



Catarina Pereira
Catarina Amorim
Inês Neves
Leonardo Martins

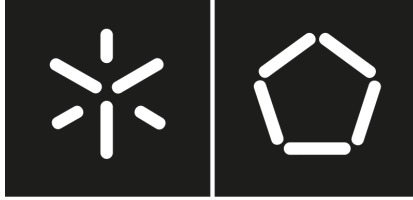
**Comunicação pela Luz: Li-Fi em Ação para
Conversação e Transferência de Ficheiros**



Universidade do Minho
Escola de Engenharia

Catarina da Cunha Malheiro da Silva Pereira
Catarina Marinho Amorim
Inês Cabral Neves
Leonardo Dias Martins

**Comunicação pela Luz: Li-Fi em Ação
para Conversação e Transferência de
Ficheiros**



Universidade do Minho

Escola de Engenharia

Catarina da Cunha Malheiro da Silva Pereira
Catarina Marinho Amorim
Inês Cabral Neves
Leonardo Dias Martins

**Comunicação pela Luz: Li-Fi em Ação
para Conversação e Transferência de
Ficheiros**

Relatório Final
Projeto Integrador em
Telecomunicações e Informática
Mestrado em Engenharia
Telecomunicações e Informática

Trabalho efetuado sob a orientação de:

Professor Doutor Joaquim Melo Henriques Macedo

Professor Doutor Adriano Jorge Cardoso Moreira





**Professor Doutor Bruno Daniel Mestre Viana
Ribeiro**

e de

Professor Doutor José Augusto Afonso

Identificação do Grupo

O Grupo 03 é constituído por quatro membros, sendo que três deles são do 1º ano do Mestrado em Engenharia de Telecomunicações e Informática (METI), sendo identificados por Pós-Graduação (PG) seguido dos seus números mecanográficos, enquanto o quarto membro pertence ao 4º ano de Mestrado Integrado em Engenharia de Telecomunicações e Informática (MIETI) e é identificado pelo código Aluno (A) seguido do seu número mecanográfico.

Imagem	Nome / Número Mecanográfico / E-mail institucional
	Catarina da Cunha Malheiro da Silva Pereira PG53733 pg537336@alunos.uminho.pt
	Catarina Marinho Amorim A93094 a93094@alunos.uminho.pt
	Inês Cabral Neves PG53864 pg53864@alunos.uminho.pt
	Leonardo Dias Martins PG53996 pg53996@alunos.uminho.pt

Resumo

Este relatório apresenta o desenvolvimento de um sistema de comunicação baseado em tecnologia Li-Fi. A tecnologia Li-Fi, que utiliza luz visível para transmitir dados, oferece vantagens como maior capacidade de banda e segurança em comparação ao Wi-Fi. O projeto abrange a concepção e implementação dos circuitos emissor e recetor, a criação de protocolos de comunicação e a execução de testes de validação. Foram utilizados componentes como o ESP32, amplificadores operacionais e fotodetetores. Os resultados dos testes demonstraram a viabilidade da comunicação por Li-Fi em várias distâncias, com análise detalhada de desempenho e identificação de possíveis melhorias.

Índice

Identificação do Grupo	ii
Resumo	iii
Índice de Figuras	vi
Índice de Tabelas	vii
Lista de Acrónimos	viii
Acrónimos	viii
1 Introdução	1
2 Revisão da Literatura	2
3 Recursos e Instrumentos Utilizados	4
3.1 Software	4
3.2 Hardware	4
3.2.1 ESP32	4
3.2.2 Amplificador operacional UA741CN	5
3.2.3 Fototransistor PL-51P3C	5
3.2.4 Fotodíodo PL-53F3BT	6
3.2.5 Transistor 2N2222A	6
3.2.6 Circuito Integrado 7404	7
3.2.7 Outros Materiais	7
4 Arquitetura do Sistema	8
4.1 Circuito Emissor	8
4.2 Circuito Recetor	9
5 Protocolo de Comunicação	13
5.1 Data Link Layer	13
5.2 Comunicação entre PCs	14
5.3 Aplicação-Aplicação	14
6 Interface Gráfica	16
6.1 Aplicação de Transmissão	16
6.2 Aplicação de Receção	16
7 Testes e Resultados	18
7.1 Distâncias entre os Links	18
7.1.1 Distância de 20 cm	18
7.1.2 Distância de 80 cm	19
7.1.3 Distância de 1,2 m	20
7.2 Testes de Tempo de Transmissão	20
8 Requisitos Funcionais e Não Funcionais	22
9 Conclusão	23
10 Autoavaliação	24

Índice de Figuras

1	Placa ESP32 [13].	5
2	<i>Pinout</i> Amplificador operacional UA741CN	5
3	<i>Pinout</i> Fotodetector PL-51P3C	6
4	<i>Pinout</i> Fotodetector PL-53F3BT	6
5	<i>Pinout</i> 2N2222A	6
6	Circuito Integrado 7404 [15].	7
7	Arquitetura Geral da fase C.	8
8	Pinout do Circuito Emissor.	8
9	Pinout do Circuito Recetor.	9
10	Pinout do recetor, filtro e amplificador transimpedância.	10
11	Gráfico da tensão de saída consoante a tensão de entrada do comparador não inversor com histerese.	11
12	Pinout do Comparador.	12
13	Pinout do Retificador de meia onda.	12
14	Enquadramento do Protocolo.	13
15	Mecanismo de Controlo de Fluxo <i>Stop-and-Wait</i>	14
16	Especificação das Tramas da Camada de Aplicação.	15
17	Aplicação do emissor.	16
18	Verificação das portas na aplicação do emissor.	16
19	Aplicação do recetor.	17
20	Sinal do cátodo do LED do circuito emissor.	18
21	Sinal à saída do amplificador de transimpedância com filtro.	18
22	Sinal à saída do comparador com filtro.	19
23	Sinal da saída do circuito frontend.	19
24	Saída do circuito frontend a 80 cm de distância.	19
25	Saída do circuito frontend a 1,20 m de distância.	20
26	Tempo de transmissão de uma mensagem de texto.	20
27	Tempo de transmissão de uma imagem.	20

Índice de Tabelas

Tabela 1: Materiais necessários para o projeto.	7
Tabela 2: Requisitos Funcionais do projeto.	22
Tabela 3: Requisitos Não Funcionais do projeto.	22
Tabela 6: Autoavaliação de cada elemento do grupo: Catarina Pereira (CP); Catarina Amorim (CA); Inês Neves (IN) e Leonardo Martins (LM).	24

Acrónimos

A	Aluno
AmpOp	Amplificador Operacional
ARQ	Automatic Repeat Request
CRC	Cyclic Redundancy Check
GPIO	General Purpose Input/Output
I2C	Inter-Integrated Circuits
IDE	Integrated Development Environment
IR	Infrared
Li-Fi	Light Fidelity
METI	Mestrado em Engenharia de Telecomunicações e Informática
MIETI	Mestrado Integrado em Engenharia de Telecomunicações e Informática
MISO	Master In Slave Out
MOSI	Master Out Slave In
NPN	Negativo-Positivo-Negativo
OSI	Open Systems Interconnection
PG	Pós-Graduação
RF	Requisitos Funcionais
RNF	Requisitos Não Funcionais
SCK	Serial Clock
SPI	Serial Peripheral Interface
SS	Slave Select
TINA	Toolkit for Interactive Network Analysis
TTL	Transistor-Transistor Logic
UART	Universal Asynchronous Receiver-Transmitter
UC	Unidade Curricular

1 Introdução

Este relatório faz parte da Unidade Curricular (UC) Projeto Integrador em Telecomunicações e Informática, do 2º semestre do 1º ano do Mestrado (Integrado) em Engenharia de Telecomunicações e Informática. Este projeto foi desenvolvido como resposta a um problema apresentado pelos docentes.

A comunicação sem fios é um campo em rápida evolução, com diversas tecnologias emergentes que oferecem soluções inovadoras para os desafios atuais. Uma dessas tecnologias promissoras é o Light Fidelity (Li-Fi), que utiliza o espectro de luz visível para transmissão de dados.

O Li-Fi apresenta várias vantagens em comparação ao Wi-Fi tradicional, como maior capacidade, largura de banda e segurança. Além disso, pode aproveitar a infraestrutura LED existente para comunicação, tendo diversas aplicações, como acesso à Internet através de iluminação e comunicação subaquática.

Apesar do grande potencial, o Li-Fi ainda enfrenta alguns desafios técnicos e de implementação. Para que esta tecnologia se torne amplamente adotada, é necessário superar obstáculos como a necessidade de linha de visão entre transmissor e recetor, a interferência de luz ambiente e a padronização de protocolos.

