



Sistema de Controlo de Luminosidade Auto-Adaptativo usando Tecnologias de IoT

João Luís Pessegueiro Costa - a31146

Dissertação apresentada à Escola Superior de Tecnologia e Gestão de Bragança para
obter o grau de Mestrado em Engenharia Industrial.

Trabalho Orientado por:

Professor José Barbosa

Professor Paulo Leitão

Bragança

2019-2020



Sistema de Controlo de Luminosidade Auto-Adaptativo usando Tecnologias de IoT

João Luís Pessegueiro Costa - a31146

Dissertação apresentada à Escola Superior de Tecnologia e Gestão de Bragança para obter o grau de Mestrado em Engenharia Industrial.

Trabalho Orientado por:
Professor José Barbosa
Professor Paulo Leitão

Bragança
2019-2020

Agradecimentos

Primeiramente gostaria de agradecer à bonita cidade de Bragança e a toda academia do IPB, por todos os momentos passados ao longo desta caminhada, momentos esses que irei recordar ao longo da minha vida.

A todos os meus docentes que me acompanharam e me transmitiram os melhores ensinamentos ao longo do meu percurso académico, sem o esforço e dedicação destes nada disto seria possível.

Ao meu orientador, Professor Doutor José Barbosa e co-orientador, Professor Doutor Paulo Leitão, pela disponibilidade e cooperação para comigo ao longo da minha dissertação.

Aos meus colegas, que se tornaram grandes amigos, um agradecimento especial, sem dúvida que foram importantes ao longo destes anos.

À minha namorada, por todo o amor e cumplicidade, pelos bons conselhos e pela disponibilidade em ajudar-me sempre que mais necessitei.

Por último e não menos importante, aos meus pais, por todo o esforço e confiança que depositaram em mim, com toda a certeza, as pessoas mais importantes e influentes em todo o meu percurso académico.

Resumo

Os sistemas de iluminação em sistemas robóticos são controlados de forma automática e reprogramável usando tecnologias de IoT, tecnologias essas capazes de se identificar em uma rede e de comunicar entre si, ou seja, todo o tipo de objectos e aparelhos com capacidade computacional e de ligação à web.

A aplicação de IoT consiste na sensorização de objectos, com o objectivo de captura e transmissão de dados para um conjunto de aplicações e usuários conectados em rede. Com isto, para usufruir de todos benefícios que a tecnologia oferece, é necessário não só sensorizar objectos e colectar dados, como também obter aplicações práticas para os dados colectados, permitindo análises e tomada de acções com base nos dados obtidos.

Contudo, a aplicação de IoT é complexa, isto é, a tecnologia está directamente relacionada a um grande volume de dados colectados e sua necessidade de processamento. Por intermédio da aplicação combinada de IoT com soluções de Inteligência Artificial (algoritmos e Machine Learning), é possível identificar padrões e emitir alertas para intervenção humana ou acções programadas.

Neste caso, foi desenvolvida uma aplicação em Node-RED, esta comunica por Wi-Fi, através do protocolo de comunicação MQTT, tendo como finalidade controlar e regular a intensidade do LED através dos valores detectados pelo sensor, de modo a obter uma maior eficiência energética, diminuindo assim consumos desnecessários do meio envolvente.

Palavras-chave: IoT, Sensorização, LED.

Abstract

The lighting systems in robotic systems are controlled automatically and reprogrammable using IoT technologies, technologies that are able to identify themselves in a network and communicate with each other, that is, all types of objects and devices with computational and connection capacity to the web.

The application of IoT consists of object sensing, with the objective of capturing and transmitting data to a set of applications and users connected in a network. With this, to take advantage of all the benefits that technology offers, it is necessary not only to sense objects and collect data, but also to obtain practical applications for the collected data, allowing analysis and action taking based on the data obtained.

However, the application of IoT is complex, that is, the technology is directly related to a large volume of collected data and its need for processing. Through the combined application of IoT with Artificial Intelligence solutions (algorithms and Machine Learning), it is possible to identify patterns and issue alerts for human intervention or programmed actions.

In this case, an application was developed in Node-RED, which communicates via Wi-Fi, using the MQTT communication protocol, with the purpose of controlling and regulating the LED intensity through the values detected by the sensor, in order to obtain greater efficiency. thus reducing unnecessary consumption of the evolving medium.

Keywords: IoT, Sensing, LED.

Conteúdo

1	Introdução	1
1.1	Contexto	1
1.2	Objectivos	2
1.3	Estrutura do Documento	2
2	Estado de Arte	5
2.1	Luz e a Visão Humana	6
2.1.1	História da Teoria da Luz	6
2.1.2	Espectro de Luz Visível	7
2.1.3	Acuidade Visual	9
2.2	Internet das Coisas - IoT	9
2.2.1	O que é a IoT?	10
2.2.2	Impacto da IoT na Indústria	11
2.3	Protocolo de Comunicação	12
2.4	Automatização e Robótica Industrial	13
2.4.1	A Automatização	13
2.4.2	A Robótica Industrial	14
2.4.3	Tipos de Automatização e Robótica Industrial	15
2.4.4	Vantagens da Automatização e Robótica Industrial	16
2.4.5	Controlo de Iluminação	16
2.4.6	Importância da Iluminação em Sistemas Robóticas	16

2.4.7	A Visão Robótica	17
2.4.8	Aplicações do Uso de Controlo Dinâmico de Iluminação em Sistemas Robóticos	18
3	Arquitectura do Sistema	21
3.1	Visão Geral	21
3.2	Sistema de Controlo Luminoso em Ambiente Laboratorial	22
3.3	MQTT	23
3.4	Arduino	25
3.4.1	Hardware do Arduino Uno	25
3.4.2	Especificações da Placa Arduino UNO	25
3.5	Shield Wi-Fi	26
3.6	LED	27
3.7	Smart Phone	27
3.8	ESP32	27
3.8.1	ESP-WROOM-32	28
3.9	Sensor de Luminosidade	29
3.9.1	Especificações	30
3.10	Bateria	30
3.11	Regulador de Tensão LM1117-3.3	31
3.11.1	Diagrama Funcional	31
3.11.2	Especificações Técnicas	32
4	Desenvolvimento	33
4.1	Programação de Software	33
4.1.1	Controlo PID	37
4.2	Desenvolvimento de Software de Interface Gráfica do Utilizador	38
4.2.1	Node-RED	38
4.3	Design de Placa de Circuito Impresso	40
4.3.1	EAGLE	40

4.4	Impressão 3D	42
4.4.1	Desenho do Invólucro	43
5	Resultados e Discussão	45
5.1	Protótipo Final	45
5.2	Aplicação Node-RED	47
5.2.1	Funcionalidade do Sistema de Controlo de Luminosidade	47
6	Conclusões e Trabalho Futuro	51
6.1	Trabalhos Futuros	52
6.1.1	Proposta 1	52
6.1.2	Proposta 2	52
A	Código	A1
A.1	Código 1	A1
A.2	Código 2	A6
B	Proposta de Dissertação	B1

Lista de Tabelas

3.1	Meios de comunicação no MQTT [14].	24
3.2	Especificações da Placa Arduino UNO [21].	26
3.3	Especificações do Sensor de Luminosidade TSL2561 [25].	30
3.4	Especificações Técnicas do LM1117 [27].	32

Lista de Figuras

1.1	Internet das Coisas [1].	2
2.1	Efeito Fotoelétrico [6].	7
2.2	Comprimentos de onda do espectro electromagnético e do espectro visível [8].	8
2.3	Onda sinusoidal periódica.	8
2.4	Esquematização da acuidade visual [10].	9
2.5	Internet das Coisas - IoT [1].	11
2.6	Evolução Industrial [12].	12
2.7	MQTT [14].	13
2.8	Automatização e Robótica Industrial [16].	14
3.1	Arquitectura do Sistema de Controlo Luminoso.	22
3.2	Sistema de Controlo Luminoso em Ambiente Laboratorial	23
3.3	Arquitectura do MQTT [14].	24
3.4	Arduino UNO Rev3 [21].	26
3.5	ESP32 PINOUT [23].	28
3.6	Posição e Conexão do Interruptor.	31
3.7	Diagrama Funcional LM1117-3.3 [26].	32
4.1	Configuração do Sensor TSL2561.	34
4.2	Conectar o microcontrolador à rede Wi-Fi.	35
4.3	Reconectar o microcontrolador à rede Wi-Fi.	35

4.4	Ligar PID.	35
4.5	Porta analógica de saída (OUTPUT).	36
4.6	PWM [28].	37
4.7	Controlo PID.	38
4.8	Esquema da Aplicação de Controlo de Luminosidade no Node-RED.	39
4.9	Interface de desenvolvimento de PCI no Eagle.	40
4.10	Esquemático da PCI.	41
4.11	Layout da PCI.	42
4.12	Desenho CAD 3D em SOLIDWORKS.	43
5.1	PCI Final.	46
5.2	Protótipo Final.	46
5.3	Publicar tópico por MQTT.	47
5.4	Aplicação do Sistema de Controlo de Luminosidade em NodeRED.	48
5.5	Sistema pronto para o controlo de luminosidade.	48
5.6	Início do controlo de luminosidade.	49
5.7	Desligar o controlo de luminosidade.	50
5.8	Retomar o controlo de luminosidade.	50

Capítulo 1

Introdução

A presente dissertação foi desenvolvida a fim de concluir o Mestrado em Engenharia Industrial com Ramo em Engenharia Electrotécnica e de Computadores. Para tal, foi proposto o tema, Sistema de controlo de Luminosidade Auto-Adaptativo usando Tecnologias de Internet das Coisas. Neste capítulo introdutório, é feita uma reflexão do contexto onde se insere esta dissertação. Posteriormente são apresentados os objectivos, bem como a estrutura da mesma.

1.1 Contexto

A Internet das Coisas, fornece um conceito actual e transformador sobre a conexão entre objectos físicos utilizando sensores, chips e softwares. Quando se fala de “coisas” em IoT, refere-se qualquer objecto que teve a implementação de sensores e outros sistemas digitais de modo a funcionar de forma mais inteligente por meio da troca de informações com pessoas e outros objectos, não sendo necessariamente utilizadas conexões de Internet. Estas trocas de informações podem ocorrer por via radiofrequência (RFID), Wi-Fi, Ethernet, Bluetooth, entre outras formas de conexão existentes actualmente. Com isto, as “coisas” são capazes de se conectar a outros objectos e às pessoas, permitindo o envio de comandos, a devolução de dados sobre uso e a identificação da presença um do outro.

visão do ser humano. De seguida, é explicado todo o funcionamento e os meios de conectividade face à tecnologia Internet das Coisas. Posteriormente, e com base nas informações fornecidas directamente por essas empresas, os produtos mais relevantes são apresentados e estudados separadamente em termos de implementação tecnológica. É também realizada uma avaliação comparativa e são apresentadas as conclusões fundamentais sobre o facto de o controlo em si ser adequado ou não ao contexto. Por fim, são estudadas soluções inovadoras emergentes nesse mercado.

No capítulo 3, é realizada uma descrição detalhada de toda a arquitectura do problema em questão. É apresentada uma visão geral de todo o trabalho prático desenvolvido, referindo onde se insere cada componente, as suas especificações, bem como todo o sistema de ligação.

No capítulo 4, é relatado todo o trabalho que foi desenvolvido ao longo deste projecto, bem como todas as ferramentas utilizadas para a execução dos mesmos.

No capítulo 5, são apresentados todos os resultados práticos obtidos com esta dissertação. Posteriormente, serão analisados e discutidos de modo a clarificar todos os procedimentos.

Por fim, no capítulo 6, são apresentadas conclusões, sendo apresentadas propostas que poderão vir a ser desenvolvidas em trabalhos futuros.

Capítulo 2

Estado de Arte

Neste capítulo, várias soluções existentes e propostas para controlo de luminosidade, são examinadas e comparadas com o objectivo de obter uma maior sustentabilidade energética e maior facilidade de comunicação com os sensores de luminosidade.

Numa fase inicial, é realizado um breve estudo sobre a luz e o sistema de visão do ser humano. De seguida, é explicado todo o funcionamento e os meios de conectividade face à tecnologia Internet das Coisas. É também explicado em que consiste a automatização e robótica industrial e qual a sua importância num controlo de iluminação. Posteriormente, e com base nas informações fornecidas directamente por essas empresas, os produtos mais relevantes são apresentados e estudados separadamente em termos de implementação tecnológica. É também realizada uma avaliação comparativa e são apresentadas as conclusões fundamentais sobre o facto de o controlo em si ser adequado ou não ao contexto. Por fim, são estudadas soluções inovadoras emergentes nesse mercado.

Considerando que esta pesquisa foi realizada em Março de 2020 e, à medida que o mercado está em constante evolução, novos produtos são constantemente introduzidos, além de melhorias nos já existentes.

2.1 Luz e a Visão Humana

2.1.1 História da Teoria da Luz

Inicialmente as teorias dos pensadores Gregos acerca da luz foram limitadas pela falta de conhecimento do funcionamento do olho humano. Pitágoras no século V a.C. pensava que a luz era originada pelos objectos visíveis. Platão no século III a.C. defendia a teoria de que o olho é que emitia um raio, que ao tocar no objecto o tornava visível. Esta teoria foi mais tarde (século X) contradita pelo cientista árabe Alhazen, ao afirmar que se tínhamos de fechar as pálpebras ou desviar os olhos quando olhávamos para o Sol, então era porque “algo entrava e não porque algo saía” [2]. No entanto algumas ideias iniciais dos Gregos estavam corretas, por exemplo:

- Empédocles no século V a.C. acreditava que a luz tinha velocidade finita [2];
- Aristóteles no século IV a.C. explicou o fenómeno do arco-íris como uma espécie de reflexão das gotas de água[2];
- Euclides no século III a.C. entendeu a lei da reflexão e as propriedades dos espelhos [2].
- Isaac Newton tornou-se defensor da **Teoria Corpuscular**, onde afirmava que a luz se comporta como uma partícula. Corpos luminosos emitem energia radiante na forma de partículas, lançadas intermitentemente em linha recta [3].
- Cristiaan Huygens defende na mesma altura a **Teoria Ondulatória**, na qual a luz se comporta como uma onda. A luz era resultante da vibração molecular de materiais luminosos, emitida através de uma substância invisível e sem peso, existente no ar e no espaço [3].
- James Maxwell, no século XIX, formula a **Teoria dos Campos Electromagnéticos**, sendo que Hertz a confirma, com a descoberta das ondas electromagnéticas de baixa frequência, cuja teoria previa. Os campos eléctricos e magnéticos afectam-se mutuamente de tal modo que permitem que as ondas viajem pelo espaço. A luz emitida pelos corpos radiantes propaga-se desta forma [4].
- Max Planck, no século XX, propõe a existência de um quantum de luz, isto é, um

pacote finito de energia denominado fóton. Assim, surge a **Teoria Quântica**, que apenas foi compreendida quando Einstein mostrou que podia ser usada para explicar o fenómeno fotoeléctrico, sugerindo que a energia de um fóton está relacionada com a sua frequência, através da constante de Planck, cujo seu valor é de $6,63 \cdot 10^{-34} (J.s)$ [5].

