

# U.C. de Projeto Integrador em Telecomunicações e Informática

Ano Letivo: **2022/2023**



## Especificação da Fase A

### Grupo 5

- Luís Oliveira, a89380
- Francisco Martins, a93079
- José Gomes, a93083
- Rui Cunha, a93093
- Diogo Cerqueira, a93108

28/02/2023

## Índice

1. Índice de figuras.....	3
2. Lista de siglas e acrónimos.....	4
3. Introdução.....	5
4. Contextualização da Fase A.....	6
5. Algoritmos do sistema.....	7
5.1. Algoritmo do Emissor.....	7
5.2. Algoritmo do Recetor.....	8
6. Protocolo de comunicação.....	9
7. Ferramentas.....	10
7.1. Hardware.....	10
7.2. Software.....	12
8. Planificação do projeto.....	14
9. Conclusão.....	15
10. Bibliografia.....	16

# 1.Índice de figuras

Figura 1 - Arquitetura da Fase A.....	6
Figura 2 - Algoritmo do emissor. ....	7
Figura 3 - Algoritmo do recetor. ....	8
Figura 4 - Estrutura de um datagrama de acordo com o protocolo definido.....	9
Figura 5 - Diagrama de Gantt.....	14

## 2. Lista de siglas e acrónimos

<b>PC</b>	<i>Personal Computer</i>
<b>BLE</b>	<i>Bluetooth Low Energy</i>
<b>Wi-Fi</b>	<i>Wireless Fidelity</i>
<b>ISP</b>	<i>Internet Service Provider</i>

### 3.Introdução

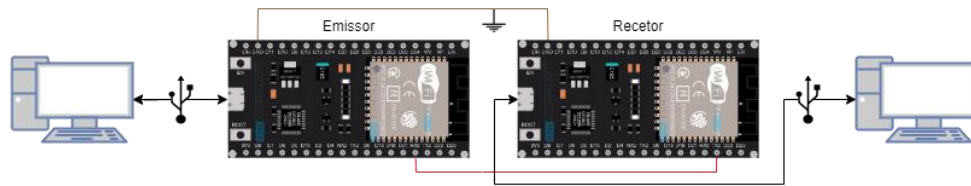
O relatório de especificação está inserido no âmbito da Unidade Curricular de Projeto Integrador em Telecomunicações e Informática.

Serve o presente relatório como introdução e descrição dos critérios de *hardware* e de *software* a serem implementados na fase A, em junção com a apresentação e planeamento temporal das tarefas convenientes à sua construção.

O relatório referido será iniciado com as especificações impostas pela fase A, nomeadamente: arquitetura, requisitos e funcionalidades do sistema; identificação das tecnologias necessárias e planificação horária. Seguido de uma ligeira conclusão.

## 4.Contextualização da Fase A

A figura 1 ilustra a arquitetura solicitada pela fase A.



*Figura 1 - Arquitetura da Fase A.*

A fase A concentra-se na transmissão unidirecional de dados, ou seja, no fluxo de *bits* em sincronismo entre as placas ESP32, com auxílio a fios de ligação.

De forma a assegurar uma transmissão fiável e eficaz é pertinente a definição de protocolos de comunicação (baseados em tramas), quer entre cada PC (*Personal Computer*) e a respetiva placa ESP32, quer para a comunicação entre placas ESP32.

Em adição à conceção dos formatos das tramas, os protocolos referidos devem incorporar mecanismos de deteção e correção de erros e mecanismos inerentes, e se necessário, incluir mecanismos de controlo de fluxo.

Para além do estabelecimento de diversos protocolos, é oportuno o desenvolvimento de mecanismos de codificação e decodificação dos *bits* e de mecanismos responsáveis pela sincronização do fluxo de *bits* entre as placas referidas, em especial, a nível do *bit*, do *byte* e da trama.

## 5. Algoritmos do sistema

### 5.1. Algoritmo do Emissor

Na figura 2, é possível observar o algoritmo incorporado pelo emissor.