Este projeto visa explorar o desenvolvimento de um sistema de comunicação Li-Fi, abordando aspetos como a arquitetura do sistema, o projeto dos circuitos emissor e recetor, a implementação de protocolos de comunicação e a realização de testes e validações.

2 Revisão da Literatura

Li-Fi é uma tecnologia de comunicação sem fios que utiliza espectro de luz visível para transmissão de dados. Esta tecnologia opera com a modulação de luzes LED para transmitir dados binários, imperceptíveis ao olho humano. Comparado ao Wi-Fi, o Li-Fi oferece maior capacidade, largura de banda e segurança [1, 2]. Pode aproveitar a infraestrutura LED existente para comunicação e tem diversas aplicações, incluindo acesso à Internet através da iluminação e comunicação subaquática.

Quanto aos protocolos de comunicação de série, pode-se dividir a terminologia em três partes. A comunicação é uma terminologia muito conhecida que envolve a troca de informações entre dois ou mais meios. Em sistemas embarcados, a comunicação significa a troca de dados entre dois microcontroladores na forma de bits [3]. Esta troca de bits de dados no microcontrolador é feita por algum conjunto de regras definidas conhecidas como protocolos de comunicação. Agora, se os dados forem enviados em série, ou seja, um após o outro, o protocolo de comunicação é conhecido como Protocolo de Comunicação de Série. Mais especificamente, os bits de dados são transmitidos um de cada vez de maneira sequencial através do barramento de dados ou canal de comunicação na Comunicação de Série. Os vários tipos de protocolos são:

- O protocolo Inter-Integrated Circuits (I2C) permite a conexão de múltiplos dispositivos escravos a um ou mais mestres numa placa de circuito impresso (PCB), usando apenas dois fios bidirecionais para transmissão e recepção de dados. Com uma velocidade de até 3,4 Mbps, é bastante rápido. As vantagens incluem flexibilidade no uso de múltiplos dispositivos mestres, endereçamento de *chips* sem a necessidade de linhas de seleção e um mecanismo robusto de tratamento de erros. No entanto, I2C ocupa mais espaço na PCB devido ao uso de resistências de *pull-up*. As aplicações incluem ajustes de cores em monitorização, controlo de *displays* em telemóveis e controlo de energia de componentes internos [4].
- O protocolo Serial Peripheral Interface (SPI) opera com uma arquitetura de mestre-escravo e utiliza quatro sinais: Master Out Slave In (MOSI), Master In Slave Out (MISO), Serial Clock (SCK) e Slave Select (SS). Oferece alta velocidade de transmissão, com um alcance de taxas de dados acima de 1 Mbps, e é especialmente útil para interfaces simples de *hardware*. No entanto, as desvantagens incluem a ausência de mecanismos de verificação de erros e de confirmação de recepção pelos escravos. O SPI é aplicado numa variedade de cenários, desde a comunicação com sensores e controladores de vídeo jogos até a interface com dispositivos de armazenamento como cartões SD e memórias FLASH. É também utilizado para leitura de dados de relógios em tempo real e para controlar LCDs [4].
- Universal Asynchronous Receiver-Transmitter (UART) são componentes fundamentais na implementação de comunicação de série [5]. Estes atuam como intermediários entre interfaces paralelas e de séries, convertendo dados entre os formatos paralelo e de série. Enquanto alguns microcontroladores incluem UARTs integrados, outros exigem *chips* externos para essa funcionalidade. Os UARTs são responsáveis por enviar e receber dados, convertendo-os entre formatos paralelo e de série de acordo com a taxa de transmissão definida. Além disso, UARTs mais avançados podem armazenar dados recebidos em *buffers*, garantindo uma transferência eficiente de dados entre dispositivos.

O modelo Open Systems Interconnection (OSI), uma estrutura fundamental para comunicação em rede, consiste em sete camadas, cada uma com funções e interações específicas. Estas camadas incluem as camadas Física, Ligação de Dados, Rede, Transporte, Sessão, Apresentação e Aplicação, que controlam coletivamente a transferência de dados e garantem uma comunicação segura. É destacado o papel do modelo em facilitar a transferência de informações entre diferentes endereços lógicos, com os *gateways* a desempenhar um papel crucial neste processo [6]. Também é enfatizada a importância do modelo OSI no estabelecimento de protocolos de comunicação para redes de computadores [7].

A camada 7 do modelo OSI, conhecida como Camada de Aplicação, é a camada superior do modelo e é responsável por fornecer serviços de rede diretamente às aplicações dos utilizadores [8]. Esta camada permite

a comunicação entre diferentes aplicações e é onde os utilizadores interagem com a rede. Esta camada define os protocolos e métodos que as aplicações usam para trocar dados pela rede [9]. A camada de aplicação inclui protocolos como HTTP, FTP, SMTP e DNS, que são essenciais para vários tipos de comunicação de rede e troca de dados. Por definição, a Camada 7 concentra-se nos requisitos de comunicação das aplicações de *software* e garante que os dados sejam apresentados ao utilizador de forma significativa [10].

3 Recursos e Instrumentos Utilizados

Neste capítulo explora-se em detalhe os elementos que desempenham um papel fundamental na condução deste projeto, com um foco predominante em ferramentas conforme a sua categoria: *Software* e *Hardware*. O conjunto de ferramentas e recursos utilizados abrange uma ampla variedade de aplicações, sendo que cada uma desempenha um papel específico e vital no desenvolvimento do projeto.

3.1 Software

As ferramentas e recursos da categoria de *Software* utilizados são as seguintes:

- **Smartsheet:** Aplica-se o programa Smartsheet para o planeamento temporal das tarefas do grupo, o que garante uma gestão eficaz do cronograma de trabalho.
- **Miro:** Utiliza-se a plataforma Miro para criar diagramas de blocos e fluxogramas, o que facilita a visualização e a comunicação de conceitos complexos.
- **Arduino Integrated Development Environment (IDE):** Para a edição, compilação e envio do código para a placa ESP32.
- **PyCharm:** Para desenvolvimento e execução de código Python.
- **Simulador Toolkit for Interactive Network Analysis (TINA) (versão 9) e Circuit Diagram:** Permitem fazer simulações tanto de circuitos analógicos como digitais.
- **Plataformas de Comunicação:** Para facilitar a comunicação, colaboração e organização do código desenvolvido pelo grupo, utiliza-se as plataformas Discord e Whatsapp.
- **OverLeaf:** Para a elaboração de relatórios em formato \LaTeX , utiliza-se a plataforma OverLeaf, que simplifica a formatação e a colaboração em documentos técnicos.

3.2 Hardware

Para as ferramentas de *hardware* utiliza-se uma placa ESP32, Amplificador operacional UA741CN, Fotodetector PL-51P3C, Fotodiodo PL-53F3BT e Transistor 2N2222A para o projeto. Os *datasheet* e a representação *pinouts* de cada componente eletrónico está também representado.

3.2.1 ESP32

A placa ESP32, Figura 1, é uma placa de desenvolvimento de *hardware* que utiliza o microcontrolador ESP32. O ESP32 é um microcontrolador de baixo custo e baixo consumo de energia que possui recursos avançados, como conectividade Wi-Fi e Bluetooth, memória *flash* integrada, entrada e saída de dados digitais e analógicos, entre outros [11, 12].

Existe duas placas ESP32, uma para o emissor e outra para o recetor. Para o emissor a placa é ESP32S WROOM 32E enquanto que, a placa ESP32 do recetor é ESP32 Express 32E. Por isso, os *pinouts* destas duas placas ESP32 serão diferentes.

3.2. HARDWARE

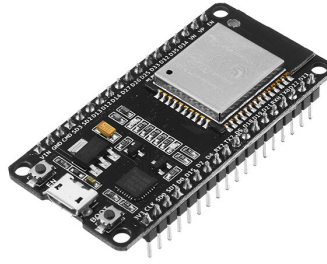


Figura 1: Placa ESP32 [13].

3.2.2 Amplificador operacional UA741CN

O amplificador operacional UA741CN é um dispositivo eletrónico amplamente utilizado em circuitos eletrónicos para amplificar sinais elétricos. O UA741CN é um modelo específico de amplificador operacional fabricado pela Texas Instruments, mas o termo “amplificador operacional” refere-se a uma classe geral de dispositivos.

O UA741CN possui características como alta impedância de entrada, alta ganho de tensão, largura de banda suficiente para muitas aplicações de áudio e controlo, entre outras propriedades desejáveis em circuitos de amplificação e processamento de sinais. É frequentemente utilizado numa variedade de aplicações, incluindo processamento de sinais de áudio, controlo de motor, instrumentação, filtros ativos, entre outros [14].