Fenómeno Fotoeléctrico

O efeito fotoeléctrico ocorre quando há emissões de electrões em determinados materiais. Geralmente, este efeito é produzido em materiais metálicos os quais são expostos a uma radiação electromagnética, como a luz.

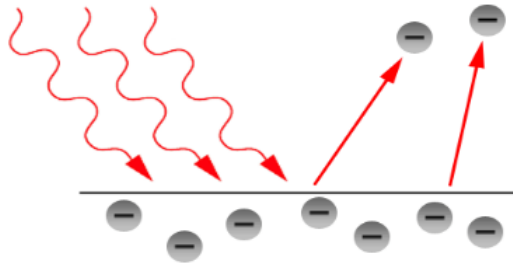


Figura 2.1: Efeito Fotoeléctrico [6].

Por sua vez, Os fótons são minúsculas partículas elementares que possuem energia e são mediadoras do efeito fotoeléctrico. A energia do fóton é calculada pela seguinte fórmula, $E = h.f$, em que o E corresponde à Energia do Fóton, o h à Constante de Planck e o f à Frequência do Fóton [6].

2.1.2 Espectro de Luz Visível

A luz corresponde à radiação da gama de comprimentos de onda, medida em nanómetros (nm), do espectro da radiação electromagnética visível pelo olho humano. Esta situa-se entre a radiação infravermelha e a ultravioleta. O espectro de luz visível está referenciado na 2.2 [7].

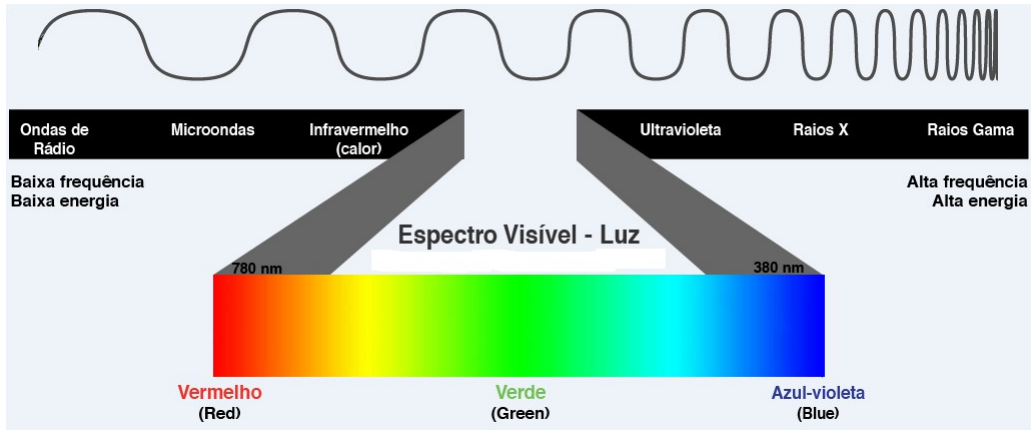


Figura 2.2: Comprimentos de onda do espectro eletromagnético e do espectro visível [8].

O comprimento de onda (λ) é a distância entre dois valores repetidos numa forma de onda periódica.

No vácuo, a velocidade da onda eletromagnética é a velocidade da luz, ou seja, 300 000 km/s. As três grandezas físicas básicas da luz e de toda a radiação eletromagnética são:

- Frequência (cor);
- Amplitude (brilho);
- Polarização (ângulo de vibração).

A frequência de uma onda é o número de ciclos executados por segundo que é inversamente proporcional ao comprimento de onda. O período da onda (T) é o inverso da frequência. Este é o tempo que a onda periódica demora a repetir-se.

A medida, entre o ponto médio e o pico da onda é denominada de amplitude (A) [9]. Na figura 2.3 é ilustrada uma onda sinusoidal periódica.

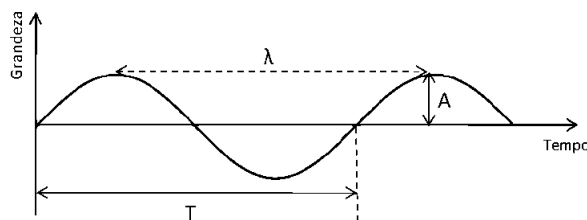


Figura 2.3: Onda sinusoidal periódica.

2.1.3 Acuidade Visual

A acuidade visual é a capacidade que o olho humano tem de identificar com nitidez, pequenos objectos próximos entre si. As distâncias na retina são referidas em termos de ângulo visual (θ). Assim, dizemos que a capacidade do olho em distinguir dois pontos está associada a um certo valor de ângulo visual. A acuidade visual é tanto maior quando menor for o ângulo (θ) [10].

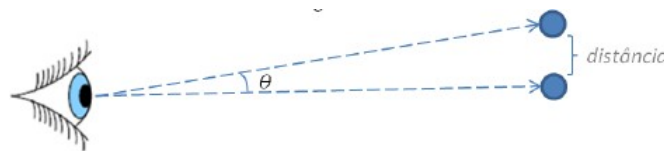


Figura 2.4: Esquemática da acuidade visual [10].

Esta, pode ser afectada por diversos factores:

- Adaptação - é a capacidade que o olho humano possui para se ajustar a diferentes níveis de intensidade luminosa, mediante os quais a pupila irá dilatar ou contrair;
- Acomodação - é o ajustamento das lentes do cristalino do olho de modo a que a imagem esteja permanentemente focada na retina;
- Contraste - é a diferença de luminância entre um objecto que se observa e o seu espaço envolvente;
- Idade - a capacidade visual de uma pessoa vai diminuindo com a idade, uma vez que, com o passar dos anos o cristalino endurece perdendo assim a sua elasticidade, o que complica a tarefa de focalização das imagens dos objectos.

2.2 Internet das Coisas - IoT

A Internet das Coisas - IoT é um termo criado em Setembro de 1999 por Kevin Ashton, um pioneiro tecnológico britânico que concebeu um sistema de sensores omnipresentes ao conectar o mundo físico à Internet, enquanto trabalhava em identificação por rádio frequência (RFID). Embora a Internet, as "coisas"(things) e a conectividade entre elas

sejam os três principais componentes da Internet, o valor acrescentado está no preenchimento das lacunas entre os mundos físico e digital em sistemas. O primeiro dispositivo IoT foi desenvolvido por Simon Hackett e John Romkey, após um desafio lançado por Dan Lynch, este último presidente da INTEROP (feira anual de tecnologia da informação organizada pela empresa britânica UBM) que citava, se eles conseguissem desenvolver uma torradeira que pudesse ser ligada através da Internet, o aparelho seria colocado em exposição durante a INTEROP 1990. Motivados pelo desafio, Hackett e Romkey desenvolveram uma torradeira conectada a um computador com rede TCP/IP que acabou por ser o grande sucesso do evento. No entanto ainda faltava desenvolver um dispositivo que colocasse o pão na torradeira. Essa dificuldade foi superada um ano depois, acrescentando um pequeno guindaste robótico ao protótipo. Esse guindaste, controlado pela Internet, pegava na fatia de pão, que metia dentro da torradeira, tornando assim o sistema totalmente automático [11].

2.2.1 O que é a IoT?

A Internet das Coisas embora definida pela Internet Society como “a extensão da conectividade de rede e capacidade de computação para objectos, dispositivos, sensores e outros artefactos que normalmente não são considerados computadores”, segundo o Centro Nacional de CiberSegurança (CNCS), autoridade nacional especialista em matéria de cibersegurança junto das entidades do Estado, a IoT compreende todos os aparelhos e objectos que se encontram habilitados a estarem permanentemente ligados à Internet, sendo capazes de se identificar na rede e de comunicar entre si, isto é, se quando pensamos na Internet, pensamos em computadores ou em terminais móveis para nos ligarmos entre nós e à Internet, quando nos referimos à IoT estamos a referir-nos a todo o tipo de objectos e aparelhos com capacidade computacional e de ligação à web. Contudo, todos esses objectos, com ou sem o envolvimento activo do ser humano, tanto podem fornecer dados relacionados com o seu funcionamento e o mundo que os rodeia, como receber e armazenar informações, criando, deste modo, condições para a génese de um mundo inteligente, em

que tudo está em comunicação com tudo e com todos [1].



Figura 2.5: Internet das Coisas - IoT [1].

2.2.2 Impacto da IoT na Industria

A IoT está associada à Indústria 4.0, também conhecida por Industry IoT (IIoT), cujo conceito vai muito para além da mera implantação de sistemas electrónicos e de TIC (Tecnologia da Informação e Comunicação) nos processos de produção nas fábricas, que caracterizou a chamada Indústria 3.0. Transpondo este conceito para a própria linha de montagem de uma fábrica, o Centro Nacional de CiberSegurança (CNCS) refere que esta nova vaga tecnológica viabiliza uma grande interacção entre os diversos dispositivos instalados ao longo da cadeia de produção no ‘chão de fábrica’, incluindo a cadeia logística, proporcionando que os processos de fabrico resultem de uma comunhão entre o mundo físico e o virtual. Quer os equipamentos nas linhas de produção, quer os produtos que estão a ser fabricados, quer os centros logísticos são capazes de interagir autonomamente, mais uma vez com o objectivo de melhorar o processo produtivo e assim fabricar produtos de maior qualidade, mais alinhados com os requisitos do cliente e com uma melhorada eficiência em toda a cadeia de valor. Essencialmente, a tecnologia digital em que se baseia

a Indústria 4.0 contempla a simbiose da informação digital proveniente de várias fontes e assente em diferentes protocolos de comunicação, como por RFID, Wireless Sensor Networks, Bluetooth ou outros. Os responsáveis do CNCS, organismo que funciona no âmbito do Gabinete Nacional de Segurança (GNS), salientam que esta união dos sistemas TIC com as OT (Operational Technologies), é caracterizada por uma forte interacção digital-físico-digital, envolvendo um conjunto de tecnologias que vão muito para além do IoT, como é o caso da análise massiva de dados (Big Data Analytics), impressão 3D, robótica e inteligência artificial, entre outros, e que completam o ciclo que digitaliza todo o processo produtivo e logístico [12].

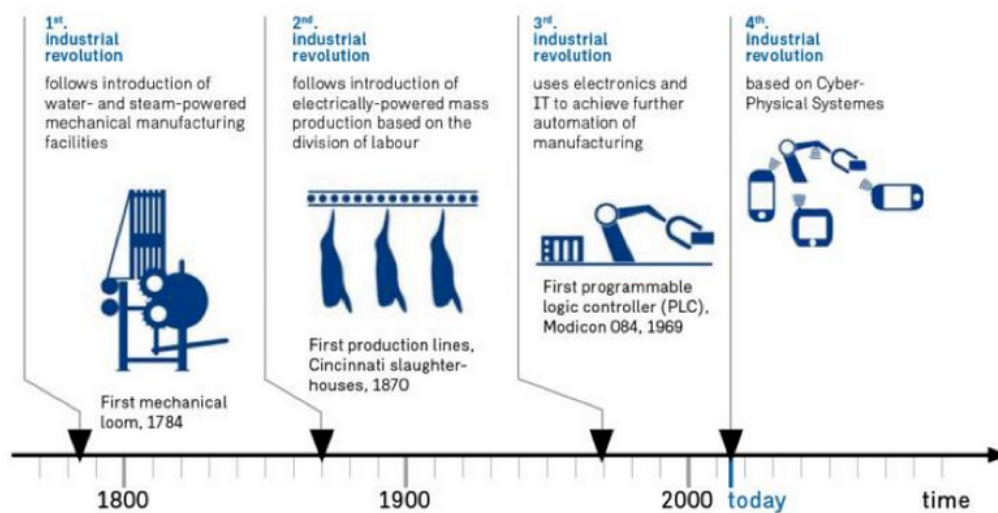


Figura 2.6: Evolução Industrial [12].

2.3 Protocolo de Comunicação

A Internet das Coisas (IoT) é um campo de pesquisa que está em grande crescimento. Basicamente, no mundo real, as coisas com capacidade de sensor, fonte de alimentação e conectividade suficientes à Internet tornam possíveis campos como a Internet das Coisas (IoT). Para esta tecnologia, é necessário ter um protocolo de largura de banda muito leve, barato e mínimo, como o Protocolo MQTT (Message Queuing Telemetry Transport).

Caso a largura de banda da rede seja baixa ou uma rede tenha alta latência, e para dispositivos com recursos e memória limitados de processamento, o MQTT poderá distribuir informações de telemetria usando um padrão de comunicação de publicação/assinatura, ou seja, é fácil para os clientes publicar ou assinar o tópico desejado por meio do host que atua como servidor da rede, também conhecido por ser o intermediário [13].

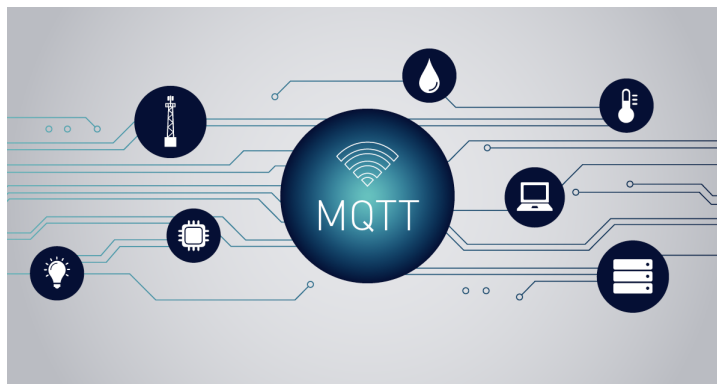


Figura 2.7: MQTT [14].

2.4 Automação e Robótica Industrial

A divisão de Automação e Robótica está capacitada para responder às inúmeras solicitações existentes na área de Automação e Controlo. Desenvolvendo parcerias estratégicas nesta área da Automação Industrial e Robótica, com Centros de Investigação e Desenvolvimento, esta divisão dispõe actualmente de saber como fazer para responder com eficácia aos desafios propostos pelo mercado [15].