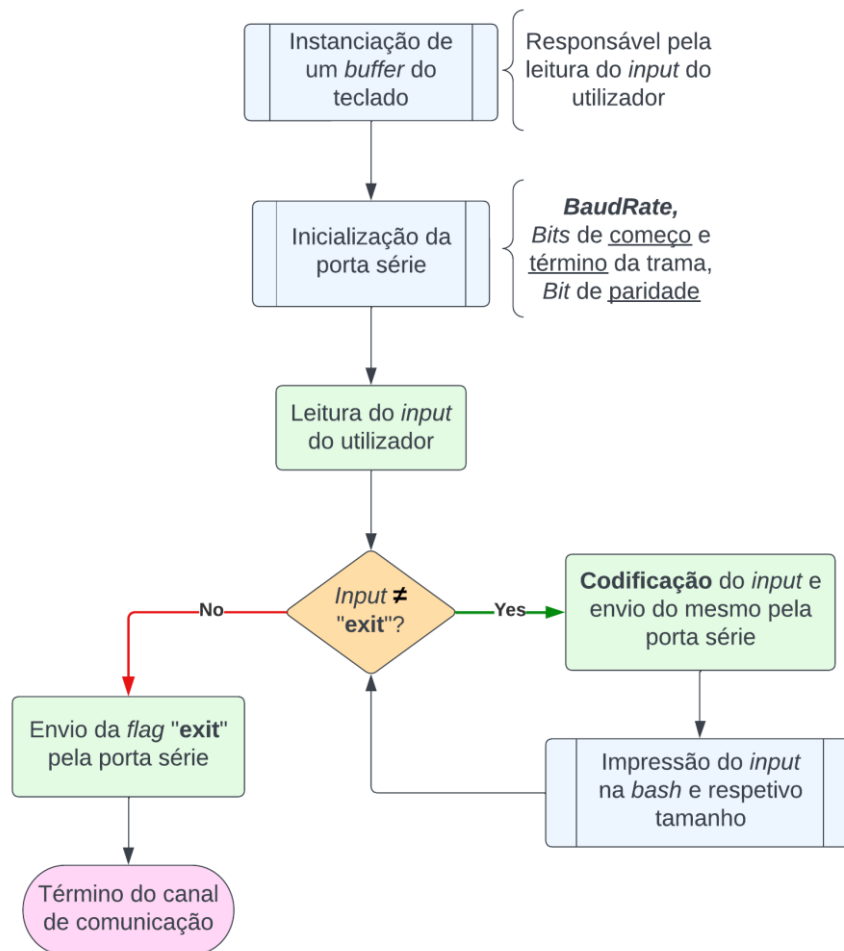


Figura 2 - Algoritmo do emissor.

## 5.2. Algoritmo do Recetor

Seguidamente, na figura 3, é ilustrado o algoritmo integrado pelo recetor.