A representação do *datasheet* do Amplificador operacional UA741CN.

A Figura 2 representa o *Pinout* Amplificador operacional UA741CN.

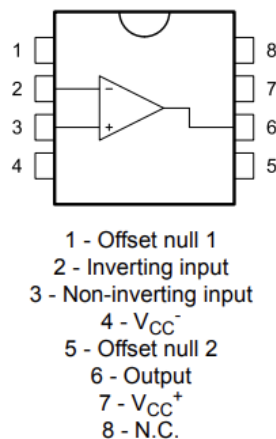


Figura 2: *Pinout* Amplificador operacional UA741CN

3.2.3 Fototransístor PL-51P3C

O PL-51P3C é um tipo específico de fototransístor. Um fototransístor é um dispositivo eletrónico que converte a luz incidente numa corrente elétrica ou noutra forma de sinal elétrico. Geralmente, fotodetectors são usados numa variedade de aplicações, incluindo sistemas de segurança, comunicações ópticas, equipamentos médicos, sensores industriais, entre outros.

A representação do *datasheet* do Fotodetector PL-51P3C.

A Figura 3 representa o *Pinout* Fotodetector PL-51P3C.

3.2. HARDWARE

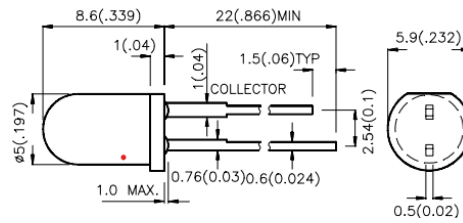


Figura 3: *Pinout* Fotodetetor PL-51P3C

3.2.4 Fotodíodo PL-53F3BT

O PL-53F3BT é um tipo específico de fotodiodo. Um fotodiodo é um dispositivo eletrônico usado para detectar luz ou outras formas de radiação eletromagnética. No caso do PL-53F3BT, usado em aplicações industriais, de pesquisa ou de consumo.

A representação do *datasheet* do Fotodetetor PL-53F3BT.

A Figura 4 representa o *Pinout* Fotodetetor PL-53F3BT.

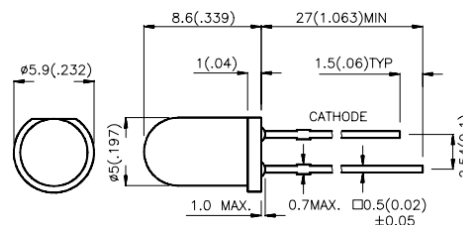


Figura 4: *Pinout* Fotodetetor PL-53F3BT

3.2.5 Transistor 2N2222A

O Transistor bipolar de junção Negativo-Positivo-Negativo (NPN) 2N2222A é um dispositivo eletrônico amplamente utilizado numa variedade de aplicações, desde amplificação de sinais até comutação de circuitos.

O 2N2222A é encapsulado no formato TO-92, uma embalagem comum para dispositivos eletrônicos pequenos. As dimensões mecânicas são padronizadas, facilitando a integração em placas de circuito impresso e outros sistemas eletrônicos.

A representação do *datasheet* do Transistor 2N2222A.

A Figura 5 representa o *Pinout* 2N2222A

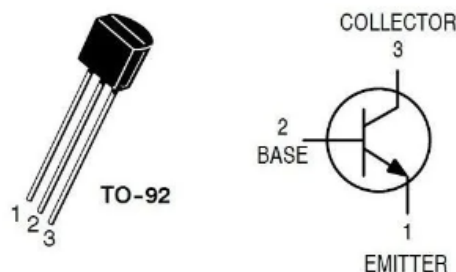


Figura 5: *Pinout* 2N2222A

3.2.6 Circuito Integrado 7404

O circuito integrado 7404 é um componente digital muito utilizado em eletrônica e pertence à família de circuitos Transistor-Transistor Logic (TTL). Este circuito é um inversor hexagonal, ou seja, contém seis portas lógicas inversoras independentes. Cada uma dessas portas tem a função de inverter o sinal de entrada. Isto significa que, se a entrada de uma porta é um nível lógico alto (1), a saída será um nível lógico baixo (0), e vice-versa.

A representação do *datasheet* do circuito integrado 7404 é representado na Figura 6.

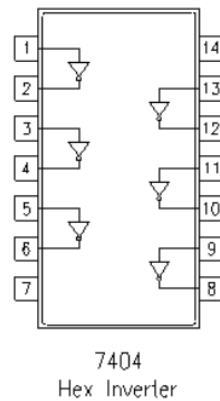


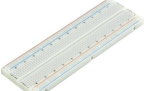




Figura 6: Circuito Integrado 7404 [15].

3.2.7 Outros Materiais

Na Tabela 1 referem-se os materiais necessários para o desenvolvimento do projeto. As imagens que se encontram na Tabela 1 são ilustrativas.

Tabela 1: Materiais necessários para o projeto.

Componente	Função	Quantidade	Figura do Componente
Computador	Emissor e Recetor	2	
Cabos Micro USB-C	Estabelecer ligações entre o computador e a placa ESP32	2	
BreadBoard	Estabelecer ligações entre todos os materiais	2	
Fios de Ligação	Estabelecer ligações entre a placa ESP32 e o circuito	Indefinido	
Resistência	Controlar a passagem de corrente elétrica nos circuitos (68 Ω , 5.4 $k\Omega$, 16 $k\Omega$ e 33 $k\Omega$)	4	

4 Arquitetura do Sistema

A arquitetura geral é delineada na Figura 7, destacando-se os componentes essenciais e os protocolos envolvidos. Nesta figura, são apresentados o emissor, o circuito driver, o LED de infravermelhos, o fotodetector, o PC, a ESP32 e a conexão ótica, juntamente com o protocolo UART.

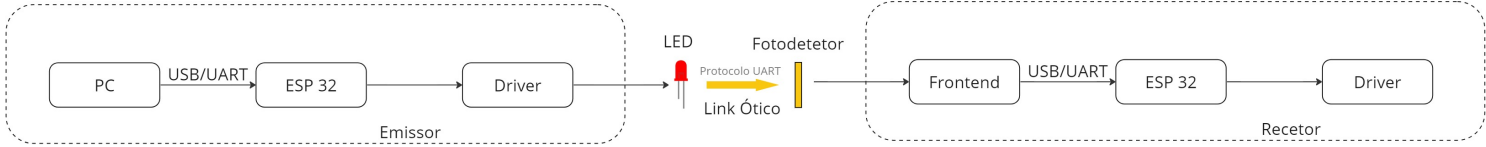


Figura 7: Arquitetura Geral da fase C.

4.1 Circuito Emissor

O circuito driver é um circuito eletrônico projetado para amplificar e controlar a corrente ou tensão num dispositivo, como um LED infravermelhos, neste caso. Este circuito será conectado ao Digital Lab, fornecendo os sinais ao driver, que os amplificará e controlará de acordo com as especificações do dispositivo. A Figura 8 mostra a conexão do circuito Driver ao ESP32 através do pino General Purpose Input/Output (GPIO) 3 para o uso do protocolo UART.

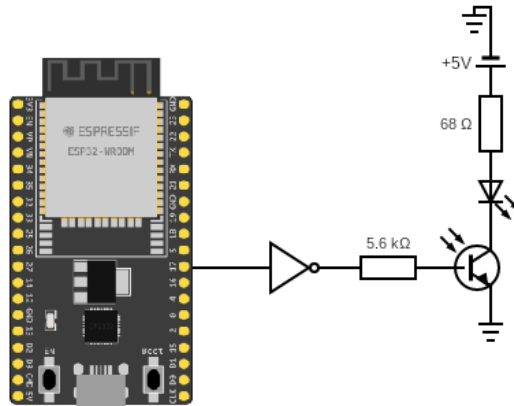


Figura 8: Pinout do Circuito Emissor.

O circuito driver é fundamental para a conversão do sinal elétrico em sinal ótico. Este circuito é composto por um transistor bipolar NPN, um LED emissor de infravermelhos e resistências. Para determinar os valores dessas resistências e correntes, é necessário realizar alguns cálculos.

