens do controlo de fluxo digital é ser possível desligar a luminária através do sinal de comando, ter uma maior resolução e linearidade ao longo do intervalo de regulação de fluxo, o que

facilita muito a sua implementação em sistemas de controlo automatizado de iluminação.



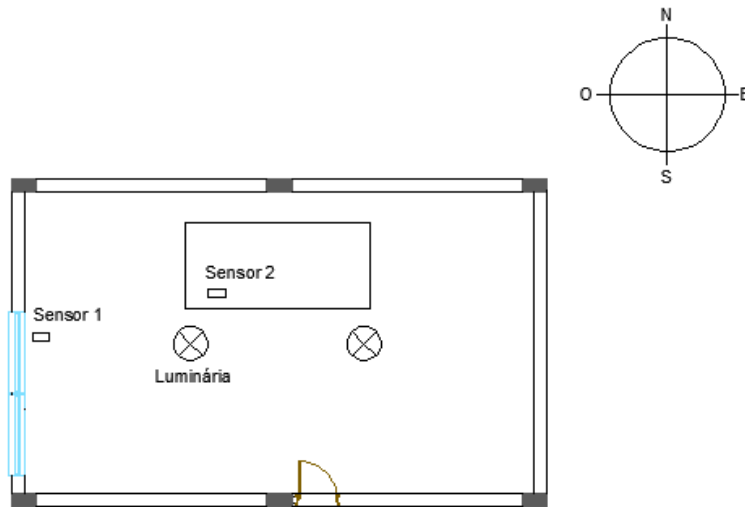
**Figura 4.3 – Consumo/ Luminosidade - luminária LED VS luminária fluorescente**

A Figura 4.3 apresenta uma comparação entre o consumo da luminária LED e da luminária fluorescente, em que verificamos que a luminária LED apresenta consumos de energia 35% inferiores para o mesmo nível de luminosidade o que posiciona os LED como a melhor solução para sistemas de iluminação de edifícios em termos de eficiência energética. Com estes ensaios verificamos que o controlador é compatível com os equipamentos de regulação de fluxo existentes no mercado tanto analógicos como digitais e concluímos sobre a eficiência energética das fontes de luz artificiais mais usadas.

#### **4.4. CARACTERIZAÇÃO DO ESPAÇO**

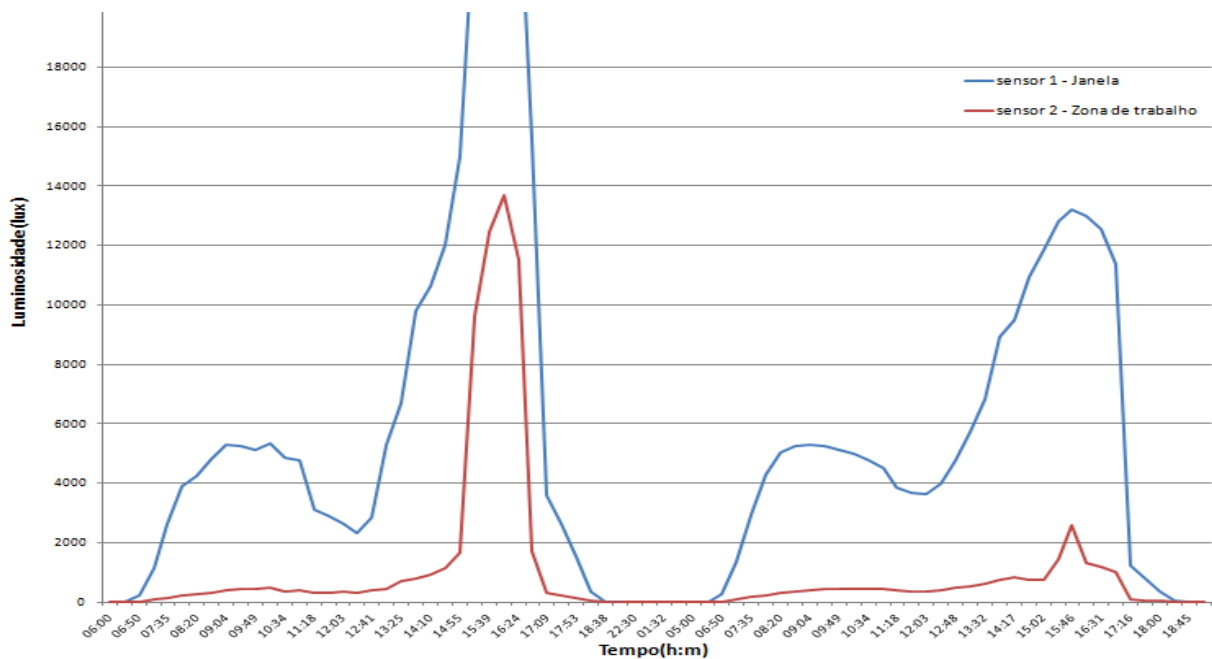
O protótipo foi testado num espaço interior com 22.7m<sup>2</sup>, situado num 9º andar com duas janelas viradas a oeste. A planta do escritório vem apresentada na Figura 4.4 onde

podemos verificar o posicionamento das luminárias e das janelas em relação á área de trabalho.



**Figura 4. 4 – Planta do espaço em estudo**

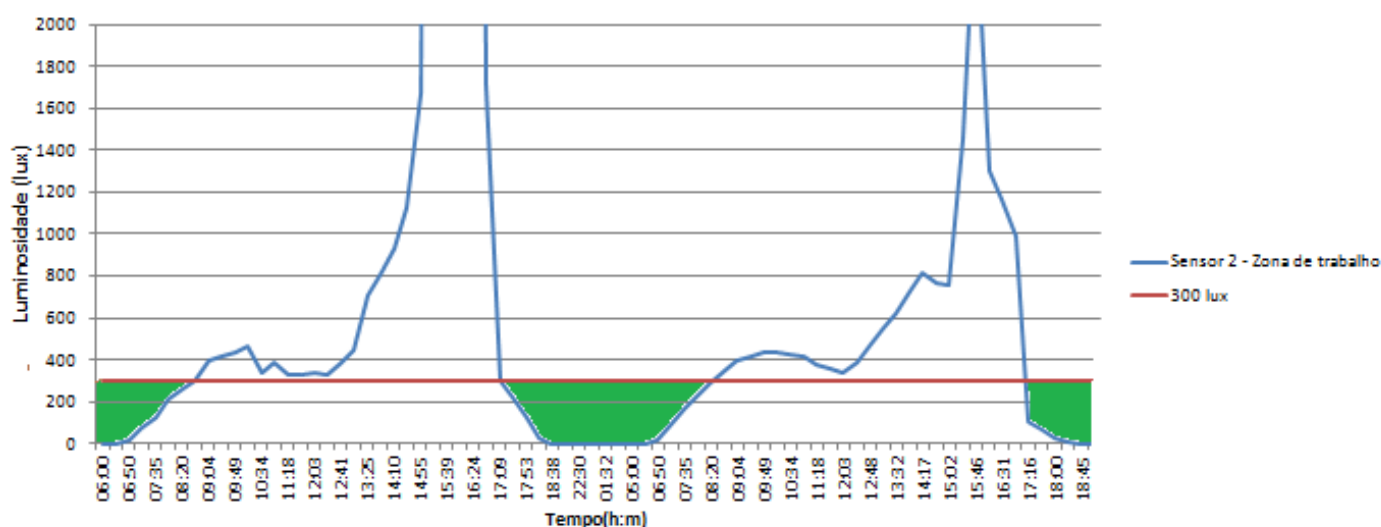
Para estudar a influência da luz natural nesse espaço foram instalados dois sensores de luminosidade, um instalado junto às janelas, e outro instalado sobre a área de trabalho desse espaço. Com o auxílio do serial monitor do Arduino IDE foram registados os valores de luminosidade que incidiu nos dois sensores durante 48 horas. Os resultados experimentais obtidos são apresentados na Figura 4.5.