2.4.1 A Automação

A automação e a robótica industrial são o processo em que uma máquina ou sistema são capazes de realizar uma actividade sozinhos, isto é, sem a necessidade de que um ser humano esteja a supervisionar o seu funcionamento correto. Isto é especialmente útil para os automatismos industriais, onde haverá mais aplicações [15].



Figura 2.8: Automatização e Robótica Industrial [16].

2.4.2 A Robótica Industrial

Quando se fala de robótica industrial, a referência é ao design, estudo e utilização dos robôs com a finalidade de que executem processos industriais de qualquer tipo. Deste modo, a normativa o define como um manipulador que pode ser programado em três ou mais eixos com múltiplos propósitos, controlado de forma automática e reprogramável.

Actualmente, as fábricas e as indústrias de grande magnitude costumam ter um grande interesse pela robótica. O que é lógico, especialmente se pensamos que este tipo de robôs podem realizar processos repetitivos e tediosos que devem ser realizados para completar a cadeia de montagem. De facto, se não fosse pelos robôs, este tipo de trabalho deveria ser completado por trabalhadores humanos, que podem apresentar mais deficiências na hora de realizar tarefas de carácter repetitivo. Deste modo, graças à automatização de alguns processos, existem fábricas muito mais eficientes, onde a equipa humana se encarrega de trabalhos mais complexas, como a programação dos robôs e a supervisão dos processos produtivos [15].

2.4.3 Tipos de Automação e Robótica Industrial

É importante levar em conta que os tipos de automação que se pode incorporar ao sector industrial são variadas. Deste modo, de acordo com as necessidades e demandas de produção, é possível mencionar três diferentes modelos de automação [15].

Automação Fixa

A automação fixa envolve um robô que repete de maneira consecutiva o mesmo processo automático. Com isto, pode parecer pouco prático, já que a sua acção parece muito limitada. Entretanto, pode trazer benefícios ao tratar de produções em grande escala. De facto, com uma única função automatizada, estes robôs são mais caros de projectar e construir, pois, como explicado anteriormente, é um tipo de automação extremamente eficiente para processos repetitivos e de produção elevada.

Automação Programável

A automação programável está entre a automação fixa e a automação flexível. Como seu nome indica, caracteriza-se porque pode ser programada. Em outras palavras, mesmo que conte com um mesmo tipo de robô que utiliza um software específico e desenhado para tal finalidade, é possível reprogramar a acção automatizada, possibilitando a sua utilidade em diferentes tarefas além da programação inicial. É uma automação especialmente útil no caso dos processos de produção em que o produto apresente variantes.

Automação Flexível

A automação flexível é aquela em que a parte robótica está constituída por uma série de estações que, mesmo que conectadas entre si, são independentes ao mesmo tempo. Isso é possível através de um computador encarregado de controlar os processos de automação e cada um dos robôs em tempo real, para que realizem as diferentes tarefas automatizadas de forma conjunta e organizada.

2.4.4 Vantagens da Automação e Robótica Industrial

A automação e robótica industrial, apresentam algumas vantagens que nos motivam à sua utilização. Como tal, são processos mais eficientes, onde a redução dos tempos de produção, dos custos e de perdas/desperdícios são notórias. Por sua vez, originam um aumento da qualidade do produto final, tendo uma maior flexibilidade na produção e uma maior eficiência energética [15].

2.4.5 Controlo de Iluminação

Um sistema de controlo de iluminação consiste num equipamento, tipicamente constituído de um processador embutido ou computador industrial, que controla luzes eléctricas em construções ou residências.

Sistemas de controlo de iluminação usualmente incluem um ou mais painéis de controlo (keypads) ou interfaces de toque (touchscreen). Estas interfaces permitem ao usuário ligar e desligar as luzes bem como controlar e programar diferentes níveis de iluminação.

A maior vantagem de um sistema de controlo de iluminação em relação a um sistema eléctrico convencional é a habilidade de controlar qualquer luz de qualquer interface e ainda poder controlar cenários de iluminação. Por exemplo, um painel touchscreen pode permitir ao usuário controlar qualquer luz do projecto, não apenas do compartimento em que se encontra. Na realidade, qualquer luz pode ser controlada de qualquer localização. Com os cenários de iluminação é possível controlar várias luzes com um mesmo accionamento [17].

2.4.6 Importância da Iluminação em Sistemas Robóticas

Nas aplicações de visão, os sistemas de iluminação ganham uma importância especial e que, muitas vezes, pode ser subestimada. A iluminação pode ser considerada a parte mais crítica dentro de um sistema de visão. As câmaras, por enquanto, são muito menos sensíveis e versáteis que a visão humana, e as condições de iluminação devem ser optimizadas

ao máximo para que uma câmara possa capturar uma imagem a qual o olho humano poderia distinguir sem necessidade de uma iluminação tão especializada. Isto torna-se muito mais evidente quando o objecto a iluminar apresenta formas complexas, ou superfícies muito reflexivos.

As câmaras de visão artificial capturam a luz reflectida dos objectos. O propósito da iluminação utilizada nas aplicações de visão é controlar a forma em que a câmara verá o objecto. A luz é reflectida de forma distinta ao iluminar, por exemplo, uma bola de aço ou uma folha de papel branco. Deste modo, os sistemas de iluminação devem ser adaptados e ajustados ao objecto em questão que será iluminado [15].

2.4.7 A Visão Robótica

Os sensores fotoeléctricos de luz são uma forma de visão para a robótica e têm diferentes formas. Estas são classificadas como, fotoresistências, fotodíodos e fototransístores [18].

Estes sensores mudam a resistência, o díodo, ou o transistor conforme detecta luz, isto é, quando um feixe de luz é detectado eles respondem seja para criar ou para trocar um sinal eléctrico que será analisado e o dispositivo tomará uma decisão. Com o uso de filtros um sensor de luz pode criar respostas selectivas com as quais o robô unicamente poderá ver determinadas cores. O uso de sensores de luz permite que os robôs se comuniquem entre si.

Para sistemas mais complexos os sensores de luz não são suficientes. Com isto, os humanos não usam somente a visão, mas sobretudo o cérebro, na tomada de decisão. Nestes casos os robôs necessitam do auxílio do computador para fazer a selecção com base em informações que os sensores de luz produzem. Para isto o computador muitas vezes tem que usar técnicas de Inteligência Artificial que simulam o funcionamento do nosso cérebro na tomada de decisões. O computador processa e envia uma informação de volta para o dispositivo robótico com uma ordem de aceitar ou rejeitar o produto.

Os sensores de infra-vermelho são usados para comportamentos simples dos robô, como por exemplo, evitar obstáculos ou mesmo para os robôs se deslocarem. A visão

robótica é ainda muito imperfeita e portanto, um dos grandes desafios para a engenharia de hoje em dia. Para poder gerar imagens tridimensionais a partir de duas imagens muito semelhantes num tempo curto requer-se uma grande quantidade de memória e de um bom processador. É difícil programar um robô para que ele saiba o que deve ou não ignorar das imagens que ele “vê”. Os robôs têm problemas para interpretar sombras, trocas de luzes, e brilho. Além disso, para poder ter percepção da profundidade é necessário que tenham visão estereoscópica. Outro dos grandes inconvenientes é ter que interpretar imagens tridimensionais [18].

2.4.8 Aplicações do Uso de Controlo Dinâmico de Iluminação em Sistemas Robóticos

A KEYENCE aposta no Sistema de Visão, com o propósito de inspeções visuais superficiais. Recentemente a empresa lançou a tecnologia multi-espectro, que conta com uma iluminação em diversas faixas do espectro de luz, para que se tenha o melhor contraste e estabilidade da aplicação. Desta forma, o método de captura é um diferencial no mercado, e que se torna interessante para os clientes para que tenham robustez, além de uma inspeção livre da orientação da interpretação subjectiva. A inspeção automática não está sujeita ao acompanhamento humano, trazendo maior repetibilidade à etapa do processo e velocidade também. A redução de manutenção também é um factor importante, uma vez que a evolução da tecnologia fez com que os instrumentos de inspeção se tornassem mais robustos às intempéries do ambiente industrial, logo a produtividade da linha é preservada e melhorada [19].

A HONEYWELL utiliza várias tecnologias nos seus sensores para medições de diversas grandezas, tais como pressão, humidade, temperatura, posição, etc. Dentre elas, a MEMS (Microelectromechanical) e a ASIC (Application Specific Integrated Circuits) com o objectivo de obter dados de forma precisa e confiável. O sensor Smart Position (Superior Measurement. Accurate. Reliable. Thinking.), por exemplo, é um ASIC com uma confiabilidade por agregar diversos circuitos integrados para detecção de campos

magnéticos (sensores de efeito-hall) para a medição precisa da posição de um objecto [19].

Já a tecnologia de sensores inteligentes da ROCKWELL AUTOMATION é habilitada através de três componentes, a sensores com diagnósticos avançados (IODD Advanced), a tecnologia de conexão IO-Link, para envio de mais informações aos sistemas de controlo e a integração avançada que existe entre os sensores e controladores. Os sensores inteligentes têm a capacidade de reduzir os custos de manutenção por permitirem a troca automática de sensores, reduzindo-se assim o MTTR (Mean Time to Repair) da máquina/linha, e são capazes de enviar informações contextualizadas sobre a situação real que acontece no campo, permitindo o usuário final a tomar decisões de forma mais rápida e eficiente [19].

Capítulo 3

Arquitectura do Sistema

Neste capítulo é realizada uma descrição detalhada de toda a arquitectura do problema em questão. É apresentada uma visão geral de todo o trabalho prático desenvolvido, referindo onde se insere cada componente, as suas especificações, bem como todo o sistema de ligação. Contudo, é pretendido que nesta arquitectura haja um controlo luminoso adaptado ao meio envolvente, isto é, a intensidade de luz do LED vai depender consoante a necessidade de luminosidade que o sensor absorve de modo a atingir os valores ideais do set-point do controlo PID.

3.1 Visão Geral

Esta visão geral do sistema de controlo de luminosidade está dividida essencialmente em três pontos de comunicação, nas quais todas por Wi-Fi pelo MQTT. A montagem desta arquitectura foi realizada na plataforma digital Lucidchart e está referenciada na figura 3.1.

A comunicação entre os controladores (ESP32, Arduino e Smart Phone), está representada por uma linha azul tracejada, sendo esta realizada por Wi-Fi pelo protocolo de comunicação MQTT. Posteriormente, as linhas de ligação a preto, "ligam" no caso do Arduino, o controlador ao actuador (LED) e no caso do ESP32, o controlador ao sensor (TSL2561).

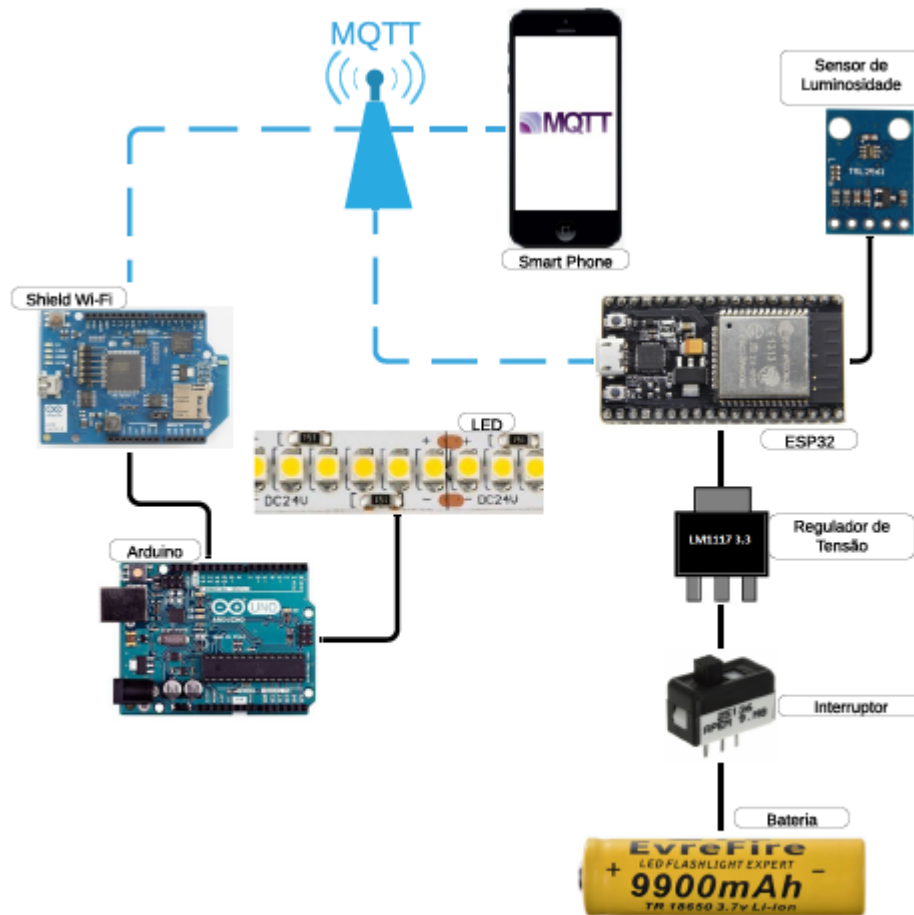


Figura 3.1: Arquitectura do Sistema de Controlo Luminoso.

3.2 Sistema de Controlo Luminoso em Ambiente Laboratorial

O Sistema de Controlo de Luminosidade, é controlado remotamente por Wi-Fi através do protocolo MQTT. Na 3.2, está representado o controlo luminoso em ambiente laboratorial.

Posto isto, o ESP32 lê os valores de intensidade luminosa que o sensor absorve, em lux, através do sensor TSL2561. Este microcontrolador é alimentado por uma bateria de 3,7V, que quando recarregada atinge um pico superior a 4V. Para tal, foi necessário o uso de um regulador de tensão LM1117-3.3, com a finalidade de o microcontrolador ser

alimentado a 3,3V. Contudo para que a bateria não tenha consumos desnecessários, foi implementado um interruptor.

O arduino para fazer a comunicação com o MQTT necessita obrigatoriamente de uma shield Wi-Fi. O controlo da luminosidade realizada por este microcontrolador, trata de regular uma luz LED de forma a obter uma luminosidade ideal. Este LED é alimentado por uma fonte de alimentação a 24V.

Posteriormente, todo este sistema é regulado através da aplicação móvel produzida em Node-RED.