Primeiramente, considerando que a corrente máxima do LED Infrared (IR) é de 50 mA para obter a máxima radiação, e assumindo um valor de β (ganho do transistor) igual a 100, pode-se calcular a corrente da base do transistor (I_{BT}) e a corrente do coletor (I_C) usando a expressão (4.1):

$$I_{BT} = \frac{I_C}{\beta} \quad (4.1)$$

Substituindo os valores, onde I_C é 50 mA e β é 100, obtém-se:

$$I_{BT} = \frac{0.05}{100} = 0.5 \text{ mA}$$

4.2. CIRCUITO RECETOR

Para calcular a resistência da base do transistor (R_{BT}), utiliza-se a expressão (4.2):

$$R_{BT} = \frac{V_I - V_{BE}}{I_{BT}} \quad (4.2)$$

Onde V_I é 3,3 V (valor máximo de tensão de um pino GPIO da placa ESP32) e V_{BE} é 0,6 V (queda de tensão típica da base para o emissor do transistor). Substituindo esses valores, tem:

$$R_{BT} = \frac{3.3 - 0.6}{0.5 \times 10^{-3}} = 5.4 \text{ k}\Omega$$

Para calcular o valor da resistência do LED (R_{LED}), considera-se a potência máxima que pode ser aplicada na resistência. Se utilizar uma resistência com potência máxima de 0,25 W, e a corrente que a atravessa é I_C , a queda de tensão máxima na resistência é de 5 V, conforme a expressão (4.3):

$$P = V \times I \quad (4.3)$$

Como V_{cc} (tensão de alimentação) é 5 V, a potência máxima suportada pela resistência não é um problema.

Quando a corrente que atravessa o LED é máxima ($I_C = 50 \text{ mA}$), a queda de tensão no LED é de $V_{LED} = 1,38 \text{ V}$. Assumindo que a queda de tensão do coletor para o emissor é $V_{CE} = 0,3 \text{ V}$, quando o transistor opera em saturação, pode-se calcular o valor de R_{LED} usando a expressão (4.4):

$$R_{LED} = \frac{V_{cc} - V_{LED} - V_{CE}}{I_C} \quad (4.4)$$

Substituindo os valores, tem:

$$R_{LED} = \frac{5 - 1.38 - 0.3}{0.05} = 66.4 \Omega$$

Será utilizado um valor de 68Ω , pois não existem resistências com o valor exato calculado. Opta-se, portanto, por um valor superior aproximado ao calculado.

4.2 Circuito Recetor

O circuito recetor é projetado para receber, processar e converter sinais de entrada de uma fonte externa, o emissor, amplificando, filtrando e adaptando os sinais para o processamento subsequente. A conexão do circuito emissor ao ESP32 é estabelecida através do pino GPIO16, Figura 9.

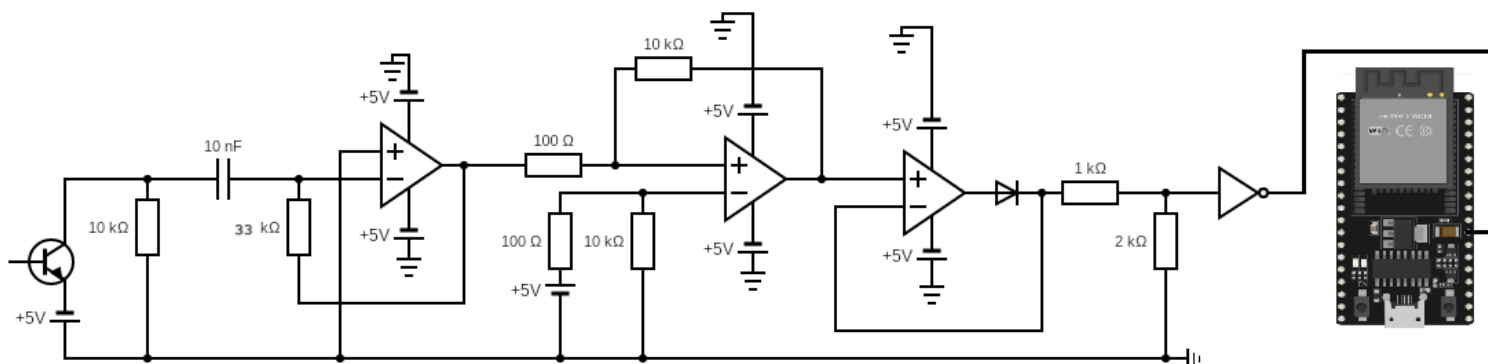


Figura 9: Pinout do Circuito Recetor.

No circuito emissor, projetado para a conversão do sinal ótico de volta para o sinal elétrico, utiliza-se um fototransistor seguido por um amplificador de transimpedância. Este amplificador converte corrente em tensão,

4.2. CIRCUITO RECETOR

onde a relação entre essas grandezas é dada pela expressão (4.5):

$$R_f = \frac{V_{out}}{I_{in}} \quad (4.5)$$

Considerando que o valor máximo para V_{out} é 3,3 V e o valor mínimo da corrente do coletor no foto-transistor é 0,1 mA, pode-se calcular o valor da resistência de feedback (R_f) como sendo 33 k Ω .

É importante observar que a tensão máxima de entrada na placa ESP32 é 3,3 V, o que implica que a tensão de saída do amplificador de transimpedância não pode exceder esse valor. Portanto, a tensão de alimentação do Amplificador Operacional (AmpOp) será de -3,3 V a 3,3 V.

Outro requisito crucial neste circuito é a implementação de uma estratégia para mitigar as interferências da luz ambiente. Para isso, será usado um filtro passa-alto. No entanto, é importante observar que a luz ambiente geralmente apresenta um sinal relativamente constante, enquanto os sinais de interesse possuem uma frequência específica. Sendo o filtro passa-alto passivo.

Considerando que a frequência do ruído (luz ambiente) é de cerca de 100 Hz e deseja-se uma frequência de corte de aproximadamente 1 kHz, com uma atenuação de 20 dB por década, pode-se calcular a frequência de corte (f_c) usando a equação (4.6):

$$f_c = \frac{1}{2\pi \times R \times C} \quad (4.6)$$

Substituindo os valores de $R = 16$ k Ω e $C = 10$ nF, obtêm-se $f_c = 994,72$ Hz.

A frequência de corte pode ser ajustada posteriormente para um valor superior, se necessário, visando aprimorar a qualidade do sinal recebido. Entretanto, é importante notar que aumentar a frequência de corte implica em uma redução na largura de banda disponível.

O circuito da Figura 10 começa com o fototransistor, que é responsável por converter o sinal ótico num sinal elétrico. Este foto-transistor (PL-51P3C) está em série com um capacitor de 10 nF. Esta configuração em série ajuda a captar e filtrar os sinais óticos recebidos. Em paralelo a estes dois componentes, há uma resistência de 16 k Ω . Essa resistência é importante para limitar a corrente que flui através do circuito e garantir que o sinal seja devidamente processado.

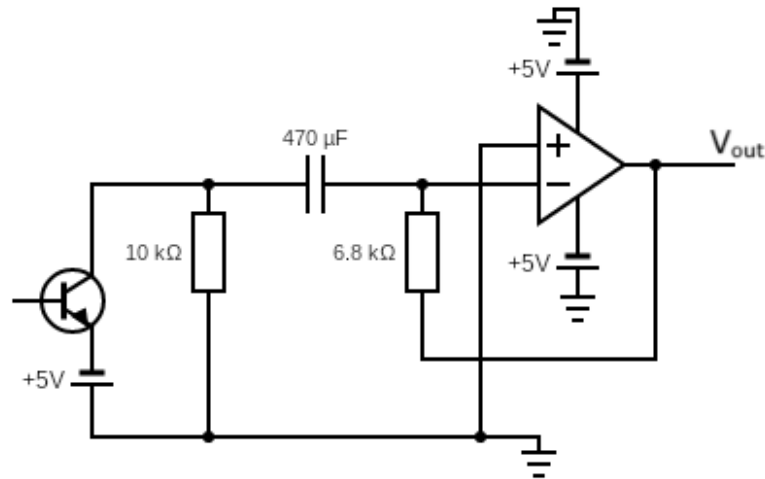


Figura 10: Pinout do recetor, filtro e amplificador transimpedância.

A conexão da resistência de 16 k Ω e do capacitor de 10 nF está ligada a uma resistência de 33 k Ω . Essa ligação é crucial para o funcionamento do amplificador de transimpedância. O amplificador de transimpedância converte a corrente de entrada do foto-transistor numa tensão de saída proporcional. A resistência de feedback (R_f), que é a resistência de 33 k Ω neste caso, é fundamental para determinar a relação entre a corrente de entrada e a tensão de saída, conforme especificado pela relação da expressão (4.5).

4.2. CIRCUITO RECETOR

A resistência de $33\text{ k}\Omega$ também está conectada à saída do AmpOp. O AmpOp amplifica o sinal de saída do circuito para garantir que ele tenha amplitude adequada e possa ser processado corretamente pelo dispositivo de destino, neste caso, a placa ESP32.

As fontes de alimentação VS1 e VS2, com tensões de -3.3 V e 3.3 V , respetivamente, são conectadas às entradas 4 e 7 do AmpOp. Estas fontes de alimentação fornecem energia necessária para o funcionamento do AmpOp.

