



Instituto Politécnico De Tomar

Escola Superior de Tecnologias de Tomar

PARAMONITORING

MONITORIZAÇÃO DE DADOS BIOMÉTRICOS E DE LOCALIZAÇÃO
DE PARAQUEDISTAS EM OPERAÇÃO MILITAR

Relatório de Projeto

Trabalho Final de Mestrado

Pedro Miguel Pinheiro de Matos

Mestrado em Engenharia Informática

Internet das Coisas

TOMAR/ NOV/ 2018



Instituto Politécnico De Tomar

Escola Superior de Tecnologia de Tomar

Pedro Miguel Pinheiro De Matos

PARAMONITORING

MONITORIZAÇÃO DE DADOS BIOMÉTRICOS E DE LOCALIZAÇÃO
DE PARAQUEDISTAS EM OPERAÇÃO MILITAR

Relatório de Projeto

Trabalho Final de Mestrado

Orientado por:

Prof. Pedro Correia, IPT

Prof. Luís Oliveira, IPT

Relatório de Projeto de Trabalho Final de Mestrado apresentado ao Instituto Politécnico de Tomar para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Informática-Internet das Coisas

*Agradeço à minha família e em especial ao meu filho
Miguel e a minha mulher Cláudia pelo apoio
constante e incondicional durante esta investigação.*

RESUMO

As operações militares de paraquedismo com saltos a 35000 pés de altitude são realizadas em condições extremas e no limite da sobrevivência. Atualmente são usados métodos empíricos para controlo dos sinais vitais, baseados na observação e experiência dos operacionais.

O trabalho desenvolvido neste projeto consistiu em criar uma infraestrutura de comunicação com tecnologias Low Power Wide Area Network (LPWAN), que permita a transmissão dos sinais biométricos, ambientais e de localização recolhidos. São gerados sinais de alarme em caso de emergência, é realizado o seguimento da localização de cada nó em tempo real, assim como a transmissão dos dados que são guardados para uma posterior análise especializada.

Foi desenvolvida uma rede de sensores com um protótipo composto por três nós, um por cada paraquedista e por um gateway localizado remotamente. Cada nó de sensores tem a capacidade de aquisição de dados de oximetria, da pressão barométrica e da temperatura no ambiente envolvente, assim como das coordenadas de geolocalização. O sistema desenvolvido permite o envio de alertas para os outros nós, assim como o envio de mensagens dos nós para o gateway. O gateway tem a capacidade de realizar a gestão dos utilizadores, de receber e guardar os dados de cada um, assim como permitir a monitorização em tempo real da localização de cada nó em mapas offline.

Foram realizados testes em laboratório e em ambiente real. Os resultados em laboratório obtidos foram bastante satisfatórios e com grande potencial, demonstrando uma grande estabilidade dos dispositivos desenvolvidos. Os testes em ambiente real foram bastante esclarecedores quanto às limitações do hardware desenvolvido tendo em conta as condições severas de operação. No entanto, os resultados foram bastante conclusivos sobre as possíveis abordagens para o desenvolvimento de uma versão do protótipo mais robusta.

Palavras chave: IoT em aplicações militares, Sensores Biométricos, Geolocalização, LoRaWAN, MySQL, NodeJS.

ABSTRACT

Military skydiving operations with 35,000-foot-high jumps are performed under extreme conditions and survival limits. Empirical methods for vital signs control are currently used, based on observation and operational experience.

The work developed in this project consisted in creating a communications infrastructure with Low Power Wide Area Network (LPWAN) technologies, which allows the transmission of biometric, environmental and localization signals. Alarm signals are generated in case of an emergency, the tracking of the location of each node in real time is carried out, as well as the transmission of the data, which are stored for further specialized analysis.

A sensor network was developed, using a prototype composed of three nodes, one for each user and a gateway, located remotely. Each sensor node has the ability to acquire oximetry data, barometric pressure, and temperature in the surrounding environment, as well as the geolocation coordinates. The developed system allows sending alerts to the other nodes, and also sending messages from the nodes to the gateway. The gateway has the ability to perform the user management, receive and save the data of each one, and also, allows real-time monitoring of the location of each node based on offline maps.

The tests were performed in the laboratory and in a real environment. The obtained results in the laboratory were quite satisfactory and with great potential, showing the stability of developed devices. The tests in the real environment were very informative about the limitations of the hardware developed taking into account the hard operating conditions. However, the results were quite conclusive on possible approaches for the development of an improved and robust prototype version.

Key words: IoT in military applications, Biometric Sensors, Geolocation, LoRaWAN, MySQL, NodeJS.

AGRADECIMENTOS

Em primeiro lugar quero manifestar o meu profundo reconhecimento à minha **família** pelo apoio, pelo tempo que estive ausente, no tempo e no espaço, para poder-me dedicar a mais uma investigação no desenvolvimento na área da informática.

Quero também, agradecer aos meus orientadores, **Professor Pedro Correia** e **Professor Luís Oliveira**, pelo tempo que dedicaram a orientar-me nesta investigação. Respostas constantes, correções atempadas e caminhos abertos no conhecimento de uma tecnologia ainda nascente.

Também agradeço o empenho e o ensino da Diretora do Mestrado de engenharia informática **Professora Ana Cristina Lopes**, pelo empenho e dedicação que sempre dispensou aos alunos em formação. Deixando marcas profundas no ensino do Latex, onde este relatório foi escrito.

O meu reconhecimento e agradecimento ao **Professor Pedro Neves**, pelo seu tempo e dedicação no desenvolvimento da placa de ligação ente os NÓS de Sensores.

Agradecimento ao Regimento de Paraquedistas pelas facilidades concedidas e em particular aos meus companheiros de armas, **Sargento-Ajudante Nuno Santos**, **Sargento-Chefe Modesto**, **Sargento-Ajudante Ramalho** e **Sargento-Ajudante Pereira**, que efetuaram o salto em paraquedas com o equipamento, podendo assim testar os NÓS de sensores.

O tempo voa e as grande reflexões fazem a humanidade evoluir, o **Instituto Politécnico de Tomar** e em particular a **Escola Superior de Tecnologias de Tomar**, tem sem duvida, contribuindo para o grande avanço tecnológico que se vem verificando.

ÍNDICE

Resumo	v
Abstract	vii
Agradecimentos	ix
Acrónimos	xv
Índice de Figuras	xvi
1 Introdução	1
1.1 Contexto	1
1.2 Enquadramento tecnológico	2
1.3 Motivação e objetivos	3
1.4 Descrição da solução proposta	4
1.5 Organização do documento	5
2 Revisão da Literatura	7
2.1 A IoT no contexto Mundial	7
2.2 Redes de Sensores Sem Fios	10
2.2.1 Aplicações das RSSF	11
2.2.2 Segurança em RSSF	14
2.3 Tecnologias de Redes Sem Fios	15
3 Tecnologia LoRa e protocolo LoRaWAN	19
3.1 Tecnologia LoRa®	19
3.1.1 LoRaWAN	22
3.1.2 Segurança LoRaWAN	24
3.1.3 Dispositivos	25
3.1.4 Módulos certificados para LoRa®	27
3.1.5 Alcance em LoRa - Cooking-hacks	28

3.1.6	Trabalhos Relacionados	29
4	Sistema de Aquisição de dados Biométricos e de Localização - GateWay	31
4.1	Arquitetura Geral	31
4.2	Gateway Server	33
4.2.1	Hardware	33
4.2.2	Software	34
4.2.3	Front-End	42
4.2.4	Inserir dados dos SDCards	47
5	Sistema de Aquisição de dados Biométricos e de Localização - NÓ de Sensores	49
5.1	Arquitetura Geral do NÓ de Sensores	49
5.2	Hardware	50
5.2.1	Módulo de processamento e sensores	51
5.2.2	Arquitetura de Hardware	57
5.2.3	Circuito desenvolvido	58
5.3	Software	59
5.3.1	Módulo de comunicações (MASTER)	60
5.3.2	Slave (Sensores)	65
5.3.3	microSD	68
6	Testes e Resultados	71
6.1	Teste no Desenvolvimento	71
6.1.1	Teste na leitura dos sensores	71
6.1.2	Teste de comunicação entre Master-Slave	73
6.1.3	Teste de envio (TX) e recepção (RX) de SOS	73
6.1.4	Teste de envio NÓ de sensor - Geteway	74
6.1.5	Testes de Alcance em LoRa	74
6.2	Teste em Laboratório	76
6.2.1	Teste de energia do NÓ de sensores	77
6.2.2	Comunicação entre NÓS com SOS	78
6.2.3	Comunicação entre NÓS e GateWay	79
6.2.4	Envio de coordenadas de GPS para Gateway	81
6.3	Teste em Ambiente real	82
6.3.1	Resultados obtidos como um NÓ de Sensor - 1ºSALTO	82
6.3.2	Resultados obtidos com um NÓ de Sensores - 2ºSALTO	83

6.3.3	Resultados obtidos como três NÓS de Sensores - 3ºSALTO . . .	85
6.3.4	Resultados obtidos como três NÓS de Sensores - 4ºSALTO . . .	87
6.3.5	Resultados obtidos como três NÓS de Sensores - 5ºSALTO . . .	89
7	Conclusões e trabalho futuro	95
7.1	Conclusões	95
7.2	Trabalho futuro	96
	Bibliografia	97

ACRÓNIMOS

4G	Quarta Geração Telecomunicação móvel
CDMA	Code Division Multiple Access
CSS	Chirp Spread Spectrum
CT	concurrent transmission
DARPA	Defense Advanced Research Projects Agency
Dash7	DASH7 Alliance Protocol
DSN	Distributed Sensor Network
Front-End	Aplicação Web do lado do utilizador
FSK	Frequency Shift Keying
GPS	sistema de posicionamento global
GSM	Global System for Mobile Communications
HAHO	High Altitude-High Opening
HALO	High Altitude-Low Opening
HSPA	High Speed Packet Access
IdC	Internet das Coisas
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
LAN	Local Area Network
LoRa	longa distância e baixa potência sem fio

LPWAN	Low Power Wide Area Network
LTE	Long Term Evolution
M2M	Machine-to-Machine
MAN	Metropolitan Area Network
MEMS	Micro Eletro Mecânico
MySQL	open-source relational database management system
NB-IoT	Narrow Band IoT
NOAA	National Oceanic and Atmospheric Administration
PI	Ponto de Impacto
RF	Radio Frequência
RSSF	Redes de Sensores Sem Fios
SAA	Salto de Abertura em Automático
SensIT	Sensor Information Technology
SNS24	Serviço Nacional de Saúde 24 Horas por dia
SISop	Sistema de Informação de Saltadores Operacionais
SO FAR	sound fixing and ranging channel
SOGA	Saltadores Operacionais de Grande Altitude
SOSUS	Sound Surveillance System
SpO2	Saturação periférica de Oxigénio
TO	Teatro de Operações
WAN	Wide Area Network
WLAN	Wireless Local Area Network
WMAN	Wireless Metropolitan Area Network
WPAN	Wireless Personal Area Network
WSN	wireless sensor networks
WWAN	Wireless Wide Area Network

ÍNDICE DE FIGURAS

1.1	Crachá SOGA	1
1.2	Saltadores Operacionais de Grande Altitude	3
1.3	Esquema geral da arquitetura proposta.	5
2.1	Análise da procura no Google do termo "Wireless Sensor Networks"[1].	8
2.2	Análise da procura no Google do termo "internet of things"[2].	8
2.3	IoT 5 a 10 Anos	9
2.4	Enxame de Drones (UAV)	14
2.5	Segurança na RSSF [3].	15
3.1	LoRaWAN	20
3.2	Comparação entre tecnologias [4].	21
3.3	Camada lógica da Rede (LoRaWAN) [5]	22
3.4	Exemplo de Up-chirp [6]	23
3.5	Quadro de Frequências LoRa [7]	24
3.6	Estrutura do Canal 868 MHz [8]	24
3.7	AES em LoRaWAN [9]	25
3.8	Exemplo de Comunicação NÓ - Gateway [10]	26
3.9	Adafruit [11].	27
3.10	Módulo SigFox [12]	27
3.11	Módulo Libelium [13].	28
3.12	Modelos certificados para LoRa	28
3.13	Modulo Libelium SX1272 [14]	29
4.1	Arquitetura Geral	32
4.2	Placa Raspberry PI 3B	33
4.3	Display Touchscreen 7".	34
4.4	Representação do Modulo Libelium.	34
4.5	Arquitetura do Gateway (Raspberry PI).	35

4.6	Estrutura da Base de Dados.	36
4.7	Algoritmo Gateway.	37
4.8	Algoritmo Backup.	38
4.9	Ficheiros de Backup.	40
4.10	Fluxograma de Arranque	42
4.11	Início da aplicação	42
4.12	Página de navegação	43
4.13	CRUD de utilizadores	43
4.14	Lista do Utilizador por salto	44
4.15	CRUD de NÓS	44
4.16	CRUD de NÓS	45
4.17	Listagem da atividade	45
4.18	NÓS no Mapa	46
4.19	API backup SD	47
4.20	Front-End - Aviso SD	48
4.21	Front-End - Clicar SD	48
5.1	Arquitetura do NÓ de Sensores	50
5.2	Placa Arduino MEGA [15]	51
5.3	Sensor de Oxímetria [16]	52
5.4	Sensor de Pressão e Altura [17]	52
5.5	GPS NEO6MV2 [18]	53
5.6	Display OLed [19]	54
5.7	Led RGB [20]	54
5.8	TCA9548A I2C Expander [21]	55
5.9	Modulo microSD [22]	55
5.10	Modulo RTC DS1307 [22].	56
5.11	Modulo Libelium [23]	56
5.12	Arquitetura de hardware.	57
5.13	NÓ de Sensores.	58
5.14	Shields dos NÓS	59
5.15	Implementação das placas	59
5.16	Algoritmo MASTER-SLAVE	61
5.17	Ficheiros criados no cartão SD	69
6.1	Teste Sensor MAX30105	71
6.2	Teste Sensor MAX30105 - Aparelho do INEM	72
6.3	Teste Sensor MPL3115A2	72

6.4	Teste GPS	72
6.5	Teste de comunicações TX	73
6.6	Receção dos Pedidos MASTER-SLAVE	73
6.8	Teste TX e RX NÓ e Gateway	74
6.9	Pontos da emissão da mensagem (GPS)	75
6.10	Mensagem enviada 20Km	75
6.11	Perfil do Relevo	76
6.12	Representação NÓ de sensor e Gateway	76
6.13	Representação dos três NÓS de sensores prontos.	77
6.14	Teste da capacidade das baterias.	78
6.15	Emissor ACK	79
6.16	Três NÓS de Sensores em Laboratório (SOS)	79
6.17	Teste de Comunicação NÓS e Gateway	80
6.18	Representação do Teste	80
6.19	Teste de envio de SOS.	81
6.20	Teste de envio das coordenadas.	81
6.21	Marcação em mapa das coordenadas registadas.	82
6.22	Teste Salto com NÓ de Sensor (GPS).	83
6.23	Resultados obtidos com um NÓ de Sensores 2ºSALTO	83
6.24	Colocação do Gateway na Zona de Aterragem	84
6.25	3ºSalto com os três NÓS de Sensores.	85
6.26	NÓ de Sensor - 1A.	87
6.27	NÓ de Sensor - 2A.	87
6.28	NÓ de Sensor - 3A.	88
6.29	SpO2 3A - 4 Salto.	88
6.30	NÓS de Sensores dentro do avião	90
6.31	Colocação do Gateway na Zona de Aterragem	91

Capítulo 1

Introdução

1.1 Contexto



Figura 1.1: Crachá SOGA

Os Saltadores Operacionais de Grande Altitude (SOGA) com suporte a oxigénio fazem parte do destacamento de percursores aero-terrestres das tropas paraquedistas do Exército Português. Estes têm como características únicas, serem responsáveis por missões de alto risco, estando na primeira linha de ação através da infiltração nos campos de batalha (TO). Tem como símbolo, o crachá aero-terrestre da (Fig. 1.1).

As missões de atuação são variadas e incluem busca e salvamento, sabotagem de edifícios, pontes, colunas militares, colocação de equipas de atiradores especiais (sniper), normalmente nas linhas inimigas ou na sua retaguarda. Outra das missões onde podem atuar consiste no apoio aos batalhões de para-quedistas que efetuam Salto de Abertura em Automático. Neste caso é realizada a preparação do terreno para o lançamento dos saltadores.

A formação de um SOGA requer um grande investimento por parte das Forças Armadas, necessitando de uma instrução específica e cuidada. Estes militares tem uma formação inicial de treino e manutenção que se aproxima das condições em operações reais. Neste sentido, existe a necessidade de enquadrar as tecnologias em tempo de paz de modo a garantir um melhor desempenho e segurança tendo em conta as condições extremas de atuação (Fig. 1.2c).

O salto SOGA executa-se em dois tipos, High Altitude-High Opening (HAHO) e High Altitude-Low Opening (HALO), sendo o HAHO um salto de infiltração a grande altitude em calote, isto é, o saltador ao abandonar o avião a 30.000Ft (aproximadamente 10Km) abre o paraquedas e vem a navegar até ao Ponto de Impacto (PI), efetuando de seguida a missão atribuída. Esta distância pode ser percorrida em calote até 50km no terreno (Santarém - Tomar), desde que as condições meteorológicas estejam favoráveis (Fig.1.2a). Já o HALO, é um salto a 30.000Ft em que o saltador sai do avião e vem abrir a baixa altitude (3.500Ft), estando o menos tempo possível em

calote (paraquedas aberto)(Fig. 1.2b).

Este tipo de saltos são efetuados com suporte de oxigénio. As máscaras de oxigénio utilizadas antigamente eram chamadas de débito. Estas estavam sempre a debitar oxigénio, no entanto como os consumos eram elevados, restringiam o tipo de missões a efetuar. As novas máscaras, denominadas de máscaras a pedido, funcionam apenas quando o saltador inspira, tornando mais críticas as falhas de oxigénio que possam existir. Estas podem ocorrer quando existe um mau posicionamento ou quando por algum motivo existe uma fuga na conexão, na traqueia ou no regulador. Estas falhas levam a que o saltador respire ar ambiente sem que seja notado pelo próprio.

Nestas falhas, podem aparecer várias patologias derivadas da *desnitrogenação*. Existem outros fatores relevantes como a *hipoxia* ou déficit de oxigénio, a *hipercapnia* que corresponde ao aumento do dióxido de carbono (CO₂) no sangue, provocando *hiperventilação* e o aumento da frequência cardíaca. Adicionalmente, existem patologias por *descompressão*, que são doenças relacionadas com a pressão atmosférica [24]. A 30.000ft, quando o saltador não consegue respirar oxigénio a 100% durante cinco segundos provoca a morte.

Atualmente são usados métodos empíricos para controlo dos sinais vitais, baseados na observação e experiência dos operacionais. Assim, existe a necessidade de controlar, verificar e testar os dados biométricos do saltador de forma precisa para que se possa salvaguardar a sua saúde assim como para permitir o estudo de causas possíveis das patologias inerentes a esta atividade.

1.2 Enquadramento tecnológico

O uso das redes sem fios (*wireless*) são um elemento indispensável para as pessoas, para as organizações, para o meio social e cultural. A rede *wireless* são das tecnologias mais populares dos dias de hoje, sendo certo que a sua utilização cresce rapidamente e abrange cada vez mais o dia a dia das pessoas e organizações, seja nas áreas de bem-estar, como no setor industrial.

Atualmente existem tecnologias que permitem obter dados significativos, desde dados ambientais (temperatura, humidade) e dados biométricos, tais como oximetria e batimento cardíaco. Uma destas tecnologias recentes em desenvolvimento é conhecida por Low Power Wide Area Network (LPWAN). A sua chegada ao mercado, veio abrir novos horizontes tecnológicos, voltada para o Internet of Things (IoT). Esta está enquadrada num grupo de várias tecnologias de rede de baixa potência que pode usar frequências licenciadas ou livres.



Figura 1.2: Saltadores Operacionais de Grande Altitude

Neste cenário de operações, onde é necessário salvaguardar, proteger e controlar os sinais vitais dos saltadores, é fundamental o uso das tecnologias adequadas para permitir obter os dados para analisar e decidir em tempo real o procedimento a efetuar num teatro de operações onde a existência de sinais de alarme pode ser essencial para garantir o suporte de vida dos intervenientes.

1.3 Motivação e objetivos

Pretende-se com este trabalho desenvolver novas tecnologias para salvaguardar, prevenir e salvar militares que executam este tipo de saltos (SOGA), dando assim um novo enquadramento às missões executadas. Neste sentido, pretende-se desenvolver e investigar novas tecnologias e sistemas ainda não implementados em redes de IoT. Esta é uma abordagem encorajadora para desenvolver uma rede para a recolha de dados da monitorização de sinais vitais, com recurso à troca de informação entre os vários nós da rede.

Considerando a **motivação**:

- Salvar o militar de possíveis anomalias do seu bem-estar;
- Prevenir perdas de vidas humanas a alturas, não fisiológicas, por falta de oxigênio;
- Prevenir doenças por desnitrogenação, hipoxia, hipercapnia, hiperventilação e descompressão;
- Guardar os dados obtidos dos sensores para que possam ser analisados no contexto de especialidades médicas.