**Figura 4.5 - Influência da luz natural na área de trabalho do espaço em estudo**

Como se verifica na Figura 4.5, a luminosidade que incide no sensor 1, colocado na janela, é 80% superior à luminosidade que incide no sensor 2 instalado na área de trabalho. Também se verifica a grande variação da luminosidade natural ao longo do dia, que está dependente da direção das janelas, da hora, período do ano e condições climáticas. Deste modo concluímos sobre a importância do estudo do impacto da luz natural na iluminação de espaços interiores, no que toca ao dimensionamento de sistemas de gestão da iluminação.

Porém, mesmo assim, a luminosidade que incidiu no sensor 2 é suficiente para suprir as necessidades de luminosidade do espaço (300 lux [9]) durante grande parte do dia, como podemos analisar na Figura 4.6.



**Figura 4.6 – Análise da influência da luz natural na área de trabalho**

Na Figura 4.6 podemos analisar em pormenor a influência da luz natural na área de trabalho sendo que as luminárias apenas necessitam de estar a funcionar nas horas em que os níveis de luminosidade são inferiores à linha que representa os 300 lux. Como podemos verificar no figura, no espaço em estudo, para garantir um nível de luminosidade mínimo de 300 lux as luminárias desse mesmo espaço apenas

necessitam de estar ligadas até por volta das 9h00m da parte da manhã e a partir das 17h00m da parte da tarde, intervalos esses que vêm representados a verde na Figura 4.5. Isto verifica-se para os dias em estudo, porém num outro período os resultados poderiam ser diferentes visto que a luminosidade natural varia ao longo do ano e está dependente das condições climáticas. Deste modo é possível verificar o potencial da iluminação natural e o quanto esta pode contribuir para uma boa eficiência de um sistema de iluminação visto que a grande maioria dos sistemas não tem em consideração a luminosidade natural.

#### 4.5. CUSTO DO AUTOMATO

Como mencionado anteriormente no capítulo III, um dos objetivos do presente projeto é desenvolver um controlador a baixo custo, ao contrário dos sistemas de controlo automatizado já existentes no mercado, que apresentam um elevado custo e que por sua vez, a sua utilização em edifícios de pequena escala ou domésticos se torna inviável. A Tabela 4.1 apresenta em detalhe o custo de cada componente usado para a construção do controlador para uma, cem ou mil unidades.

Componente	Qtd(un)	Preço (€)	Qtd(un)	Preço (€)	Qtd(un)	Preço (€)
Arduino Deumilanove	1	9	100	8,77	1000	7,45
LCD 16 x 2 Hitachi	1	3,2	100	2,95	1000	2,43
Shift register	1	0,62	100	0,16	1000	0,11
Rotary Encoder	1	0,88	100	0,8	1000	0,72
Tiny RTC	1	3,96	100	2,94	1000	2,5
PIR sensor	1	2,4	100	1,85	1000	1,76
DS 1820 sensor	1	1,34	100	1,2	1000	0,97
BH1750FVI sensor	1	2,64	100	1,75	1000	1,21
ULN2802	1	0,77	100	0,52	1000	0,36
Outros(*)	1	16,41	100	14,5	1000	10
Total		41,22	100	35,44	1000	27,51 €

(\*) - Resistências, condensadores, cablagens e construção circuito impresso

**Tabela 4. 1 – Custo do sistema para 1, 100 e 1000 unidades [16] [17]**

Como podemos verificar na tabela 4.1 o custo do equipamento para o desenvolvimento do protótipo foi de 41,22 €, o que neste tipo de sistemas se pode considerar bastante económico. De importância mencionar que não foram contabilizados os custos do desenvolvimento do *software*, que no caso de este ser desenvolvido ao nível empresarial iria aumentar o custo do sistema, visto que este é o coração do autómato, que garante o correto funcionamento dos sensores, atuadores e interface. Se o sistema fosse comercializado, teria que passar para uma linha de montagem, onde em vez de se encomendar um componente de cada vez, a compra dos equipamentos seria em grande escala o que levaria ainda a uma redução maior do preço final do autómato como podemos verificar na tabela 4.1 em que o estimado do sistema para mil unidades é 33% inferior ao custo de produção de uma unidade.



---

## **CAPÍTULO**

# **5**

---

### **CONCLUSÃO**

*Neste capítulo enuncia-se uma síntese do trabalho e apresentam-se as principais conclusões que se extraíram do estudo desenvolvido para a resolução do problema de otimização da eficiência energética em sistemas de iluminação de edifícios. Apontam-se ainda algumas direções em que pode ser desenvolvido o protótipo de controlo de iluminação apresentado neste trabalho.*



## 5.1. CONCLUSÕES

Nos últimos anos os sistemas de iluminação têm vindo a melhorar a sua eficiência, com a introdução das lâmpadas fluorescentes compactas e, mais recentemente, com os LED. Com a introdução destas tecnologias conseguem-se alcançar reduções até cerca de 80% no consumo de eletricidade na iluminação. Se as lâmpadas fluorescentes compactas facilmente substituíram as lâmpadas incandescentes, os LED conseguem ser uma alternativa a praticamente todo o tipo de lâmpadas, desde incandescentes, fluorescentes a lâmpadas de alta pressão. O controlo de iluminação tem um grande impacto no que diz respeito à poupança de energia em sistemas de iluminação, a conjugação dos diversos métodos de controlo com luminárias, balastros eletrónicos ou fontes de alimentação eficientes permite obter uma grande redução nos consumos de energia dos sistemas de iluminação.