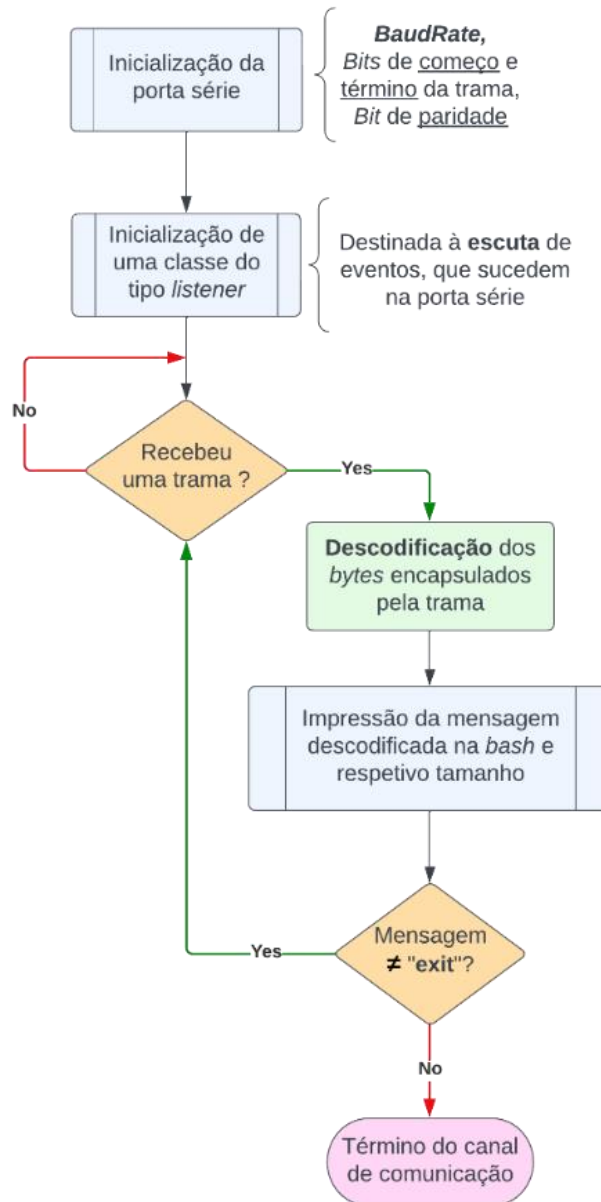


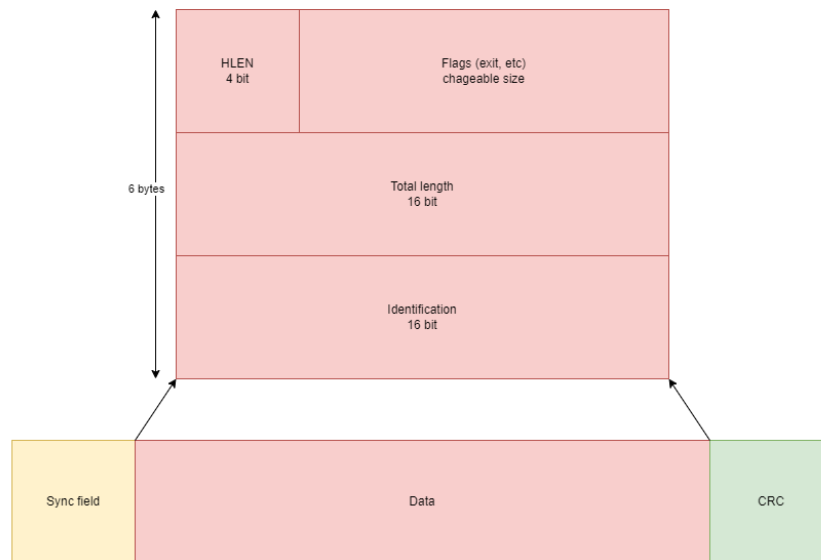
Figura 3 - Algoritmo do recetor.



## 6. Protocolo de comunicação

A figura 4 descreve a fisionomia de um datagrama, que se enquadra no comportamento do protocolo de comunicação concebido pelo grupo.

O protocolo concebido implica a codificação dos datagramas, antes dos mesmos serem encaminhados pela porta série. Tal como a figura sugere, o datagrama consiste na combinação de três campos essenciais: o campo de sincronização; o campo de dados e o campo destinado à correção de erros.



*Figura 4 - Estrutura de um datagrama de acordo com o protocolo definido.*

- **Sync field:** Campo responsável pela sincronização dos pacotes.
- **Data:** Campo que armazena a mensagem transmitida.
- **CRC:** Campo utilizado para efetuar o controlo de erros.






## 7.Ferramentas

Para a realização deste projeto, o grupo dispõe de vários recursos de forma a ser possível a finalização do projeto prático com sucesso. Parte dos recursos a serem utilizados podem ser classificados como físicos, constituindo desta forma a componente *hardware* a ser usufruída, e os restantes sem formato físico, enquadram-se no domínio do *software*.

### 7.1. Hardware

A tabela seguinte apresenta todo o *hardware* necessário ao desenvolvimento da fase A.

*Tabela 1 - Hardware relativo à fase A.*

Imagem	Designação	Descrição
	ESP32-DevKitC-32D	2 placas: uma para a implementação do sistema sensor e outra para o gateway BLE/Wi-Fi
	Cabos USB	Cabos responsáveis pela conexão entre o PC e a placa ESP32
	Fios de ligação	Fios responsáveis pela conexão dos componentes necessários
	Computador	Desenvolvimento do código e relatórios
	<i>Breadboard</i>	Interface de conexão entre os circuitos

### 7.1.1 ESP32-DevKitC-32D

O ESP32 é um módulo genérico de Wi-Fi (*Wireless Fidelity*), *Bluetooth* e BLE (*Bluetooth Low Energy*), que possibilita uma grande variedade de aplicações como por exemplo redes de sensores de baixa potência, decodificação de MP3 e codificação de voz.

A tabela 2 apresenta as características da placa em questão.

*Tabela 2 - Características da placa ESP32-DevKit-32D.*

Módulo	ESP32-DevKit-32D
SPI <i>flash</i>	32 Mbits, 3.3 V
Core	ESP32-D0WD
Crystal	40 MHz (apenas para a funcionalidade do Wi-Fi e do <i>Bluetooth</i> )
Antena	Conector U.FL (que precisa de estar conectado a uma antena IPEX externa)
Dimensões (Unidade: mm)	(18.00±0.10) × (19.20±0.10) × (3.20±0.10)

O *Bluetooth*, o BLE e o Wi-Fi são diferentes tipos de comunicação, que são suportados pelo ESP32. A utilização do Wi-Fi proporciona um grande alcance físico e conexão direta com a *internet* através de um ISP (*Internet Service Provider*).


A *sleep current* (corrente característica, quando a placa opera em modo *standby*) da placa é inferior a 5 microamperes, o que torna este componente ideal para sistemas eletrónicos alimentados por baterias.

O *chip* da placa suporta uma taxa de transmissão até 150 Mbps e 20 dBm de potência de saída de antena para proporcionar o melhor alcance físico possível.

## 7.2. Software

A tabela seguinte contém o *software* necessário à implementação da fase A.