O **objetivo** geral desta investigação é o desenvolvimento de um sistema baseado em rede de sensores sem fios, cujos nós estão localizados em cada paraquedista, e que permita a transmissão de dados a grandes distâncias, usando tecnologias emergentes nomeadamente as LPWAN (Low Power WAN).

1.4 Descrição da solução proposta

Levando em consideração o descrito nos itens anteriores, bem como os objetivos que foram definidos, o trabalho a realizar consiste em desenvolver uma infraestrutura de comunicação com tecnologias LPWAN, que permita a aquisição de sinais biométricos, ambientais e de localização, o lançamento de sinais de alarme em caso de emergência, que permita a transmissão dos dados para análise posterior ou o seguimento da localização de cada nó em tempo real.

A figura 1.3 apresenta o esquema geral da arquitetura proposta. Cada nó está localizado num saltador. Este possui um módulo de comunicações, que permite enviar os dados tanto para o gateway como para os outros nós.

Cada nó possui associado um conjunto de sensores, nomeadamente, temperatura, pressão, altitude, SpO2 e GPS (Localização, altitude, dia e hora). Os dados recolhidos são guardados localmente, mas também enviados ao gateway. Cada nó possui a capacidade de analisar os dados de SpO2 e gerar alertas caso os valores registados estejam abaixo do valor admissível. O alerta é enviado para o gateway e para os outros nós, que caso estejam no seu alcance, têm a possibilidade de o receber. Localmente é gerado um sinal de alarme visual que permite a identificação imediata do elemento em dificuldades.

O gateway tem como função recolher os dados de cada nó e enviar alertas em modo broadcast. Previamente, existe um registo de utilizadores associados a cada nó, cuja gestão é feita usando um base de dados. Adicionalmente, o gateway possui a capacidade de visualização dos dados de cada saltador, em cada salto. Esta visualização

e análise pode ser feita em modo offline. Relativamente aos dados de localização por GPS, é possível realizar o seguimento em tempo real, a partir de uma pagina web, usando mapas previamente instalados, sem necessidade de comunicações de dados para carregamento dos mapas. O gateway pode estar ligado a um serviço de Cloud que permite uma análise mais aprofundada e sistemática dos dados recolhidos no teatro de operações.

Por exemplo, sabendo-se que a oximetria ideal para o ser humano está entre 98 a 100% (SpO2). Se o valor do sensor de oximetria baixar dos 95%, considera-se que existe perigo para a saúde do saltador. Assim o gateway envia uma mensagem a todos os NÓS de sensores, e avisa que o NÓ em causa está com valores abaixo da referência.

1.5 Organização do documento

O documento está organizado da seguinte forma. No capítulo 1 é realizado uma introdução, motivação e descrição da proposta de trabalho. No capítulo 2 faz uma revisão da literatura e um enquadramento geral da implementação de IoT. No capítulo 3 aborda a tecnologia LoRa e protocolos LoRaWAN, assim como os trabalhos relacionados. Posteriormente, o Capítulo 4 apresenta o desenvolvimento do gateway e do sistema de informação associados. O Capítulo 5 apresenta o sistema de aquisição de dados biométricos e de localização. O Capítulo 6 apresenta os resultados obtidos e finalizando, o Capítulo 7 expõe uma conclusão do trabalho realizado e apresenta as perspectivas para trabalho futuro.

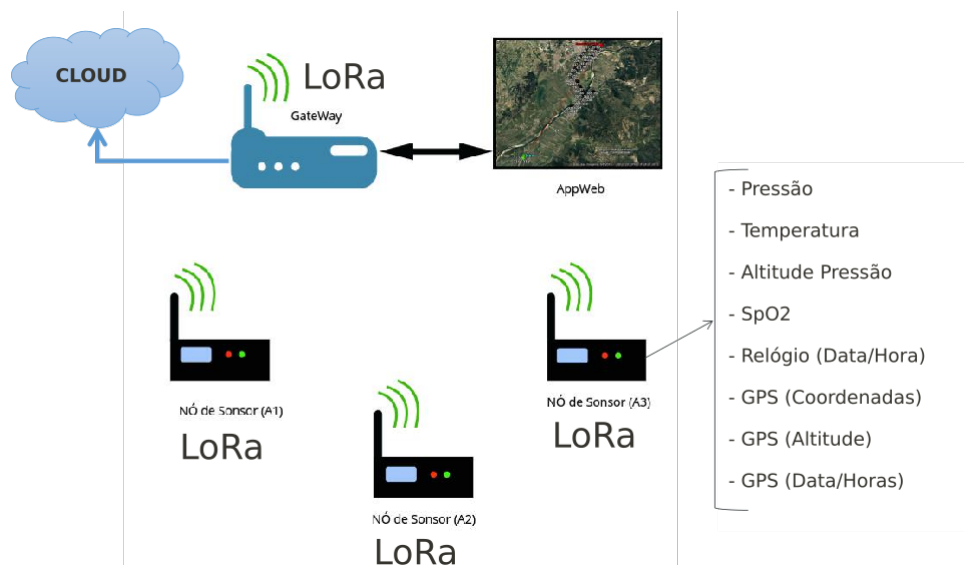


Figura 1.3: Esquema geral da arquitetura proposta.

Capítulo 2

Revisão da Literatura

Este capítulo realiza uma descrição sucinta sobre a IoT. É apresentado o contexto actual da IoT, realiza uma descrição geral das redes de sensores sem fios e suas aplicações, assim como das tecnologias de rede utilizadas.

2.1 A IoT no contexto Mundial

A internet das coisas (IoT), é um conceito cada vez mais utilizado e ouvido nos dias de hoje, no entanto estamos a falar de evolução dos conceitos informáticos e da internet ao longo da história. A humanidade vive num ritmo elevado de desenvolvimento tecnológico, saciada com aparecimento de novas soluções. Segundo "Daniel Smihula", o mundo atravessa a chamada "The 5th wave of the technological innovations started in 1980s", com uma alta presença da internet no dia-a-dia do ser humano. De fato, a IoT tem ganho importância no meio académico e industrial, porque se espera que o seu impacto seja significativo. A IoT é uma extensão da internet atual. Esta proporciona uma conectividade ao mundo, que possibilita uma oportunidade para diferentes serviços e negócios, alterando aos poucos o paradigma das redes de computadores. Assim, poderemos definir **IoT** como um sistema de dispositivos inter-relacionados, máquinas mecânicas e digitais, objetos, animais ou pessoas com identificadores únicos com capacidade de transferir dados através de uma rede, sem a necessidade de interferência humana no processo [25].

Kevin Ashton começou a desenvolver um sistema de sensores (RFID) no seu projeto, e aí nasce o termo "Internet of Things" demonstrando na sua apresentação "I made at Procter and Gamble" em 1999. Na época, a IoT era relacionada com o uso da tecnologia RFID[26]. Por volta de 2005, tornou-se um termo bastante procurado apresentando a relação entre IoT e as Redes de Sensores Sem Fios (Wireless Sensor Networks).

Relativamente ao termo "Wireless Sensor Networks", podemos analisar com base nas QUERY'S efetuadas no motor de pesquisa Google, que a sua procura tem vindo a decair como mostra a figura 2.1.

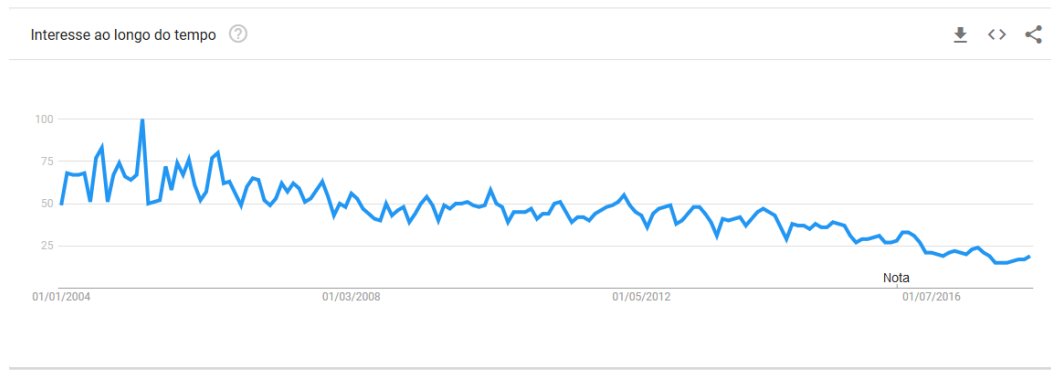


Figura 2.1: Análise da procura no Google do termo "Wireless Sensor Networks"[1].

Entre 2014 e 2015, o termo "Internet das Coisas" ganhou popularidade rapidamente (fig.2.2). Esta, deve-se ao amadurecimento das WSN e ao crescimento das expectativas sobre IoT.



Figura 2.2: Análise da procura no Google do termo "internet of things"[2].

A IoT foi identificada como uma tecnologia emergente em 2015. Foi previsto que a IoT levaria entre 5 e 10 anos (fig.2.3) para se adaptar ao mercado mundial e nota-se uma grande procura no âmbito académico e industrial no decorrer dos últimos anos. Embora o número de pesquisas tenha vindo a decrescer até aos dias de hoje, não quer dizer que estejam a ficar de parte pela comunidade científica, mas sim, que a comunidade já se adaptou a estes conceitos e já faz parte dos seus projetos.

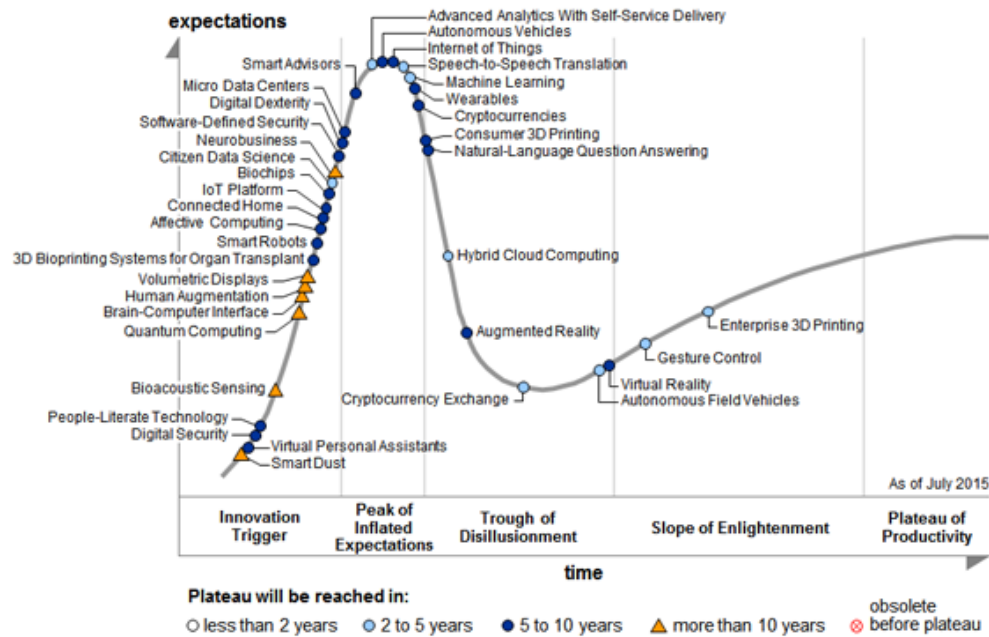


Figura 2.3: IoT 5 a 10 Anos

Fonte: Gartner (July 2016) [27]

Contudo, as redes de sensores ainda tem um caminho árduo a percorrer. Existem muitos sensores que podem ser conectados à rede, mas a população mundial não está confortável com a ideia. Este desconforto remete para o problema da segurança dos dados adquiridos, armazenados e visualizados, uma vez que a humanidade não está confiante na sua salvaguarda, alterando as suas vidas. Este sentimento remete para os problemas de segurança (saber onde estou?). As sondagens mostram que mais de 70% dos europeus se manifestam preocupados com o que acontece com os dados disponíveis na Internet [28], a maioria não tem conhecimentos da área e remete para os valores humanos. Em Portugal os dados pessoais estão salvaguardados pela Constituição da Republica Portuguesa (CRP) no Artigo 35º, na alínea nº2 refere que “A lei define o conceito de dados pessoais, bem como as condições aplicáveis ao seu tratamento automatizado, conexão, transmissão e utilização, e garante a sua proteção, designadamente através de entidade administrativa independente“, casos mais específicos, são remetidos para a própria área. No caso dos Países Europeus que pertencem à União, existe uma Diretiva 95/46/CE do parlamento Europeu, relacionada com a salvaguarda dos dados pessoais.

Deste modo, a aceitação deste tipo de tecnologias passa por uma boa informação do que são as IoT e as WSN, podendo-se explicar aos cidadãos, que estas não são um mal em si, nem pretente ser uma vigia da vida privada ou pública, mas sim um bem público que pode permitir a informação, sobre um estado de segurança. Podemos

considerar segurança, saber o estado do tempo, as marés, o trânsito, entre outras, com diferentes níveis de acesso, no entanto disponível a todos os cidadãos.

A Internet das Coisas (IoT) é considerada a terceira onda das tecnologias de informação depois do desenvolvimento da internet e das comunicações móveis. Prevê-se que até 2025, os nós (sensores) na Internet possam residir em aspetos da vida quotidiana, desde pacotes de comida, móveis, documentos em papel e muito mais. Destacam-se as oportunidades futuras que surgirão, partindo da ideia de que a questão popular combinada com os avanços tecnológicos podem impulsionar a difusão generalizada de uma Internet das Coisas, que por si só, pode contribuir de forma inestimável para o desenvolvimento económico [29].

2.2 Redes de Sensores Sem Fios

No que respeita a Redes de Sensores Sem Fios (RSSF) ou em inglês wireless sensor networks (WSN), poderemos dizer que estas nasceram em 1896 com **Alexander Popov**, com a implementação de uma transmissão sem fios em rádio entre dois edifícios. Em 1898 faz uma transmissão de 5km, entre um navio e um posto em terra. Ao serviço da Marinha Russa coordenou a instalação de uma estação rádio na ilha de Gogland, no Golfo da Finlândia, e outra em Kotka, na costa finlandesa. Estas estações deram uma ajuda preciosa ao navio Apraksin no inverno de 1899/1900. [30]

Ainda num contexto militar, na década de 50 do século XX, com um emprego militar (**US Navy**) o projeto Sound Surveillance System (SOSUS) da US Navy, utilizava um sistema para monitorizar o som de baixa frequência no canal sound fixing and ranging channel (SOFAR) para detetar os submarinos inimigos. Este canal utiliza regiões profundas nos oceanos onde o som se propaga a milhares de Km até se dissipar. Este projeto foi criado do programa Distributed Sensor Network (DSN) pela Defense Advanced Research Projects Agency (DARPA). [31]

Em 1980 a DARPA com o projeto DSN alarga as pesquisas sobre RSSF [32], em que o SOSUS é utilizado pela National Oceanic and Atmospheric Administration (NOAA). Esta nova abordagem usa nós de sensores colocados numa arquitetura distribuída, cooperativa, autónoma e com um baixo custo, com o objetivo de encaminhar a informação entre os diversos nós.

Estamos numa época em que não existiam computadores pessoais, nem estações de trabalho. O processamento era feito em minicomputadores como o PDP-11 e VAX sistemas operativos Unix e OpenVMS (Open Virtual Memory System). O projeto DSN

foi um grande sucesso, para detetação de aviões a baixa altitude.

Com a investigação a avançar, em 1999 os investigadores começam a pesquisar redes com pequenos sensores com possibilidade de interligar-se entre si. No contexto militar surge o sistema de defesa Americana, Sensor Information Technology (SensIT), com duas finalidades principais: desenvolver redes Ad Hoc em ambientes dinâmicos e realizar processamento da informação.

Em aplicações num âmbito doméstico, destaca-se por exemplo, no ano 2000 o primeiro frigorífico ligado à internet pela LG. Em 2002 é apresentado o Ambient Orb. Este sistema consiste numa luz que mostra várias cores. Cada cor representa informação baseada em dados da internet relacionada com as finanças pessoais, informações meteorológicas, entre outras. Já em 2005 a "The United Nations first mentions IoT", publica pela primeira vez informação sobre o IoT. Em 2008 a IPSO alliance, promove o uso do protocolo IP nos dispositivos. Mais recentemente em 2011, IPv6 internet protocol version 6, renasce com uma alternativa (nova geração do IP - dual stack), que vem substituir o IPv4 que se encontra esgotado (limitado na sua capacidade de endereços). Em 2013, a Intel lança "internet of things solutions Group", com grande referências e adeptos.

Mais recentemente surgiram aplicações com RSSF, desde "Mobile-Controlled", Temperature Learning, Mobile Keyless entry e Wi-Fi Security camera, Connected Fitness Tracker", procurando usar hardware mais pequeno e flexível entretanto desenvolvido. Prevê-se que em 2025 a evolução seja feita com a implementação de "Efficiente Smart home", "Artificial Assistant Hologram", "Smart Walls" e "Connected Wellness Monitoring". [33]

A evolução das RSSF tem vindo a crescer devido aos custos de implementação mais reduzidos, processadores de baixa potência e mais precisos nas leituras. Com sistemas Micro Eletro Mecânico (MEMS), acessíveis com uma grande variedade de tipos (Motes, Pico, WINS, Z1 Mote, WaspMote, JPL, Ubi-Mote, AMPS, TinyNode). Todos eles têm finalidades semelhantes, no entanto com diferenças significativas, dependendo sempre do seu objetivo. Os sistemas MEMS, hoje em dia estão fortemente implementados no desenvolvimento de sensores em automóveis, na área militar, na medicina, na agricultura, na indústria em geral, no estudo das condições ambientais e na engenharia com várias aplicações em desenvolvimento, como por exemplo a área da robótica [34].

2.2.1 Aplicações das RSSF

Com o desenvolvimento do hardware de suporte, as aplicações de sensores em redes sem fios, têm vindo a aumentar bruscamente. Existe neste momento uma grande

variedade de aplicações que podem ser divididas por âmbito de atividade. Destacam-se;

1. Aplicações Domésticas

Estas aplicações são conhecidas por Smart Home, com um grande desenvolvimento da IoT nesta área. Podemos assim considerar a automação doméstica, com sensores nos eletrodomésticos e ambientes inteligentes, usando uma interação baseada em smartphone, tablets ou SmartTV, criando uma rede entre eles e facilitando a vida dos utilizadores [35].

2. Aplicações de controlo Ambientais

As redes de sensores podem ser uma grande ajuda no controlo da floresta, permitindo a prevenção e monitorização de incêndios [36]. Outra aplicação possível é no controlo da poluição industrial, minimizando o seu impacto nas plantas e animais selvagens [37]. Na prevenção de catástrofes naturais [38], os sensores podem prevenir e atempar acontecimentos, de modo que se possam proteger vidas e bens.

3. Aplicações Industriais e de Agricultura

Se em relação ao ambiente existe uma variedade de sensores, a sua aplicação pode também ser realizada em ambientes industriais, onde as redes de sensores podem ser usadas para monitorização e controlo de processos, controlo ambiental, controlo de qualidade e de gestão. Por outro lado, na agricultura, estas podem ser usadas na motorização de animais, no controlo de regas, na automação de máquinas de precisão, que permitem melhorar as técnicas antigas e otimizar a produção [39].

4. Aplicações Médicas

A IoT na medicina tem vindo a crescer ao longo dos anos, em particular nestes últimos três (3) anos. Não só a nível hospitalar mas também em ambiente doméstico "*iHome Health-IoT*". Este conceito vem auxiliar a vida das pessoas em vários aspetos, tais como a "*iMeBox*", uma plataforma que envolve uma caixa de medicamentos inteligentes, com conectividade de integração de dispositivos e serviços, o "*iMedPack*" com capacidade de comunicação passiva de radiofrequência (RFID) ou dispositivos biomédicos "*Bio-Patch*"[40], indo assim ao encontro das necessidades de cada pessoa. Aumentam-se assim as possibilidades de assistência aos utentes relativamente aos sistemas já existentes, por exemplo, baseados em telefone (SNS24).

Podemos também observar desenvolvimentos em rotinas diárias da pílula das mulheres, controlando tanto o seu consumo, como os pedidos de novas embalagens,

como é o caso do *Smart Pill Box* [41].

Não só a IoT se encontra nas casas, mas também nos hospitais. Um hospital inteligente será sem dúvida uma ajuda aos seus profissionais. Assim, o armazenamento dos dados e a sua utilização em função de diagnóstico, tratamento, gestão e decisão, sendo sem dúvida uma opção para um melhoramento dos serviços prestados, tanto do lado do doente como dos profissionais [42].

5. Aplicações Militares

Desde a década de 50, que as tecnologias de rede de sensores está presente nos meios militares [31]. Contudo não têm evoluído tanto como desejável quando comparando com outras áreas. Por um lado é compreensível, uma vez que a sociedade está mais voltada para aspetos sociais do que para as aplicações militares. Contudo existe a necessidade de introduzir as novas tecnologias, existindo espaço para melhorias no controlo da segurança com sistemas *RFID* ou por vídeo vigilância, por exemplo [43].

Um Relatório do Seminário da Divisão de Tecnologia Militar da Universidade Sueca da Defesa (SEDU), fez um estudo sobre as novas tecnologias nas suas Forças Armadas da atualidade até 2040. Não se refere apenas a IoT mas também a outras áreas.