Através do recurso à tecnologia eletrónica digital os sistemas de controlo automatizado de iluminação, possibilitam que os ambientes possam ser controlados de forma mais eficiente, económica e com inúmeros recursos, trazendo flexibilidade e conforto às pessoas que utilizam esses ambientes. Estes modernos sistemas em conjugação com sensores de luminosidade ambiente regulam a intensidade luminosa das luminárias em função de configurações feitas pelo utilizador, permitindo deste modo um melhor aproveitamento da luz natural existente e evitando desperdícios de energia.

Tendo em conta os objetivos e características propostas para o projeto e construção do protótipo, pode dizer-se que o sistema de controlo de iluminação de edifícios foi desenvolvido com sucesso, pois todos os pressupostos iniciais foram alcançados.

Com o autómato desenvolvido é possível controlar luminárias LED e luminárias com lâmpadas fluorescentes, ambas com ou sem regulação de fluxo, acionar ou desativar qualquer tipo de luminária em função de um horário definido no interface, em função da ocupação da sala ou em função da luminosidade exterior.

O sistema é aplicável a diferentes tipos de espaços, desde comerciais, domésticos a empresariais, sendo que permite colmatar praticamente todos os requisitos de controlo, regulação e monitorização impostos pela norma EN12464-1 e EN 15193.

Tem uma interface local que permite a configuração e gravação dessas mesmas configurações na memória EEPROM, bem como a visualização do estado das luminárias no interface LCD.

Ao nível da eficiência energética as poupanças que o sistema pode trazer estão diretamente relacionadas com o tipo de controlo usado.

Os casos mais evidentes são o controlo por ocupação que permite reduzir até 40% dos consumos [5] e o controlo por regulação de fluxo que permite obter até 80% dos consumos [5]. O controlo horário também permite uma poupança de energia pois garante que as luminárias não estão ativas mais tempo do que o necessário.

Os testes efetuados foram realizados de modo a garantir o correto funcionamento do autómato no que diz respeito à robustez dos métodos de controlo, funcionamento correto do interface, sensores e atuadores.

## **5.2. TRABALHO FUTURO**

Tendo como base o estudo realizado, é de apontar perspetivas futuras de desenvolvimento do presente projeto.

As possibilidades de um sistema deste tipo são praticamente infinitas estando sempre dependentes das funcionalidades pretendidas pelo utilizador.

Considerando que o presente trabalho foi desenvolvido para a área de energia, incidiu-se principalmente na parte de controlo de potência para sistemas de iluminação. Com o código genérico desenvolvido em função de variáveis, os métodos de controlo implementados são facilmente escalonáveis para controlo de estores, temperatura, etc.

A parte do projeto onde se pode incidir no futuro é a parte de comunicação. Já existem no mercado placas de comunicação *Ethernet* para o Arduino, para se desenvolver *software* de monitorização e interação com um sistema deste tipo através de uma plataforma online.

Outra funcionalidade possível de integrar no Arduino é a comunicação entre sensores e atuadores através de rádio frequência. Deste modo minimiza-se bastante os custos de

instalação visto que o número de cablagens e material necessário para instalar o sistema será inferior.



## Referências Bibliográficas

- [1] Manual prático de iluminação, Osram.
- [2] Teixeira, Armínio., *Iluminação Interior - O Projecto Luminotécnico*. Porto : FEUP.
- [3] EDP Eficiência Energética <http://www.edp.pt/>
- [4] <http://www.dgge.pt/>
- [5] Manual Philips iluminação energeticamente eficiente <http://www.lighting.philips.pt/>
- [6] <https://navelocidadedaluz.wordpress.com/tag/co2/>
- [7] [http://www.ecocasa.pt/energia\\_content.php?id=2](http://www.ecocasa.pt/energia_content.php?id=2)
- [8] [www.dali-ag.org](http://www.dali-ag.org)
- [9] EN 12464-1. 2002. Norma Europeia EN 12464-1. Light and lighting - Lighting of work places - Part 1: Indoor. 2002. Bruxelles. EN 12464-1:2002.
- [10] EN 15251. 2007. European Standard EN 15251. Indoor environmental input parameters for design and assessment of energy performance of buildings addressing indoor air quality, thermal environment, lighting and acoustics. s.l., Brussels, Belgium : International Organization for Standardization, 2007. EN 15251:2007:E.
- [11] Arduino Team. 2012. Arduino. [Online] ©Arduino , 2012. <http://www.arduino.cc/>.
- [12] <https://adafruit.com/pir-passive-infrared-proximity-motion-sensor/how-pirs-work>  
[www.adafruit.com](http://www.adafruit.com)
- [13] <http://forum.arduino.cc/index.php?topic=162555.0>
- [14] 2012. Github. harlequin-tech / WiFlyHQ. [Online] GitHub Inc., 2012. [Citação: 15 de 06 de 2012.] <https://github.com/>.
- [15] SparkFun Electronics. 2012. Sparkfun Electronics. [Online] 2012. <http://www.sparkfun.com/>.
- [16] <http://www.ebay.com/>
- [17] <http://pt.farnell.com/>
- [18] [http://www.ecocasa.pt/energia\\_content.php?id=1](http://www.ecocasa.pt/energia_content.php?id=1)
- [19] [http://www.lighting.philips.com/main/prof/lamps/compact-fluorescent-non-integrated/pl-c/pl-c-cluster-4-pin-base-energy-advantage/927906883520\\_NA/product](http://www.lighting.philips.com/main/prof/lamps/compact-fluorescent-non-integrated/pl-c/pl-c-cluster-4-pin-base-energy-advantage/927906883520_NA/product)

[20] <http://www.ebay.co.uk/itm/18-Watt-Electronic-Ballast-Control-Gear-Fluorescent-18W-VOSSLOH-SCHWABE-188219-/300879818012>

[21] <http://rohms.rohm.com/en/products/databook/datasheet/ic/sensor/light/bh1750fvi-e.pdf>