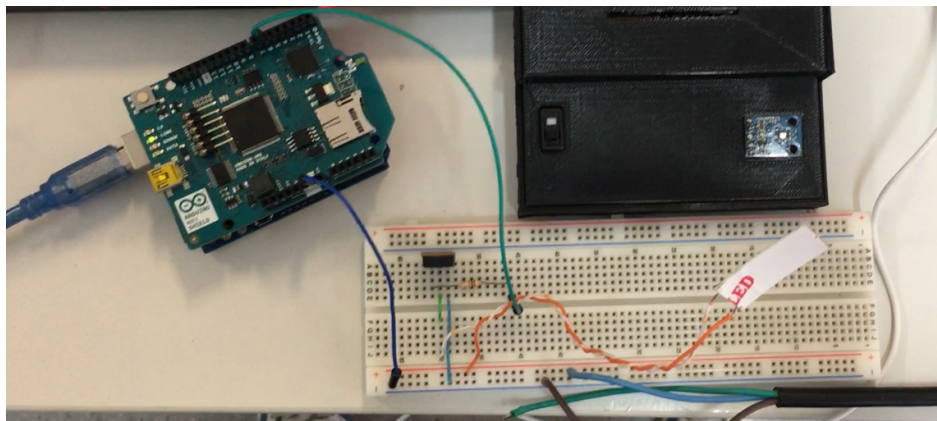


Figura 3.2: Sistema de Controlo Luminoso em Ambiente Laboratorial .

3.3 MQTT

O MQTT - Message Queuing Telemetry Transport, é um protocolo de mensagens simples, projectado para dispositivos restritos com baixa largura de banda. Portanto, é a solução perfeita para aplicativos da Internet das Coisas. O MQTT permite enviar comandos para controlar saídas, ler e publicar dados de sensores, o que torna muito fácil estabelecer uma comunicação entre vários dispositivos [14]. Na figura 3.3 é referenciada a arquitectura deste protocolo.

No protocolo MQTT os servidores e clientes podem enviar mensagens uns para o outros. Na tabela 3.1 é referenciado os meios de comunicação entre os servidores e os clientes.

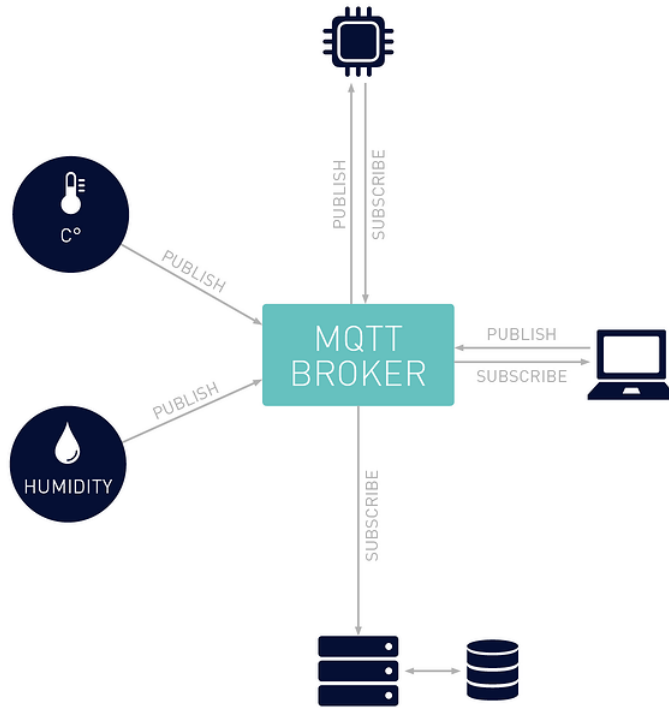


Figura 3.3: Arquitectura do MQTT [14].

Tabela 3.1: Meios de comunicação no MQTT [14].

Tipos de Comunicação	Descrição
Topics	Esta é uma maneira de categorizar o tipo de mensagens que podem ser enviadas. Por exemplo: se um sensor mede a luminosidade, o tópico pode ser definido como "LUZ" e o sensor envia mensagens rotuladas como "LUZ".
Publishers	Os dispositivos podem ser configurados para enviar mensagens contendo dados.
Subscribers	Dispositivos ou sistemas podem ser configurados para receber apenas mensagens de um tópico específico.
Broker	Este é o servidor que transmite as mensagens publicadas para servidores ou clientes que se inscreveram em tópicos específicos.

3.4 Arduino

Neste sistema de arquitectura, o arduino é alimentado pelo computador através do conector USB, cuja tensão é 5V. Este para comunicar via MQTT, necessita de uma Shield Wi-Fi. Depois desta comunicação com o MQTT, o arduino vai receber os valores lidos pelo sensor de luminosidade TSL2561 que esta alimentado pelo microcontrolador ESP32. Após receber os valores, é aqui que o arduino faz o PWM manobrando o set-point do controlo PID. Este set-point, são os valores que o sensor absorve, fazendo assim com que a intensidade luminosa do LED aumente ou diminua consoante a necessidade.

o Arduino é uma plataforma de prototipagem electrónica, criado por Massimo Banzi e David Cuartielles, em 2005, com objectivo de permitir o desenvolvimento de controlo de sistemas interactivos, de baixo custo e acessível. Todo o seu material, software, bibliotecas e hardware, é open-source, ou seja, qualquer indivíduo o pode usar sem necessidade de pagamentos de direitos de autor. A linguagem de programação do Arduino é uma implementação do Wiring, uma plataforma computacional física semelhante, que é baseada no ambiente de programação Processing, feito em Java. A plataforma do arduino é composta essencialmente de duas partes, o Hardware e o Software [20].

3.4.1 Hardware do Arduino Uno

O Hardware é uma placa electrónica que possui todos os componentes necessários para a maioria dos projectos académicos. A electrónica deste permite usar a placa com diversas fontes de energia, como por exemplo, baterias e fontes de alimentação. Por sua vez, o Hardware, permite o acoplamento de circuitos externos através de pinos de conexão em posições padronizadas, as Shields. Este arduino esta referenciado na figura 3.4 [21].

3.4.2 Especificações da Placa Arduino UNO

As especificações da Placa do Arduino Uno estão referenciadas na tabela 3.2.

A memória flash é o local onde o programa é guardado, já a SRAM é a memória na qual as variáveis são guardadas. Caso o arduino seja desligado ou reiniciado, a memória

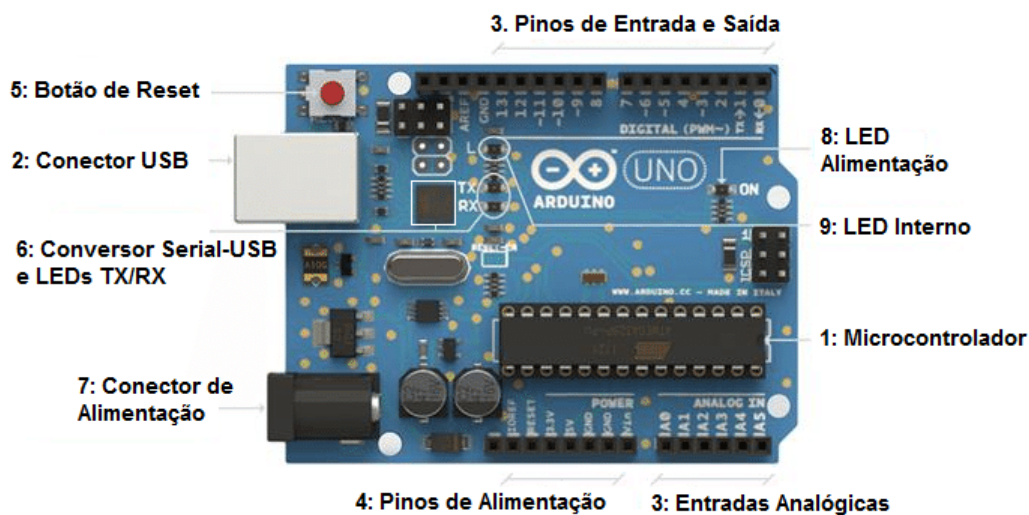


Figura 3.4: Arduino UNO Rev3 [21].

Tabela 3.2: Especificações da Placa Arduino UNO [21].

Especificações	Valores
Microcontrolador	ATmega328
Memória flash	32 kB
Memória SRAM	2kB

flash não perde os seus dados, contrariamente a memória SRAM quando desligada ou reiniciada perde os seus dados [21].

3.5 Shield Wi-Fi

O Arduino Wi-Fi Shield permite que uma placa Arduino se conecte a uma rede sem fios. Neste caso, o arduino para controlar a intensidade luminosa, que é realizada através do set-point do controlo PID, terá que estar conectado pelo MQTT.

3.6 LED

O LED-Light Emitting Diode, é um Díodo Emissor de Luz. Este é um componente electrónico semiconductor, composto de cristal semiconductor de silício ou germânio e possui a capacidade de transformar energia eléctrica em energia luminosa, este fenómeno é denominado por electroluminescência. Posto isto, O LED é um dispositivo capaz de emitir luz de forma eficiente e económica, daí a ser uma alternativa interessante para a iluminação pública e residencial. Também possui grande vida útil, podendo ultrapassar facilmente as 100 mil horas de uso [22].

A intensidade do LED depende do meio envolvente em que se encontra, isto é, da luminosidade absorvida pelo sensor. Este actua consoante a necessidade do consumidor em obter luz, ou seja, se a necessidade for o aumento/diminuição de luminosidade, o set-point do controlo PID vai ser alterado obtendo então a sua luminosidade ideal, a intensidade de Luz projectada pelo LED.

3.7 Smart Phone

O Smart Phone é o dispositivo tecnológico mais usado pelo ser humano. Como tal, para controlar a luminosidade absorvida pelo sensor, foi criada uma Aplicação em Node-RED. Nesta, podemos manobrar o set-point do controlo PID do arduino pelo protocolo MQTT. Para além do controlo, também é possível observar o comportamento da luminosidade absorvida graficamente.

3.8 ESP32

O ESP32 é utilizado para alimentar um sensor de luminosidade TSL2561. O objectivo pretendido, seria enviar os valores absorvidos pelo sensor, por Wi-Fi pelo MQTT. Depois de realizado o PWM pelo arduino, este comunica com o ESP-WROOM-32, manobrando assim o sensor através do set-point do controlo PID realizado no arduino, ou seja, permite

manobrar a luminosidade pretendida. Este, é alimentado por uma bateria a 3,3V.

A placa do ESP32 contém um conversor USB serial integrado e uma porta micro USB para alimentação e programação. Este microcontrolador, para um bom funcionamento, actua com uma tensão entre 2,2V a 3,6V em nível lógico e só pode ser alimentado a 5V através do conector micro USB. A sua corrente típica é de 80mA, sendo que a corrente máxima por pino é de 12mA, o recomendado é 6mA. Este possui um módulo ESP-WROOM-32. Esta placa está referenciada na figura 3.5 [23].

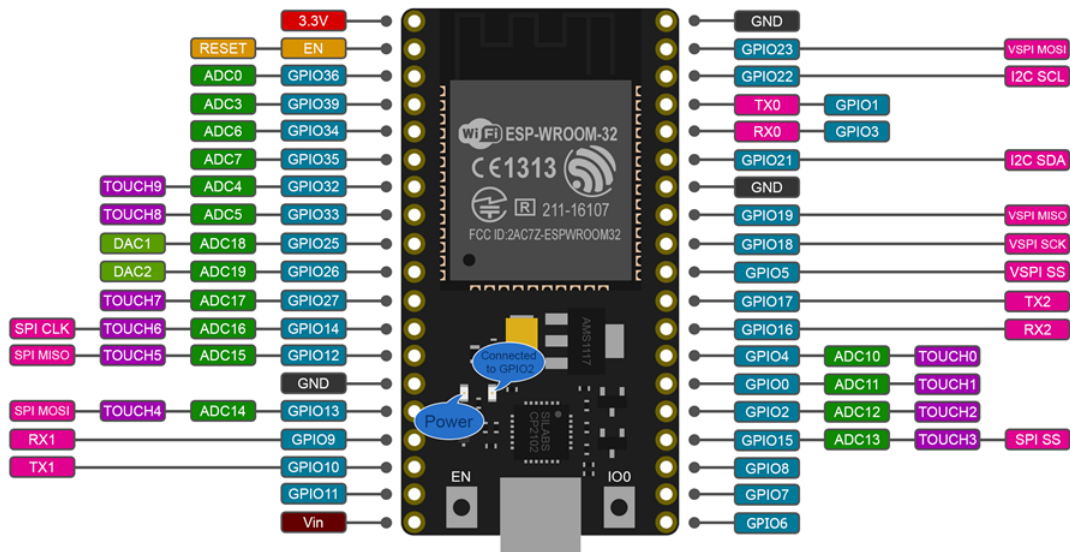


Figura 3.5: ESP32 PINOUT [23].

3.8.1 ESP-WROOM-32

O ESP-WROOM-32 é um dispositivo IoT que consiste de um microprocessador de baixa potência, dual core Tensilica Xtensa 32-bit LX6, com suporte embutido à rede Wi-Fi, Bluetooth e memória flash integrada. Esta arquitectura permite que ele possa ser programado de forma independente, sem a necessidade de outras placas microcontroladoras como, por exemplo, o Arduino [24].

CPU e Memória Interna

O ESP-WROOM-32 contém dois microprocessadores (Dual Core) Xtensa de 32 bits LX6 de baixa potência [24].

A memória interna inclui:

- 448 kB de ROM para inicialização e funções principais;
- 520 kB de SRAM no chip para dados e instruções;
- 8kB de SRAM no RTC, chamado Memória RTC SLOW e podem ser acedidos pelo co-processador durante o modo Deep-sleep;
- 8kB de SRAM no RTC, denominado RTC FAST Memory e podem ser usados para armazenamento de dados; é acedido pela CPU principal durante a inicialização do RTC a partir do modo de suspensão profunda.

Flash SPI Externo

O ESP-WROOM-32 integra 4 MB de flash SPI externo. O flash SPI de 4 MB pode ser mapeado na memória para o espaço de código da CPU, suportando acesso de 8, 16 e 32 bits. A execução de código é suportada. O flash SPI integrado está conectado ao GPIO6, GPIO7, GPIO8, GPIO9, GPIO10 e GPIO11. Esses seis pinos não podem ser usados como GPIO comum [24].