Por fim, a saída do AmpOp está conectada à resistência de $33\text{ k}\Omega$ e ao V_{out} . Esta saída é onde o sinal elétrico processado e amplificado é para uso posterior.

A segunda parte, Figura 11, presente à saída do amplificador transimpedância, é um comparador não inversor com histerese. Para uma boa leitura por parte da ESP32 e também para a sua segurança, é necessário que o sinal de entrada da placa seja uma onda quadrada, o que nem sempre se verifica à saída do amplificador transimpedância, pois esta varia muito consoante a distância entre o emissor e o recetor. Com o comparador, verifica-se sempre que à saída deste está presente uma onda quadrada, que varia entre -5 V e 5 V (alimentação do AmpOp).

Em consequência ao ruído, não seria ideal usar um comparador centrado em zero, ou seja, quando a tensão de entrada é superior a zero, a tensão de saída é positiva e quando a tensão de saída é inferior a zero, a tensão de saída é negativa, pois, mesmo quando não é recebido um sinal no recetor, o comparador estaria sempre a gerar valores de saída aleatórios, resultantes do ruído. Para evitar este problema, poderá-se simplesmente centrar o comparador a um valor de referência superior ao valor de pico do ruído, assim, a saída do comparador não irá variar consoante o ruído.

Porém, após a realização de alguns testes, observa-se que alterando o valor de referência do comparador afetava demasiado o *duty cycle* e que, como consequência, o sistema não funcionava tão bem para maiores distâncias.

Com esta informação, decidiu-se introduzir histerese ao comparador. Com isto, a saída do nosso comparador é positivo quando o sinal sobe dos 100 mV (superior ao valor de pico do ruído) e é negativo quando desce para valores inferiores a 0 V , como se pode observar na Figura 11.

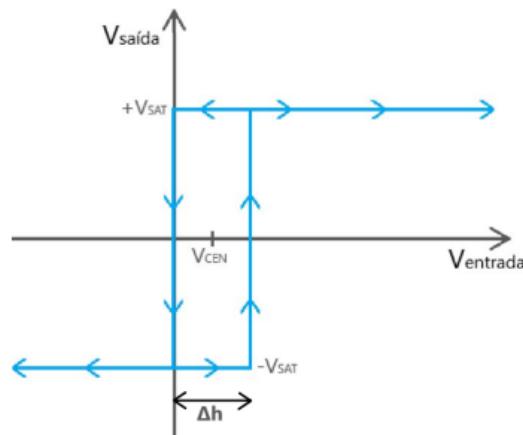


Figura 11: Gráfico da tensão de saída consoante a tensão de entrada do comparador não inversor com histerese.

Foram utilizadas as seguintes fórmulas para projeção das quatro resistências necessárias:

$$\Delta h = 2 \times V_{SAT} \times \frac{R1}{R2} \quad (4.7)$$

$$R2 = 100 \times R1$$

Sendo que $V_{SAT} = 5\text{ V}$ E $\Delta h = 100\text{ V}$.

$$V_{CEN} = \frac{V_{REF}}{2} \quad (4.8)$$

4.2. CIRCUITO RECETOR

$$V_{CEN} = 50 \text{ mV}$$

$$V_{REF} = V_{REF} \times \frac{R1 + R2}{R2} \quad (4.9)$$

$$V_{REF} = 49 \text{ mV} \approx 50 \text{ mV} \approx V_{CEN}$$

Para obtermos o R_{REF} na entrada negativa do AmpOp, recorremos a um divisor de tensão.

$$V_{REF} = 5 \times \frac{R4}{R4 + R3} \quad (4.10)$$

$$R_3 = 100 \times R_4$$

Então obtêm-se os seguintes valores $R_2 = R_2$ e $R_1 = R_{14}$

Na Figura 12, está presente a montagem do comparador não inversor com histerese.

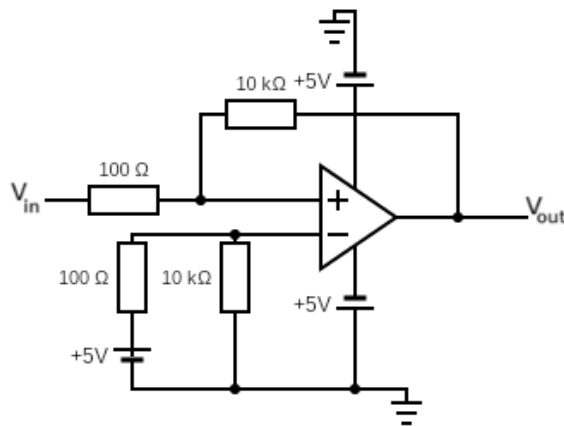


Figura 12: Pinout do Comparador.

Na parte final, Figura 13, onde a saída está ligada diretamente à placa, temos o retificador de meia onda. Este circuito é necessário pois a placa não lê valores de tensão negativas. Como a saída do comparador é uma onda quadrada de valores pico a pico de -5V e 5V, é necessário transformar os valores de -5V em 0V e obtemos esse comportamento com o retificador de meia onda, presente na Figura 13.

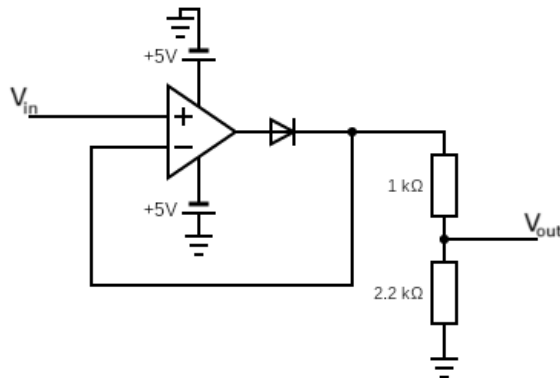


Figura 13: Pinout do Retificador de meia onda.

5 Protocolo de Comunicação

Um protocolo de comunicação é um conjunto de regras e convenções que definem a forma como os dados são transmitidos e recebidos entre dispositivos numa rede. Estas regras garantem a integridade, confiabilidade e eficiência da comunicação, permitindo que os dispositivos se comuniquem de forma organizada e padronizada.

5.1 Data Link Layer

A camada de ligação de dados é fundamental para o estabelecimento e controlo da comunicação entre dispositivos de uma rede. Nesta secção, aborda-se os principais aspetos que compõem essa camada, desde o enquadramento dos protocolos de comunicação até o controlo de erros e fluxo [16].

- **Enquadramento**

Para o enquadramento do protocolo de comunicação, usa-se os seguintes campos, representados na Figura 14:

- **Preâmbulo:** Responsável por marcar o início da trama, assegurando assim a sincronização entre o emissor e o recetor;
- **Número de Sequência:** Valor que indica o pacote atual a ser recebido, usado para a identificação do pacote na sua receção, sendo assim possível o envio de um pacote ACK para a confirmação da mesma. O número de sequência varia de 0 a 255 (1 byte), quando atinge o valor máximo este recomeça a contagem;
- **Endereço de origem/destino:** Identifica o emissor e o recetor, como a comunicação é feita de ponto a ponto não seria necessário endereços, no entanto serão implementados por uma questão de escalabilidade. Serão usados valores inteiros entre 0 e 255 (1 byte) para identificação;
- **Tamanho real de dados:** Como opta-se por um enquadramento de tamanho fixo (para evitar erros que poderiam acontecer em enquadramentos de tamanho dinâmico), foi adicionado um campo que indica o tamanho real do *payload* sem contar com o *padding*, para que este seja mais facilmente excluído na receção do pacote;
- **Dados**
- **CRC:** Bits usados para a deteção de erros através da Verificação cíclica de redundância;

Preâmbulo (8 bytes)	Número de Sequência (1 byte)	Endereço de origem (1 byte)	Endereço de destino (1 byte)	Tamanho real de dados (1 byte)	Dados (128 bytes)	CRC (4 bytes)
------------------------	------------------------------------	-----------------------------------	------------------------------------	--------------------------------------	----------------------	------------------

Figura 14: Enquadramento do Protocolo.

- **Endereçamento**

No endereçamento o emissor é o dispositivo responsável por enviar os dados, enquanto o recetor é o dispositivo que os recebe. Embora a comunicação ocorra diretamente entre estes dois dispositivos e, teoricamente, não seja necessário o uso de endereços, estes serão implementados por uma questão de escalabilidade.

- **Controlo de Erros e de Fluxo**

Para o controlo de erros será usado o mecanismo Stop-And-Wait Automatic Repeat Request (ARQ) e a técnica Cyclic Redundancy Check (CRC).