## **Anexos**

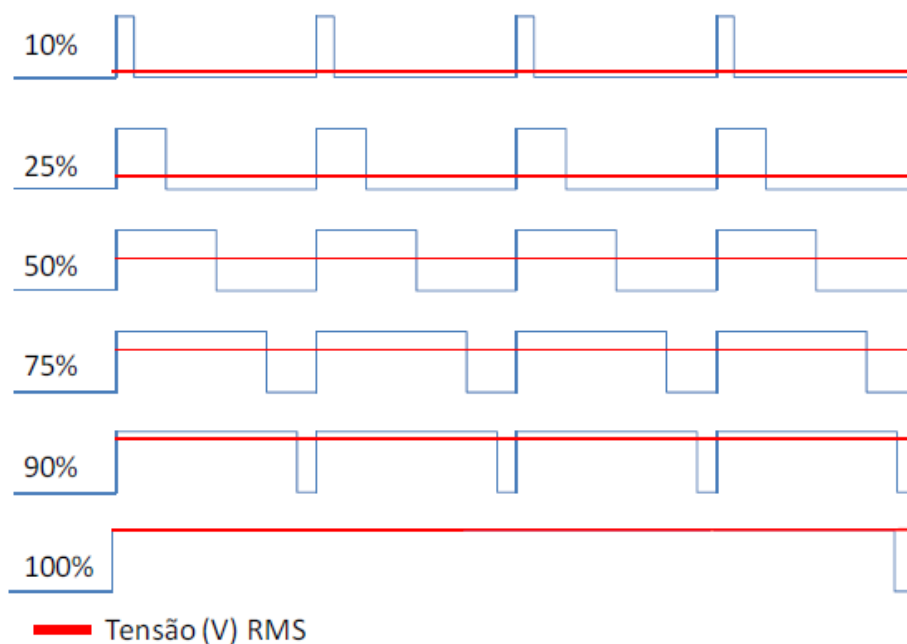
### **Anexo A – PWM**

#### **- Pulse-Width Modulation**

A modulação por largura de pulso (MLP) mais conhecida pela sigla em inglês PWM (Pulse-Width Modulation) de um sinal envolve a modulação de sua razão cíclica (duty cycle) para transportar qualquer informação sobre um canal de comunicação ou controlar o valor da alimentação entregue à carga.

A modelação PWM é uma maneira de codificar digitalmente níveis de um sinal analógico, através da utilização de contadores precisos, o ciclo de trabalho de uma onda quadrada é modulado de forma a codificar um nível específico de um sinal analógico. O sinal PWM é digital devido ao fato de que em um dado instante de tempo, o sinal é máximo ou mínimo (zero). Desse modo, a fonte de tensão ou corrente fornece energia para o dispositivo analógico através de uma série de pulsos que se repetem ao longo do tempo e podem assumir valores máximos e mínimos, mas nunca assumem valores intermediários. O período em que o sinal é máximo é aquele no qual a fonte DC fornece energia para o circuito. Já o período no qual o sinal é mínimo é aquele no qual não ocorre fornecimento de energia. Dado um número suficiente de bandas de frequência, qualquer valor analógico pode ser codificado com PWM.

A Figura 2.17 mostra diferentes sinais PWM. O primeiro sinal da figura mostra um PWM com ciclo de trabalho de 10%. Isso significa que o sinal é máximo em 10% do período e mínimo no restante. Os restantes sinais representam sinais PWM com saídas de 25%, 50%, 75%, 90% e 100% de ciclo de trabalho respectivamente. Esses sinais PWM codificam diferentes valores de sinais analógicos, um com 10% do valor máximo que pode ser obtido e outros com 25%, 50%, 75%, 90% e 100% respectivamente. Se, por exemplo uma tensão máxima de 9V e o ciclo de trabalho de 10%, isso significa que o sinal analógico resultante é de 0,9V.



**Figura 2. 18 – Exemplo de sinal PWM**

Vários microcontroladores incluem controladores PWM. Como exemplo podemos citar o PIC16C67 ou o ATmega368 em que é possível seleccionar o período e o tempo ativo. Para iniciar a operação do PWM, o datasheet sugere que o software deve:

- Definir o período no timer/contador do chip que fornece a onda quadrada modulada.
- Definir o tempo ativo no registo de controlo do PWM.
- Definir a direcção da saída do PWM, a qual é um dos pinos de I/O de propósito geral.
- Iniciar o Contador.
- Iniciar o controlador PWM.

## Anexo B – Source Code

### Anexo B.1 – Prototipo Base

```
#include <avr/pgmspace.h> // library for keeping variables in Flash RAM
#include <LiquidCrystal.h> // i2c liquid crystal library
#include <ooPinChangeInt.h>
#include <AdaEncoder.h> // adafruit.com eoncoder routine.
#include <Wire.h> // wire library
#include <SPI.h> // SPI library
#include <Time.h> // Time library
#include <EEPROM.h> // eeprom library
#include <Arduino.h>
#include <OneWire.h> // DS1820 library

#define DEBUG 1
#define menuTimeout 10000 //delay before settings menus time out & return

#define SIZE_ZONE_EEPROM 10
#define Pos_EEPROM_STATE 0
#define Pos_EEPROM_HOURON 1
#define Pos_EEPROM_MINUTEON 2
#define Pos_EEPROM_HOUROFF 3
#define Pos_EEPROM_MINUTEOFF 4
#define Pos_EEPROM_PWM 5
#define Pos_EEPROM_LUXSETPOINT 6

#define SENSOR1_ADDRESS 0x23 // i2c Address sensor lux
#define SENSOR2_ADDRESS 0x5C

#define SENSOR1_dlux 5// i2c Address sensor lux
#define SENSOR2_dlux 15

#define EncoderPin1 3 // Rotary Encoder Pins
#define EncoderPin2 4 //
#define buttonPin 2 // pin for encoder press button
#define SENSOR1_pwmPin 9
#define SENSOR1_relayPin A3
#define SENSOR2_relayPin A5
#define SENSOR2_pwmPin 6
#define PIR1Pin 12
#define PIR2Pin 12
int DS18S20_Pin = 7; //DS18S20 Signal pin on digital 7
int ldr = A1; // ldr pin
```

```

#define On 255
#define Off 0
#define Cenario1 10
#define Cenario2 100
#define Cenario3 220
#define NUMBER_OF_ZONES 2

int sensorsAddress[NUMBER_OF_ZONES]={SENSOR1_ADDRESS,SENSOR2_ADDRESS};
int luxs[NUMBER_OF_ZONES];
int dlux[NUMBER_OF_ZONES]={SENSOR1_dlux,SENSOR2_dlux};
int luxsSetPoint[NUMBER_OF_ZONES];
int pwm_value[NUMBER_OF_ZONES];
int zone_relay[NUMBER_OF_ZONES]={SENSOR1_relayPin,SENSOR2_relayPin};;
int pwm_pin[NUMBER_OF_ZONES]={SENSOR1_pwmPin, SENSOR2_pwmPin};
int pir_pin[NUMBER_OF_ZONES]={PIR1Pin,PIR1Pin};
int stateControlZone[NUMBER_OF_ZONES];

int calibrationTime = 30; //pir
int temp; // temp var
int r;
int total = 0;
unsigned long lastTime;
unsigned long lastMills=0;
unsigned long timeout;

byte buff[2]; // sensor lux
byte buff1[2];// sensor lux1

byte time[2]; // hr, min
byte LightOn[NUMBER_OF_ZONES][2] = {{9, 00},{9,00}}; // hr, min
byte LightOff[NUMBER_OF_ZONES][2] = {{ 19, 00},{19,00}}; // hr, min
#define HOUR_OFFSET 0
#define MINUTE_OFFSET 1

char buffer[18];

int8_t clicks;
OneWire ds(DS18S20_Pin);
LiquidCrystal lcd(10); // LCD start
char *LCDline1[4]; // pointer array for pointers to LCD line 1 strings
AdaEncoder encoderA = AdaEncoder('a', EncoderPin1, EncoderPin2);
AdaEncoder *knob=NULL;

void setup()