3.9 Sensor de Luminosidade

O TSL2561 é programado pelo ESP32. Os valores absorvidos por este sensor são enviados pelo microcontrolador para o MQTT. Posteriormente, o set-point do controlo PID realizado pelo arduino, é precisamente o valor da luz digital, em lux, que o sensor pretende absorver. A luz absorvida pode ser controlada através da aplicação MQTT-Spy, bem como do Smart Phone através da aplicação andróide realizada no Node-RED.

O sensor de luminosidade TSL2561 é um sensor de luz digital avançado, ideal para uso em uma ampla gama de situações de luz. Este, contém díodos de espectro infravermelho, isto significa que é possível medir separadamente a luz infravermelha, de um espectro total ou visível ao ser humano.

3.9.1 Especificações

O sensor TSL2561 tem uma resposta aproximada ao olho humano, este mede com precisão a luminosidade. As suas especificações estão referenciadas na tabela 3.3.

Tabela 3.3: Especificações do Sensor de Luminosidade TSL2561 [25].

Especificações	Valores
Temperatura mínima	-30 °C
Temperatura máxima	80 °C
Intensidade de luz mínima absorvida (Lux)	0,1 Lux
Intensidade de luz máxima absorvida (Lux)	40000 Lux
Tensão mínima	2.7 V
Tensão máxima	3.6 V
Interface	I2C

3.10 Bateria

A bateria é utilizada para alimentar o microcontrolador ESP32. Esta é recarregável e quando esta operação é feita, atinge um pico de sensivelmente 4,2 V. Esta tem uma capacidade de 9900 mAh.

O interruptor utilizado no circuito, tem um tipo de funcionamento ON/OFF, este vai abrir e fechar o circuito eléctrico, ou seja, se estiver fechado, a bateria consegue alimentar o controlador, por outro lado, se estiver aberto, a bateria não consegue alimentar o controlador. Este componente foi inserido no sistema para que a bateria não tivesse gastos desnecessários.

O interruptor quando se encontra na posição 2, não alimenta o ESP32, ou seja, abre

o circuito eléctrico. Contrariamente, quando se encontra na posição 1, alimenta o controlador, estando assim fechado o circuito eléctrico. Esta posição e conexão do interruptor esta referenciada na figura 3.6.

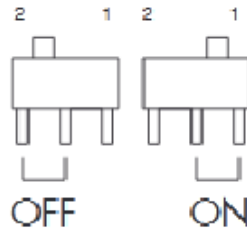


Figura 3.6: Posição e Conexão do Interruptor.

3.11 Regulador de Tensão LM1117-3.3

O regulador de tensão utilizado neste circuito é o LM1117. A bateria, para além de ser de 3,7V, quando recarregada atinge sensivelmente um pico de 4,2V. Como o ESP32 tem que ser alimentado a uma tensão entre 2,2V a 3,6V em nível lógico, este regulador actua entre a bateria e o controlador, tendo uma tensão de saída de 3,3V.

3.11.1 Diagrama Funcional

O diagrama funcional do regulador de tensão LM1117, está referenciado na figura 3.7. O LM1117, permite uma regulação estável para a tensão desejada 3,3V evitando grande quedas de tensão. Por sua vez, permite ainda um limite de corrente e um controlo térmico. Para uma melhor estabilidade das tensões de entrada V_{in} e saída V_{out} , devem ser usados dois condensadores de acoplamento. Estes devem ser colocados no layout o mais próximo possível dos pinos de entrada e de saída.

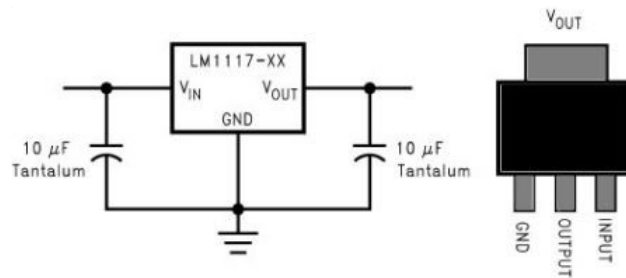


Figura 3.7: Diagrama Funcional LM1117-3.3 [26].

3.11.2 Especificações Técnicas

Este regulador de tensão tem algumas especificações técnicas que estão referenciadas na tabela 3.4.

Tabela 3.4: Especificações Técnicas do LM1117 [27].

Especificações	Valores
Precisão da tensão de saída	1%
Número de saídas	1
Tipo de saída	Fixa
Polaridade	Positiva
Tensão de Dropout	1.2 V
Tensão máxima de entrada	15 V
Tensão máxima de saída	3.3 V
Tensão máxima de alimentação	10 V
Tensão mínima de entrada	4.5 V
Tensão de alimentação mínima	4.5 V
Tensão nominal de saída	3.3 V
Tensão de saída	3.3 V
Corrente de saída máxima	800 mA
Corrente de saída	800 mA
Corrente de repouso	5 mA
Temperatura operacional máxima	125 °C
Temperatura operacional mínima	0 °C

Capítulo 4

Desenvolvimento

Este capítulo relata todo o trabalho prático que foi desenvolvido ao longo deste projecto, bem como todas as ferramentas utilizadas para a execução dos mesmos. Com base na figura 3.1, é demonstrado e explicado todo o procedimento desenvolvido em software.

Posteriormente, para que este controlo fosse realizado à distância, surgiu a necessidade de criação de uma aplicação móvel em Node-RED com a mesma finalidade, manobrar o set-point do controlo PID. Esta é uma ferramenta de programação baseada em fluxo.

Contudo, depois de concluído todo este processo de código, o circuito eléctrico do ESP32 é transposto para uma placa de circuito impresso, em que o seu esquemático e o seu layout foram desenvolvidos no EAGLE. Para que esteticamente a PCI fosse melhorada, foi produzida uma peça no SOLIDWORKS, trata-se de uma impressão CAD 3D que teve um papel fundamental na sua apresentação.

4.1 Programação de Software

Para programar os microcontroladores, ESP32 e arduino, é necessário um software. Ambos são programáveis no software do arduino em linguagem C++. A linguagem de programação do arduino pode ser dividida em três partes principais, isto é, estruturas, valores (variáveis e constantes) e funções.

O ponto de partida inicial seria programar o ESP32 com a finalidade de ler os valores

de intensidade luminosa que o sensor absorvia, em lux. Na figura 4.1 é demonstrada a configuração do sensor TSL2561.

```
//sensor Tsl2561
void displaySensorDetails(void)
{
  sensor_t sensor;
  tsl.getSensor(&sensor);
  Serial.println("-----");
  Serial.print ("Sensor:      "); Serial.println(sensor.name);
  Serial.print ("Driver Ver:   "); Serial.println(sensor.version);
  Serial.print ("Unique ID:    "); Serial.println(sensor.sensor_id);
  Serial.print ("Max Value:    "); Serial.print(sensor.max_value);
  Serial.println(" lux");
  Serial.print ("Min Value:    "); Serial.print(sensor.min_value);
  Serial.println(" lux");
  Serial.print ("Resolution:   "); Serial.print(sensor.resolution);
  Serial.println(" lux");
  Serial.println("-----");
  Serial.println("");
  delay(500);
}
```

Figura 4.1: Configuração do Sensor TSL2561.

De seguida, surgiu a necessidade de enviar e receber valores entre microcontroladores. Para tal, foi necessário o uso do protocolo MQTT, este permite enviar comandos para controlar saídas, ler e publicar dados de sensores, através do software MQTT-Spy. Na figura 4.2 é realizada a conexão dos microcontroladores à rede Wi-Fi.

Caso haja uma perda de ligação à rede, o sistema vai proceder a uma reconexão de uma durabilidade de cinco segundos, até voltar a subscrever os valor pretendidos 4.3.

Posto isto, o receptor desta mesma informação e que mais tarde vai fazer o controlo da luminosidade exposta no meio em que se encontra, é o arduino. Este para fazer a comunicação com o MQTT necessita obrigatoriamente de uma shield Wi-Fi. Como referido anteriormente, o controlo da luminosidade realizada por este último microcontrolador, trata de regular uma luz LED de forma a obter uma luminosidade ideal. Essa luminosidade, depende do valor do set-point do controlo PID desenvolvido no código do arduino. Inicialmente, este controlo é iniciado a zero 4.4.

```

void setup_wifi() {
  delay(10);
  // Começo de conexão à rede WiFi
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}

```

Figura 4.2: Conectar o microcontrolador à rede Wi-Fi.

```

void reconnect() {
  // Loop até à reconexão
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection...");
    // Tentativa de conexão
    if (client.connect("ESP8266Client")) {
      Serial.println("connected");
      // Subescreber
      client.subscribe("esp/output");
    } else {
      Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      // Espera de 5 segundos antes de tentar novamente
      delay(5000);
    }
  }
}

```

Figura 4.3: Reconectar o microcontrolador à rede Wi-Fi.

```

//Set point : Brilho
Setpoint = 0;
//Ligar PID
myPID.SetMode(AUTOMATIC);

```

Figura 4.4: Ligar PID.

Posto isto, são recebidos dados que são enviados pelo ESP32, dados esses da intensidade luminosa que o sensor TSL2561 absorve. Para que haja um controlo dessa luminosidade, está programado no arduino um controlo PID, este faz com que a intensidade do LED aumente ou diminua alterando o seu set-point. Este set-point é a luminosidade que o sensor pretende absorver. Para manobrar a intensidade de luz emitida pelo LED, foi utilizada uma porta digital que possibilita uma modelação através de largura de pulso (PWM) 4.5.

```
void loop() {  
  
    if (!client.connected()) {  
        reconnect();  
    }  
    client.loop();  
  
    myPID.Compute();  
    analogWrite(6, Output);  
    Serial.print("Setpoint: ");  
    Serial.println(Setpoint);  
}
```

Figura 4.5: Porta analógica de saída (OUTPUT).

A modulação por largura de pulso, ou do inglês "Pulse Width Modulation"(PWM), é uma técnica para obter resultados analógicos com meios digitais. O controlo digital é usado para criar uma onda quadrada, ou seja, um sinal alternado entre ligado e desligado. Assim, quando o valor do "duty cycle" deste sinal PWM é 100% a carga luminosa receberá a tensão máxima do sistema, ou seja os 24V. Já quando o valor do "duty cycle" é 0% nenhuma tensão será entregue ao elemento luminoso, como representado na figura 4.6. Este padrão on-off pode simular tensões entre ligado a 24V, e desligado a 0V, alterando a parte do tempo em que o sinal passa. A duração do tempo é chamada de largura do pulso. Para obter valores analógicos variados, é alterada essa largura de pulso.

Na figura 4.6, as linhas verdes representam um período de tempo regular. Esta duração ou período é o inverso da frequência PWM. O analogWrite está numa escala de 0 a 255, de modo que analogWrite a 255, tem um duty-cycle de 100%, está sempre activo, e analogWrite a 127, tem um duty-cycle de 50%, na metade do tempo.

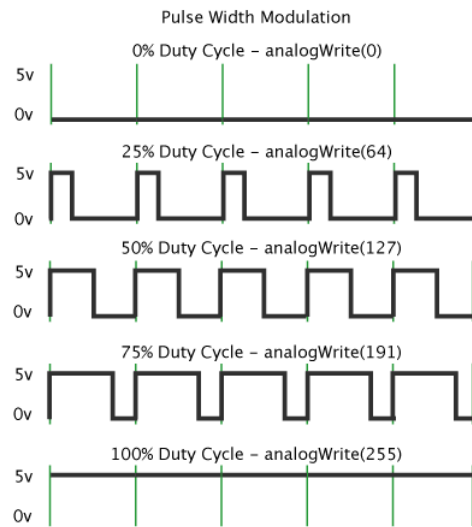


Figura 4.6: PWM [28].

4.1.1 Controlo PID

Como o próprio nome indica, o algoritmo PID é composto por três coeficientes, Proporcional-Integral-Derivativo (PID) que são variados para obter a resposta ideal. Este é o algoritmo de controlo mais usado na indústria para sistemas de controlo industrial. A popularidade de controladores PID pode ser atribuída em parte ao seu desempenho robusto de uma ampla gama de condições de funcionamento e em parte à sua simplicidade funcional, que permite ao utilizador operar de uma forma simples e directa.

Os ganhos de um controlador PID podem ser obtidos pelo método de tentativa e erro. Neste método, os termos I e D são definidos para zero e o ganho proporcional é aumentado até a saída do loop começar a oscilar. Quando se aumenta o ganho proporcional, o sistema torna-se mais rápido, mas deve-se ter em atenção para não torná-lo instável. Uma vez que P foi definido para obter uma resposta rápida desejada, o termo integral é aumentado a fim de parar as oscilações. O termo integral reduz o erro de estado estacionário, mas aumenta o overshoot. Um certo valor de overshoot é sempre necessário para um sistema rápido de modo que possa responder às mudanças imediatamente. O termo integral é novamente ajustado para atingir um mínimo erro de steady state. Uma vez que o P e I foram definidos para que o sistema de controlo seja rápido com o steady state mínimo e constante, o termo

derivativo é aumentado até que o loop seja relativamente rápido em relação ao seu ponto de referência. Aumentar o termo da derivada diminui o overshoot, aumentando o ganho, mantendo a estabilidade e ainda faz com que o sistema seja altamente sensível ao ruído [29] 4.7.

```
//Variáveis
double Setpoint, Input, Output;

//PID Parâmetros
double Kp = 0.02, Ki = 0.4, Kd = 0.002;

//Start PID instance
PID myPID(&Input, &Output, &Setpoint, Kp, Ki, Kd, DIRECT);
```

Figura 4.7: Controle PID.

De acordo com a figura 4.7, os valores de "Input", são os valores da intensidade luminosa que o sensor TSL2561 absorve, em lux. Por outro lado, os de "Output" são os valores da luminosidade que o LED emite.

4.2 Desenvolvimento de Software de Interface Gráfica do Utilizador

Para o desenvolvimento de software gráfico, é necessário um modelo de Programação Baseada em Fluxo, que se presta muito bem a uma representação visual e a torna mais acessível a uma ampla gama de usuários. Neste modelo é possível dividir um problema em etapas discretas, observar um fluxo e ter uma noção do que se está a fazer, sem precisar de entender as linhas de código individuais em cada node. Este modelo designa-se por Programação Baseada em Fluxo [30].

4.2.1 Node-RED

O Node-RED é uma ferramenta de programação baseada em fluxo, originalmente desenvolvida pela equipa de Serviços de Tecnologia Emergente da IBM e agora parte da JS

Foundation. Esta ferramenta consiste num tempo de execução baseado no Node.js. e projecta para um navegador Web o editor de fluxo. No navegador, é criado um aplicativo arrastando nodes para um espaço de trabalho e começar a conectá-los. Com um único clique, o aplicativo é implantado novamente no tempo de execução em que é executado. O Grupo de nodes pode ser facilmente estendido, instalando novos nodes criados pela comunidade e os fluxos criados podem ser facilmente compartilhados como arquivos JSON [30].