Em relação à IoT, estudam os displays transparentes com grafeno ou nanotubos de carbono, câmaras de imagem térmica produzidas em massa para soldados, pequenas câmaras de imagem térmica para veículos aéreos não tripulados (UAV), telas de toque flexíveis integradas em uniformes de soldados. Num tempo próximo e no futuro, teremos óculos de proteção com suporte de realidade aumentada e possível função anti-névoa, câmara com bandas de comprimento de onda muito ampla para suporte ao combate dos soldados e sensores que podem ser usados numa variedade de ambientes em veículos de combate. [44]

Os drones (veículos aéreos não tripulados) (UAV) são cada vez mais uma ferramenta fundamental no campo de batalha, podendo assim dar apoio à decisão, como também realizar missões específicas. Estas, podem ir desde a vigilância até ao ataque com mísseis de precisão. Com base nos UAV, e num contexto mais alargado, foi estudado o cenário de utilizar estes drones em grupo (Fig:2.4), reduzindo o seu custo e operando em várias missões atribuídas. Neste caso existe a possibilidade de comunicação à estação base de controlo ou entre si, sendo

possível que dezenas de drones possam estar em combate. [45]

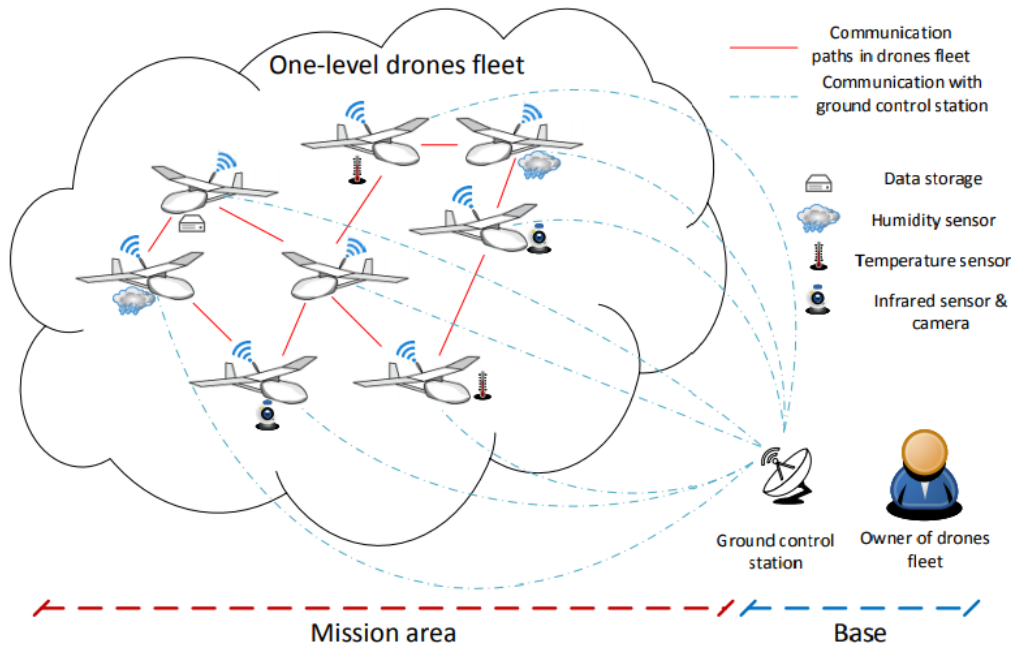


Figura 2.4: Enxame de Drones (UAV)

Fonte: Artigo [45]

2.2.2 Segurança em RSSF

As redes de sensores sem fios (WSN), tem cada vez mais um lugar assegurado no futuro. Em aplicações militares, comerciais e na agricultura, são fundamentais para uma boa gestão de conteúdos e processos nessas áreas. Contudo, existem aspetos a ter em conta, desde a sua autonomia como a sua capacidade de memória. Estes são pontos fundamentais para uma boa segurança dos dispositivos. Neste tipo de redes existem fatores que não se podem desprezar, tais como a sua segurança. A segurança em redes (WSN), é fundamental por vários motivos, desde proteger o sistema dos ataques, da alteração de dados ou para recolha de dados não autorizada. Segundo [3], os processos de gestão de chaves devem incorporar quatro (4) fatores importantes: **a confidencialidade, integridade, autenticação e não-repúdio** (Fig. 2.5).

- **Autenticação do remetente** - O destinatário deverá ser capaz de identificar o remetente e verificar que foi mesmo ele quem enviou a mensagem.
- **Confidencialidade da mensagem** - Só o destinatário autorizado deve ser capaz de extrair o conteúdo da mensagem da sua forma cifrada. Além disso, a obtenção de informação sobre o conteúdo da mensagem não deverá ser possível, tornando a análise criptográfica, mais facilitada.

- **Integridade da mensagem** - O destinatário deverá ser capaz de determinar se a mensagem foi alterada durante a transmissão.
- **Não-repúdio** - Não deverá ser possível ao emissor negar a autoria da mensagem.

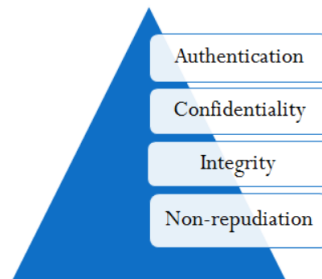


Figura 2.5: Segurança na RSSF [3].

2.3 Tecnologias de Redes Sem Fios

Em IoT todas as tecnologias são abordadas e utilizadas, dependendo da sua arquitetura e das suas características. Assim faremos uma breve abordagem aos tipos mais utilizadas na indústria atual;

1. Wi-fi

Uma tecnologia bastante popular na rede sem fios. Encontra-se em muitos aparelhos ou mesmo quase todos, desde telefones móveis, computadores, impressoras, na indústria em geral (máquinas), na extensão da rede por cabo, em routers em casa, na televisão (SmartTV), ou em alguns eletrodomésticos da cozinha. Considera-se a evolução da Ethernet [46]. As suas características são:

- Protocolo IEEE 802.11a,b,g,n,ac;
- Frequências de 2,4 GHz a 5 GHz (54Mbps e 150 a 600Mbps);
- Segurança de acesso (WPA e WPA2);
- Potência de 100 a 1000 mW.

2. WiMax

Esta tecnologia WiMax, oferece acesso de banda larga a grandes distâncias e funciona de forma semelhante ao Wi-Fi. O seu funcionamento consiste num terminal e numa estação base e utilizada em redes Wireless Metropolitan Area Network (WMAN) [47]. As suas características são:

- Protocolo IEEE 802.16;
- Frequências de 3,5 GHz a 5,8 GHz;
- Distancia acima 50 km.

3. LTE

O Long Term Evolution (LTE), é uma rede de comunicações móveis Quarta Geração Telecomunicação móvel (4G). Esta tecnologia de rádio permite débitos até 1 GBs. Utilizada pelas operadoras em várias frequências, para várias aplicações, como telefone, video e dados. Existe LTE Cat-1,3,4 e Cat-M1. [48]

4. ZigBee

O ZigBee é um protocolo de comunicação de alto nível, usado para criar redes pessoais (Casa - WPAN), com baixa potência, curto alcance, hardware de baixo custo e de baixo consumo de energia. Esta utiliza topologia ad hoc sem fios, podendo ser usada em atuadores para ligar e desligar aparelhos. Com a topologia em MESH permite alcançar maiores distâncias, ligando-se uns aos outros e transmitindo a informação desejada ao atuador (aparelho - rele) [49]. As suas características são:

- Protocolo IEEE 802.15.4;
- Frequências de 2,4 GHz;
- Distância de 75-100 metros dentro de casa;
- Distância de mais 300 metros sem obstáculos;
- Segurança AES-128 Network Layer e Application Layer;
- Topologia de rede em MESH.

5. Bluetooth Low Energy (BLE)

O BLE é uma tecnologia cada vez mais utilizada, já na sua versão cinco (5), pois permite interligar dispositivos (emparelhar) facilmente. Na área da informática temos teclados, ratos, já na área da IoT temos os telefones móveis com troca de dados para outros telefones e dispositivos. Uma aplicação muito popular é a sua utilização em relógios para treino físico e a sua ligação com o telefone móvel [50].

As suas características são:

- Protocolo IEEE 802.15.1;
- Frequências de 2,4 GHz;
- Distância 100 metros;
- Segurança 128-bit AES with Counter Mode CBC-MAC (cipher block chaining message authentication code) and application layer user defined;
- Topologia de rede P2P.

6. RFID

O RFID é um método de identificação automática, que utiliza rádio- frequência para receber dados dos seus identificadores (etiquetas RFID). Estas etiquetas contêm um ID (identificador numérico) e ainda pode conter dados. Quando este é detetado pelo leitor de RFID, o leitor recolhe dele a informação contida no

mesmo, guarda em base de dados ou então executa uma ação (fazer algo, após reconhecer o ID) [51]. As suas características são:

- Frequência: 120-150kHz (LF), 13.56 MHz(HF), 433 MHz(UHF), 865-868 MHz (Europeu) 902-928 MHz (North America) UHF, 2450-5800 MHz(microwave), 3.1–10 GHz (microwave);
- Distância Curta;
- Segurança *protecting the chip tags, cryptography e middleware*;
- Topologia de rede P2P;
- Outro Protocolo RFID: DASH7 (ISO/IEC 18000-7) (>1Km).

7. LoRa(WAN)

LoRa é a tecnologia e LoRaWAN é o nome dado ao protocolo que define a arquitetura do sistema bem como os parâmetros de comunicação usando a tecnologia LoRa. É uma tecnologia de rádio-frequência que permite uma comunicação a longas distâncias com um consumo mínimo de energia. Foi desenvolvida a pensar em IoT, com uma comunicação bidirecional (Node-gateway), mobilidade e longa vida da bateria [52]. As suas características são:

- Frequências de 863-870 MHz e 433MHz (EU) e 902-928MHz(US);
- Distância de 5 Km ambiente urbano;
- Distância, mais 22 Km sem obstáculos;
- Segurança AES-128 Network;
- Topologia de rede M2M;
- Bateria mais de 10 anos.

8. SigFox

O *SigFox* é uma empresa Francesa, criada em meados do ano 2009, com a finalidade de construir redes sem fios de baixa energia para IoT. Poderíamos dizer que seria uma concorrente de LoRa Alliance, mas não é. O seu conceito de negócio assenta em *mobile network*, criando a sua própria rede LPWAN, com servidores, repetidores, e dispositivos IoT. Contudo, outros dispositivos podem ser compatíveis. A utilização da rede (network) por parte de outros, necessita de pagamento para acesso ao serviço [53] [54]. Tem como características:

- Frequências de 863-870 MHz e 433MHz (EU) e 902-928MHz(US);
- Distância de 10 Km ambiente Urbano;
- Distância de 50 Km Rural;
- Segurança AES-128 Network;
- Topologia de rede **MESH ou M2M!**; - Bateria mais de 5 anos.

9. NB-IoT

O Narrow Band IoT (NB-IoT) foi criado em meados do ano 2016, pelo organismo

de normas internacionais 3GPP. O NB-IoT usa frequências que requerem licença, iguais às dos telefones móveis 4G, no entanto mais barato que a LTE [55] [56]. Opera em LTE Cat NB1. (a) Stand-alone: utilizando suporte independente. O espectro atualmente utilizado pelos sistemas GERAN como uma substituição de uma ou mais operadoras GSM. (b) Guard band: utilizando os blocos de recursos não utilizados dentro da banda de guarda de uma transportadora LTE. (c) In-band: utilizando blocos de recursos dentro de um transportador LTE normal. [57]. A distância utilizada é a das antenas das telecomunicações e a sua bateria tem uma duração de mais de 10 anos.

10. Outras tecnologias

Existem mais tecnologias em redes sem fios, no entanto vamos só abordar algumas mais relevantes na indústria e que tem sobrevivido ao tempo. A **DASH7!** está presente na área da logística. Outras implementações, por exemplo **Wireless-shart** (2007) que é o líder da automação industrial e o ISA100 (2009), utilizando topologias de rede em *Mesh*. [58] [59]

A **Telensa**, é uma tecnologia criada para as cidades (*Smart Cities*). A "Smart City API" permite que os dados dos sensores e das aplicações sejam aproveitados, para análise e visualização dos mesmos. Tem sido um grande sucesso e ainda permite que outros dispositivos contribuíssem para o armazenamento de dados. Utiliza uma tecnologia de rádio de banda ultra estreita (UNB), tendo duas (2) tipologias: uma do livre licenciamento e outra paga, numa comunicação bidirecional, baixo custo, longo alcance e comunicação flexível. Esta tecnologia está na grande corrida da LPWAN, já estando em 30 países e com nove (9) milhões de dispositivos. No entanto, consideramos uma tecnologia que não se encontra em "*open source*" sendo de difícil iteração para a investigação. [60]

Capítulo 3

Tecnologia LoRa e protocolo LoRaWAN

Este capítulo apresenta uma descrição da tecnologia LoRa e protocolo LoRaWAN, usados para a transmissão de dados no protótipo apresentado neste trabalho.

É também apresentada uma descrição de trabalhos relacionados no contexto de IoT usando a mesma tecnologia de rede.

3.1 Tecnologia LoRa®

LoRa® é uma tecnologia emergente vocacionada para a IoT, com vários microprocessadores no mercado. Utiliza uma frequência rádio que permite longas distâncias com o mínimo consumo de energia. Esta tecnologia baseia-se em comunicação M2M (Machine-to-Machine), em que os nós comunicam com um Gateway e este com os módulos (rede estrela). As comunicações em malha estão ainda numa fase embrionária, pelo menos para os microprocessadores emergentes (SX1272[EU], RN2903[USA], RN2483[EU], F8L10D, MIPOT 32001353). O LoRa baseia-se numa tecnologia conhecida por “chirp spread spectrum modulation (CSS)” utilizada em sistemas militares, que possibilita longo alcance e grande imunidade ao ruído [61].

LoRa é uma tecnologia de radio frequência que permite comunicações a longas distâncias com um consumo mínimo de energia. Baseia-se em uma rede com topologia em estrela, similar a uma rede de telefone móvel, como mostra a figura 3.1. Esta permite a ligação de milhões de nós, a distância pelo menos até 20Km. É uma tecnologia com baixos custos de infraestrutura e de fácil implementação.

LoRaWAN™ é o nome dado ao protocolo que define a arquitetura do sistema, bem como os parâmetros de comunicação da tecnologia LoRa®. O protocolo LoRaWAN™ implementa os detalhes de funcionamento, segurança, qualidade do serviço, ajustes de potência visando maximizar a duração da bateria dos módulos, e os tipos de aplicações tanto do lado dos NÓs como do Gateway.

Como representado na figura 3.1, os NÓs ligam-se ao gateway e enviam os dados recolhidos dos seus sensores. Por sua vez, o gateway concentra os dados, e envia-os a um servidor com base de dados. Este dados podem ser utilizados por aplicações para as diversas finalidades.

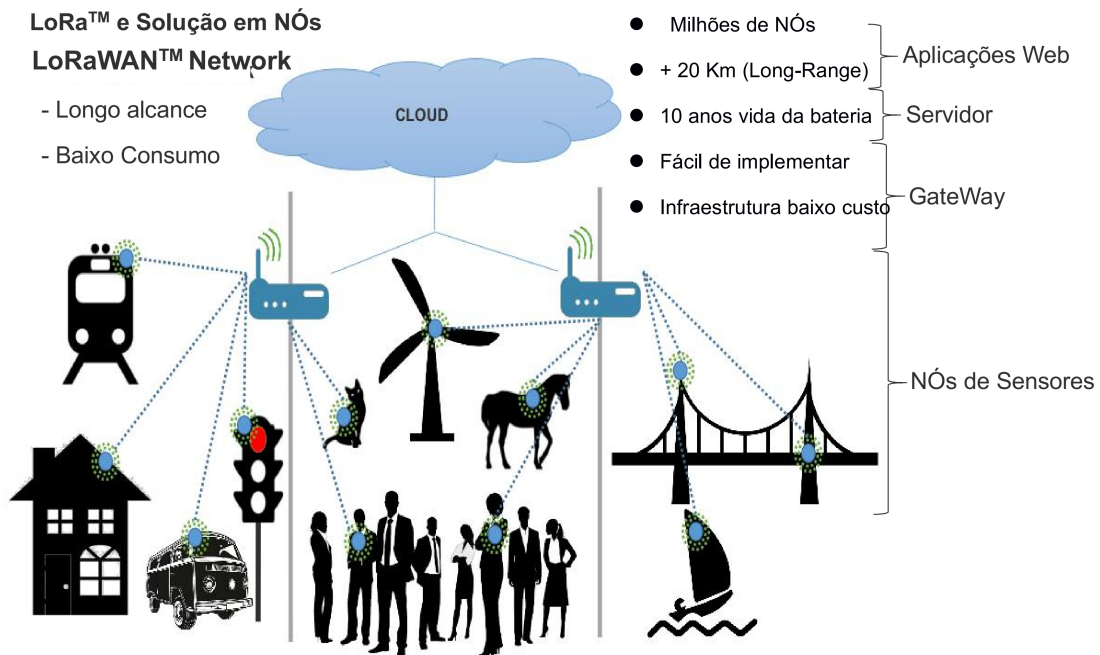


Figura 3.1: LoRaWAN

No conceito LoRa Semtech, esta basea-se nas LPWAN, com uma modulação de espectro (CSS) e permite diferentes débitos de transmissão, com a largura de banda e a distância de propagação confiáveis, acrescentando ao baixo consumo de energia em cada módulo (NÓ). Segundo refere o artigo [62], o consumo é de $10\mu A$ (rede: 25 mA TX e 10 mA RX).

Este protocolo é de código aberto (*open source*) para desenvolvimento de aplicações. Pertence à **LoRa® Alliance** que também é uma associação aberta sem fins lucrativos, criada e fundada por líderes da indústria (Semtech, IBM e mais 160 membros) [63].

A grande vantagem desta tecnologia está no seu longo alcance. Um único dispositivo pode permitir transmissões numa cidade, no entanto, está dependente do meio envolvente. Mesmo assim, permite comunicações a distâncias maiores dos alcances pelas comunicações padronizadas. Uma determinada tecnologia não consegue ser desejada em todas as aplicações e volumes de dados. Por exemplo, podemos ver que o WiFi e BLE são padrões de comunicação para dispositivos em casa e em redes de telefones para altas taxas de dados com grandes consumos de energia. As LPWAN ao contrário das anteriores, permitem garantir uma energia duradoura, e vocacionadas para redes

de sensores.

Como exemplo na figura 3.2, podemos observar que o mercado das IoT já possui 55% de utilização e com tendência para aumentar. A tecnologia LPWAN não foi pensada para grande volume de dados, por isso considera-se um envio das mensagens de pequenos volumes (10kbps), enquanto que as outras tecnologias tem maior capacidade. No que se refere à distância de envio e recepção, esta está ao nível das redes de telefones (20Km). A eficiência na bateria é a melhor em relação às outras tecnologias (10 anos vida), como também o custo de implementação deste tipos de redes.

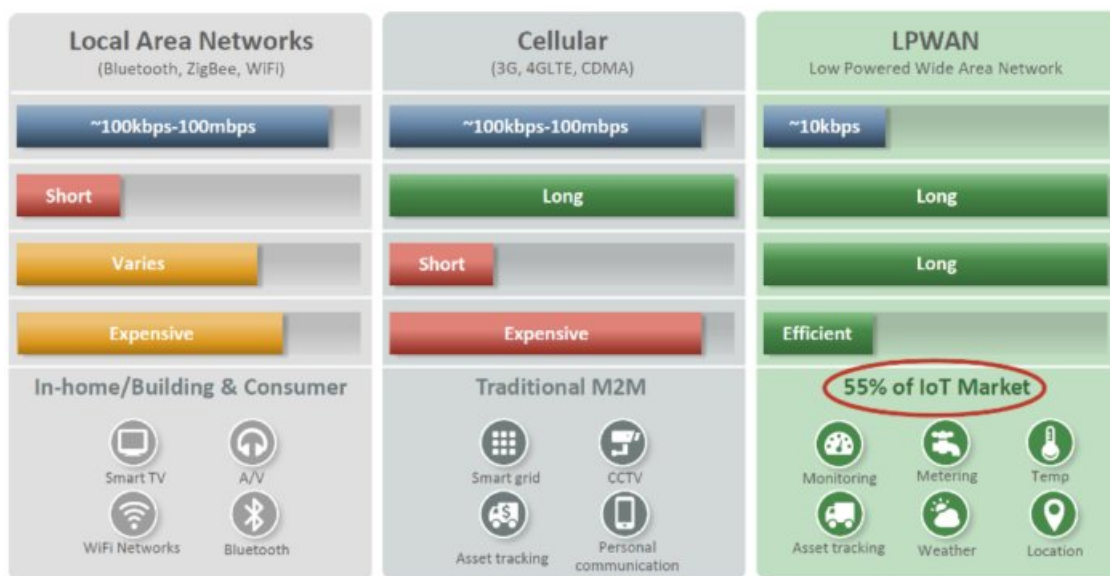


Figura 3.2: Comparação entre tecnologias [4].

Características do LORA®:

- Longo alcance tornando imune a interferências;
- Longa vida útil para as baterias, devido ao baixo consumo;
- M2M (Machine-to-Machine);
- Multiusos, pode operar em rede pública (aberta) e rede privada;
- Usando protocolos para cada uma das redes;
- Funciona via rádio (RF – EU 868);
- Segurança – AES encryption;
- Utilização de vários sensores (GPS-free, Temperatura, Humidade, etc.);
- Capacidade de utilizar 1 Milhão de nós em comunicação com gateway;
- Protocolo LPWAN (Low Power Wide Area Network).