```

```

{
  Serial.begin(9600);
  lcd.begin(16,2);
  Wire.begin();
  for( int i=0; i < NUMBER_OF_ZONES ; i++)
    EEPROMread(i);
  for( int i=0; i < NUMBER_OF_ZONES ; i++)
    initBH1750(sensorsAddress[i]);
  delay(200);
  initRTC();
  rtc();
  for( int i=0; i < NUMBER_OF_ZONES ; i++)
    initPIR(i);
}

void loop()
{
  lcd.clear();
  digitalClockDisplay();
  TemperaturaDisplay();
  sensorlux();
  ldrread();
  time[0] = hour();
  time[1] = minute();
  for(int i=0;i<NUMBER_OF_ZONES;i++)
    controlZone(i);
  entermenu();
  delay(1000);
}

```

## **Anexo B.2 – BH1750**

```

void initBH1750(int address){ // BH1750 sensor init
  Wire.beginTransmission(address);
  Wire.write(0x10); // 1 [lux] auflösung
  Wire.endTransmission();
}

```

```

int readBH1750(int address){ // BH1750 sensor read
  byte i=0;
  byte buf[2];
  Wire.beginTransmission(address);
  Wire.requestFrom(address, 2);
  while(Wire.available() && i<=2){
    buf[i] = Wire.read();
  }
}

```

```

    i++;
  }
  Wire.endTransmission();
  if(i==2){
    return (int)((buf[0]<<8)|buf[1])/1.2;
  }
  return -1;
}

```

## Anexo B.3 – Base Control

```

void writePWM( int zone, int value)
{
  analogWrite(pwm_pin[zone], pwm_negative[zone]?255-value:value);
  if(value==0)
  {
    analogWrite(zone_relay[zone],LOW);
  }
  else
  {
    analogWrite(zone_relay[zone],HIGH);
  }
}

```

```

void on(int zone) //light on
{
  // analogWrite(pwm_pin[zone], On);
  writePWM(zone,On);
}

```

```

void off(int zone) //light off
{
  //analogWrite(pwm_pin[zone], Off);
  writePWM(zone,Off);
}

```

```

void cenario1(int zone) //light mode 1
{
  //analogWrite(pwm_pin[zone], Cenario1);
  writePWM(zone,Cenario1);
}

```

```

void cenario2(int zone) //light mode 2
{
  //analogWrite(pwm_pin[zone], Cenario2);
}

```

```

    writePWM(zone,Cenario2);
}

void cenario3(int zone) //light mode 3
{
    //analogWrite(pwm_pin[zone], Cenario3);
    writePWM(zone,Cenario3);
}

void outputs ()
{
    pinMode(SENSOR1_pwmPin, OUTPUT);
    pinMode(SENSOR2_pwmPin, OUTPUT);
    pinMode(PIR1Pin, INPUT);
    pinMode(buttonPin, INPUT);
    pinMode(ldr, INPUT);
    digitalWrite(buttonPin,LOW);
}

void controlZone(int zone)
{
    switch (stateControlZone[zone]) {
        case 0:
            off(zone);
            break;
        case 1:
            on(zone);
            break;
        case 2:
            analogWrite(pwm_pin[zone], Cenario1);
            break;
        case 3:
            analogWrite(pwm_pin[zone], Cenario2);
            break;
        case 4:
            analogWrite(pwm_pin[zone], Cenario3);
            break;
        case 5:
            luxcontrol(zone);
            horario(zone);
            break;
        case 6:
            pir(zone);
    }
}

```

```

        break;
    case 7:
        lr(zone);
        return;
        break;
    }
}

```

int lcdNumberChg (int number, int numberMin, int numberMax, int loc) // LCD change number

```

{
    lcd.setCursor(loc,1);
    lcd.print(F("^^"));
    timeout = millis() + menuTimeout;
    byte inByte;
    do
    {
        AdaEncoder *thisEncoder=NULL;
        thisEncoder=AdaEncoder::genie();
        if (thisEncoder != NULL){
            timeout = millis() + menuTimeout;
            int clicks = thisEncoder->query();
            number += clicks;
            clicks = 0;
            if (number > numberMax) number = numberMin;
            if (number < numberMin) number = numberMax;
            lcd.setCursor(loc,0);
            if (number < 10) lcd.print('0');
            lcd.print(number);
        }
        if(digitalRead(buttonPin))
        {
            while (digitalRead(buttonPin))
            {
                delay(20);
            }
            delay(20);
            timeout = millis() - 1;
        }
    }
    while (timeout > millis());
    lcd.setCursor(loc,1);
    lcd.print(F(" "));
    timeout = millis() + menuTimeout;
    return number;
}

```

```

}

int freeRam ()
{
extern int __heap_start, *__brkval;
int v;
return (int) &v - (__brkval == 0 ? (int) &__heap_start : (int) __brkval);
}

```

## Anexo B.4 – Controlo Luminosidade

```

void sensorlux() // lux sensor read
{
for(int i=0;i<NUMBER_OF_ZONES;i++){
luxs[i]=readBH1750(sensorsAddress[i]);
#if DEBUG == 1
lcd.setCursor(5*(i+1), 0);
lcd.print(luxs[i]);
lcd.println("lx ");
#endif
}
}

void luxcontrol(int zone) //Dimming function
{
luxs[zone]=readBH1750(sensorsAddress[zone]);
#if DEBUG == 1
Serial.println(luxs[zone]);
#endif
if(luxs[zone] >= luxsSetPoint[zone] - dlux[zone] && luxs[zone] <=
luxsSetPoint[zone] + dlux[zone])
{
Serial.println(pwm_value[zone]);
Serial.println("Pfixo");
return;
}
if (luxs[zone] < luxsSetPoint[zone] - dlux[zone])
{
pwm_value[zone]=pwm_value[zone]+1;
Serial.println(pwm_value[zone]);
Serial.println("Psubir");
}
}

```

```

}
if (luxs[zone] < luxsSetPoint[zone] - dlux[zone] && pwm_value[zone]>=249)
{
  pwm_value[zone]=pwm_maxValue[zone];

  Serial.println(pwm_value[zone]);
  Serial.println("Pmaximo");
}
if (luxs[zone] > luxsSetPoint[zone] + dlux[zone] && pwm_value[zone]<=1)
{
  pwm_value[zone]=0;

  Serial.println(pwm_value[zone]);
  Serial.println("Pminimo");
}
else if (luxs[zone] > luxsSetPoint[zone] + dlux[zone])
{
  pwm_value[zone]=pwm_value[zone]-1;
  Serial.println(pwm_value[zone]);
  Serial.println("Pdescer");

}
writePWM(zone,pwm_value[zone]);
}