Este protocolo deteta erros nos dados transmitidos através de *checksums* ou verificações de

5.3. APLICAÇÃO-APLICAÇÃO

- Fim de transmissão - indica ao recetor que a transmissão daquele fluxo de dados terminou.
- Dados - mensagem com os dados que o emissor quer fazer chegar ao recetor, pode ser texto, um ficheiro, etc.

A Figura 16 apresenta a especificação das tramas da camada de aplicação que se encontra encapsulada dentro do campo "Dados" da trama da camada de ligação.

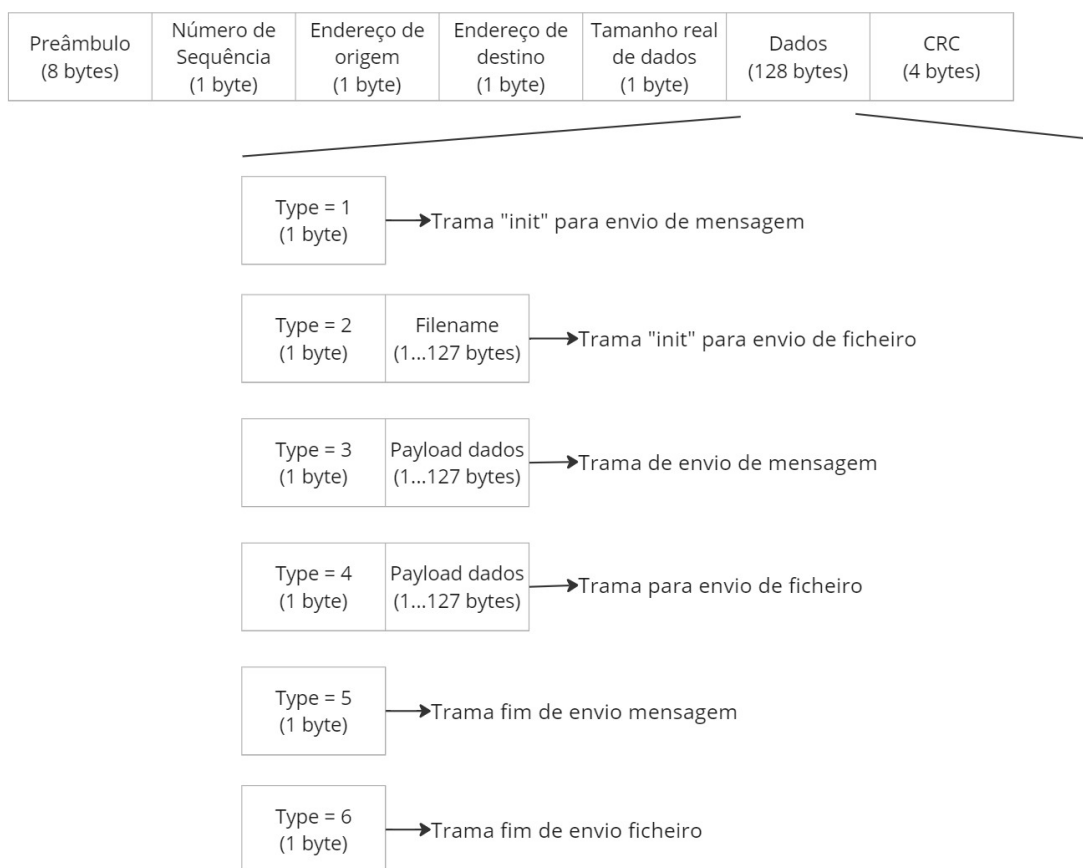


Figura 16: Especificação das Tramas da Camada de Aplicação.

6 Interface Gráfica

Nesta fase do projeto, serão desenvolvidas duas aplicações diferentes, ambas com uma interface gráfica, que permita ao utilizador efetuar a comunicação de uma forma mais intuitiva, sendo uma para o lado do transmissor e outra para o lado do recetor, tendo características a nível de funcionalidades muito diferentes entre elas. Nos pontos abaixo serão explicadas as funcionalidades e as diferenças entre cada programa.

6.1 Aplicação de Transmissão

A aplicação de transmissão, observada na Figura 17 irá permitir enviar dados, como por exemplo, mensagens de texto e até mesmo ficheiros do modulo emissor para o recetor. A aplicação permite também selecionar a porta COM associada à placa ESP32 que se encontra conectada ao PC.

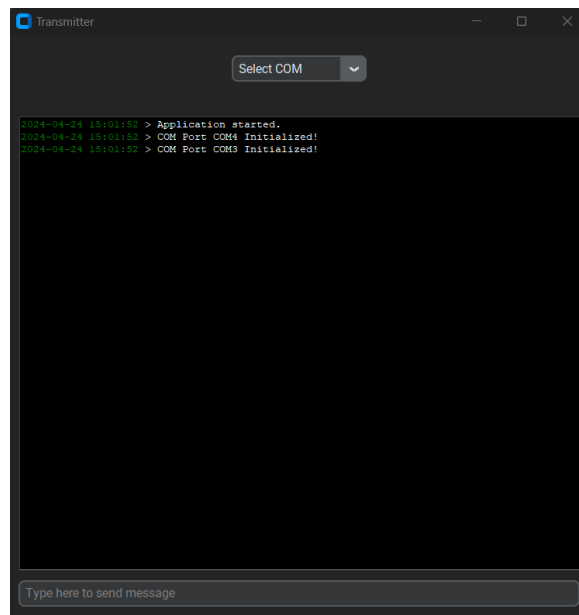


Figura 17: Aplicação do emissor.

No início desta aplicação como observado na Figura 18 é verificada se existe uma ou mais portas COM (Communication), disponíveis que permitam efetuar a comunicação com a placa ESP32, sendo mostrado um popup que apresenta a verificação dessas portas

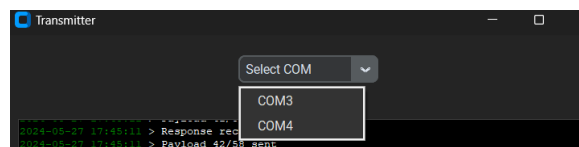


Figura 18: Verificação das portas na aplicação do emissor.

6.2 Aplicação de Receção

A aplicação de receção, observada na Figura 19 apenas irá fazer a receção dos dados enviados a partir da aplicação de transmissão e fazer a correção dos erros dos blocos de dados que lhe chegam. No caso de ser uma simples mensagem, disponibilizar a mesma na caixa de texto da aplicação ou no caso de ser um ficheiro aguardar por todas as tramas correspondentes a ele e disponibilizar o ficheiro ao utilizador. Tal como

6.2. APLICAÇÃO DE RECEÇÃO

na aplicação de transmissão, é também possível selecionar a porta COM associada à placa ESP32 que se encontra conectada ao PC.

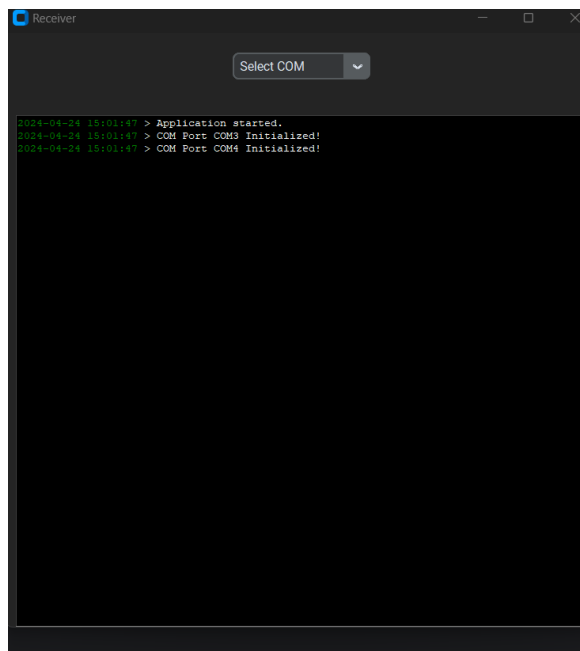


Figura 19: Aplicação do recetor.

7 Testes e Resultados

Nesta secção serão abordados todos os testes efetuados, bem como os resultados obtidos.

Foram efetuados testes que avaliam a distância máxima que se podia atingir sem que houvesse grande distorção do sinal e testes para analisar o tempo de transmissão de ficheiros.

7.1 Distâncias entre os Links

Para este tipo de testes, foram capturadas as formas de onda, com o auxílio do osciloscópio, à entrada do circuito driver e na saída do circuito frontend, a distâncias de 20 cm, 80 cm e 1.20 m. Nestes testes é possível reparar que à medida que se aumenta a distância o duty cycle diminui.