3.1.1 LoRaWAN

LoRaWAN™ define o protocolo de comunicação e a arquitetura do sistema para a rede, enquanto a camada física LoRa® habilita o link de comunicação de longo alcance. O protocolo e a arquitetura de rede têm a maior influência na determinação da vida útil da bateria de um nó, da capacidade da rede, da qualidade do serviço, da segurança e da variedade de aplicações suportadas pela rede.

Podemos observar a sua composição na figura 3.3, onde descrevemos a sua estrutura;

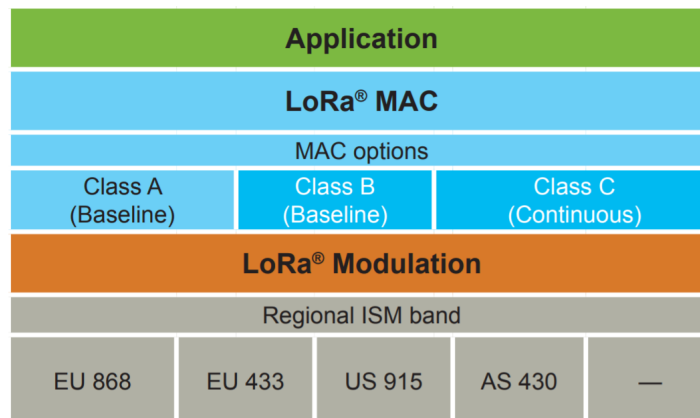


Figura 3.3: Camada lógica da Rede (LoRaWAN) [5]

- Camada Aplicação - Esta camada é responsável por promover serviços para os clientes. Poderemos observar o seguinte exemplo: uma aplicação solicita medições de humidade e envia para o utilizador através das camadas inferiores.
- Camada Mac - Considerada um subcampo do protocolo, que se encontra em cima da camada física, segunda camada do modelo OSI. Podemos assim considerar as funções da camada MAC:
 - Atribuir a cada dispositivo uma frequência, código e taxa de dados;
 - Eliminar receções duplicadas;
 - Reconhecimento do fuso horário;
 - Adapta as taxas de dados;
 A camada de aplicação pode executar solicitações à camada MAC e esta confirmar ou vice-versa [64].
- Tipo de Classes - No protocolo LoRaWAN [62] foram definidas três classes de dispositivos (módulos).
 - Classe A - Sensores:
 - Sensores;

- Operados a bateria;
- Comunicação bidirecional, recepção após transmissão;
- Otimizado para reduzir o consumo;
- Classe B - Atuadores:
 - Atuadores;
 - Operados a bateria;
 - Comunicação bidirecional, com janelas de recepção agendadas;
 - Módulo e gateway trocam informação (beacon way) de forma a que o gateway saiba quando o módulo está pronto para receber dados.
- Classe C – Bidirecional, recepção de dados praticamente sem atrasos;
 - Nesta classe o módulo está sempre apto a receber dados do gateway;
 - Não é recomendado o uso só da bateria, devido ao consumo de energia;
- Camada Modulação - LoRa utiliza a modulação do tipo **Frequency Shift Keying (FSK)**, que se baseia em **Chirp Spread Spectrum (CSS)**. A CSS ajuda a transmitir sinais para distâncias muito grandes. [65]

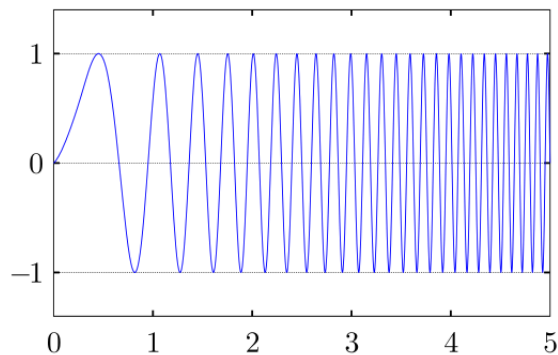


Figura 3.4: Exemplo de Up-chirp [6]

O CSS foi desenvolvido para aplicações de radar. Os sinais de *Chirp* têm amplitude constante e passam toda a largura de banda de forma linear ou não linear, de uma extremidade a outra. O espectro *Chirp* usa a largura de banda completa para transmitir os sinais. Se a frequência varia da mais baixa para a mais alta, é chamada "*up-chirp*" e se a frequência muda da mais alta para a mais baixa, chamamos de "*down-chirp*". A figura 3.4, mostra um "*up-chirp*".

- Frequências - Os Canais de LoRaWAN são de licença livre de utilização, em frequência Sub-GHz. Os canais livres variam de país para país e de continente para continente. No entanto, existem países que para normalizarem procedimentos, adotaram algumas bandas idênticas. Como podemos observar na figura 3.5,

na Europa, a frequência livre é de 868 MHz ou de 433 MHz, enquanto que na América é de 915 MHz. A estrutura da banda de 868 MHz, está representada na figura 3.6.

	Europe	North America	China	Korea	Japan	India
Frequency band	867-869MHz	902-928MHz	470-510MHz	920-925MHz	920-925MHz	865-867MHz
Channels	10	64 + 8 + 8	In definition by Technical Committee	In definition by Technical Committee	In definition by Technical Committee	In definition by Technical Committee
Channel BW Up	125/250kHz	125/500kHz				
Channel BW Dn	125kHz	500kHz				
TX Power Up	+14dBm	+20dBm typ (+30dBm allowed)				
TX Power Dn	+14dBm	+27dBm				
SF Up	7-12	7-10				
Data rate	250bps- 50kbps	980bps-21.9kbps				
Link Budget Up	155dB	154dB				
Link Budget Dn	155dB	157dB				

Figura 3.5: Quadro de Frequências LoRa [7]

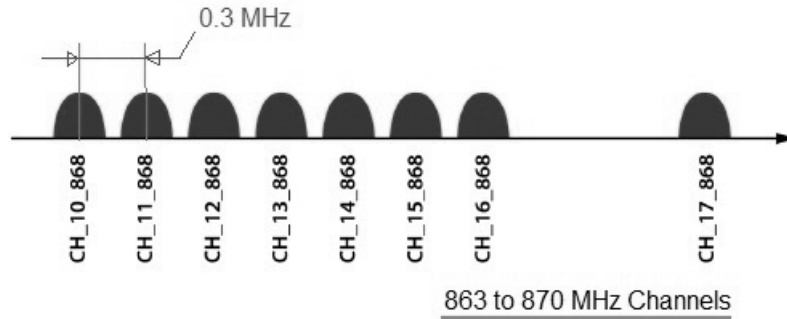


Figura 3.6: Estrutura do Canal 868 MHz [8]

3.1.2 Segurança LoRaWAN

Para garantir a privacidade dos dados enviados existem vários níveis de segurança:

- Segurança para a informação (Chaves);
- Segurança na transmissão dos dados (Encriptação).

AES - Encryption

Utiliza duas camadas de segurança: uma para a rede e outra para a aplicação. A segurança da rede assegura a autenticidade do nó na rede enquanto que a camada de segurança da aplicação garante que o operador de rede não tenha acesso aos dados da aplicação do utilizador final. É usada a encriptação AES. [66]

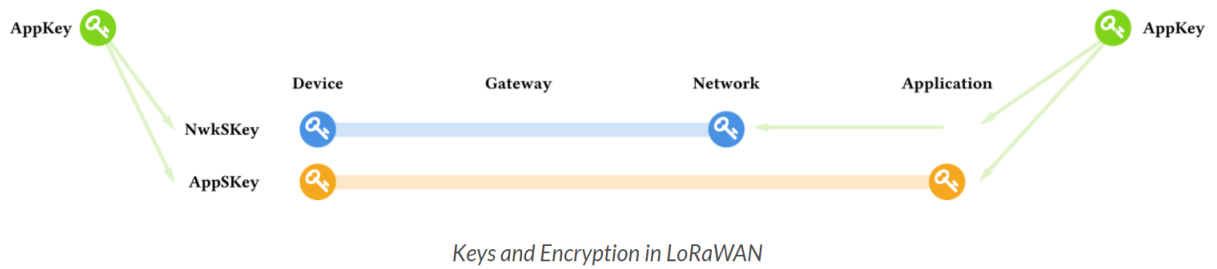


Figura 3.7: AES em LoRaWAN [9]

Por exemplo na figura 3.7, temos uma implementação da segurança em AES, que funciona do NÓ para o servidor (Cloud), usando o gateway com meio de transporte da mensagem codificada. A proteção de segurança é fornecida nos níveis de dados, usando o NWSKey. O protocolo LoRaWAN especifica um número de chaves de segurança que se podem usar na implementação, a NwkSKey, a AppSKey e a AppKey em que todas as chaves têm um comprimento de 128 bits.

A chave de sessão de rede (NwkSKey) é usada para interação entre o nó e a rede, verificando a validade das mensagens. A chave de sessão do aplicativo (AppSKey) é usada para encriptação e desencriptação da mensagem. A mensagem é totalmente encriptada entre o nó e o servidor (Cloud). Estas duas chaves de sessão (NwkSKey e AppSKey) são exclusivas do dispositivo, por sessão. Em cada sessão será gerada novamente um par de chaves. Os dispositivos ativados dinamicamente (OTAA) usam a chave do aplicativo (AppKey) para gerar as duas chaves de sessão durante o procedimento de ativação. Na aplicação "Front-End", também é feita uma validação (DevEUI - 64 bit end-device identifier, DevAddr - 32 bit device address, AppEUI - 64 bit application identifier, GatewayEUI - 64 bit gateway identifier).

3.1.3 Dispositivos

Os módulos de sensores podem ser basicamente (*end-points ou end-devices*), este permitem uma comunicação entre o *gateway* e o NÓ e vice-versa.

Estes módulos podem ser:

- Módulos de sensores (temperatura, humidade...)
- Gateway (recetor da comunicação e interligação com servidor). Como por exemplo, a figura 3.8, representa uma estrutura de um NÓ de sensor (Sensor), um gateway (gateway Host) e um servidor (Network Server), desde a tecnologia as implementações da estrutura funcional:

- Sensor - **Estrutura do NÓ**: As camadas dos dispositivos definem-se pela camada

do próprio hardware. A camada de aplicação (Customer Application) define o seu processamento (código), e a camada LoRaWAN os protocolos para envio da mensagem. A camada de abstração de hardware (HAL) fornece uma abstração de hardware e a camada de ligação dos sensores ao hardware (SPI). NA camada física (PHY) e na camada de LoRa FSK encontra o seu protocolo de comunicação.

- **Gateway Host - Estrutura do GateWay:** O gateway tem as seguintes camadas, a camada de receção de pacotes/mensagens (packet forward), a camada de abstração de hardware (HAL), a camada de ligações (SPI/USB), a camada física (PHY), a camada de protocolo de IP (backhaul ip stack) e a ligação radio LoRa FSK
- **Network Server - Estrutura do servidor:** O servidor baseia-se numa estrutura simples de receção das mensagens vindas do gateway. Se forem encriptadas, serão desencriptadas. Este divide-se em camadas do servidor, a camada do protocolo LoRaWAN protocolo e a camada de protocolo de IP.
- **LoRa FSK - "Frequency Shift Keying",** é uma técnica de modulação na qual o sinal digital varia a frequência de uma onda analógica de acordo com valores pré-determinados. Esta comunicação é feita entre os NÓS e o Gateway.
- **Secure -** Utiliza os protocolos de segurança e ligação existentes em IP (Ethernet, 3G, WiFi), nas comunicações entre o gateway e o servidor.
- **Encrypted -** Pode utilizar mensagens encriptada dos NÓS até ao servidor.

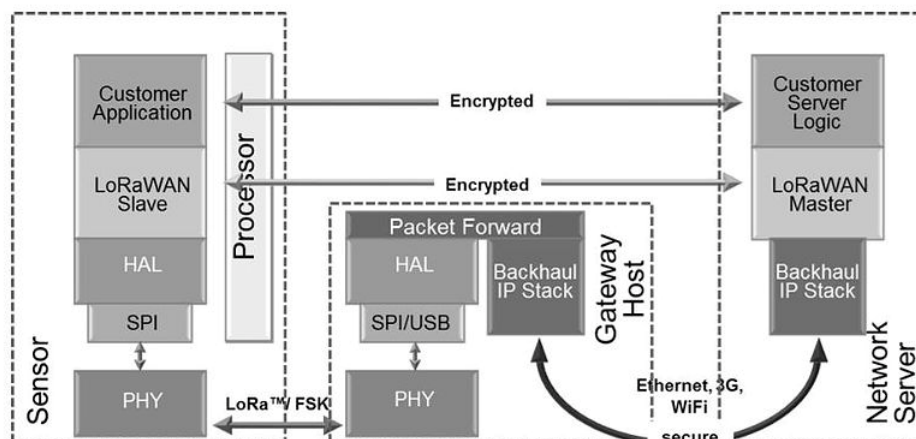


Figura 3.8: Exemplo de Comunicação NÓ - Gateway [10]

3.1.4 Módulos certificados para LoRa®

Existe uma grande variedade de módulos de rádio para a tecnologia LoRa, embora muitos não tenham as mesmas características, no que refere a distâncias máximas de transmissão. Neste sentido, iremos fazer uma abordagem aos que oferecem maior potência de envio e com frequências de acordo com as normas europeias:

Fabricantes de Chip para LoRa:

- **Microchip** - Módulo - RN2483 [67]
- **Semtech** - SX1272 [68].

Placas Radio com Chip:

- **Adafruit** - Adafruit Feather 32u4 RFM95 LoRa Radio - 868 MHz [11]

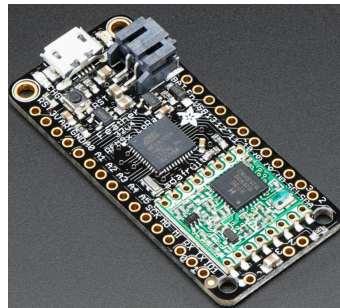


Figura 3.9: Adafruit [11].

- **SigFox** - [12]



Figura 3.10: Módulo SigFox [12]

- **Libelium** - Libelium RF SX1272 - [13]



Figura 3.11: Módulo Libelium [13].

Embora não se encontrem todos os módulos representados na descrição, poderemos observar na figura 3.12 em termos gerais;

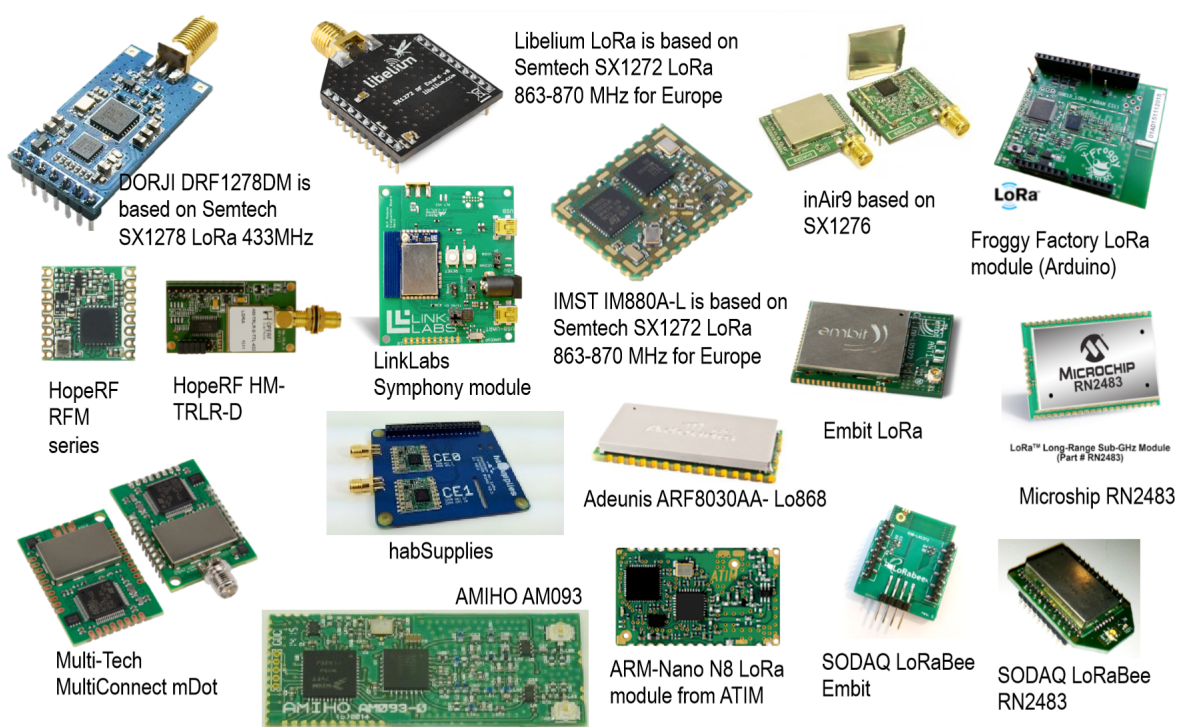


Figura 3.12: Modelos certificados para LoRa

3.1.5 Alcance em LoRa - Cooking-hacks

Módulo Libelium SX1272, é um radio com tecnologia LoRa, foram feitos testes por parte da equipa *Libelium Comunicaciones Distribuidas SL "ES"* [14]. Com este módulo obtiveram resultados impressionantes. Em Ambiente Urbano conseguiram comunicações a 2 Km sem perda de mensagens. Em ambiente rural a distância chegou a 22 Km, como podemos verificar na figura 3.13. A figura 3.13a demonstra uma implementação com várias distâncias, em que obtiveram os dados vindos dos NÓS de sensores. A figura 3.13b, sem obstáculos obtiveram uma distância impressionante de 21 Km.

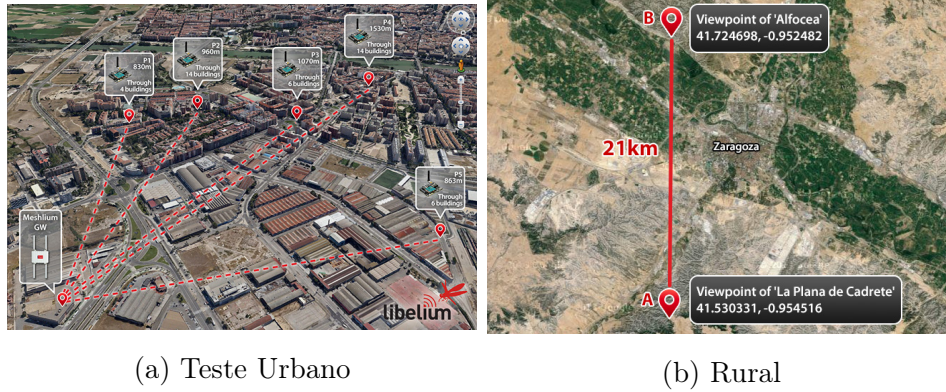


Figura 3.13: Modulo Libelium SX1272 [14]

3.1.6 Trabalhos Relacionados

Numa tecnologia nova, existem trabalhos de investigação relevantes, no entanto abordaremos os mais consistentes e atuais para o nosso projeto.

Em 2017 poderemos destacar as investigações feitas, na área da saúde, onde foram utilizadas técnicas de transmissão de dados em ambiente urbano, transmitidos via LoRa [69]. O trabalho conclui que a tecnologia pode ser uma alternativa para transmissão de dados a longas distâncias em áreas urbanas que não impliquem um grande volume de dados.

Ainda na área da saúde, em [70] é apresentado um projeto de apoio em áreas geográficas onde as comunicações moveis (GPRS / 3G / 4G) não chegam ou têm cobertura fraca. Assim podem ser recolhidos dados referentes à pressão arterial, glicose e temperatura, enviando os resultados para análise do profissional de saúde, evitando viagens aos centros de saúde por parte dos utentes mantendo o registo médico atualizado.

Aplicado à agricultura, em [71] são recolhidos dados de temperatura, de humidade do ar, da humidade da folha e da humidade do solo para a gestão de deteção e automatização dos sistemas. Com base nos dados recolhidos, o sistema atua no processo de rega.

Aplicado à gestão de tráfego urbano, foi desenvolvido um conceito novo com suporte a sensores magnéticos, que permite detetar o semáforo que tem maior volume de tráfego [72]. Assim este terá mais tempo de luz verde para uma melhor fluidez do trânsito. A tecnologia LoRa é usada para interagir entre os NÓs com o gateway e o servidor .

Já no corrente ano (2018), foram investigados alguns aspetos importantes, como as tendências das novas tecnologias em IoT. Considerando o custo da energia, foi desenvolvido e testado um gateway LoRaWAN alimentado com energia renováveis (painéis solares) [73]. Analisando os custos de implementação para uma autonomia maior, con-

sideraram uma viabilidade sustentável e verde para o ambiente. Em [74] é apresentado um sistema de controlo de qualidade da água baseado na tecnologia LoRa. Este recorre a sensores e gestão com suporte com serviços cloud. Realiza a monitorização da temperatura da água, turbidez (presença de matéria em suspensão na água, como argila, silte, organismos microscópicos e outras partículas), condutividade elétrica na água e pH. Utiliza energia renovável para a alimentação dos nós.