```

```

void lumi(char* label,int zone) //select lux
{
  lcd.clear();
  lcd.print(label);

  if (pwm_value[zone] < 10) lcd.print(F("00"));
  lcd.print(pwm_value[zone]);
  analogWrite(pwm_pin[zone], pwm_value[zone]);
  lcd.setCursor(6,1);
  lcd.print(F(" "));
  pwm_value[zone] = pwmChg(pwm_value[zone], 0, 255, 5,zone);
  writeParamEEPROM(Pos_EEPROM_PWM,pwm_value[zone],zone);
  delay(2000);
}

```

```

int pwmChg (int number, int numberMin, int numberMax, int loc, int zone) // lcd
pwm change
{
  lcd.setCursor(loc,1);
  lcd.print(F(" "));
  timeout = millis() + menuTimeout;
  byte inByte;
  do
  {
    luxs[zone] =readBH1750(sensorsAddress[zone]);
    luxsSetPoint[zone]= luxs[zone];
    lcd.setCursor(10, 0);
    lcd.print(luxsSetPoint[zone]);
    lcd.println("lx ");
    AdaEncoder *thisEncoder=NULL;
    thisEncoder=AdaEncoder::genie();
    if (thisEncoder != NULL)
    {
      timeout = millis() + menuTimeout;
      int clicks = thisEncoder->query(); // int instead of byte to make compatible with
number var
      number += clicks;
      clicks = 0;
      if (number > numberMax) number = numberMin;
      if (number < numberMin) number = numberMax;
      if (number < 10);
      {
        lcd.setCursor(7,0);
        lcd.print(" ");}
      lcd.setCursor(loc,0);
      if (number < 10) lcd.print('0');
      lcd.print(number);
      analogWrite(pwm_pin[zone],number);
    }
    if(digitalRead(buttonPin))
    {
      luxsSetPoint[zone] =readBH1750(sensorsAddress[zone]);
      writeParamEEPROM(Pos_EEPROM_LUXSETPOINT,luxsSetPoint[zone],zone);
      while (digitalRead(buttonPin))
      {
        delay(20); // wait for it to be released
      }
      delay(20); // delay for button debounce
      timeout = millis() - 1; // set timeout to exit while loop
    }
  }
}

```

```

    }
    while (timeout > millis());
    lcd.setCursor(loc,1);
    lcd.print(F(" "));
    timeout = millis() + menuTimeout;
    pwm_value[zone]=number;
    EEPROM.write(SIZE_ZONE_EEPROM*zone+Pos_EEPROM_PWM,pwm_value[zone]);
    return number;
}

```

## Anexo B.5 – DS1820

```

void TemperaturaDisplay(){ // display temp funtion
    float temperature = getTemp();
    temp = temperature;
    lcd.setCursor(10,1);
    lcd.print("T=");
    lcd.print(temp);
    lcd.print("C");
}

float getTemp() { //returns the temperature from one DS18S20 in DEG Celsius
    byte data[12];
    byte addr[8];
    if ( !ds.search(addr))
    {
        ds.reset_search();
        return -1000;
    }
    if ( OneWire::crc8( addr, 7) != addr[7])
    {
        Serial.println("CRC is not valid!");
        return -1000;
    }
    if ( addr[0] != 0x10 && addr[0] != 0x28)
    {
        Serial.print("Device is not recognized");
        return -1000;
    }
    ds.reset();
    ds.select(addr);
    ds.write(0x44,1);
    byte present = ds.reset();
    ds.select(addr);
    ds.write(0xBE);

```

```

    for (int i = 0; i < 9; i++)
    {
        data[i] = ds.read();
    }
    ds.reset_search();
    byte MSB = data[1];
    byte LSB = data[0];
    float tempRead = ((MSB << 8) | LSB);
    float TemperatureSum = tempRead / 16;
    return TemperatureSum;
}

```

## Anexo B.6 – EEPROM

```

void writeParamEEPROM(int paramPos,int param,int zone){ //keeping vars EEPROM
    EEPROM.write(SIZE_ZONE_EEPROM*zone+paramPos,param);
}

```

```

void EEPROMread(int zone){ //read variables EEPROM
    stateControlZone[zone] = EEPROM.read(Pos_EEPROM_STATE);
    LightOn[zone][0] = EEPROM.read(Pos_EEPROM_HOURON);
    LightOn[zone][1] = EEPROM.read(Pos_EEPROM_MINUTEON);
    LightOff[zone][0]= EEPROM.read(Pos_EEPROM_HOUROFF);
    LightOff[zone][1]= EEPROM.read(Pos_EEPROM_HOUROFF);
    luxsSetPoint[zone]=EEPROM.read(Pos_EEPROM_LUXSETPOINT);
    pwm_value[zone]=EEPROM.read(Pos_EEPROM_PWM);
}

```

## Anexo B.7 – PIR

```

#include <Time.h>
int stateControl_PIR[NUMBER_OF_ZONES];
time_t timeStamp[NUMBER_OF_ZONES];
#define STATECONTROL_PIR_IDLE 0
#define STATECONTROL_PIR_ACTIVE 1
#define STATECONTROL_PIR_TIMEOUT 2
#define STATECONTROL_PIR_UP 3
#define STATECONTROL_PIR_DOWN 5
#define PIR_TIMEOUT 5

void initPIR(int zone){ // PIR sensor init
    pinMode(pir_pin[zone], INPUT);
    digitalWrite(pir_pin[zone], LOW);
    for(int i = 0; i < NUMBER_OF_ZONES; i++)
        stateControl_PIR[i]= STATECONTROL_PIR_IDLE;
}

```

```

}

void pir(int zone){ //detection function
int pira;
pira = digitalRead(pir_pin[zone]);
switch(stateControl_PIR[zone]){
case STATECONTROL_PIR_IDLE:
    if (pira == HIGH){
        Serial.println("detection ON");
        analogWrite(pwm_pin[zone], On);
        stateControl_PIR[zone]=STATECONTROL_PIR_ACTIVE;
    }
    break;
case STATECONTROL_PIR_ACTIVE:
    if(pira == LOW)
    {
        Serial.println("Non detection");

        timeStamp[zone] = now();

        stateControl_PIR[zone]=STATECONTROL_PIR_TIMEOUT;
    }
    break;
case STATECONTROL_PIR_TIMEOUT:
    if ((now()-timeStamp[zone])>PIR_TIMEOUT){
        Serial.println("OFF");
        analogWrite(pwm_pin[zone], Off);
        stateControl_PIR[zone]=STATECONTROL_PIR_IDLE;
    }
    if (pira == HIGH){
        Serial.println("detection ON");
        analogWrite(pwm_pin[zone], On);
        stateControl_PIR[zone]=STATECONTROL_PIR_ACTIVE;
    }
    break;
}
return;
}