No Node-RED, os programas são fluxos compostos por uma coleção de nodes conectados para trocar mensagens por MQTT. Estas mensagens transmitidas entre nodes são, por convenção, objectos JavaScript chamados de “msg”. Numa mensagem podem ser anexadas outras propriedades pelos nodes, que pode ser usada para transportar outras informações para outro node do fluxo. Os nodes, são o principal bloco de construção dos fluxos no Node-RED, quando um fluxo esta em execução, as mensagens são geradas(input), consumidas(output) e processadas(function) por estes. Para que haja conexão entre nodes de entrada e saída, são necessários fios. Na figura 4.8 é possível observar todo o processo de montagem para este projecto.

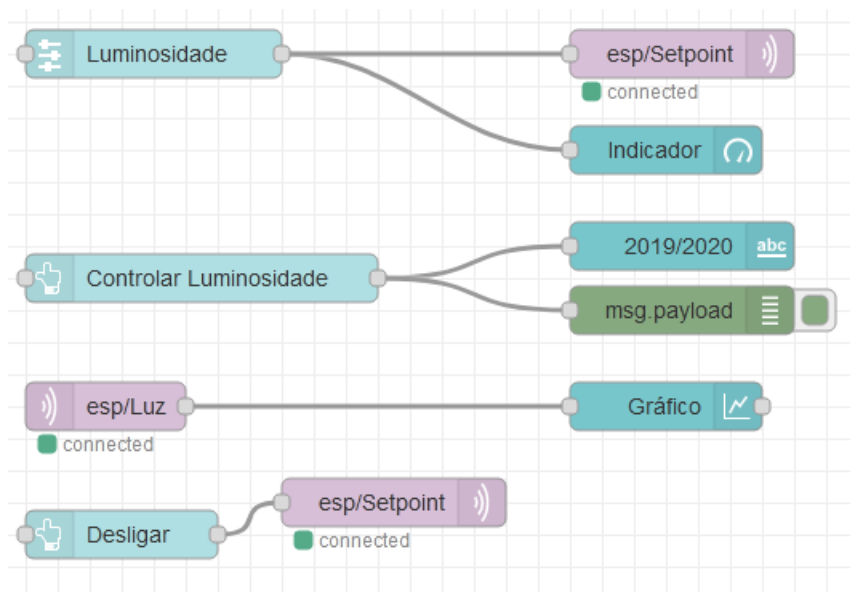


Figura 4.8: Esquema da Aplicação de Controlo de Luminosidade no Node-RED.

4.3 Design de Placa de Circuito Impresso

Para a execução da Placa de Circuito Impresso - PCI, foi necessário o uso do EAGLE. Este é um aplicativo de automação de design electrónico programável com captura esquemática, layout da PCI, programa automático e recursos de fabricação auxiliada por computador. EAGLE significa Editor de Layout Gráfico Facilmente Aplicável, tem origem alemã, (Einfach Anzuwendender Grafischer Layout-Editor) e é desenvolvido pela CadSoft Computer GmbH. A empresa foi adquirida pela Autodesk Inc. em 2016 [31].

4.3.1 EAGLE

O design de uma PCI no EAGLE é um processo de duas etapas. Primeiro, projecta-se o esquema, depois define-se uma PCI com base neste esquema. Um esquema bem projectado é fundamental para o processo geral de design de placas de circuito impresso. Este ajudará a detectar erros antes da fabricação da placa e a certificar-se quando algo não funcionar.

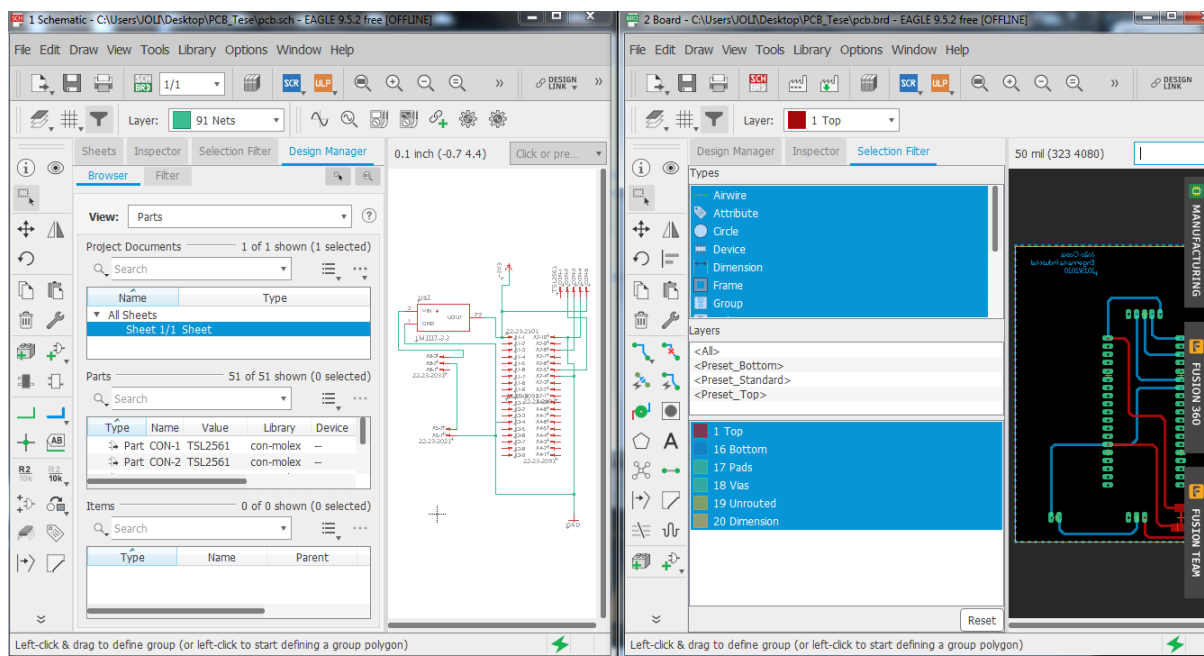


Figura 4.9: Interface de desenvolvimento de PCI no Eagle.

Esquemático da PCI

O EAGLE contém um editor esquemático, para projectar diagramas de circuitos. Esquemas são armazenados em arquivos com extensão .SCH, partes são definidas nas bibliotecas de dispositivos com extensão .LBR. As peças podem ser colocadas em várias folhas e conectadas por meio de portas. O esquema eléctrico deste projecto é referenciado na figura 4.10.

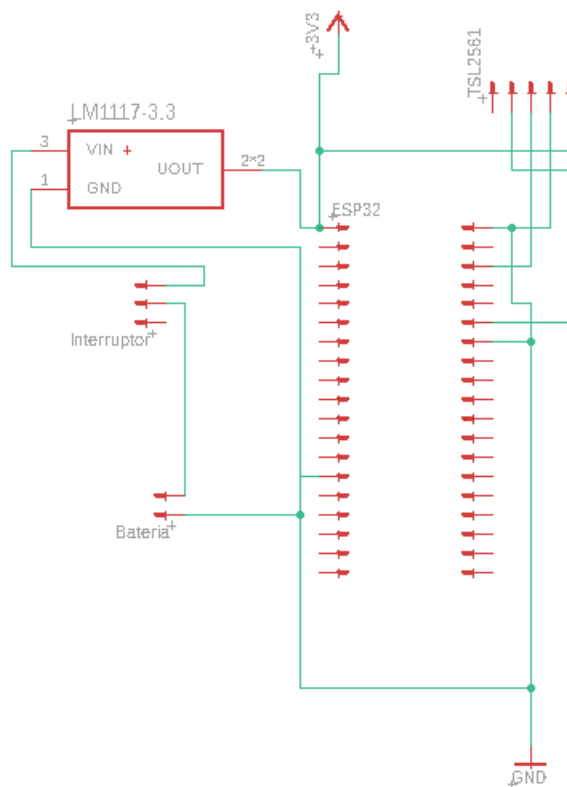


Figura 4.10: Esquemático da PCI.

Layout da PCI

No EAGLE, o editor de layout da PCB armazena os arquivos da placa com a extensão .BRD. Este permite que a anotação retroactiva no esquema e o roteamento automático conectem automaticamente os rastreamentos com base nas conexões definidas no esquema. O layout do esquema eléctrico deste projecto é referenciado na figura 4.11.

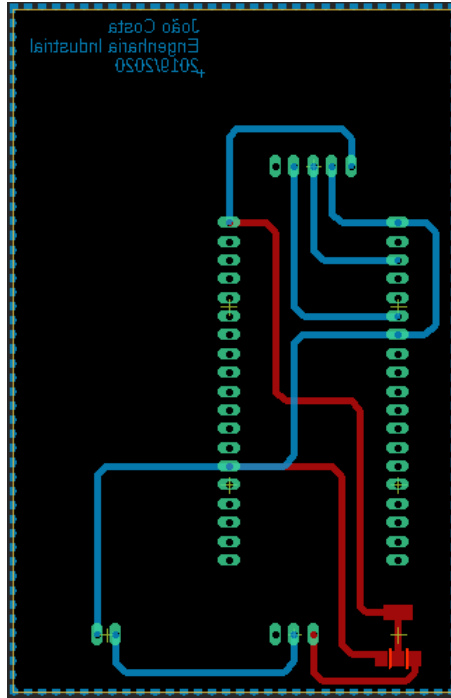


Figura 4.11: Layout da PCI.

Trata-se de uma layout de dupla face. Nesta layout da PCI, o circuito a azul é o bottom e o circuito a vermelho é o top. Para que a placa seja impressa, o EAGLE gera os arquivos de layout em ficheiros Gerber, para que com estes na CNC seja produzida a PCI.

4.4 Impressão 3D

Trazer produtos inovadores para o mercado mais rapidamente ao fornecer as melhores ferramentas totalmente integradas da classe para todas as áreas de desenvolvimento e gerir produtos, o SOLIDWORKS permite que as empresas acelerem cada estágio do ciclo de desenvolvimento, permitindo que todas as disciplinas trabalhem simultaneamente, começando com os conceitos até a fabricação. Este Melhora drasticamente a maneira como se desenvolve e fabrica produtos. Como base para todo o conjunto de soluções de desenvolvimento de produtos SOLIDWORKS, os pacotes CAD do SOLIDWORKS cobrem design, simulação, estimativa de custos, verificações de fabricação, CAM, design sustentável e

gerir dados [32].

4.4.1 Desenho do Invólucro

O desenho do invólucro foi realizado no SOLIDWORKS. Este trabalha com um tipo de tecnologia chamada CAD. O termo significa Computer Aided Design ou desenho assistido por computador, em tradução literal. Ele funciona tendo como base objectos sólidos, que podem ser modelados para a criação de uma variedade de arquivos. Este software baseia-se em computação paramétrica, criando formas tridimensionais a partir de operações geométricas elementares. No ambiente do programa, a criação de um sólido ou superfície tipicamente começa com a definição de um sketch 2D que depois é transformado através de uma operação num modelo tridimensional [33].

Para este projecto surgiu a necessidade de fabricação de uma peça CAD 3D, para tal, o software SOLIDWORKS foi essencial. Com este é possível criar esboços simulando, neste caso, movimentos para encaixe da peça, tornando mais fácil todo o seu processo de execução. O desenho desta está referenciado na figura 4.12.

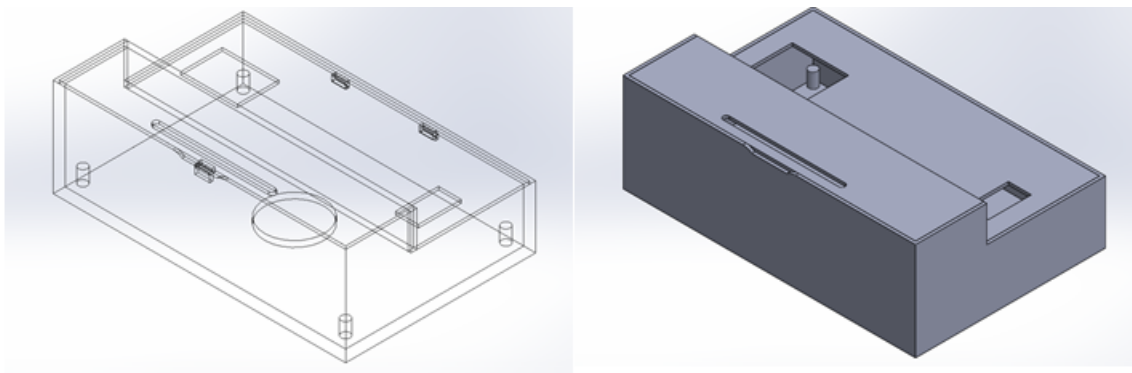


Figura 4.12: Desenho CAD 3D em SOLIDWORKS.

Capítulo 5

Resultados e Discussão

Neste capítulo são apresentados todos os resultados práticos obtidos com esta dissertação. Posteriormente, serão analisados e discutidos de modo a clarificar todos os procedimentos. De salientar, que foram cumpridos todos os objectivos assumidos no início do projecto.

5.1 Protótipo Final

Depois de o código implementado nos microcontroladores, arduino e ESP32, estar a funcionar correctamente, como referido anteriormente, foi projectado no EAGLE um esquema eléctrico de todo o circuito, com a finalidade de obter uma layout da PCI. Para que esta seja produzida, são criados ficheiros Gerber para ser impressa na CNC.

Na figura 5.1, podemos observar a PCI já com os respectivos componentes soldados à placa.

Contudo, surgiu a necessidade da criação de uma peça CAD 3D para a PCI e respectivos componentes. Esta foi esboçada no SOLIDWORKS.

Na figura 5.2, é possível observar a peça com a PCI lá dentro.