Relacionada, com o trabalho proposto neste documento, em [75] foi testado um sistema de gestão dos militares em operações táticas com suporte a GPS e LoRa, num teatro de operações. Este sistema localiza os combatentes no terreno através de um front-End com mapa do tipo google map, sendo apenas um sistema de monitorização GPS.

A tecnologia LoRa fornece uma grande cobertura em áreas amplas. No entanto, em meios urbanos esta cobertura diminui. Estudos recentes revelam que uma topologia de rede em estrela não é fiável em meio urbano, podendo-se perder um número elevado de pacotes de dados. Um desenvolvimento em malha (mesh) será uma solução para LoRa. Este foi testado na universidade em Taiwan (Chung Cheng University, Chiayi 62102, Taiwan) [76], com a implementação de uma rede em malha e uma rede em estrela com a tecnologia LoRa. Após a recolha dos dados observaram que a rede em estrela obteve 58.7% dos dados e a rede em malha obteve 88.49% dos dados.

Outro estudo realizado sobre a cobertura de LoRa em urbanizações [77], foi testada num edifício de sete (7) andares. Foi concluído que a rede esteve à altura dos desafios, no entanto está limitado no tamanho dos pacotes (242 bytes). Significa que quanto maior for o pacote de dados, maior será a taxa de erros da transmissão.

Num sistema ainda em desenvolvimento, todos os aspetos são importantes e a segurança é uma delas. Em [78] foi testado os conceitos de segurança, em que definiram como teste os aspetos, de um ataque de repetição, recuperação do texto enviado, modificação e falsificação da mensagem assim como um ataque à bateria. Com esta amostra, obtiveram as vulnerabilidades no protocolo LoRa. Por exemplo, a construção do AES-CTR pode levar a uma reutilização do fluxo de chaves. Na confirmação das mensagens (ACK) os invasores podem colidir com mensagens, mas para isso tem de conseguir controlar o gateway. A classe B não é protegida com encriptação, permitindo que o invasor esgote a sua bateria. Também encontraram falhas nos links que se servem do servidor para a aplicação, dependendo sempre da forma como é desenvolvido.

Capítulo 4

Sistema de Aquisição de dados Biométricos e de Localização - GateWay

Este capítulo apresenta o desenvolvimento do sistema de motorização de sinais e redes de sensores utilizando a tecnologia LoRa e sua comunicação, abrangendo ainda a metodologia de apresentação dos dados em tempo real do ponto de vista do gateway e da interface com o utilizador.

Apresenta também funções adicionais à rede, tais como o evento de alertas, apresentando uma proposta de desenvolvimento nas redes sem fios utilizando uma arquitetura de longo alcance.

4.1 Arquitetura Geral

A figura 4.1 apresenta a arquitetura geral do sistema proposto. Esta baseia-se em dois modelos fundamentais: o GateWay| Server e o(s) NÓ(S) de Sensor(es). Todo o sistema pode estar interligado a um servidor de CLOUD, para armazenamento e análise dos dados. É utilizado o protocolo LoRaWAN para comunicação sobre a tecnologia LoRa a 868MHz.

O **GateWay| Server** é o responsável pela receção mensagens e envio de alertas, o registo na base de dados e a apresentação dos dados ao utilizador. Este é constituído por dois (2) módulos distintos, mas que se interligam. O Gateway é o responsável pelas comunicações entre o Gateway e os NÓS, como também pela ativação do backup dos dados na base de dados e em ficheiros. O Server (servidor de base de dados) é responsável pela base de dados ativa. No mesmo módulo um Front-End, mostra os dados, comunicando com a base de dados.

Existe também a possibilidade de se configurar uma ligação a um servidor externo (CLOUD) para a salvaguarda dos dados e para a consulta por parte de outros intervenientes, nomeadamente pessoal médico.

Cada **NÓ de Sensores** é responsável pela recolha de dados, pelo envio dos mesmo ao Gateway e pelo envio de alertas. Também possui a capacidade de receção de alertas de outros utilizadores, assim como de visualizar os seus indicadores biometricos através

de um pequeno ecrã. Os seus sensores recolhem os valores de Oxímetria (SpO2), pressão atmosférica (PA), temperatura ambiente, coordenadas, hora, dia, altitude GPS e guarda estes valores no cartão de memória. O valor da PA ainda é usado para calcular a altitude pressão (QNE), que é diferente da altitude GPS.

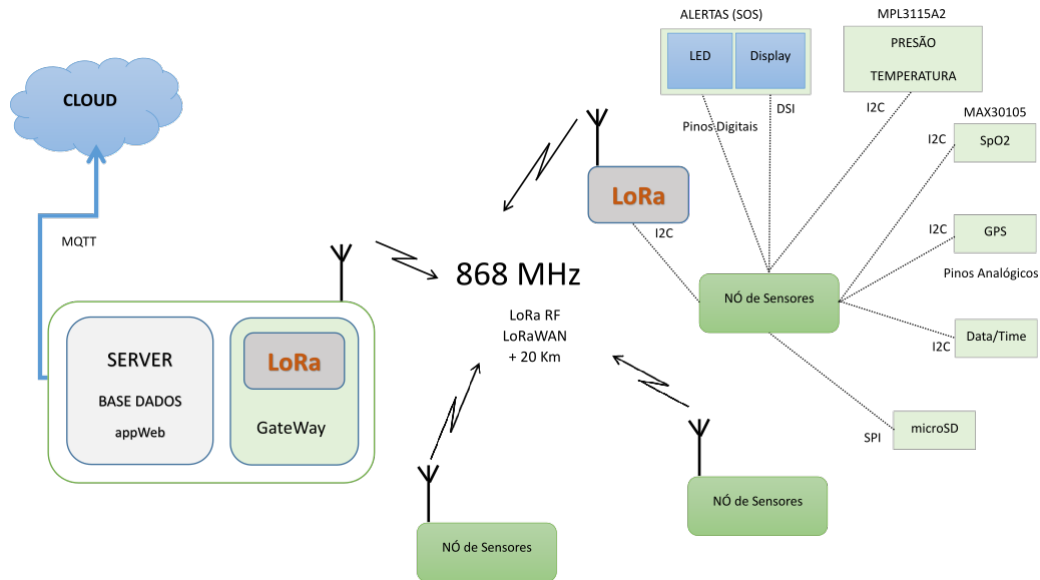


Figura 4.1: Arquitetura Geral

A arquitetura foi desenvolvida com base nas necessidades de uma comunicação bi-direcional, utilizado como referencia uma topologia de rede em estrela. Considerando que todos podem transmitir mensagens e todos podem receber, facilitando os alertas entre NÓS e o GateWay, se este não estiver ao alcance dos NÓS de sensores para a receção e envio de alertas, os NÓS tem a capacidade de comunicar entre si. Este sistema de rede, permite assim uma flexibilidade maior, face as necessidades do sistema, que depende da distancia entre os NÓS e o Gateway, mas também das barreiras físicas, nomeadamente da fuselagem das aeronaves (avião). No interior das aeronaves existe um bloqueio às comunicações, em qualquer frequência, seja GPS ou RF, não permitindo a entrada, nem a saída de comunicações. No entanto, é possível a comunicação entre NÓS, uma vez que estes estão dentro da aeronave.

Os alertas (SOS) são gerados pelos valores de SpO2 (sensor de Oxímetria), que o saltador (utilizador) tem em contacto com o seu corpo (pulso), controlando assim alguma anomalia ocorrida na administração do oxigénio, durante o voo na aeronave.

4.2 Gateway | Server

A implementação do *Gateway* e *Server* baseia-se no mesmo dispositivo, podendo assim ser mais versátil e abrindo o leque de opções na implementação. O dispositivo escolhido foi o Raspberry PI 3 b+, acoplando a si os restantes acessórios (Radio e ecrã).

4.2.1 Hardware

A escolha do hardware recaiu sobre versatilidade de um sistema capaz de acolher todas as tecnologias e funcionalidades necessárias e a compatibilidade do módulo das comunicações. Nesta visão, o Raspberry PI foi a melhor opção, dando a possibilidade de implementação de um Gateway e Server no mesmo dispositivo.

4.2.1.1 Raspberry PI 3 b+

A placa do Raspberry PI integra vários componentes, desde placa gráfica, ligação a periféricos (USB/RJ45/Audio/HMDI/Display/pinos GIO/DSI porta). Munido com um processador ARM Cortex-A53 de 64bits, faz deste mini computador uma escolha ideal para o projetos.



Figura 4.2: Placa Raspberry PI 3B

Fonte: <https://www.cooking-hacks.com/raspberry-pi-3-model-b>

4.2.1.2 Raspberry Pi 7" Touchscreen Display

O ecrã LCD de 7" para raspberry PI usa a porta DSI, este ecrã é uma maneira ágil de interagir com as aplicações, dando para observar os dados e a possibilidade de usar com os toques dos dedos, facilitando assim a sua utilização. [79]

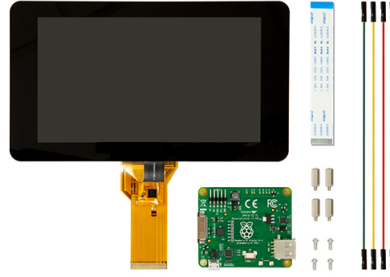


Figura 4.3: Display Touchscreen 7”.

Fonte: <https://www.electrofun.pt/display-touchscreen-7-raspberry-pi>

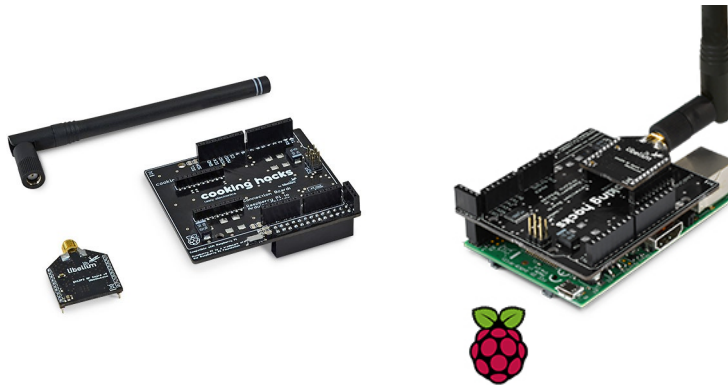
4.2.1.3 Modulo Libelium

Este modulo é constituído em três partes, a placa *Shield*, a placa radio *Libelium SX1272* e a antena bidirecional de 868MHz. [80]

A comunicação do modulo Libelium, são desenvolvidas em linguagem C, que inclui as bibliotecas do fabricante [81].

Bibliotecas utilizadas:

- #include "arduPiLoRa.h" - Biblioteca do Modulo Rádio
- #include <stdio.h> - Biblioteca de C



(a) Modulo Libelium.

(b) GateWay | Server

Figura 4.4: Representação do Modulo Libelium.

4.2.2 Software

O Raspberry PI (Gateway | Server) tem a finalidade de receber a comunicação de todos os NÓS de sensores, registar a sua informação em ficheiros (TXT) e numa base de dados, para posteriormente ser visualizada num Front-End (pagina web). Ainda trata os dados recebidos, e caso seja alguma mensagem de alerta (SOS), emite uma comunicação para todos os NÓS a avisar que existe um NÓ com o SpO2 fora dos

parâmetros normais. A figura 4.5 apresenta o diagrama de blocos da arquitetura do Gateway. Este é constituído por uma sistema operativo (OS) que sustenta toda a arquitetura do PI. Dentro deste tem uma arquitetura de Gateway e Server.

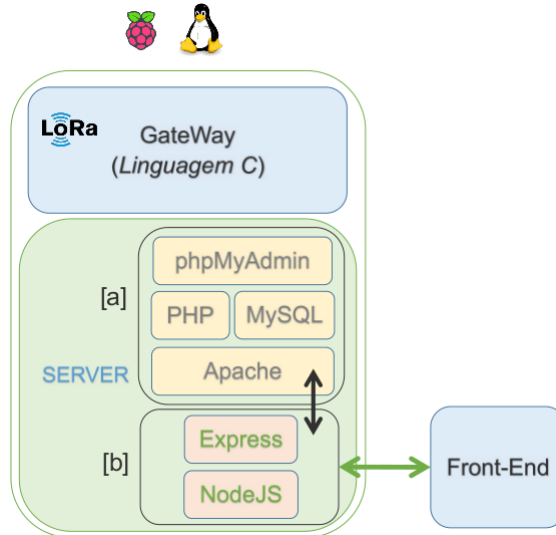


Figura 4.5: Arquitetura do Gateway (Raspberry PI).

- **Sistema Operativo (OS)** - O OS contém o Gateway | Server e o Front-End, a sua base é o Linux **JESSIE**, derivado do OS Debian, preparado pela equipa da "**Raspberry Pi Foundation**".
- **Gateway** - Utiliza a tecnologia **LoRa** na frequência de 868MHz e é programado em linguagem C (*dados.cpp_exe*). Este recebe as comunicações dos NÓS e avalia os valores. Quando os parâmetros de SpO2 estão fora dos valores normais envia um alerta a todos os NÓS de aviso com a mensagem de ID do NÓ e os caracteres de SOS (A1 SOS). O Gateway ainda recebe mais informação dos NÓS, nomeadamente, a pressão, temperatura e a localização. Essa informação é guardada em ficheiros e em Base de Dados, com a chamada do *Script* (*dados.js*). Estes Scripts são desenvolvidos e utilizam a API em NodeJS (*JavaScript*).
- **Server** - O servidor implementado foi utilizado para guardar os dados em base de dados e para apresentação dos dados ao utilizador, com um Front-End amigável.
 - (a) - **LAMPP** - Consiste num servidor local, Apache com PHP, MySQL e phpMyAdmin. A sua utilização permite a gestão fácil da base de dados com o suporte à página web.

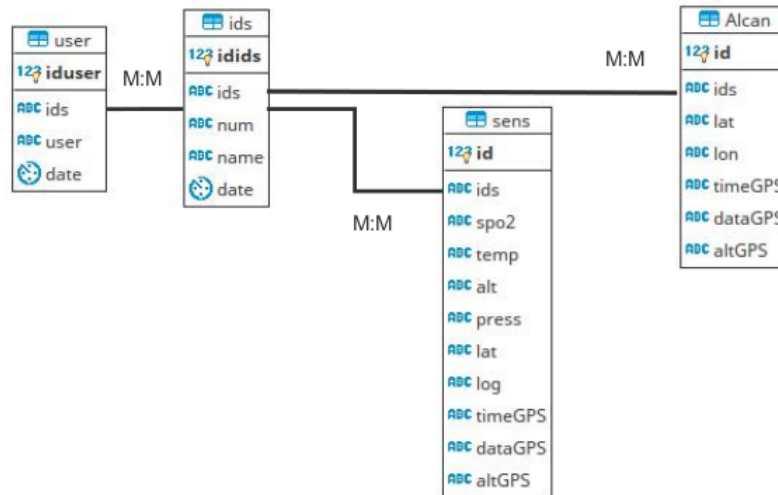


Figura 4.6: Estrutura da Base de Dados.

As tabelas representadas na figura 4.6, foram implementadas para satisfazer as necessidades da aplicação web. Existe uma tabela para utilizadores (user), uma tabela para sensores (ids), uma tabela para os valores dos sensores (sens) e uma tabela para teste de alcances (Alcan). (b) - **Framework Express** - Consiste numa API Web, com uma estrutura definida de regras. Esta framework utiliza a linguagem JavaScript, apoiada em NodeJS. O NodeJS permite a utilização como servidor e os seus módulos de interação com a framework, tais como o da base de dados MySQL.

- **Front-End** - O sistema de informação para a visualização dos dados é feita com o recurso ao Browser. Este recolhe a informação recebida pelo Gateway e apresenta-a. Esta visualização consiste em duas (2) partes fundamentais. A primeira representa os valores de SpO2, pressão atmosférica, altitude, temperatura ambiente. A segunda consiste na visualização da posição do saltador, através das coordenadas enviadas num mapa offline.

4.2.2.1 Algoritmo do GateWay | Server

Esta solução divide-se em dois algoritmos interligados, mas independentes.

- O algoritmo do Gateway está representado na figura 4.7. Este é responsável pelas comunicações, que recebe, trata e guarda os dados num ficheiro e emite os alertas recebidos. Adicionalmente, ativa os dois algoritmos do Server (dados.js e updatedb.js). Este inicia o módulo rádio, onde estão contidas todas as configurações das comunicações. Ao mesmo tempo inicia a contagem de trinta (30) segundos, para a ativação do algoritmo do backup de dados e atualização do ficheiro JSON. Após este processo, faz uma decisão, se existem mensagens para receber. Se não houver, volta ao processo de

escuta de mensagens. Se sim, compara os caracteres recebidos, para saber se é uma mensagem de SOS ou de valores normais (dados). Se forem valores normais, guarda a informação em ficheiro. Se não, envia alerta aos NÓS que recebem um SOS de um NÓ com o ID. Após fazer nove (9) envios volta para a escuta de recepção de mensagens.

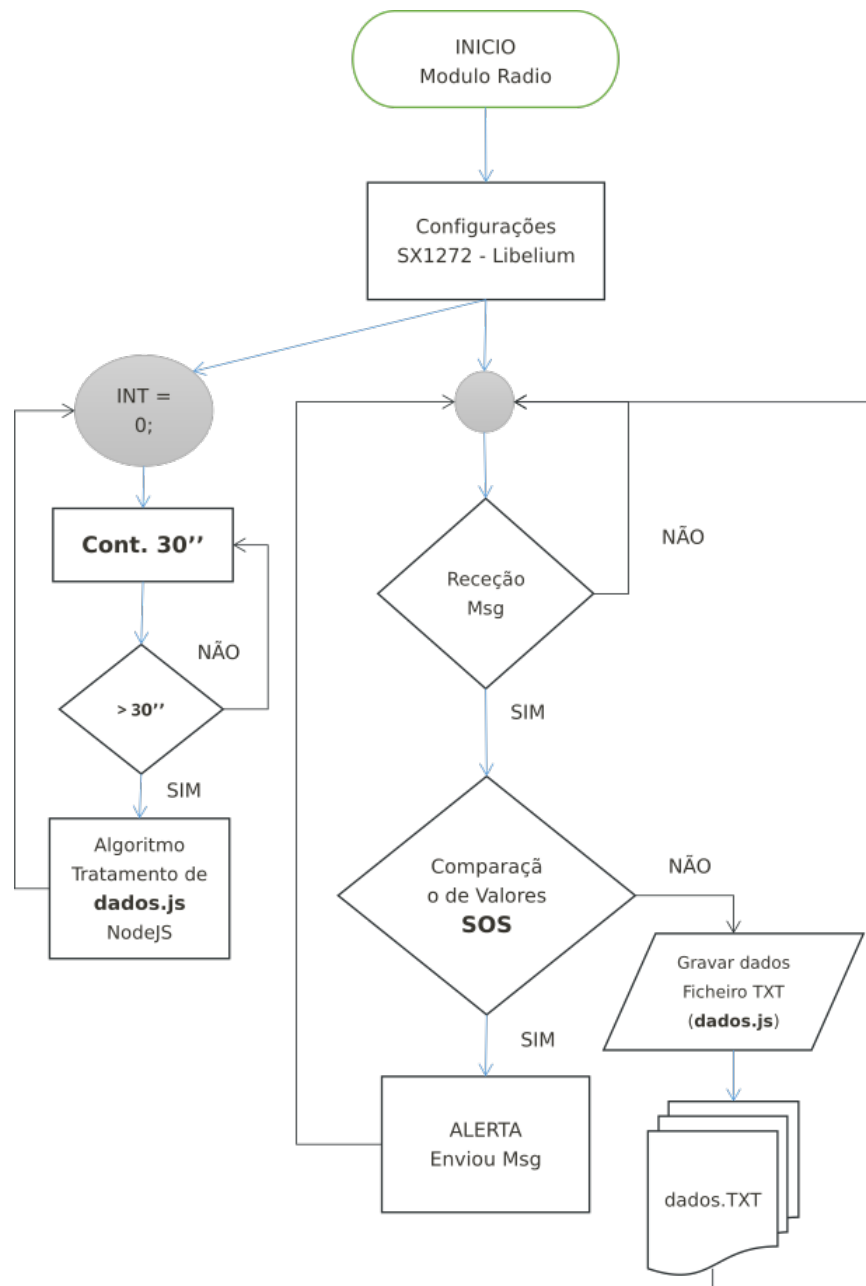


Figura 4.7: Algoritmo Gateway.

- O algoritmo de backup de dados que ocorre nos trinta (30) segundos representado na figura 4.8. Este ativa o script (dados.js) que efetua o tratamento de dados do servidor com o Gateway usando NodeJS e MySQL. É responsável por ler os ficheiros guardados das comunicações e garantir que não vão existir dados repetidos. Guarda na base de dados e faz copias (backup) dos dados em novos ficheiros de texto, separados pela sequencia recebida.