```

## Anexo B.8 – RTC

```

#include <DS1307RTC.h> //RTC library
const char *monthName[12] = {
    "Jan", "Feb", "Mar", "Apr", "May", "Jun",
    "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"
}

```

```

};

tmElements_t tm;

void rtc(){ // get the date and time the compiler was run
  bool parse=false;
  bool config=false;
  if (getDate(__DATE__) && getTime(__TIME__))
  {
    parse = true;
    if (RTC.write(tm)) // and configure the RTC with this info
    {
      config = true;
    }
  }
}

bool getTime(const char *str)
{
  int Hour, Min, Sec;
  if (sscanf(str, "%d:%d:%d", &Hour, &Min, &Sec) != 3) return false;
  tm.Hour = Hour;
  tm.Minute = Min;
  tm.Second = Sec;
  return true;
}

bool getDate(const char *str)
{
  char Month[12];
  int Day, Year;
  uint8_t monthIndex;

  if (sscanf(str, "%s %d %d", Month, &Day, &Year) != 3) return false;
  for (monthIndex = 0; monthIndex < 12; monthIndex++) {
    if (strcmp(Month, monthName[monthIndex]) == 0) break;
  }
  if (monthIndex >= 12) return false;
  tm.Day = Day;
  tm.Month = monthIndex + 1;
  tm.Year = CalendarYrToTm(Year);
  return true;
}

void initRTC() // the function to get the time from the RTC

```

```

{
  setSyncProvider(RTC.get);

  if(timeStatus()!= timeSet)
    Serial.println("Unable to sync with the RTC");
  else
    Serial.println("RTC has set the system time");
}

void digitalClockDisplay() //display time function
{
  lcd.setCursor(0, 0); // Set LCD cursor position (column, row)
  if (hour()<10)
  {
    lcd.print(0);
  }
  lcd.print(hour());
  lcd.print(":");
  if (minute()<10)
  {
    lcd.print(0);
  }
  lcd.print (minute());
  lcd.print(" ");
  lcd.setCursor(0, 1);
  if (day()<10)
  lcd.print(0); // Set LCD cursor position (column,row)
  lcd.print(day());
  lcd.print(".");
  lcd.print(month());
  lcd.print(".");
  lcd.print(year());
}

```

## Anexo B.9 – Controlo Horário

```

void horario(int zone){ //schedule control function
  int LightON_number = LightOn[zone][0]*60+LightOn[zone][1];
  int LightOFF_number =LightOff[zone][0]*60+LightOff[zone][1];
  int time_number =time[0]*60+time[1];
  if( (LightON_number<LightOFF_number) && (time_number >=
LightON_number) && (time_number<LightOFF_number));
  {
    digitalWrite(zone_relay[zone], On);
  }
}

```

```

        if( (LightON_number>LightOFF_number) && ((time_number >= LightON_number)
|| (time_number<LightOFF_number)));
        {
            digitalWrite(zone_relay[zone], On);
        }
        if( (LightON_number<LightOFF_number) && ((time_number < LightON_number) ||
(time_number>=LightOFF_number)));
        {
            digitalWrite(zone_relay[zone], Off);
        }
        if( (LightON_number>LightOFF_number) && ((time_number < LightON_number) ||
(time_number>=LightOFF_number)));
        {
            digitalWrite(zone_relay[zone], Off);
        }
    }

void MySetTime(char* label, byte *t){ //insert hour & minute function
    lcd.clear();
    lcd.print(label);
    if (t[ HOUR_OFFSET ] < 10) lcd.print(F("0"));
    lcd.print(t[ HOUR_OFFSET ]);
    lcd.print(F(":"));
    if (t[ MINUTE_OFFSET ] < 10) lcd.print(F("0"));
    lcd.print(t[ MINUTE_OFFSET ]);
    lcd.setCursor(10,1);
    lcd.print(F("^^"));
    t[ HOUR_OFFSET ] = lcdNumberChg(t[ HOUR_OFFSET ], 0, 23, 10);
    t[ MINUTE_OFFSET ] = lcdNumberChg(t[ MINUTE_OFFSET ], 0, 59, 13);
}

```

## Anexo B.10 – Menu

```

//MENUS LCD
prog_char mainMenu1[] PROGMEM = "1.Zona 1  ";
prog_char mainMenu2[] PROGMEM = "2.Zona 2  ";
prog_char mainMenu3[] PROGMEM = "3.Sair   ";

PROGMEM const char *mainMenu[] = {
    mainMenu1, mainMenu2, mainMenu3};

prog_char zonaMenu1[] PROGMEM = "1.Off    ";
prog_char zonaMenu2[] PROGMEM = "2.On     ";
prog_char zonaMenu3[] PROGMEM = "3.Cenario 1 ";
prog_char zonaMenu4[] PROGMEM = "4.Cenario 2 ";

```

```

prog_char zonaMenu5[] PROGMEM = "5.Cenario 3 ";
prog_char zonaMenu6[] PROGMEM = "6.Auto ";
prog_char zonaMenu7[] PROGMEM = "7.Deteccao ";
prog_char zonaMenu8[] PROGMEM = "8.Crepuscular ";
prog_char zonaMenu9[] PROGMEM = "9.Definicoes ";
prog_char zonaMenu10[] PROGMEM = "10.Sair ";
PROGMEM const char *zonaMenu[] = {
    zonaMenu1, zonaMenu2, zonaMenu3, zonaMenu4, zonaMenu5, zonaMenu6,
zonaMenu7, zonaMenu8, zonaMenu9, zonaMenu10};

```

```

prog_char Select0[] PROGMEM = "1. Auto ";
prog_char Select1[] PROGMEM = "2. Off ";
prog_char Select2[] PROGMEM = "3. On ";
PROGMEM const char* Select[]={
    Select0, Select1, Select2};

```

```

prog_char definicoesMenu0[] PROGMEM = "1. Horario ";
prog_char definicoesMenu1[] PROGMEM = "2. Luminosidade";
prog_char definicoesMenu2[] PROGMEM = "3. Sair ";
PROGMEM const char* definicoesMenu[]={
    definicoesMenu0, definicoesMenu1, definicoesMenu2};

```

```

prog_char lightTimeMenu0[] PROGMEM = "1.Hora Ligar ";
prog_char lightTimeMenu1[] PROGMEM = "2.Hora Desligar";
prog_char lightTimeMenu2[] PROGMEM = "3.Sair ";
PROGMEM const char* lightTimeMenu[]={
    lightTimeMenu0, lightTimeMenu1, lightTimeMenu2};

```

```

void entermenu(){ //enter menu function
    if(!digitalRead(buttonPin) == 1)
    {
        knob = AdaEncoder::genie(); // check for input from rotary encoder
        if (knob != NULL || digitalRead(buttonPin))
        {
            while(digitalRead(buttonPin));
            lcd.clear();
            menu();
        }
    }
}

```

```

void menu(){ //run menus function
    int item;
    timeout = millis() + menuTimeout;