7.1.1 Distância de 20 cm

Os primeiros testes foram efetuados a uma distância de 20 cm e também foram capturadas as formas de onda, nos vários pontos de interesse dos circuitos. As imagens que se seguem são fotos tiradas ao osciloscópio, em que o channel 1 (onda amarela) corresponde ao sinal de entrada e o channel 2 corresponde à forma de onda do ponto de interesse. No circuito driver foi capturada a forma de onda no cátodo do LED, conforme é possível observar na figura 20.

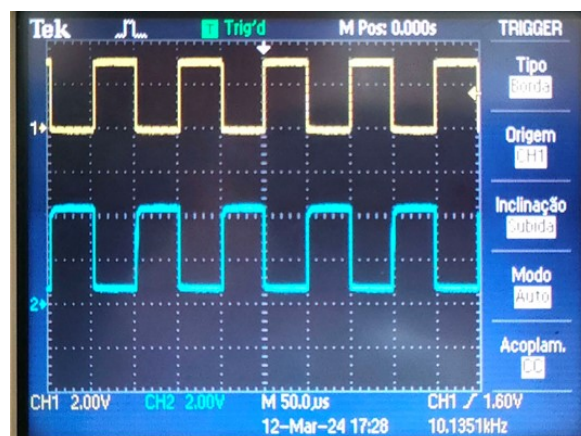


Figura 20: Sinal do cátodo do LED do circuito emissor.

No circuito frontend foram capturadas as formas de onda à saída do amplificador de transimpedância, à saída do comparador e à saída do circuito, pode-se observar nas figuras 21, 22 e 23, respetivamente.

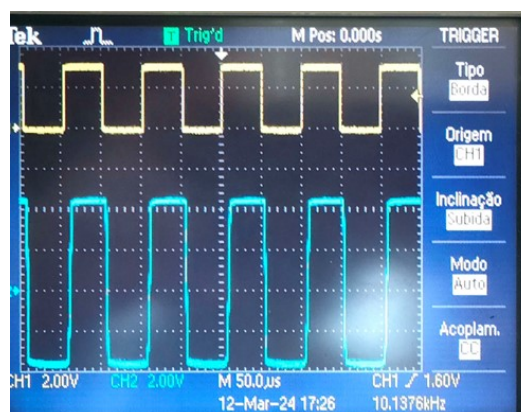


Figura 21: Sinal à saída do amplificador de transimpedância com filtro.

7.1. DISTÂNCIAS ENTRE OS LINKS

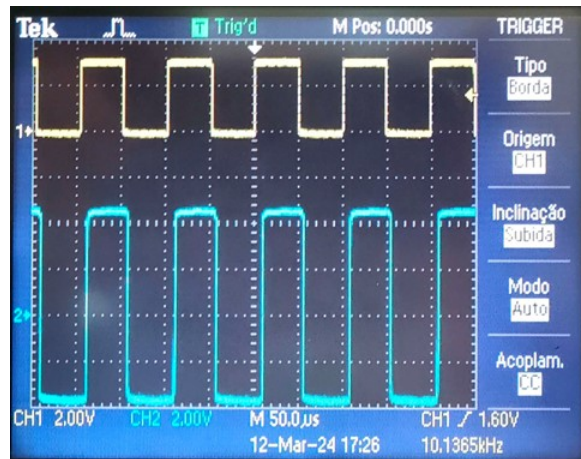


Figura 22: Sinal à saída do comparador com filtro.

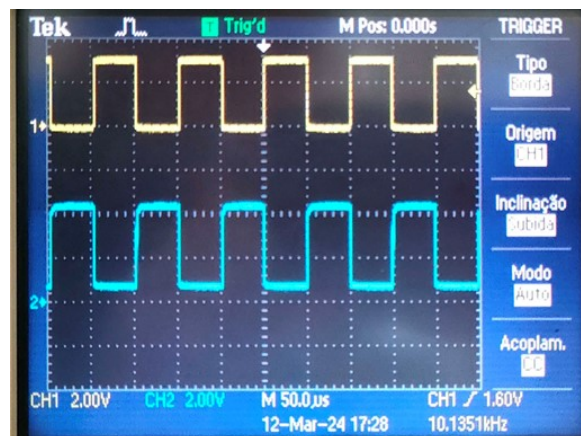


Figura 23: Sinal da saída do circuito frontend.

7.1.2 Distância de 80 cm

Na Figura 24 está registrada a forma de onda à saída do circuito frontend (channel 2) e a forma de onda do sinal de entrada (channel 1), a uma distância de 80 cm.

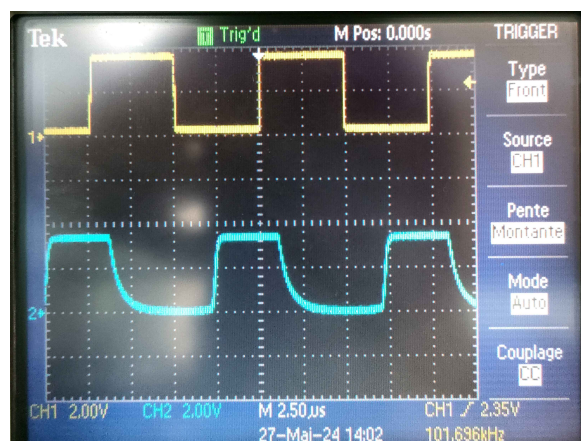


Figura 24: Saída do circuito frontend a 80 cm de distância.

7.2. TESTES DE TEMPO DE TRANSMISSÃO

7.1.3 Distância de 1,2 m

Na Figura 25 está registrada a forma de onda à saída do circuito frontend (channel 2) e a forma de onda do sinal de entrada (channel 1), a uma distância de 1,20 m.

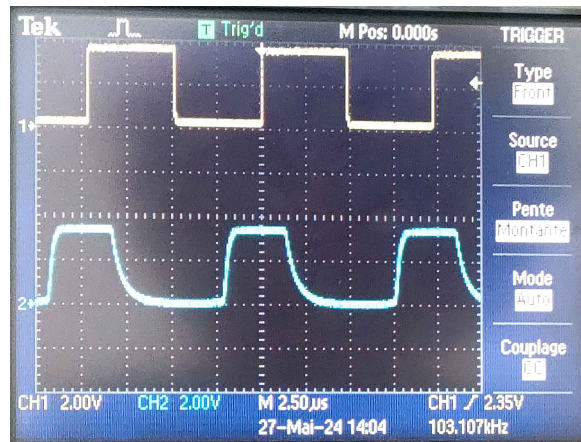


Figura 25: Saída do circuito frontend a 1,20 m de distância.

7.2 Testes de Tempo de Transmissão

A Figura 26 mostra o tempo de transmissão, em segundos, de uma mensagem de texto. O tempo de transmissão foi cerca de 0.081 s.

```
2024-05-27 17:44:43 > Mensagem a ser transmitida
2024-05-27 17:44:43 > Mensagem recebida: teassattsatsatastsa --> tempo total: 0.081
```

Figura 26: Tempo de transmissão de uma mensagem de texto.

A Figura 27 mostra o tempo de transmissão, em segundos, de um ficheiro JPG de 7 KBytes. O tempo de transmissão foi cerca de 10.152 s.

```
2024-05-27 17:45:04 > Ficheiro a ser transmitido
2024-05-27 17:45:14 > Ficheiro recebido: SampleJPGImage_50kbmb.jpg --> tempo total: 10.152
```

Figura 27: Tempo de transmissão de uma imagem.

O cálculo teórico do tempo de envio é dado pela expressão 7.11.

$$T = \frac{n^{\circ} \text{ total de bits}}{\text{taxa de transmissão (s)}} \quad (7.11)$$

Se apenas se considerasse os bytes de informação, o tempo teórico seria cerca de 4s. No entanto, tem de se levar em consideração o tempo dos ACKs introduzido ao longo da transmissão para o cálculo do tempo de transmissão teórico.

Assim, para este caso, o número total de bytes que a imagem possui é de 7000 bytes, como cada trama contém no máximo 128 bytes de informação útil, são geradas 55 tramas ($\frac{7000}{128}$), a cada uma destas tramas, no nível aplicacional, são acrescentados 16 bytes (8 bytes de preâmbulo, 1 byte de número de sequência, 1 byte de endereço de origem, 1 byte de endereço de destino, 1 byte para indicar o tamanho real de dados e 4 byte para o CRC), além disso são adicionados mais 2 bytes para as tramas de INIT e END. Assim sendo, o total de bytes produzidos a nível aplicacional é de 7482 bytes ($2 + 55 \times (128 + 8)$). Deste modo, o número total de bits enviados é de 59856. Para este valor o tempo teórico é de 6,235 s.