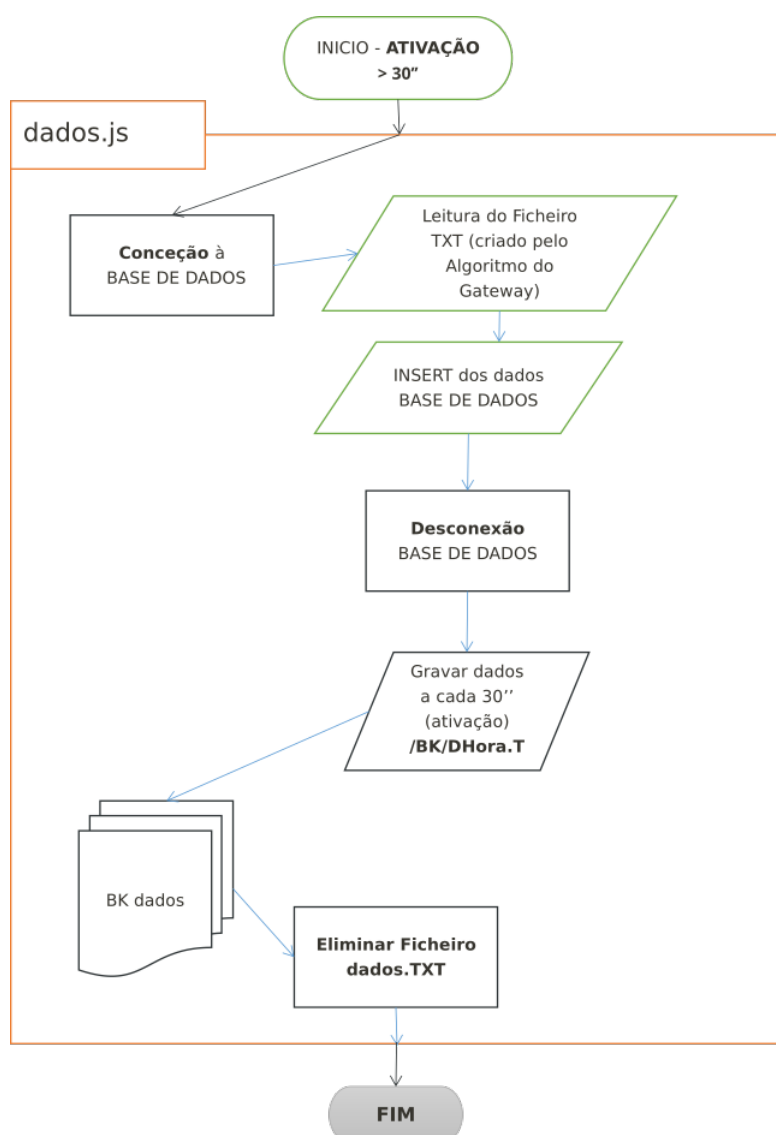


Figura 4.8: Algoritmo Backup.

Após os trinta (30) segundos, este faz a conexão à base de dados (MySQL), faz a leitura do ficheiro criado e atualizado pelo algoritmo do Gateway. Após este processo, faz a inserção dos dados na base de dados, desliga a conexão e guarda os dados num ficheiro de backup com data e hora. Depois apaga o ficheiro de leitura dos dados que foi criado pelo gateway, garantido assim que não haverá dados repetidos.

Este algoritmo de backup, ainda ativa o script de atualização de dados no ficheiro

JSON (updatedb.js). Este é ativado após o backup de dados e efetua a consulta a base de dados, pedindo a informação dos últimos registos dos NÓS, registando essa informação no ficheiro tancos.json para atualização nos mapas da posição atual.

4.2.2.2 Funcionalidades do Gateway

(i) Código do modulo rádio

O código completo do ficheiro **dados.cpp_exe**, mostrando abaixo os pontos mais relevantes e ou decisivos do código.

- Configurações do Módulo;

O módulo permite que se altere as suas configurações, assim na nossa implementação foi definido o canal 10 (865,20 MHz) e a potencia Máximo ('M') de 14 dBm, de transmissão e receção, a sua identificação na rede foi definido como oito (8) na receção de mensagem (setNodeAddress(8)) e na transmissão como global para todos os recetores na rede com o endereço de zero (0) (sendPacketTimeout(0, msg)).

(ii) Formação da mensagem (RX)

A receção e a divisão da mensagem dos NÓS é um fator importante, tanto para os alertas como para guardar os dados recebidos. No RX do GateWay foram desenvolvidas as funções de separação da mensagem, a função de guardar os dados e as configurações do rádio.

- Divisão da Mensagem;

Esta divisão da mensagem é fundamental para se saber se existem alertas (SOS). Assim, utilizou-se a função *strtok*, para a separação dos grupos criados pelos NÓS antes de enviar.

```
div = strtok(msg, sib);
```

- Detecção de SOS;

Depois da separação da mensagem, e da verificação dos caracteres, se esta coincidir com a mensagem de *SOS*, é ativada a função de alerta. Esta emite o alerta com o recurso à Transmissão (TX).

- Dados recebidos

Adicionalmente, os dados recebidos são guardados em ficheiro, criado pelo Gateway e atualizado a cada receção de novos dados. A mensagem guardada tem a informação enviada dos NÓS, que é composta por ID do NÓ de sensor (ID), Oxímetria (SpO2), Temperatura, Pressão barométrica, Data, Hora, Altitude Pressão

(Alt1), Latitude (Lat), Longitude (Log), Altitude pelo GPS (Alt2), receção Via radio (BD).

A mensagem tem o seguinte formato:

ID#SpO2#Temp#PA#DATA#HORA#Alt(1)#Lat#Log#Alt(2)#BD

Por exemplo;

1A#99#25#1010#100#15/10/2018#20:00:00#39.496707#-8.350728#100#BD

(iii) Backup de dados

O backup de dados é feito pelo *Script dados.js*, com o recurso do NodeJS. Este atua em cada sessenta (30) segundos. O *Script dados.cpp_exe* faz a sua chamada e este inicia e desliga-se após a conclusão do backup.

A chamada ao script é feita com o recurso ao *system* que atua através da API do NodeJS.

```
system ( "node_/var/www/html/SMEI/dados.js" );
```

A execução do Script cria um novo ficheiro em TXT com data e horas dos backup. Apaga o ficheiro criado pelo GateWay e o mesmo Script executa a inserção na base de dados. A figura 4.9 mostra um exemplo de uma lista dos ficheiros guardados em backup. Esta inserção será repetida no item referente as funções do Server.

Dados guardados:

ID#SpO2#Temp#PA#DATA#HORA#Alt(1)#Lat#Log#Alt(2)#BD

1B#100#23#1013#15/10/2018#20:00:00#100#39.496707#-8.350728#100#BD

Ficheiros de backup em txt (dados.js);

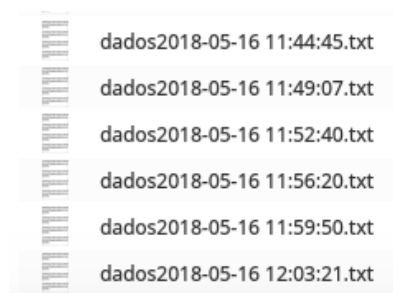


Figura 4.9: Ficheiros de Backup.

(iv) **Envio de dados (TX)** O envio de dados pelo Gateway, é feito quando este recebe um alerta (SOS).

- **Envio dos alertas (SOS)** Quando é detetado uma mensagem com alerta, este envia nove (9) vezes a mensagem de (SOS) para todos os NÓS com o ID do NÓ que deu o alerta.

4.2.2.3 Funcionalidades do Server

Os dados, recebidos pelo rádio (`dados.cpp_exe`), são guardados em `dados.TXT` e passado trinta (30) segundos são lidos e guardados pelo ficheiro **`dados.js`**. Após esta operação, o ficheiro `TXT` (`dados.TXT`) é eliminado, salvaguardando, que não existem dados repetidos a serem inseridos na base de dados. O ficheiro do rádio quando verifica que não existe ficheiro `TXT`, cria um novo e continua a escrever as mensagens recebidas.

No ficheiro (Script) `dados.js`, todo o seu código é importante para a ação a ser executada, tanto o backup com a inserção dos valores na base de dados. No entanto considera-se que o mais relevante é a inserção na base de dados.

A inserção na base de dados (BD), é feita em dois (2) processos distintos, a leitura do ficheiro `TXT` e a separação dos valores vindos na mensagem.

A separação da mensagem usou-se o recurso à API do NodeJS, chamando as funções *`fs.createReadStream`*, *`require('readline').createInterface`* e *`num.split`*;

Devido à extensão da linha a mensagem, esta foi separada em duas (2) e só foi colocada uma linha de amostra. O primeiro ID é o da base de dados, os restantes são iguais as amostras anteriores nos ficheiros `TXT`:

ID, ID, SpO2, PA, DATA, HORA, Alt(1), Lat, Log, Alt(2), BD

Por exemplo;

```
(652 , '1B' , '99' , '25' , '1305' , '15/10/2018' , '20:00:00' , '100' ,  
'39.496707' , '-8.350728' , '100' , 'BD')
```

4.2.2.4 Arranque automático no inicio

O sistema Operativo (Linux), foi configurado para o arranque automático do ficheiro Gateway (`dados.cpp_exe`). Este arranque permite ao utilizador uma facilidade de interação, sendo ou não conhecedor dos sistemas operativos linux. Assim, foi criado um ficheiro novo de interligação, chamado pelo Script de inicio da secção do OS.

A linha acrescentada ao código de inicio de secção no ficheiro *`autostart`* (`/etc/xdg/lxsession/LXDE-pi/autostart`).

Linha acrescentada;

```
@lxterminal -e "/etc/LoRa.sh"
```

Ficheiro novo de interação **LoRa.sh** em `/etc/`;

```
lxterminal --command="sudo ./dados.cpp_exe"
```

- A figura 4.10 apresenta o fluxograma de arranque no SO a secção do utilizador inicia e chama o *Script* **LoRa.sh**, que por sua vez inicia o Gateway (`dados.cpp_exe`). Após

iniciar, começa a executar o algoritmo descrito no item Gateway | Server.

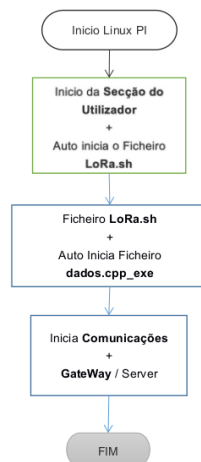


Figura 4.10: Fluxograma de Arranque

4.2.3 Front-End

O Front-End é uma aplicação web com o recurso ao servidor (server), desenvolvida sobre a API do NodeJS e com uma estrutura do ExpressJS. Nesta aplicação pode-se observar e gerir os dados dos NÓS de Sensores. Este vai dar a possibilidade dos utilizadores do Gateway visualizarem a atividade dos NÓS.

O início da aplicação é automática, renderizando para a página de opções,

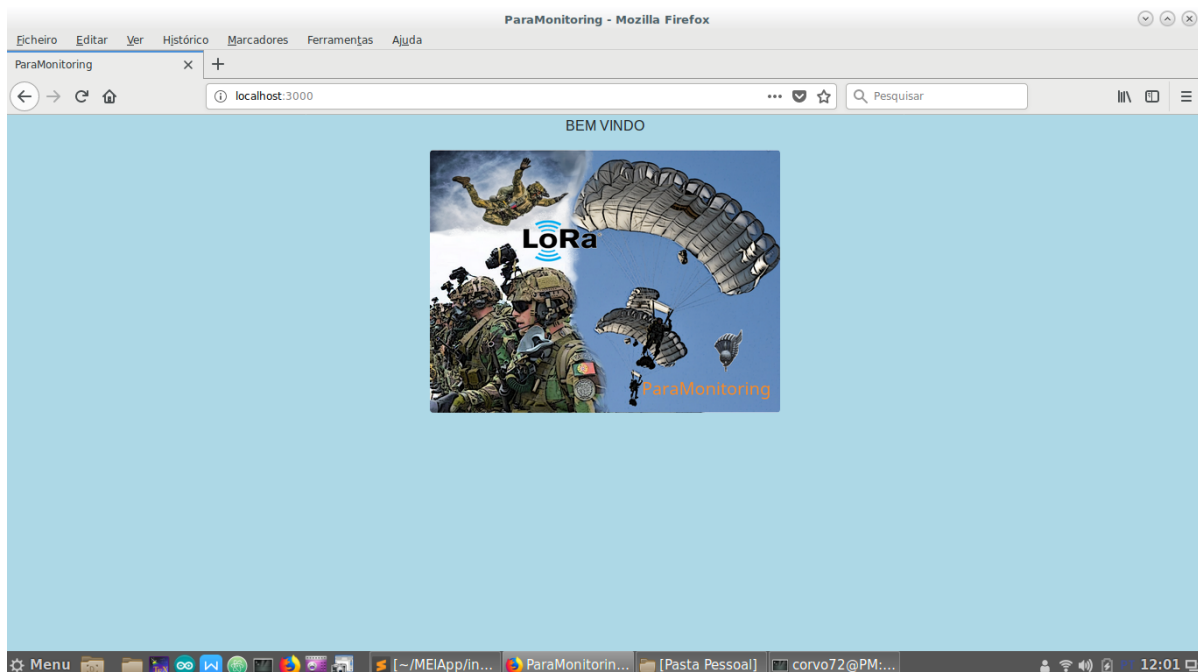


Figura 4.11: Início da aplicação

A página de opções dá a possibilidade de navegação, entre as páginas de gestão e o mapa off-line do geoposicionamento dos NÓS.

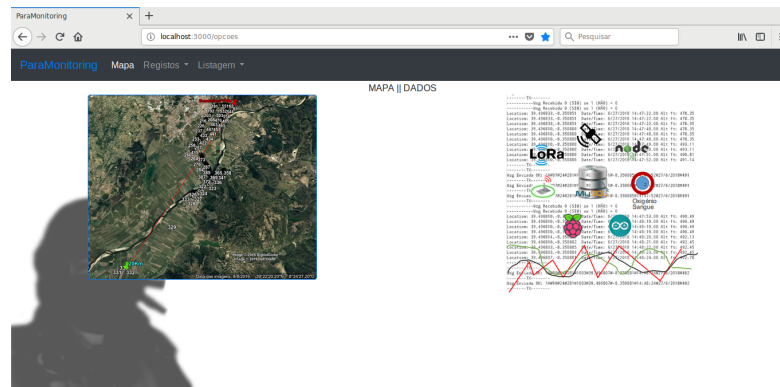


Figura 4.12: Página de navegação

(i) Gestão de dados

Na gestão foram considerados os utilizadores e os NÓS de Sensores, podendo assim atribuir ao utilizador o NÓ que vai utilizar nesse dia. Assim, podemos registar os NÓS que forem necessários e os seus utilizadores de uma maneira acessível.

Outra possibilidade é apresentar a visão geral dos dados, onde se lista a atividade dos sensores, por ordem decendente, deixando os valores mais antigos para o fim.

• Gestão de utilizadores

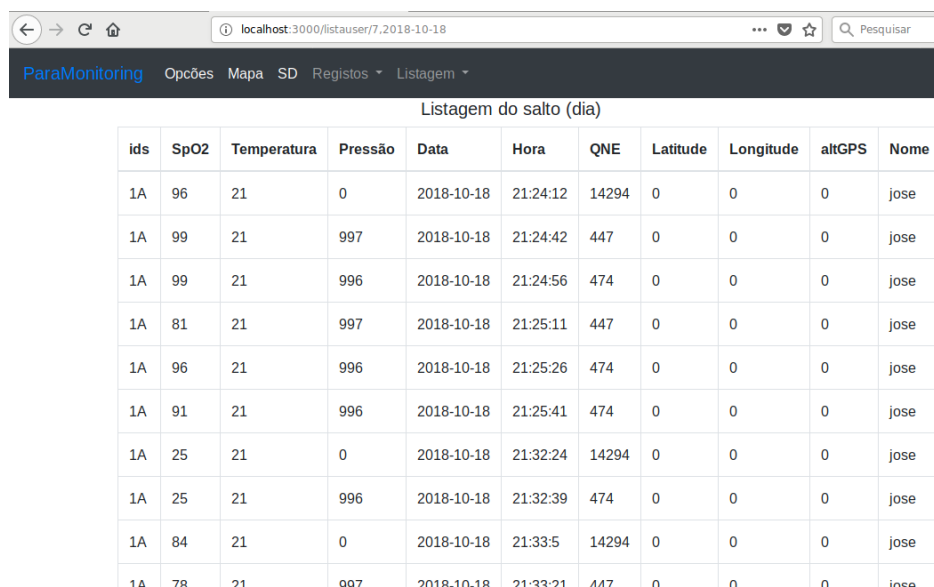
Nesta gestão podemos fazer CRUD (Create, Read, Update e Delete), nos utilizadores. A data vai ser a referencia do dia do salto, para que se possa cruzar os dados com o NÓ atribuído naquele dia.

Nº	Nome	Data	ID NÓ	
1	jose	2018-10-18	1A	<button>Edit</button> <button>Delete</button> <button>Ver</button>
2	Antonio	2018-06-01	2A	<button>Edit</button> <button>Delete</button> <button>Ver</button>
3	Pedro	2018-06-01	3A	<button>Edit</button> <button>Delete</button> <button>Ver</button>
4	Miguel	2018-06-02	1A	<button>Edit</button> <button>Delete</button> <button>Ver</button>

Registar

Figura 4.13: CRUD de utilizadores

Na opção "ver" pode-se criar a lista dos registo efetuados na base de dados para aquele saltador (user) no dia escolhido (dia do salto). Neste tipo de salto a grande altitude com suporte a Oxigénio, só é aconselhável efetuar um por dia, assim deve ser registado cada vez que o saltador utilizar o equipamento, com a respetiva data.



ids	SpO2	Temperatura	Pressão	Data	Hora	QNE	Latitude	Longitude	altGPS	Nome
1A	96	21	0	2018-10-18	21:24:12	14294	0	0	0	jose
1A	99	21	997	2018-10-18	21:24:42	447	0	0	0	jose
1A	99	21	996	2018-10-18	21:24:56	474	0	0	0	jose
1A	81	21	997	2018-10-18	21:25:11	447	0	0	0	jose
1A	96	21	996	2018-10-18	21:25:26	474	0	0	0	jose
1A	91	21	996	2018-10-18	21:25:41	474	0	0	0	jose
1A	25	21	0	2018-10-18	21:32:24	14294	0	0	0	jose
1A	25	21	996	2018-10-18	21:32:39	474	0	0	0	jose
1A	84	21	0	2018-10-18	21:33:5	14294	0	0	0	jose
1A	78	21	997	2018-10-18	21:33:21	447	0	0	0	jose

Figura 4.14: Lista do Utilizador por salto

- Gestão de NÓS

Na gestão de NÓS de sensores, serve para adicionar NÓS ao sistema, ou para eliminar algum quando este se danificar.

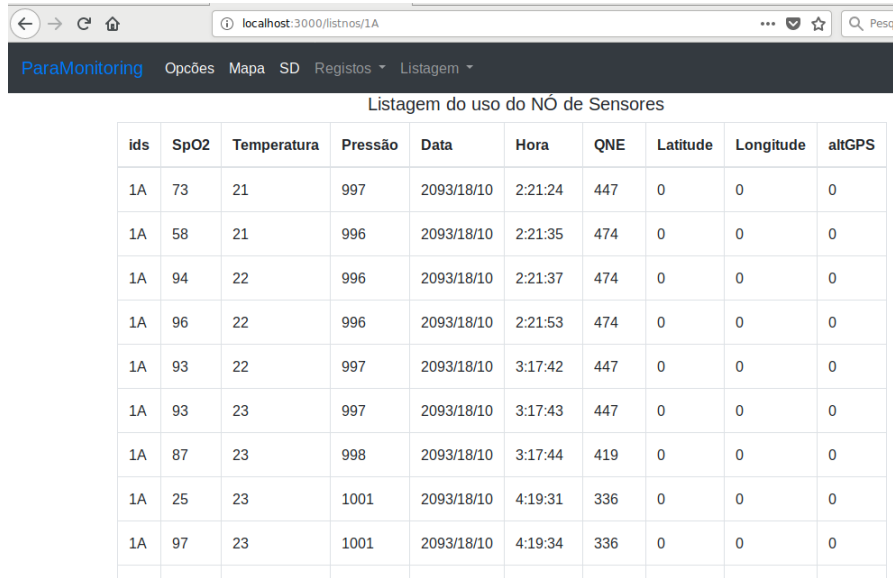


Nós de Sensores

N°	ID do NÓ	
1	1A	<button>Edit</button> <button>Delete</button> <button>Ver</button>
2	2A	<button>Edit</button> <button>Delete</button> <button>Ver</button>
3	3A	<button>Edit</button> <button>Delete</button> <button>Ver</button>

Figura 4.15: CRUD de NÓS

Na opção "ver", é possível listar todos os dados contidos na base de dados desse NÓ.