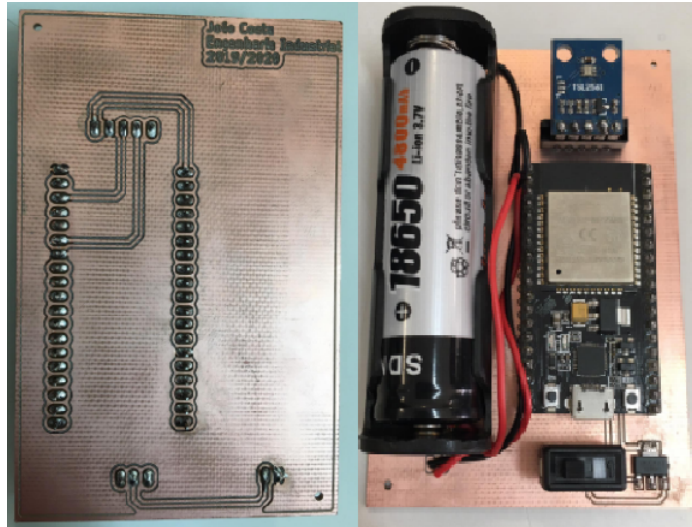


Figura 5.1: PCI Final.



Figura 5.2: Protótipo Final.

5.2 Aplicação Node-RED

O MQTT-Spy é o programa utilizado para ler os valores enviados para este protocolo. Neste caso específico, lê os valores de luminosidade, em lux, absorvidos pelo sensor TSL2561, enviados pelo ESP32. Posteriormente, se for necessário aumentar ou diminuir a intensidade luminosa, é aqui que actua o PWM do controlo PID. No MQTT-Spy é publicado um tópico "esp/Setpoint". Este tópico altera os valores do set-point do controlo PID, cujos valores, em lux, são definidos na data desta publicação. A figura 5.3 referencia este procedimento.

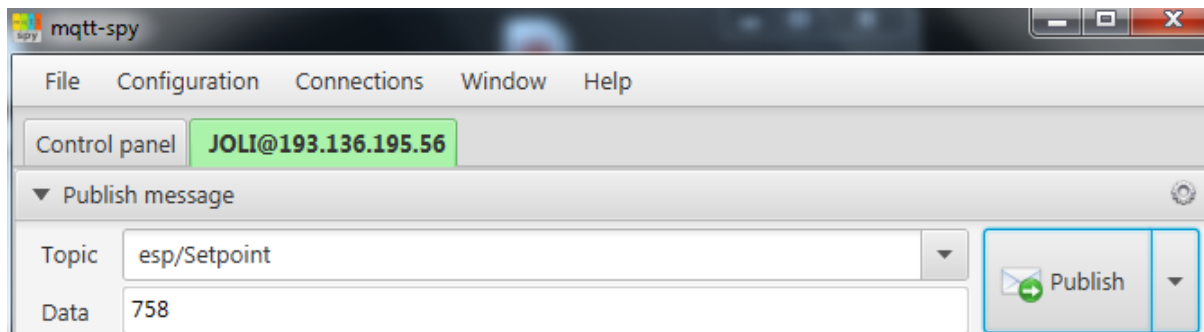


Figura 5.3: Publicar tópico por MQTT.

Posto isto, para controlar todo este sistema de luminosidade surgiu a necessidade da criação de um aplicativo de modo a que este controlo seja executado remotamente.

Na figura 5.4, é possível observar a interface do aplicativo.

5.2.1 Funcionalidade do Sistema de Controlo de Luminosidade

Para controlar este sistema luminoso, foi criada uma aplicação em Node-RED. Nesta, ao clicar no botão "Controlar Luminosidade", irá aparecer o ano lectivo e o nome do autor do projecto. Deste modo, o Sistema de Controlo de Luminosidade esta pronto a ser utilizado, como se pode observar na figura 5.5.

Todo este controlo inicia quando no slider "Luminosidade" se aplicam valores, onde estes vão aparecer em cima no indicador. Estes valores, são os pretendidos para o sensor absorver. Deste modo, a intensidade do LED vai alterar-se consoante a luminosidade

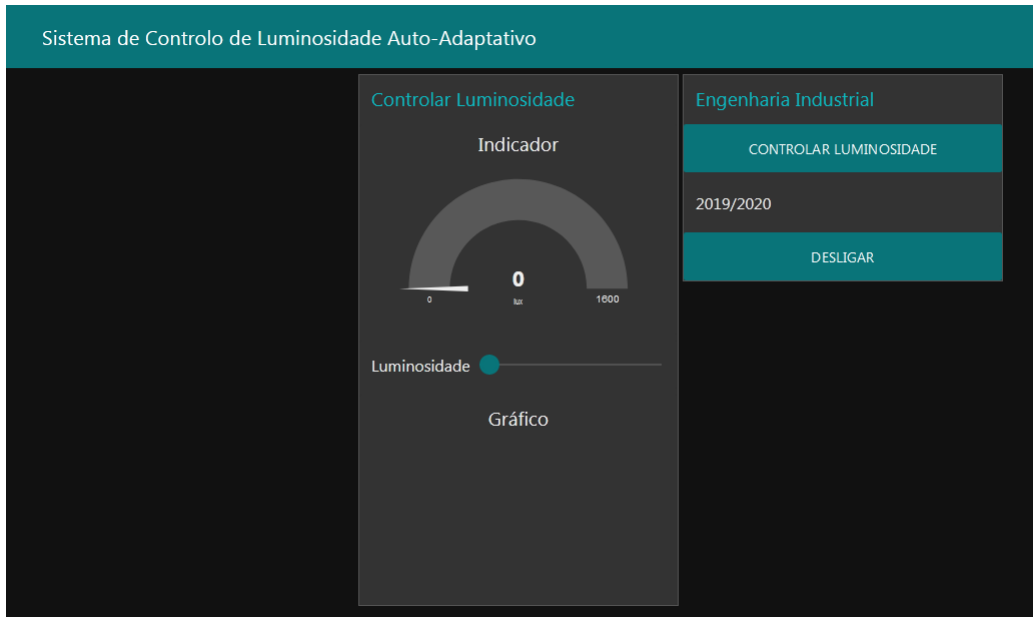


Figura 5.4: Aplicação do Sistema de Controlo de Luminosidade em NodeRED.

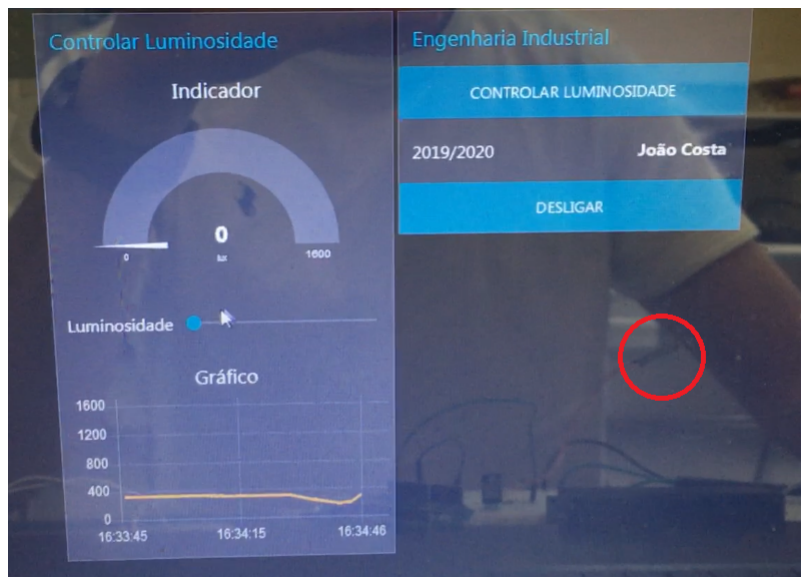


Figura 5.5: Sistema pronto para o controlo de luminosidade.

pretendida na aplicação onde a sua variação é de 0 a 1600 Lux. O gráfico da aplicação mostra a luminosidade que o sensor absorve em Lux e a respectiva hora a que este controlo foi submetido. Na figura 5.6, é possível observar o controlo em funcionamento.



Figura 5.6: Início do controlo de luminosidade.

Para que não haja gastos desnecessários de energia, quando é pretendido terminar o controlo de luminosidade, basta clicar no botão "Desligar", este irá terminar o controlo deixando de alimentar o LED. Na figura 5.7, é possível observar o LED desligado, mesmo que o slider não esteja no "0".

Posto isto, para retomar este mesmo controlo, é apenas necessário voltar a introduzir valores no slider de "Luminosidade", deste modo, o sistema volta a estar pronto para o controlo luminoso. Na figura 5.8, é possível observar que o controlo de luminosidade está ligado.



Figura 5.7: Desligar o controlo de luminosidade.



Figura 5.8: Retomar o controlo de luminosidade.

Capítulo 6

Conclusões e Trabalho Futuro

A IoT envolve uma variedade de dispositivos, possuindo estas uma capacidade interagir e cooperar uns com os outros a fim de alcançar objectivos comuns. A maior parte de dispositivos decorrentes da IoT são fornecidas por soluções de controlo dos diversos serviços e posteriormente o uso de plataformas, que integram esses serviços. Através desta tecnologia, podemos usar os recursos oferecidos pelos dispositivos da IoT para fornecer serviços de vastas áreas.

O controlo luminoso, usando aplicações IoT, apresentam uma ferramenta de fácil adaptação e implementação, pois estas dependem de plataformas e dispositivos conectados entre si. Para além disto, apresentam inúmeras vantagens se comparados com outros sistemas do mercado actual, ou seja, os materiais utilizados são de baixo custo e consequentemente baixo custo operacional se comparados com outros sistemas electrónicos.

Deste modo, este Controlo de Luminosidade Auto-Adaptativo usando Tecnologias de IoT, realizado em laboratório, é de fácil funcionalidade e rapidamente adaptável consoante a necessidade de quem o usa, isto é, com o uso da aplicação criada em Node-RED, facilmente se controla a luminosidade do LED do meio envolvente.

Contudo, ao longo do desenvolvimento da presente dissertação, foram superados todos os objectivos, correspondendo às expectativas nos resultados obtidos. Por outro lado, surgiram algumas dificuldades maioritariamente ao longo do desenvolvimento do código, de modo a obter uma resposta rápida e exacta no controlo, até que, com alguns testes de

diversas formas de controlo, o PID foi o que satisfiz às necessidades pretendidas.

6.1 Trabalhos Futuros

Por fim, depois de estar perfeitamente funcional este controlo luminoso, este poderá ser utilizado em edifícios e até mesmo na rede de iluminação pública. De seguida irei apresentar um exemplo de proposta para cada situação.

6.1.1 Proposta 1

Em edifícios, este projecto poderá ser complementado com um sistema de sensores ultrasónicos e até mesmo com a abertura de estores e persianas. Durante o período diurno, se pretendermos obter uma luminosidade constante, os estores ou persianas poderão abrir e fechar de modo a que isto aconteça. Já no período nocturno, com o uso de sensores ultrasónicos, ou seja, detectores de movimento, o LED do compartimento em questão ligar e atingir a luminosidade desejada na aplicação.

6.1.2 Proposta 2

Em rede de iluminação pública, para que haja uma maior eficiência energética, as luminárias não necessitam de apresentar uma intensidade constante. Para tal, para complementar este projecto, usando sensores ultrasónicos, onde cada apoio teria que ter o seu sensor, à medida que se vai aproximando de cada apoio, a luminosidade do LED da luminária fosse aumentando de modo a que a visibilidade fosse perfeita. Contudo, de acordo com a detecção de movimento juntamente com a distância a que o atinge, seriam estabelecidos valores de intensidade luminosa para cada LED.

Bibliografia

- [1] P. Marmé. (2018). O que é a Internet das Coisas (IoT – Internet of Things)?, URL: <https://www.wattson.pt/2018/03/09/o-que-e-a-internet-das-coisas-iot-internet-of-things/> (acedido em 17/03/2020).
- [2] G. Ciência. (2011). A história da luz: há mais de 2 mil anos ela instiga filósofos e cientistas, URL: <http://redeglobo.globo.com/globociencia/noticia/2011/11/historia-da-luz-ha-mais-de-2-mil-anos-ela-instiga-filosofos-e-cientistas.html> (acedido em 07/03/2020).
- [3] Wikipédia. (2020). Luz, URL: <https://pt.wikipedia.org/wiki/Luz> (acedido em 07/03/2020).
- [4] —, (2020). Radiação eletromagnética, URL: https://pt.wikipedia.org/wiki/Radia%C3%A7%C3%A3o_eletromagn%C3%A9tica (acedido em 08/03/2020).
- [5] TodaMatéria. (2020). Constante de Planck, URL: <https://www.todamateria.com.br/constante-de-planck/> (acedido em 10/03/2020).
- [6] —, (2020). Efeito Fotoelétrico, URL: <https://www.todamateria.com.br/efeito-fotoeletrico/> (acedido em 10/03/2020).
- [7] Wikipédia. (2020). Espectro Visível, URL: https://pt.wikipedia.org/wiki/Espectro_vis%C3%ADvel (acedido em 11/03/2020).

- [8] H. M. Junior. (2012). A formação das cores – parte 1, URL: <https://blogdografico.wordpress.com/2012/04/24/a-formacao-das-cores-parte-2-2/> (acedido em 11/03/2020).
- [9] D. SantoPietro. (2019). Luz: ondas eletromagnéticas, espectro eletromagnético e fótons, URL: <https://pt.khanacademy.org> (acedido em 12/03/2020).
- [10] SENERGIA. (2018). Eficiência Energética na Iluminação Pública, URL: http://www.senergia.pt/wp-content/uploads/2018/07/DREEIP_ParteI.pdf (acedido em 15/03/2020).
- [11] Wikipédia. (2020). Internet das Coisas, URL: https://pt.wikipedia.org/wiki/Internet_das_coisas (acedido em 17/03/2020).
- [12] CNCS. (2020). A Internet das Coisas (IoT – Internet of Things), URL: <https://www.cncs.gov.pt/a-internet-das-coisas-iot-internet-of-things/> (acedido em 17/03/2020).
- [13] B. Mishra, «Performance evaluation of MQTT broker servers», em *International Conference on Computational Science and Its Applications*, Springer, 2018, pp. 599–609.
- [14] S. Behrens. (2019). Why MQTT Is Everywhere, and the Security Issues It Faces, URL: <https://blog.paessler.com> (acedido em 28/03/2020).
- [15] INFAIMON. (2018). Automatização e robótica industrial, URL: <https://blog.infaimon.com/pt/automatizacao-e-robotica-industrial/> (acedido em 27/08/2020).
- [16] MECALUX. (2020). Automatização e Robótica, URL: <https://www.logismarket.pt/automatizacao-robotica/2721697500-ci.html> (acedido em 27/08/2020).
- [17] Wikipédia. (2019). Controlo de iluminação, URL: https://pt.wikipedia.org/wiki/Controle_de_ilumina%C3%A7%C3%A3o (acedido em 28/08/2020).
- [18] J. A. M. F. de Souza. (2020). Robótica, URL: http://webx.ubi.pt/~felippe/texts5/robotica_cap2.pdf (acedido em 22/03/2020).