7.2. TESTES DE TEMPO DE TRANSMISSÃO

O tempo de transmissão real é bastante superior, isto pode dever-se a alguns fatores tais como: tempo de processamento, o facto do protocolo de comunicação entre PC-ESP32 do lado do emissor, utilizar o mecanismo stop and wait como controlo de fluxo, o que introduz atrasos, e também não está a ser considerado o tempo de envio dos dados via porta série, neste caso foi utilizado um baud rate de 250000.

8 Requisitos Funcionais e Não Funcionais

Neste capítulo, são apresentados os Requisitos Funcionais (RF), Tabela 2, e Requisitos Não Funcionais (RNF), Tabela 3, duas categorias essenciais de especificações que direcionam o desenvolvimento do sistema de software e hardware em questão. Estes requisitos delineiam o que o sistema deve realizar (RF) e as condições nas quais deve operar (RNF).

Tabela 2: Requisitos Funcionais do projeto.

ID	Requisitos Funcionais	Justificação
RF1	Implementação um protocolo da Camada 7 (Camada de Aplicação).	Desenvolver um protocolo para a Camada de Aplicação para permitir a comunicação entre diferentes aplicações.
RF2	comunicação Half-Duploex.	Implementar funcionalidades do protocolo de nível 2 para comunicação half-duplex entre emissor e recetor.
RF3	Criação um programa de conversa e transferência de arquivos	Desenvolver um programa com funcionalidades de conversa e transferência de arquivos entre dois PCs.
RF4	Projeção uma interface gráfica de utilizador (GUI)	Criar uma interface intuitiva e fácil de usar para o programa

Tabela 3: Requisitos Não Funcionais do projeto.

ID	Requisitos Não Funcionais	Justificação
RNF1	Desempenho.	O sistema deve ser capaz de transferir arquivos e mensagens de chat de forma rápida e eficiente.
RNF2	Independência de Camadas.	O sistema deve ser capaz de manter independência entre camadas, garantindo que a camadas 2 não conhece os detalhes da aplicação e vice-versa.
RNF3	Confiabilidade	Garantir a transferência confiável de dados sobre a ligação sem fios, mesmo em caso de interrupção momentânea do feixe ótico.
RNF4	Segurança	Implementar medidas de segurança para proteger a privacidade e integridade dos dados transmitidos.

Para demonstração, o sistema deve conseguir transferir com sucesso um arquivo de imagem e um arquivo de texto entre dois computadores através da camada 2 e da porta série, cumprindo todos os requisitos funcionais e não funcionais estabelecidos.

9 Conclusão

Este projeto tem como objetivo implementar circuitos óticos de sinais infravermelhos, consistindo num emissor e num recetor integrados com dois módulos ESP32, e desenvolver duas aplicações para envio e receção de dados. A conclusão do projeto proporcionou conhecimentos em eletrónica, comunicações óticas e desenvolvimento de software. Demonstrou-se a capacidade de modular a intensidade luminosa de um LED infravermelho para transmitir informações e reconstruir o sinal no recetor, utilizando componentes eletrónicos adequados e um protocolo de comunicação eficaz para garantir o sincronismo.

O desenvolvimento das aplicações resultou numa interface amigável, permitindo o envio e a receção de mensagens de texto e ficheiros, semelhante a aplicações de chat.

No entanto, enfrentou-se desafios como a sensibilidade dos circuitos óticos a interferências, a necessidade de posicionamento preciso dos componentes, taxas de transmissão limitadas e a comunicação unidirecional que requer códigos de correção de erros. Apesar destas limitações, o sistema mostrou-se eficaz para aplicações como controlo remoto de dispositivos a curta distância.

Em resumo, este projeto permitiu a aplicação de conceitos teóricos de outras unidades curriculares, o aprimoramento de competências pré-existentes e a aquisição de novas competências. Apesar das limitações, o desenvolvimento foi bem-sucedido e o sistema apresenta potencial para aplicações práticas em contextos específicos.

10 Autoavaliação

Este capítulo apresenta uma análise crítica do desempenho dos membros da equipa, destacando as contribuições em diferentes áreas do projeto. A Tabela 6 exibe uma avaliação quantitativa, onde são atribuídos percentuais de realização para cada etapa do desenvolvimento, desde a implementação dos circuitos até a entrega de relatórios e apresentações. Esta avaliação proporciona uma visão geral do progresso individual e do grupo, permitindo identificar áreas de melhoria e reforçar pontos fortes para o sucesso do projeto.

Tabela 6: Autoavaliação de cada elemento do grupo: Catarina Pereira (CP); Catarina Amorim (CA); Inês Neves (IN) e Leonardo Martins (LM).

	CP	CA	IN	LM
Implementação do Circuito Driver	20%	15%	30%	35%
Implementação do Circuito Frontend	20%	15%	35%	35%
Definição de protocolos de comunicação	25%	25%	25%	25%
Implementação de códigos ESP32 emissora	20%	15%	20%	45%
Implementação de códigos ESP32 recetora	20%	15%	20%	45%
Implementação da aplicação emissora	25%	20%	25%	30%
Implementação da aplicação recetora	25%	20%	25%	30%
Relatórios	35%	15%	35%	15%
Apresentação	30%	20%	30%	20%

Referências Bibliográficas

- [1] Kanchan Tiwari Swami e Asmita A. Moghe. "A Review of LiFi Technology". Em: *2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE)*. 2020, pp. 1–5. DOI: 10.1109/ICRAIE51050.2020.9358340.
- [2] Jasmine Mannil Abraham, Hardeep Kumar e G. Josemin Bala. "Li-Fi: Illuminating the Future of Internet". Em: *2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS)*. 2020, pp. 550–554. DOI: 10.1109/ICIIS51140.2020.9342641.
- [3] Abhimanyu Pandit. *Serial Communication Protocols*. <https://circuitdigest.com/tutorial/serial-communication-protocols>. Acedido em 06 de fevereiro de 2024. Abril de 2019.
- [4] *Serial Communication Protocols: The Basics*. <https://www.totalphase.com/blog/2017/08/serial-communication-protocols-the-basics/>. Acedido em 06 de fevereiro de 2024. Agosto de 2017.
- [5] *Serial Communication*. <https://learn.sparkfun.com/tutorials/serial-communication/all>. Acedido em 06 de fevereiro de 2024. Agosto de 2017.
- [6] Syed V. Ahamed e Victor B. Lawrence. "The Role of the OSI Model". Em: Springer US, 1997, pp. 94–123. DOI: 10.1007/978-1-4615-6341-9_4.
- [7] Sushmita Biya e Renuka Uday Kotwal. "The OSI Model: Overview of All Seven Layers of Computer Networks". Em: *International Journal of Advanced Research in Science, Communication and Technology* (set. de 2023), pp. 427–432. ISSN: 2581-9429. DOI: 10.48175/IJARST-13064.
- [8] Peter Boait et al. "Data Link Layer — Layer 2". Em: Macmillan Education UK, 1988, pp. 62–89. DOI: 10.1007/978-1-349-10306-5_5.
- [9] Gerry Howser. "The OSI Seven Layer Model". Em: Springer International Publishing, 2020, pp. 7–32. DOI: 10.1007/978-3-030-34496-2_2.
- [10] Pradeep Kumar Srivastava e Sandhya Tiwari. "An Overview of Open System Interconnection (OSI): A Seven Layered Model". Em: 2017. URL: <https://api.semanticscholar.org/CorpusID:113631165>.
- [11] Espressif Systems. *ESP32 Series Datasheet*. https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf. Acedido em 13 de fevereiro de 2023. Jan. de 2023.
- [12] <https://www.botnroll.com/pt/arduino-controladores/3540-esp32-placa-de-desenvolvimento-wifi-bluetooth-esp-32s-ai-thinker.html>. Acedido em 15 de fevereiro de 2023.
- [13] BangGood. https://pt.banggood.com/ESP32-Development-Board-WiFi+bluetooth-Ultra-Low-Power-Consumption-Dual-Cores-ESP-32-ESP-32S-Board-Geekcreit-for-Arduino-products-that-work-with-official-Arduino-boards-p-1109512.html?cur_warehouse=CN. Acedido em 13 de fevereiro de 2023.
- [14] *UA741 General-purpose single operational amplifier*. <https://datasheetspdf.com/pdf/1404688/STMicroelectronics/UA741CN/1>. Acedido em 08 de fevereiro de 2024. Setembro de 2013.
- [15] *7404 - 7404 Hex Inverter Datasheet*. <https://www.futurlec.com/74/IC7404.shtml>. Acedido a 27 de Maio de 2024. 2024.
- [16] José Augusto Afonso. "Data Link Layer". Em: *Blackboard* (2023). Acedido em 20 de março de 2024.