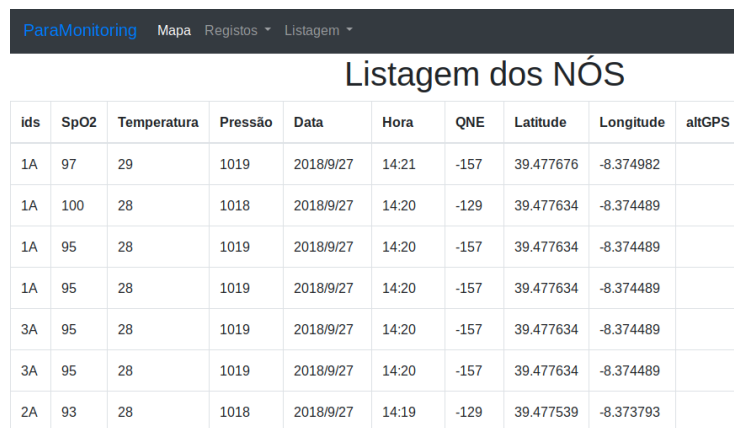
The screenshot shows a web browser at localhost:3000/listnos/1A. The application has a dark header with 'ParaMonitoring' and navigation links: 'Opções', 'Mapa', 'SD', 'Registos', and 'Listagem'. The main content is titled 'Listagem do uso do NÓ de Sensores' and displays a table with 10 columns: 'ids', 'SpO2', 'Temperatura', 'Pressão', 'Data', 'Hora', 'QNE', 'Latitude', 'Longitude', and 'altGPS'. The table contains 10 rows of data for sensor ID 1A, recorded on 2093/18/10.

ids	SpO2	Temperatura	Pressão	Data	Hora	QNE	Latitude	Longitude	altGPS
1A	73	21	997	2093/18/10	2:21:24	447	0	0	0
1A	58	21	996	2093/18/10	2:21:35	474	0	0	0
1A	94	22	996	2093/18/10	2:21:37	474	0	0	0
1A	96	22	996	2093/18/10	2:21:53	474	0	0	0
1A	93	22	997	2093/18/10	3:17:42	447	0	0	0
1A	93	23	997	2093/18/10	3:17:43	447	0	0	0
1A	87	23	998	2093/18/10	3:17:44	419	0	0	0
1A	25	23	1001	2093/18/10	4:19:31	336	0	0	0
1A	97	23	1001	2093/18/10	4:19:34	336	0	0	0

Figura 4.16: CRUD de NÓS

- Listagem

A listagem geral serve para visualizar os acontecimentos, podendo observar se os NÓS de Sensores, estão operar bem ou se existe alguma falha.



The screenshot shows a web browser at localhost:3000/listnos/1A. The application has a dark header with 'ParaMonitoring' and navigation links: 'Mapa', 'Registos', and 'Listagem'. The main content is titled 'Listagem dos NÓS' and displays a table with 10 columns: 'ids', 'SpO2', 'Temperatura', 'Pressão', 'Data', 'Hora', 'QNE', 'Latitude', 'Longitude', and 'altGPS'. The table contains 8 rows of data for sensor ID 1A, recorded on 2018/9/27.

ids	SpO2	Temperatura	Pressão	Data	Hora	QNE	Latitude	Longitude	altGPS
1A	97	29	1019	2018/9/27	14:21	-157	39.477676	-8.374982	
1A	100	28	1018	2018/9/27	14:20	-129	39.477634	-8.374489	
1A	95	28	1019	2018/9/27	14:20	-157	39.477634	-8.374489	
1A	95	28	1019	2018/9/27	14:20	-157	39.477634	-8.374489	
3A	95	28	1019	2018/9/27	14:20	-157	39.477634	-8.374489	
3A	95	28	1019	2018/9/27	14:20	-157	39.477634	-8.374489	
2A	93	28	1018	2018/9/27	14:19	-129	39.477539	-8.373793	

Figura 4.17: Listagem da atividade

(ii) Geoposicionamento dos NÓS de Sensores

O geoposicionamento é um elemento importante quando o paraquedas se encontra aberto. Deste modo, o utilizador do Gateway | Server, pode ver em tempo real na aplicação a posição de cada saltador. Esta posição ainda ganha mais relevo quando o salto é noturno. A aplicação atualiza os pontos com o recurso ao ficheiro tancos.json, e mostra a posição atual no mapa. Este ficheiro utiliza uma fotografia aérea que se encontra georreferenciada.

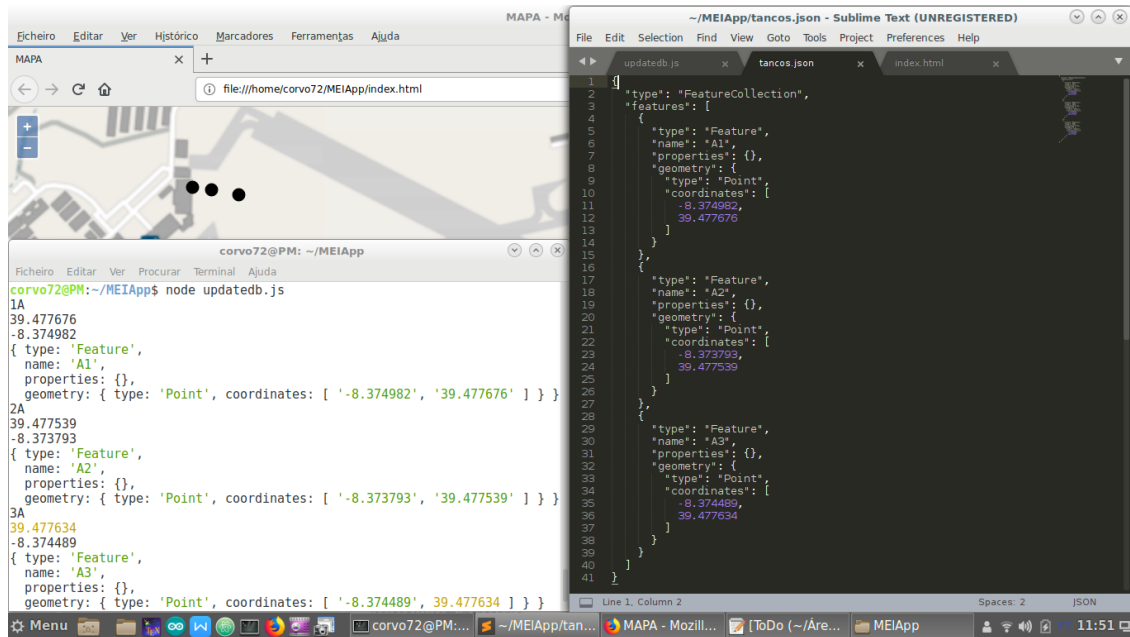


Figura 4.18: NÓS no Mapa

(iii) Atualização de dados json

A atualização de dados é feita pelo *Script* `updatedb.js`, com o recurso aos módulos de NodeJS. Este atua em cada cinco (5) segundos, num ciclo constante. O script inicia ao mesmo tempo que o server nodeJS, através da iniciação feita pelo utilizador. O utilizador clica duas vezes no ícone no ambiente de trabalho "AppWeb". Este executa um ficheiro *Shell* "JsonUpdate.sh", que inicia o servidor e o loop de atualização do ficheiro `tancos.json`. Este ficheiro "updatedb.js" encontra-se em `"var/www/html/SMEI/public/MEIApp/updatebd.js"`. A atualização do ficheiro `tancos.json`, altera as coordenadas nos NÓS de Sensores contidas na base de dados:

```
{ "type": "FeatureCollection",
  "features": [
    { "type": "Feature",
      "name": "A1",
      "properties": {},
      "geometry": {
        "type": "Point",
        "coordinates": [
          -8.374982,
          39.477676 ] }},
```

4.2.4 Inserir dados dos SDCards

A salvaguarda dos dados dos cartões SD dos NÓS de Sensores, é um fator importante para a sua observação e análise. Foi desenvolvido uma API que funciona juntamente com o Front-End. A API (*penSDBk.js*), é acionada quando o utilizador clica o botão no Front-End. A API recorre dos script (*Move_TXT*, *CopyTexto_TXT*, *insertdados* e *Delect_TXT*). Cada script tem uma função específica, tornando assim possível a cópia do cartão SD para a base de dados. Esta API tem como suporte uma manipulação de dados e utiliza as pastas (*TXT*, *CopyTXT* e *dados*) "fig.4.19".

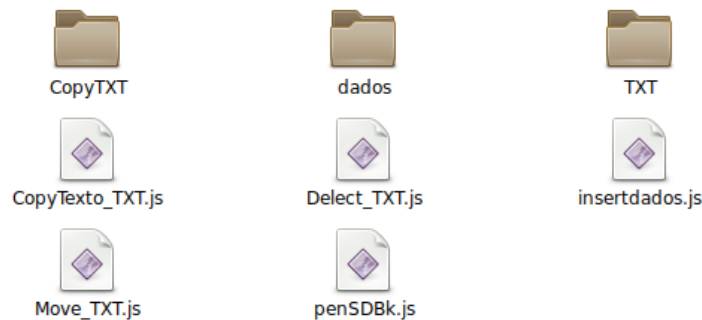


Figura 4.19: API backup SD

O seu funcionamento baseia-se em processos "child_process", que executa por ordem lógica. O Front-End indica ao utilizador o que deve fazer (colocar pen) "fig.4.20", e ao clicar no botão "fig.4.21", este executa a API. A *penSDBk.js* utiliza o script *Move_TXT*, que copia os ficheiros do SDCard para a pasta *TXT* e move-os com novo nome numérico para a pasta "CopyTXT".

O próximo processo é a execução do ficheiro *CopyTexto_TXT.js*. Este cria um ficheiro único com os dados de todos os ficheiros da pasta "CopyTXT" para a pasta "dados". O script *insertdados.js* lê o ficheiro criado pelo *CopyTexto_TXT.js* e insere os dados na base de dados. Por fim, é chamado o script "*Delect_TXT.js*", que elimina todos os ficheiros das pastas que ficam prontas para novos dados. Este processo necessita a instalação dos módulos *fs*, *path*, *node-cmd* e *shelljs*, com npm.

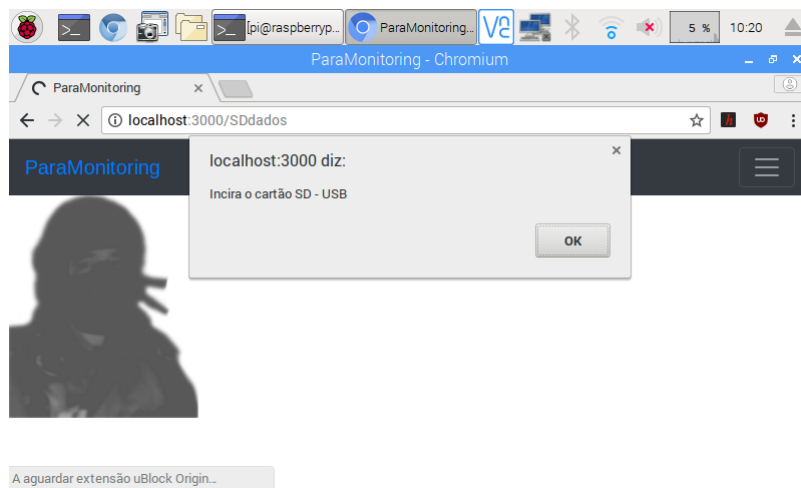


Figura 4.20: Front-End - Aviso SD

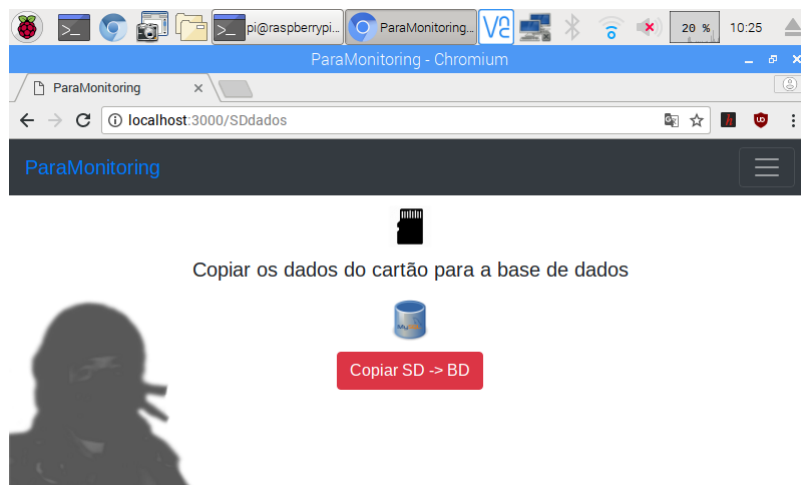


Figura 4.21: Front-End - Clicar SD

Capítulo 5

Sistema de Aquisição de dados Biométricos e de Localização - NÓ de Sensores

Este capítulo apresenta o desenvolvimento do sistema dos NÓS de sensores. Neste sistema dos NÓS abrange os sensores que efetuam a recolha dos dados, a utilização da tecnologia LoRa nas comunicações utilizadas.

Serão abordados os aspetos do desenvolvimento do hardware e do software implementado, assim como o conceito seguido para um desenvolvimento consistente.

5.1 Arquitetura Geral do NÓ de Sensores

A figura 5.1 apresenta a arquitetura do NÓ de sensores. Este usa a plataforma Arduino MEGA (NÓ de sensor) e tem a finalidade de enviar as dados dos sensores para o gateway, registar os dados em ficheiros, dar alertas e receber alertas.

- Sensores - No NÓ de sensores existem vários tipos de sensores e de dispositivos, estes utiliza vários tipos de ligações. Estas ligações utilizam I2C, SPI, DSI, pinos digitais e analógicos. Como podemos observar na figura 5.1, o MAX30105 (SpO2), o MPL3115A2 (Pressão), o Modulo do relógio (DS1307RTC) e o modulo libelium utilizam uma ligação I2C, o GPS utiliza pinos analógicos, o microSD utiliza SPI, o Display utiliza DSI, o Led e botão os pins digitais.
- NÓS de Sensores - Utiliza-se a tecnologia **LoRaWAN** em modo estrela, podendo existir comunicação entre cada nó e o gateway e entre cada nó. Esta opção torna os NÓS mais independentes do GateWay. Cada NÓ recolhe informação dos seus sensores e envia, quando o alcance ao GateWay for viável (20Km). No entanto, cada NÓ regista a informação num ficheiro (TXT), salvaguardando os dados. A receção de mensagens de alerta é meramente de aviso, acendendo um Led vermelho que apenas se desliga, quando o utilizador pressionar o botão. Mas

se o alerta for do próprio NÓ, este é avisado no *display* (ecrã) em flash. Se a informação vier do gateway, a informação visual (Led) é também ativada.

- microSD - A salvaguarda dos dados recolhidos pelos sensores é também uma função NÓS, atendendo a possibilidade de não haver comunicações com o gateway. Recorre-se a esta modalidade para garantir a integridade dos dados e também que não haverá dados perdidos. No entanto, esta salvaguarda não guarda os valores do GPS.

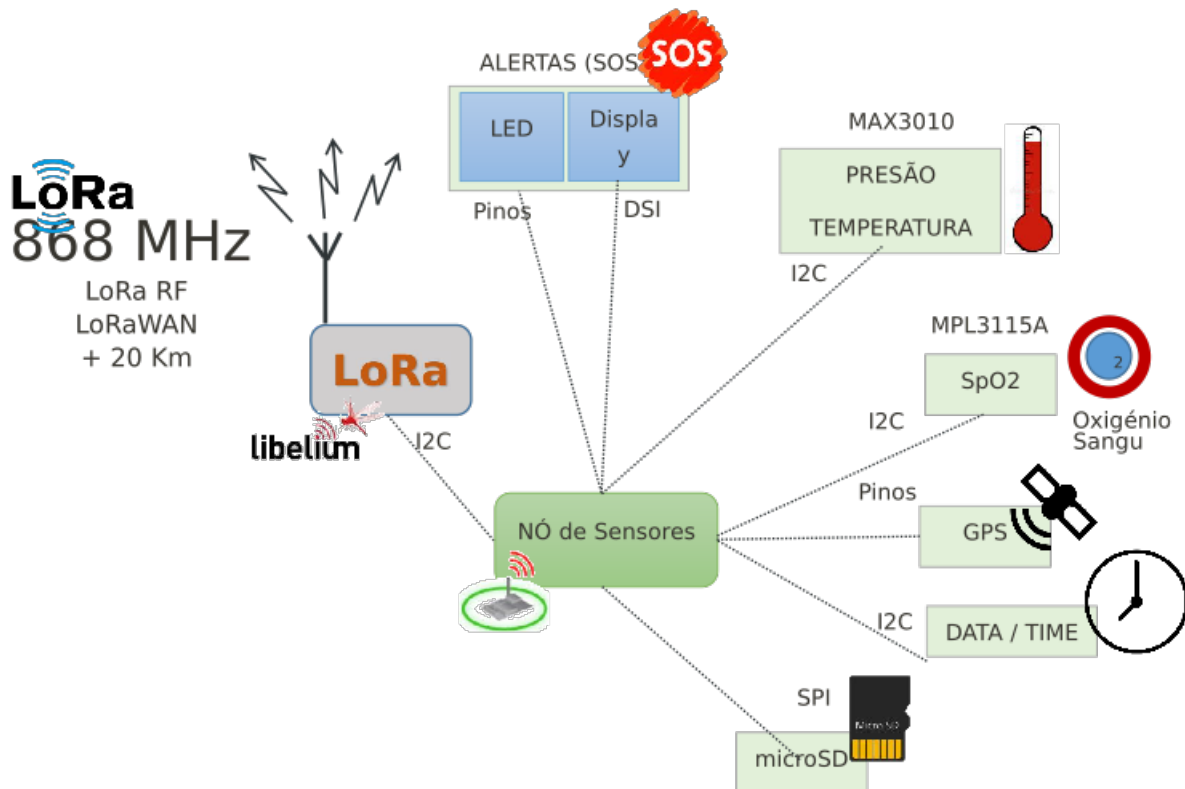


Figura 5.1: Arquitetura do NÓ de Sensores

5.2 Hardware

O projeto é baseado na plataforma Arduino MEGA. A necessidade de ter um microprocessador com a capacidade de armazenamento e RAM era fundamental, já que, as bibliotecas dos sensores eram bastantes exigentes (SpO2) nos seus requisitos de armazenamento como na velocidade de processamento. Outro dos fatores da sua escolha foi a placa intermédia de ligação ao Libelium (multiprotocolo), que tem bibliotecas muito estáveis e de fácil configuração, e nessa perspetiva não havia outra solução melhor, no momento.

5.2.1 Módulo de processamento e sensores

O Arduino MEGA é uma placa de desenvolvimento, que permite testes de programação através da porta USB, com uma boa capacidade no seu processador e armazenamento faz do MEGA uma placa versátil, tem disponível pinos para ligar vários dispositivos (sensores/relés/Leds/etc..).

A figura 5.2 apresenta o Arduino Mega 2560 é uma placa microcontroladora baseada no ATmega2560, possui 54 pinos de entrada / saída digitais, 16 entradas analógicas, 4 UARTs, 2 I2C (paralelo), 1 SPI, um conector ICSP e trabalha com um oscilador de cristal de 16 MHz. [82]

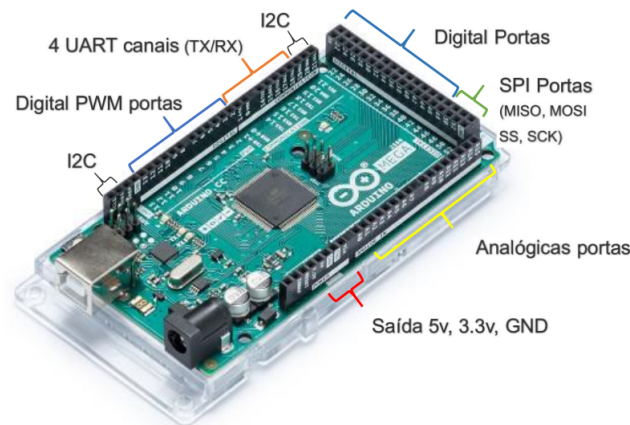


Figura 5.2: Placa Arduino MEGA [15]

5.2.1.1 Sensor de Oxímetria (MAX30105)

O módulo MAX30105 é dos sensores mais completos, permite várias leituras diferentes tais como, detetar distâncias, frequências cardíacas, partículas do tipo fumo ou sangue oxigenado (SpO₂).

O SpO₂ é a saturação do oxigénio no sangue. O ser humano tem uma taxa normal que varia de 95 a 100%. Quando a saturação atinge valores inferiores a 90%, indica que a oxigenação de sangue está deficiente e que o paciente (utilizador), está a entrar nos limites que o organismo suporta. Em termos médicos, uma pessoa que apresente valores inferiores a 90% encontra-se com problemas de saúde (asma, pneumonia, enfisema, insuficiência cardíaca ou doenças neurológicas). Num saltador de grande altitude, este valor indica que a admissão do oxigénio está deficiente (mascara com fuga, mal colocada ou consumo total da garrafa).