```

```

do{
  item = menuSelect(mainMenu, 3);
  switch (item) {
  case 0:
    zoneMenu(0);
    break;
  case 1:
    zoneMenu(1);
    break;
  case 2:
    return;
    break;
  }
}
while (item != 2 && timeout > millis());
}

```

```

void zoneMenu(int zone) //state control function
{
  int item;
  timeout = millis() + menuTimeout;
  do{
    item = menuSelectZone(zonaMenu, 10,zone);
    switch (item)
    {
  case 0:
    stateControlZone[zone]=0;
    writeParamEEPROM(Pos_EEPROM_STATE,stateControlZone[zone],zone);
    break;
  case 1:
    stateControlZone[zone]=1;
    writeParamEEPROM(Pos_EEPROM_STATE,stateControlZone[zone],zone);
    break;
  case 2:
    stateControlZone[zone]=2;
    writeParamEEPROM(Pos_EEPROM_STATE,stateControlZone[zone],zone);
    break;
  case 3:
    stateControlZone[zone]=3;
    writeParamEEPROM(Pos_EEPROM_STATE,stateControlZone[zone],zone);
    break;
  case 4:
    stateControlZone[zone]=4;
    writeParamEEPROM(Pos_EEPROM_STATE,stateControlZone[zone],zone);
    break;

```

```

case 5:
    stateControlZone[zone]=5;
    writeParamEEPROM(Pos_EEPROM_STATE,stateControlZone[zone],zone);
    break;
case 6:
    stateControlZone[zone]=6;
    writeParamEEPROM(Pos_EEPROM_STATE,stateControlZone[zone],zone);
    break;
case 7:
    stateControlZone[zone]=7;
    writeParamEEPROM(Pos_EEPROM_STATE,stateControlZone[zone],zone);
    break;
case 8:
    definicoes(zone);
    break;
case 9:
    return;
    break;
}
controlZone(zone);
}
while (item != 9 && timeout > millis());
}

```

```

void definicoes(int zone) // definitions menu funtion
{
    int item;
    timeout = millis() + menuTimeout;
    do{
        item = menuSelectZone(definicoesMenu, 3,zone);
        switch (item)
        {
            case 0:
                setLightTimes(zone);
                break;
            case 1:
                lumi("PWM: ",zone);
                break;
            case 2:
                return;
                break;
        }
    }
    while (item != 3);
}

```

```
}
```

```
void setLightTimes(int zone) //set light times funtion
```

```
{  
  int item;  
  do{  
    item = menuSelectZone(lightTimeMenu, 3,zone);  
    switch (item) {  
      case 0:  
        MySetTime("Ligar: ", LightOn[zone]);  
        writeParamEEPROM(Pos_EEPROM_HOURON,LightOn[zone][0],zone);  
        writeParamEEPROM(Pos_EEPROM_MINUTEON,LightOn[zone][1],zone);  
        break;  
      case 1:  
        MySetTime("Desligar: ", LightOff[zone]);  
        writeParamEEPROM(Pos_EEPROM_HOUROFF,LightOff[zone][0],zone);  
        writeParamEEPROM(Pos_EEPROM_HOUROFF,LightOff[zone][1],zone);  
        break;  
      case 2:  
        return;  
        break;  
    }  
  }  
  while (item != 6);  
}
```

```
int menuSelect(const char* menustring[], int menuitems) //select menus funtion
```

```
{  
  int currentItem = 0;  
  lcd.clear();  
  lcd.setCursor(0,0);  
  lcd.print(F(">"));  
  lcd.println(strcpy_P(buffer, (char*)pgm_read_word(&(menustring[0]))));  
  lcd.setCursor(0,1);  
  lcd.print(F(" "));  
  lcd.println(strcpy_P(buffer,  
(char*)pgm_read_word(&(menustring[(1)%menuitems]))));  
  timeout = millis() + menuTimeout;  
  do {  
    AdaEncoder *knob=NULL;  
    knob=AdaEncoder::genie();  
    if (knob != NULL){  
      clicks=knob->query();  
    }  
  }  
}
```

```

    currentItem += clicks;
    clicks = 0;
    timeout = millis() + menuTimeout;
    currentItem = (currentItem + menuItems)%menuItems;
    lcd.setCursor(0,0);
    lcd.print(F(">"));
    lcd.println(strcpy_P(buffer,
(char*)pgm_read_word(&(menustring[currentItem]))));
    lcd.setCursor(0,1);
    if(currentItem<menuItems-1){
        lcd.print(F(" "));
        lcd.println(strcpy_P(buffer, (char*)pgm_read_word(&(menustring[(currentItem
+1)%menuItems]))));
    }
    else
    {
        lcd.println(F(" "));
    }
}
if(digitalRead(buttonPin)){
    while (digitalRead(buttonPin)){
        delay(20);
    }
    delay(30);
    timeout = millis() + menuTimeout;
    return currentItem;
}
}
while (timeout > millis()); // end do
timeout = millis() + menuTimeout;
return (menuItems-1);
}

```

```

int menuSelectZone(const char* menustring[], int menuItems,int zone) //run menus
{
    int currentItem = 0;
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(F(">"));
    lcd.println(strcpy_P(buffer, (char*)pgm_read_word(&(menustring[0]))));
    lcd.setCursor(0,1);
    lcd.print(F(" "));
    lcd.println(strcpy_P(buffer,
(char*)pgm_read_word(&(menustring[(1)%menuItems]))));
    timeout = millis() + menuTimeout;
}

```

```

do {
  AdaEncoder *knob=NULL;
  knob=AdaEncoder::genie();
  if (knob != NULL){
    clicks=knob->query();
    currentItem += clicks;
    clicks = 0;
    timeout = millis() + menuTimeout;
    currentItem = (currentItem + menuItems)%menuItems;
    lcd.setCursor(0,0);
    lcd.print(F(">"));
    lcd.println(strcpy_P(buffer,
(char*)pgm_read_word(&(menustring[currentItem]))));
    if(currentItem==stateControlZone[zone]){
      lcd.setCursor(15,0);
      lcd.print('*');}
      lcd.setCursor(0,1);
      if(currentItem<menuItems-1){
        lcd.print(F(" "));
        lcd.println(strcpy_P(buffer, (char*)pgm_read_word(&(menustring[(currentItem
+1)%menuItems]))));
        if((currentItem+1)==stateControlZone[zone]){
          lcd.setCursor(15,1);
          lcd.print('*');}
        }
        else
        {
          lcd.println(F("      "));
        }
      }
    }
    if(digitalRead(buttonPin)){
      while (digitalRead(buttonPin)){
        delay(20);
      }
      delay(30);
      timeout = millis() + menuTimeout;
      return currentItem;
    }
  }
}
while (timeout > millis());
timeout = millis() + menuTimeout;
return(menuItems-1);
}

```