- [19] R. Automação. (2019). Indústria 4.0 e IoT expandem aplicações com sensores inteligentes, URL: <https://revista-automacao.com/market-overview/17169-ind%5C%C3%5C%BAstria-4-0-e-iot-expandem-aplica%5C%C3%5C%A7%5C%C3%5C%B5es-com-sensores-inteligentes> (acedido em 22/09/2020).
- [20] Arduino. (2017). O que é Arduino?, URL: <https://www.arduinoportugal.pt/o-que-e-arduino/arduino> (acedido em 30/03/2020).
- [21] A. Mota. (2017). O que é Arduino e como funciona?, URL: <https://portal.vidadesilicio.com.br/o-que-e-arduino-e-como-funciona/> (acedido em 30/03/2020).
- [22] A. Electronics. (2020). O que é LED? Para que serve?, URL: <https://athoselectronics.com/o-que-e-led-diodo-emissor-luz/> (acedido em 30/03/2020).
- [23] F. K. (2018). ESP32: Detalhes internos e pinagem, URL: <https://www.fernandok.com/2018/03/esp32-detalhes-internos-e-pinagem.html> (acedido em 31/03/2020).
- [24] E. Systems. (2017). ESP-WROOM-32 Datasheet, URL: <https://blogmasterwalkershop.com.br/arquivos/datasheet/Datasheet%5C%20ESP-WROOM-32.pdf> (acedido em 31/03/2020).
- [25] B. Electrónica. (2018). Sensores Ópticos, URL: <https://www.boxelectronica.com/pt/sensores-opticos/519-tsl2561-digital-luminosity-lux.html> (acedido em 02/04/2020).
- [26] N. Semiconductor. (2006). LM1117/LM1117I800mA Low-Dropout Linear Regulator, URL: <https://datasheet.octopart.com/LM1117MP-3.3-Texas-Instruments-datasheet-2078.pdf> (acedido em 01/04/2020).
- [27] T. Instruments. (2006). LM1117MP-3.3, URL: <https://octopart.com> (acedido em 01/04/2020).
- [28] Arduino. (2020). PWM, URL: <https://www.arduino.cc/en/tutorial/PWM> (acedido em 01/04/2020).

- [29] NI. (2020). Explicando a Teoria PID, URL: <https://www.ni.com/pt-pt/innovations/white-papers/06/pid-theory-explained.html> (acedido em 01/04/2020).
- [30] Node-RED. (2020). About, URL: <https://nodered.org/about/> (acedido em 02/07/2020).
- [31] Wikipédia. (2020). EAGLE (program), URL: [https://en.wikipedia.org/wiki/EAGLE_\(program\)](https://en.wikipedia.org/wiki/EAGLE_(program)) (acedido em 02/04/2020).
- [32] SOLIDWORKS. (2020). Choosing SOLIDWORKS, URL: <https://www.solidworks.com/choosing-solidworks> (acedido em 03/04/2020).
- [33] Wikipédia. (2020). SolidWorks, URL: <https://pt.wikipedia.org/wiki/SolidWorks> (acedido em 03/04/2020).

Apêndice A

Código

A.1 Código 1

Trata-se do código compilado no ESP32. Neste microcontrolador é programado o sensor de modo a que este obtenha o valor desejado de luminosidade, em lux. Para tal, foi necessário a comunicação por MQTT de modo a enviar e receber dados. Os dados enviados são os valores da intensidade luminosa absorvida pelo sensor, por outro lado, os dados recebidos, são os valores desejados que o sensor terá que absorver. Esta comunicação é realizada por Wi-Fi e testada em laboratório ligada à rede da escola.

```

// Wifi-MQTT
#include <WiFi.h>
#include <PubSubClient.h>

//sensor TSL2561
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_TSL2561_U.h>

//Conectar ao servidor
const char* ssid = "agents";
const char* password =
"NVucvZu5R9QgCzzYvaHqqOM8AVruv26cV75kByq5diJ6v07D23KG4E1801zdpbG";
const char* mqtt_server = "193.136.195.56";

WiFiClient espClient;
PubSubClient client(espClient);

//Para delay
long lastMsg = 0;
int espera = 5000;
char msg[50];

Adafruit_TSL2561_Unified tsl =
Adafruit_TSL2561_Unified(TSL2561_ADDR_FLOAT, 12345);

char LuzString[8];

//sensor Tsl2561
void displaySensorDetails(void)
{
  sensor_t sensor;
  tsl.getSensor(&sensor);
  Serial.println("-----");
  Serial.print ("Sensor:      "); Serial.println(sensor.name);
  Serial.print ("Driver Ver:   "); Serial.println(sensor.version);
  Serial.print ("Unique ID:    "); Serial.println(sensor.sensor_id);
  Serial.print ("Max Value:    "); Serial.print(sensor.max_value);
  Serial.println(" lux");
  Serial.print ("Min Value:    "); Serial.print(sensor.min_value);
  Serial.println(" lux");
  Serial.print ("Resolution:   "); Serial.print(sensor.resolution);
  Serial.println(" lux");
  Serial.println("-----");
  Serial.println("");
  delay(500);
}

```

```

//sensor Tsl2561
void configureSensor(void)
{
  /* You can also manually set the gain or enable auto-gain support */
  // tsl.setGain(TSL2561_GAIN_1X);      /* No gain ... use in bright
light to avoid sensor saturation */
  // tsl.setGain(TSL2561_GAIN_16X);    /* 16x gain ... use in low
light to boost sensitivity */
  tsl.enableAutoRange(true);          /* Auto-gain ... switches
automatically between 1x and 16x */

  /* Changing the integration time gives you better sensor resolution
(402ms = 16-bit data) */
  tsl.setIntegrationTime(TSL2561_INTEGRATIONTIME_13MS);      /* fast
but low resolution */
  // tsl.setIntegrationTime(TSL2561_INTEGRATIONTIME_101MS); /* medium
resolution and speed */
  // tsl.setIntegrationTime(TSL2561_INTEGRATIONTIME_402MS); /* 16-bit
data but slowest conversions */

  /* Update these values depending on what you've set above! */
  Serial.println("-----");
  Serial.print ("Gain:          "); Serial.println("Auto");
  Serial.print ("Timing:        "); Serial.println("13 ms");
  Serial.println("-----");
}

void setup() {
  Serial.begin(9600);

  /* Exibir informações básicas sobre este sensor Tsl2561 */
  displaySensorDetails();

  /* Configurar o ganho e o tempo de integração do sensor Tsl2561 */
  configureSensor();

  setup_wifi();
  client.setServer(mqtt_server, 1883);
  client.setCallback(callback);
}

void setup_wifi() {
  delay(10);
  // Começo de conexão à rede WiFi
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.begin(ssid, password);
}

```

```

while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}

Serial.println("");
Serial.println("WiFi connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());
}

void callback(char* topic, byte* message, unsigned int length) {
    Serial.print("Message arrived on topic: ");
    Serial.print(topic);
    Serial.print(". Message: ");
    String messageTemp;

    for (int i = 0; i < length; i++) {
        Serial.print((char)message[i]);
        messageTemp += (char)message[i];
    }
    Serial.println("s");
    espera = atoi(messageTemp.c_str()) * 1000;
    Serial.print(espera);Serial.println("ms");

    // Altera o estado de saída de acordo com a mensagem recebida no
    tópico esp32/output, a mensagem

}

void reconnect() {
    // Loop até à reconexão
    while (!client.connected()) {
        Serial.print("Attempting MQTT connection...");
        // Tentativa de conexão
        if (client.connect("ESP8266Client")) {
            Serial.println("connected");
            // Subscriber
            client.subscribe("esp/output");
        } else {
            Serial.print("failed, rc=");
            Serial.print(client.state());
            Serial.println(" try again in 5 seconds");
            // Espera de 5 segundos antes de tentar novamente
            delay(5000);
        }
    }
}
}

```

```
void loop() {
  if (!client.connected()) {
    reconnect();
  }
  client.loop();

  //delay(5000)
  long now = millis();
  if (now - lastMsg > 100) {
    lastMsg = now;

    luz();
  }
}

void luz() {

  //luz(lux)
  sensors_event_t event;
  tsl.getEvent(&event);

  dtostrf(event.light, 1, 2, LuzString);
  Serial.print("Lux: ");
  Serial.println(LuzString);
  client.publish("esp/Luz", LuzString);
}
```

A.2 Código 2

Neste caso, trata-se do código compilado no Arduino. Este microcontrolador, também ligado à rede da escola, onde a sua comunicação é feita por Wi-Fi, recebe os valores de luminosidade absorvidos pelo sensor TSL2561. É também neste código que é realizado o controlo PID, que faz com que a intensidade do LED aumente ou diminua alterando o seu set-point. Este set-point é a luminosidade que o sensor pretende absorver. Para manobrar a intensidade de luz emitida pelo LED, foi utilizada uma porta digital que possibilita uma modelação através de largura de pulso (PWM), a porta 6.

```

// Wifi-MQTT
#include <WiFi.h>
#include <PubSubClient.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <PID_v1.h>

// string para manter o input
String inString = "";

//Conectar ao servidor
const char* ssid = "agents";
const char* password =
"NVucvZu5R9QgCzzYvaHqqOM8AVrUV26cV75kByq5diJ6v07D23KG4E1801zdpbG";

const char* mqtt_server = "193.136.195.56";

WiFiClient espClient;
PubSubClient client(espClient);

//Para delay
long lastMsg = 0;
int espera = 5000;
char msg[50];
String messageTemp;
int Luz = atoi(messageTemp.c_str());

int led = 6;

//Variáveis
double Setpoint, Input, Output;

//PID Parâmetros
double Kp = 0.02, Ki = 0.4, Kd = 0.002;

//Start PID instance
PID myPID(&Input, &Output, &Setpoint, Kp, Ki, Kd, DIRECT);

void setup() {
  Serial.begin(9600);
  pinMode(led, OUTPUT);

```

```

//Set point : Brilho
Setpoint = 0;
//Ligar PID
myPID.SetMode(AUTOMATIC);

setup_wifi();
client.setServer(mqtt_server, 1883);
client.setCallback(callback);
}

void setup_wifi() {
  delay(10);
  // Começo de conexão à rede WiFi
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}
void callback(char* topic, byte* message, unsigned int length) {
  // Serial.print("Message arrived on topic: ");
  // Serial.print(topic);
  // Serial.print(". Message: ");
  String messageTemp;

  for (int i = 0; i < length; i++) {
    Serial.print((char)message[i]);
    messageTemp += (char)message[i];
  }
  if (strcmp(topic, "esp/Luz")==0) {
    Luz = atoi(messageTemp.c_str());
    Serial.print(" lux");
    Serial.print(Luz);

    Serial.print(" Output");
    Serial.println(Output);
    Input = Luz;
  }
}

```

```

    if (strcmp(topic, "esp/Setpoint")==0) {
        Setpoint = atoi(messageTemp.c_str());
        Serial.println("Atualizei...");
    }

}

void reconnect() {
    // Loop até à reconexão
    while (!client.connected()) {
        Serial.print("Attempting MQTT connection...");
        // Tentativa de conexão
        if (client.connect("ESP8266Client")) {
            Serial.println("connected");
            // Subescreber
            client.subscribe("esp/output");
        } else {
            Serial.print("failed, rc=");
            Serial.print(client.state());
            Serial.println(" try again in 5 seconds");
            // Espera de 5 segundos antes de tentar novamente
            delay(5000);
        }
    }
}

void loop() {

    if (!client.connected()) {
        reconnect();
    }
    client.loop();

    myPID.Compute();
    analogWrite(6, Output);
    Serial.print("Setpoint: ");
    Serial.println(Setpoint);
}

```

Apêndice B

Proposta de Dissertação

**Proposta de Dissertação / Projeto
Mestrado em Engenharia Industrial
2019/2020**

Título Provisório: Sistema de controle de luminosidade auto-adaptativo usando tecnologias de IoT

Aluno de mestrado:

Orientador: José Barbosa

Co-Orientador: Paulo Leitão

Principais objetivos a atingir:

O objetivo deste trabalho consiste no estudo e desenvolvimento de uma solução que visa o controle de luminosidade auto-adaptável. Desta forma, o sistema a desenvolver deverá contemplar o desenvolvimento de uma aplicação móvel, um elemento sensorizador e a posterior atuação da luz LED de uma das bancadas existentes no Laboratório de Investigação e Inovação (L2I), através do uso do protocolo MQTT, controlando o LED usando uma aplicação desenvolvida em Node-RED.

Resultados esperados:

Pretende-se desenvolver uma solução que permita, de uma forma rápida e eficiente, efetuar o ajuste das condições desejáveis de iluminação do LED existente do L2I, assim como o controlo do nível de conforto, na variável iluminação, do nível de iluminação do L2I.

Caracterização do trabalho

A utilização de tecnologias baseadas em Internet of Things (IoT) permite automatizar e agilizar a troca de informação entre dispositivos constituintes de um sistema. No caso da presente proposta, é pretendido o desenvolvimento de uma aplicação móvel que permita ao utilizador definir o nível de iluminação desejado de um determinado espaço (e.g., LED existente no L2I e/ou nível de iluminação ambiente do L2I). Uma vez definido, o sistema deverá permitir a troca de informação entre um elemento de sensorização (capaz de medir o nível de iluminação do ambiente onde está inserido) e o elemento controlador (i.e. o que permite ajustar a iluminação do espaço). Desta forma, o sistema deverá, de forma rápida e eficiente ajustar o nível de iluminação ao pretendido pelo utilizador.

Calendarização das fases do trabalho:

O desenvolvimento da presente proposta de trabalho será realizado através da execução das seguintes etapas:

1. Familiarização com o problema de sensorização de iluminação e controlo de iluminação usando técnicas de variação de tensão (M1-M2)
2. Desenvolvimento do sistema de sensorização (M2 – M3)
3. Desenvolvimento do sistema de controlo de iluminação do LED (M3 – M4)
4. Desenvolvimento da aplicação móvel (M3-M6)
5. Implementação do protocolo de envio e troca de informação entre os diversos elementos (M4 – M7)
6. Testes e validação do protótipo (M7 – M10)
7. Escrita da dissertação e defesa final do trabalho (M10 - M11)

Palavras-chave: IoT, sensorização, LED.

Infraestruturas a utilizar:

Este trabalho será desenvolvido no Laboratório de Investigação e Inovação (L2).