*Tabela 3 - Software relativo à fase A.*

Símbolo	Designação	Função
	Arduino IDE	Programação do módulo Arduino
	Microsoft Word	Editor de texto
	IntelliJ IDEA	Ambiente de desenvolvimento integrado
	Facebook	Comunicação entre os membros do grupo
	Opera	Motor de pesquisa

### 7.2.1 Linguagens de Programação

A tabela seguinte contém as linguagens de programação, que se serão necessárias ao desenvolvimento da fase A.

*Tabela 4 - Linguagens de programação necessárias à implementação da fase A.*

Linguagens de programação		
	C++	A linguagem C++ será empregue no desenvolvimento das funções <i>setup()</i> e <i>loop()</i> de ambas as placas.
	Java	A linguagem Java será aplicada na implementação dos algoritmos do emissor e recetor.

# 8. Planificação do projeto

Para garantir consistência, linearidade e para que se cumpra todos os objetivos é necessário recorrer a um planeamento bem estruturado. O planeamento do projeto encontra-se ilustrado pelo diagrama de *Gantt*, representado pela figura 5.

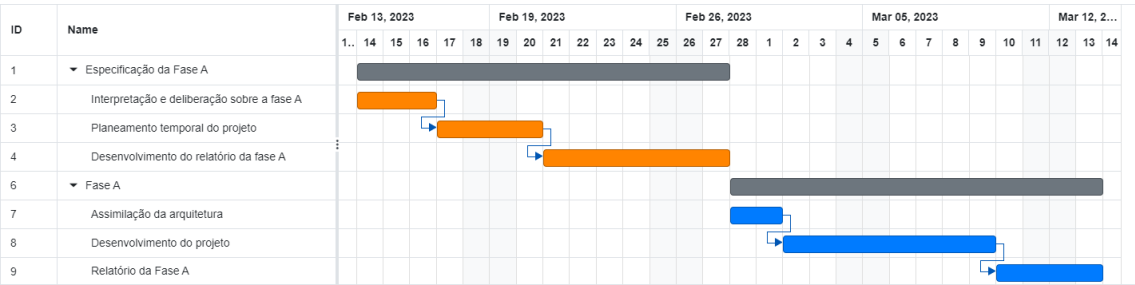


Figura 5 - Diagrama de Gantt.

## 9. Conclusão

Concluída esta exposição, na opinião do grupo, espera-se desta fase um nível de complexidade moderada, especialmente na fase de *design* inicial, visto que se ambiciona, desde o início, eficiência e modularidade para que o resto do projeto seja mais suave.

Assim sendo, acreditamos que o planeamento apresentado irá resultar numa aplicação eficaz do *hardware* e do *software*, para que demonstremos adequadamente conhecimentos a adquirir no decorrer da Unidade Curricular.

## 10. Bibliografia

- Bot n Roll*. (20 de Fevereiro de 2022). Obtido de Bot n Roll: [https://www.botnroll.com/1219-medium\\_default/sensor-de-temperatura-e-humidade-dht11.jpg](https://www.botnroll.com/1219-medium_default/sensor-de-temperatura-e-humidade-dht11.jpg)
- Bot n Roll. (22 de 2 de 2022). *botnroll*. Obtido de [https://www.botnroll.com/8958-medium\\_default/sensor-de-press-o-atmosf-rica-bmp280.jpg](https://www.botnroll.com/8958-medium_default/sensor-de-press-o-atmosf-rica-bmp280.jpg)
- botnroll. (2 de 2 de 2022). Obtido de botnroll: [https://www.botnroll.com/8958-medium\\_default/sensor-de-press-o-atmosf-rica-bmp280.jpg](https://www.botnroll.com/8958-medium_default/sensor-de-press-o-atmosf-rica-bmp280.jpg)
- dfrobot. (20 de Fevereiro de 2022). Obtido de dfrobot: [https://image.dfrobot.com/image/data/DFR0067/DFR0067\\_DS\\_10\\_en.pdf](https://image.dfrobot.com/image/data/DFR0067/DFR0067_DS_10_en.pdf)
- Espressif Systems. (21 de Fevereiro de 2022). *ESP32 Series*. Obtido de [www.espressif.com](http://www.espressif.com): [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf)
- fnac-static*. (20 de Fevereiro de 2022). Obtido de <https://static.fnac-static.com/multimedia/Images/PT/NR/67/05/62/6423911/1540-1.jpg>
- lotone*. (22 de Fevereiro de 2022). Obtido de [https://www.iotone.com/files/vendor/logo\\_Thingspeak.jpg](https://www.iotone.com/files/vendor/logo_Thingspeak.jpg)
- Sistema de Monitorização de Estações Meteorológicas*. (20 de Fevereiro de 2022). Obtido de Blackboard: [elearning.uminho.pt](http://elearning.uminho.pt)
- sparkfun*. (20 de Fevereiro de 2022). Obtido de <https://cdn.sparkfun.com/assets/parts/3/3/4/5/09567-01-Working.jpg>