O sensor utiliza três (3) Leds diferentes (Vermelho, Verde e infravermelho), para leitura e recolha dos dados. O cálculo efetuado para os valores de SpO₂, utiliza um algoritmo

de apoio. Este recolhe os registos efetuados nos ciclos de medição do sensor e disponibiliza os dados nas variáveis. Este algoritmo utiliza dois (2) ciclos de leitura, o primeiro acumula cinquenta (50) registos e efetua mais cinquenta (50) para que, com ambos os ciclos, possa calcular o SpO2. [83]

Bibliotecas Utilizadas:

```
#include <Wire.h>           - Comunicação I2C
#include "MAX30105.h"        - Biblioteca do sensor
#include "spo2_algorithm.h"  - Biblioteca SpO2 Algoritmo
```

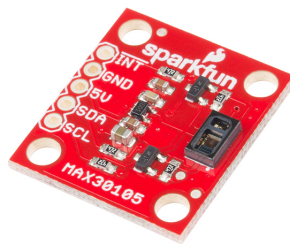


Figura 5.3: Sensor de Oxímetria [16]

5.2.1.2 Sensor de pressão, altitude e temperatura (MPL3115A2)

O módulo MPL3115A2 é um sensor de baixo consumo e de grande precisão. Deteta mudanças da pressão barométrica, altitude e temperatura. É utilizado para estações meteorológicas, para drones (UAV) e aplicações similares. [84]

Bibliotecas utilizadas:

```
#include <Wire.h>           - Comunicação I2C
#include "SparkFunMPL3115A2.h" - Biblioteca do sensor
```



Figura 5.4: Sensor de Pressão e Altura [17]

5.2.1.3 Módulo GPS (NEO-6M - UART)

O módulo NEO-6M é uma placa desenvolvida pela "Waveshare", em que é composto por um conjunto de antena com o u-blox 6. O módulo u-blox NEO-6 é da família dos receptores de GPS autônomos com o mecanismo de posicionamento de alto desempenho e precisão, deteta 50 canais e possui um Time-To-First-Fix (TTFF) inferior a 1 segundo. Permite encontrar satélites instantaneamente, elimina fontes de interferência e reduz os efeitos multi-percurso. [85]

Bibliotecas utilizadas:

```
#include <TinyGPS++.h>      - Biblioteca desenvolvida por Mikal Hart [86]
#include <SoftwareSerial.h>  - biblioteca arduino (pinMode)
```



Figura 5.5: GPS NEO6MV2 [18]

5.2.1.4 Módulo Display OLED LCD 0.96" I2C IIC

O módulo Display Oled é um visor de I2C (Vcc,GND, SCL e SDA), que para além da alimentação só necessita de duas linhas de comunicação (SCL, SDA) um para o relógio (CLK) e outra para os dados (MOSI). Fácil de implementar, com várias opções de representação no ecrã e um tamanho de 0.96", as bibliotecas utilizadas foram desenvolvidas por a equipa adafruit.com. [87]

Bibliotecas utilizadas:

```
#include <SPI.h>              - Biblioteca do Arduino para comunicação entre
.                             Master-Slave do MOSI
#include <Wire.h>              - Biblioteca do Arduino para I2C
#include <Adafruit_GFX.h>      - Biblioteca para sistema de graficos (adafruit)
#include <Adafruit_SSD1306.h>  - Biblioteca para o processamento do Oled
.                             (adafruit)
```

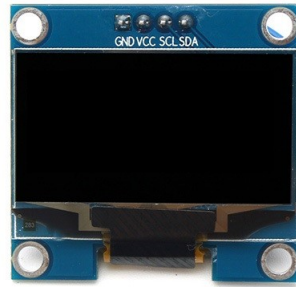



Figura 5.6: Display OLed [19]

5.2.1.5 Módulo LED RGB 5mm

O módulo Led RGB é uma placa já preparada para ligação sem preocupações com a tensão, esta incorpora o Led e as resistências necessárias à utilização de 5V. O Led faz de base as três (3) cores, o vermelho, o verde e o azul, com a possibilidade de mostrar mais cores, conjugando com o código de cores (0 a 255). [88]



Figura 5.7: Led RGB [20]

5.2.1.6 Módulo TCA9548A I2C Multiplexer

O módulo TCA9548A é uma placa de múltiplos endereços I2C (SCL,SDA), permite até oito (8) escravos (Slaves) ligados ao mesmo tempo, gerindo com uma única porta I2C ao Mestre (Master). [89]

Bibliotecas utilizadas:

#define TCAADDR 0x70 - Biblioteca desenvolvida pela equipa adafruit [89]



Figura 5.8: TCA9548A I2C Expander [21]

5.2.1.7 Módulo microSD

O módulo microSD permite a gravação de dados num cartão de memória, de pequenas dimensões e de grandes capacidades. É utilizado no projeto para salvaguardar os dados dos sensores. Este procedimento é fundamental, ficando assim salvaguardados os dados que foram transmitidos e não foram recebidos nas comunicações. [90]

Bibliotecas utilizadas:

`#include <SD.h>` - Biblioteca SDcard (dfrobot.com)

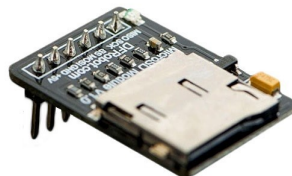


Figura 5.9: Modulo microSD [22]

5.2.1.8 Módulo RTC

O módulo do relógio é utilizado para registar a data e hora dos acontecimentos efetuados pelos os NÓS de Sensores. O módulo tem de ser programado, com dois ficheiros, um *SetTime* e *ReadTest*, este último permite confirmar se o relógio fica programado com a data e hora corretas.

Bibliotecas utilizadas

`#include <RTCLib.h>`

`#include <TimeLib.h>`

`#include <DS1307RTC.h>`



Figura 5.10: Modulo RTC DS1307 [22].

5.2.1.9 Módulo Libelium para Arduino

O módulo é composto por três (3) partes distintas, tendo uma placa (Shield Multiprotocol) de ligação ao arduino UNO, um módulo rádio SX1272 e uma antena bidirecional. A placa tem dois "Socket", um vocacionado para o módulo RF-LoRa ou equivalentes e sendo o outro vocacionado ou para modules de RFID, XBee, Bluetooth, WiFi, 3G, GPRS e CAN Bus Module. [80]

Neste projeto foi feita uma adaptação da placa multi-protocolo para a placa arduino MEGA. Os encaixes dos pinos não são os mesmos, existindo a necessidade de conectar os pinos A4 e A5 aos pinos 20 (SDA) e 21 (SCL) do MEGA.

Bibliotecas utilizadas:

- `#include <arduinoUtils.h>` - Cooking API libraries (Placa multi-protocolo)
- `#include "arduinoLoRa.h"` - inclui a biblioteca de SX1272 e SPI

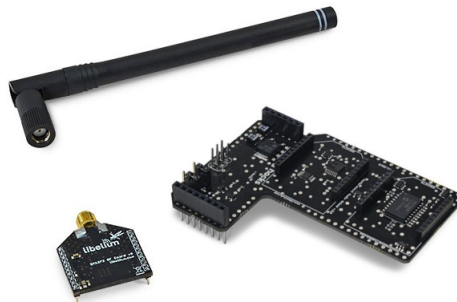


Figura 5.11: Modulo Libelium [23]

5.2.2 Arquitetura de Hardware

A arquitetura do NÓ de Sensores é constituída por um conjunto de ligações, sensores e dois (2) Arduinos. Esta arquitetura forma um NÓ de MASTER e SLAVE, onde o MASTER é responsável pelas comunicações com o gateway e outros NÓS, enquanto que o SLAVE é responsável leitura dos sensores e tratamento dos dados.

5.2.2.1 Diagrama das ligações dos pinos

A figura 5.12 representa, as ligações dos pinout dos dois (2) Arduinos (MASTER e SLAVE).

Arduino MASTER tem varias ligações, as das comunicações do modulo Libelium que utiliza uma placa com os pinos de A0 a A5 analogicos e D0 ao D7 digitais. O modulo GPS utiliza os pinos analógicos A10 e A11. O Led vermelho utiliza o pinos 35 e led RGB utiliza os pinos 30, 32 e 34. O botão utiliza o pino 39. A comunicação de dados entre o MASTER e o SLAVE é feito pela ligação I2C com os pinos no MASTER SCL1 e SDA1. Esta utiliza o protocolo I2C de mestre e escravo, recorrendo à biblioteca "wire", mestre pedindo ao escravo o dados disponíveis.

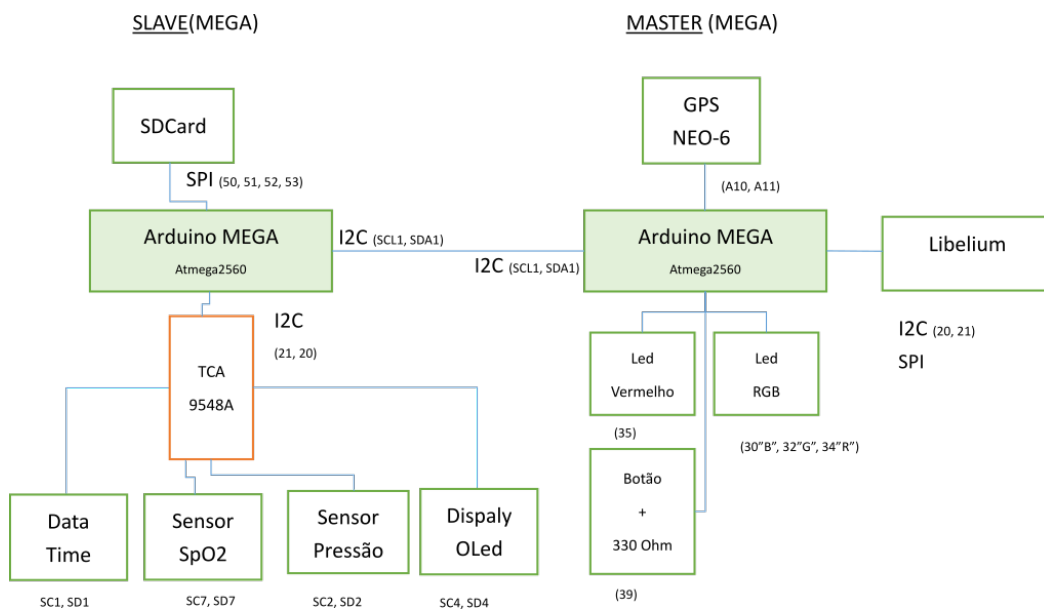


Figura 5.12: Arquitetura de hardware.

Arduino SLAVE tem conectado o TCA expander pela ligação I2C nos pinos SCL(21) e SDA(20). Este tem os sensores ligados aos pinos I2C. O SC7 e SD7 liga o sensor de oximetria, o SC2 e SD2 liga o sensor de pressão e o SC4 e SD4 liga o display.

O SC1 e SD1 liga o módulo do relógio. A comunicação de dados entre os dois Arduino é realizada nos pinos I2C. No arduino SALVE são utilizados os pinos 21 e 20.

5.2.2.2 Diagrama do NÓ - Esquema elétrico

A figura 5.13, representa o esquema elétrico de todas as ligações e hardware que compõe o NÓ de Sensores, numa representação gráfica.

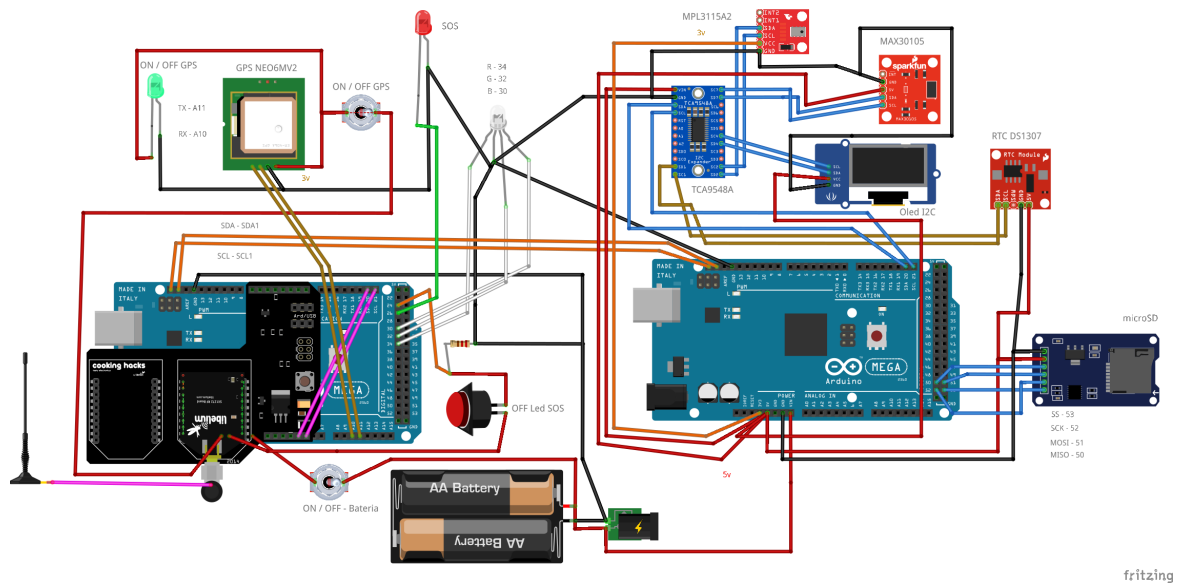


Figura 5.13: NÓ de Sensores.

- Componentes do conjunto de NÓ de sensores:

MEGA-MASTER (Radio)

- Arduino MEGA: Placa Multi-protocolo, Modulo Radio Libelium (SX1272) a 868MHz, Antena Bidirecional 868Mhz, Modulo GPS NEO-6M com ON/OFF para alimentação, Led Vermelho (SOS) com botão de desligar, Modulo Led RGB, Resistência 330 ohm;

MEGA-SLAVE (Sensores)

- Arduino MEGA: Placa Expansão I2C TCA9548A Multiplexer, Modulo MAX30105 (Oxímetria), Sensor: Modulo MPL3114A2 (Pressão barométrica), Modulo Display Oled, Modulo RTC - Relógio, ON/OFF para energia do NÓ e o Modulo microSD

5.2.3 Circuito desenvolvido

Foi desenvolvida no IPT, uma placa para as ligações entre os dois Arduinos MEGA e os sensores. Esta placa é fundamental para alojar e facilitar todas as conexões existentes.

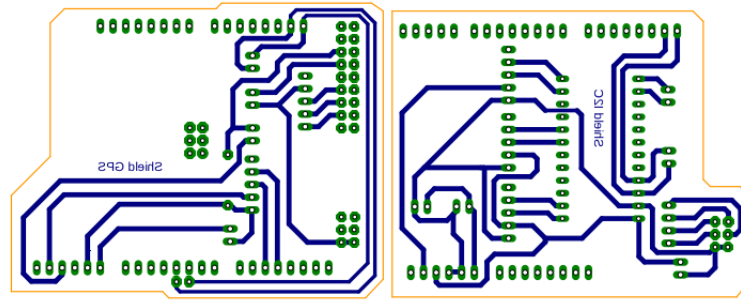
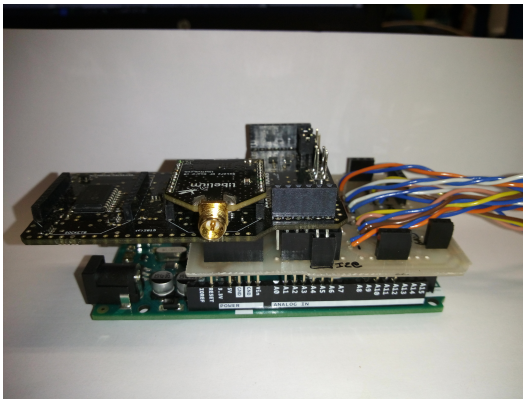
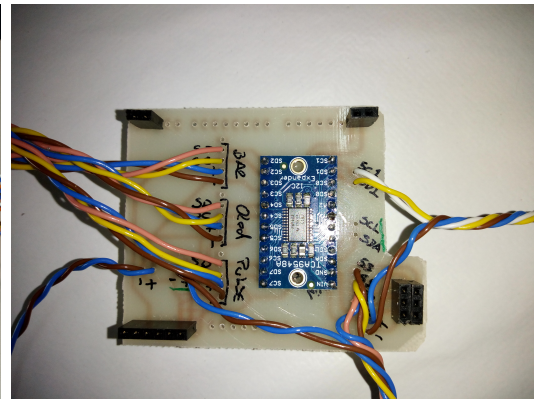


Figura 5.14: Shields dos NÓS



(a) Placa no MASTER



(b) Placa SLAVE

Figura 5.15: Implementação das placas

5.3 Software

Pretende-se que cada NÓ de Sensores, recolha dados de SpO2, pressão atmosférica, altitude, temperatura, coordenadas da posição, data e hora atual. Esta informação vai ser enviada pelo módulo de rádio. Neste pressuposto foi definido um algoritmo (figura 5.17), em que se considerou o funcionamento de dois Arduinos em separado, havendo sincronismo na transferência de informação entre os dois, que funcionam em modo MASTER-SLAVE.

- **MASTER:** Este fica vocacionado para a comunicação (TX/RX), da aquisição de dados do GPS e gestão dos pedidos ao SLAVE dos valores de sensores. Este módulo também gere a receção de possíveis SOS vindos dos outros nós, mostrando em Led, sendo possível desativar este Led com o botão de pressão.

O seu princípio de funcionamento **dentro do avião:** Inicia o módulo, faz o pedido ao SLAVE dos valores dos seus sensores. Após a receção dos valores do SLAVE, faz o tratamento da mensagem. Se os dados forem válidos (valores

normais, ERRO ou SOS) envia a mensagem para o gateway. Caso contrário, efetua a receção e escuta as mensagens. Esta escuta serve para verificar se existe algum SOS de outros NÓS ou do GateWay. Se houver algum alerta (SOS) acende o Led Vermelho que só apaga se o utilizador carregar no botão. As mensagens de SOS enviadas pelo próprio NÓ, são com suporte a ACK (confirmação da receção), o emissor (TX) fica à espera da resposta do recetor (RX). Caso receba a resposta zero (0) indica que RX recebeu a mensagem, caso receba um (1), o recetor não recebeu mensagem. Neste caso, vai enviar novamente a mensagem de SOS, até quatro tentativas. Nesta finalidade de envio de SOS o NÓ começa a piscar o Led vermelho, indicando que é aquele NÓ que o enviou. Se não houver alerta, volta a ler os valores descritos no início.

Princípio de funcionamento **fora do avião em paraquedas aberto**: A sua funcionalidade é quase idêntica ao anterior, só que é ligado o GPS. Após ser válido o sinal de GPS, as receções de SOS deixam de se efetuarem. Caso os valores do SALVE (SpO2) não forem lidos ou contenham erros, o envio (TX) é feito com as coordenadas do GPS. Assim é possível saber onde se encontra o NÓ de Sensor no mapa.

- **SLAVE**: Fica responsável pela aquisição dos dados dos sensores (MAX30105, MPL3115A2) e de mostrar no display a informação do SpO2 e alertas do próprio NÓ.

O seu princípio de funcionamento faz a leitura dos sensores (SpO2, Pressão, Temperatura, data e hora), junta os vários dados obtidos e aguarda o pedido do MASTER, para o envio dos dados por I2C. O SLAVE tem um módulo de microSD, onde são guardado todos os dados em ficheiros TXT. Ficou definido no Led RGB as Cores Verde, Azul e Vermelho. Estas cores correspondem aos processos a executar do SLAVE. A cor verde indica que o GPS já detetou os satélites e tem dados válidos, a cor azul indica o envio de mensagens e a cor vermelha indica a receção de mensagens.

5.3.1 Módulo de comunicações (MASTER)

Foi implementado um código coerente e simples, para uma comunicação fluida, dando assim ao MASTER uma capacidade de resposta aceitável e notória, tanto na sua memória, como no espaço de armazenamento.

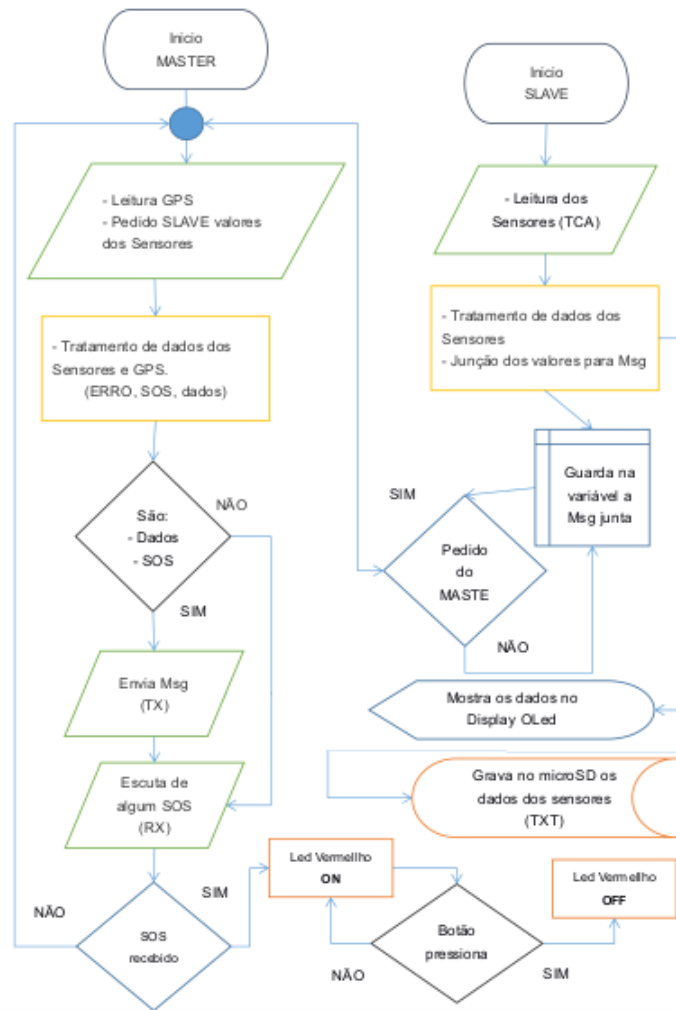


Figura 5.16: Algoritmo MASTER-SLAVE

- Configurações do Módulo Rádio

O módulo rádio foi configurado com o canal (frequência) dez (10), potência máxima de 14dBm e a sua identificação na rede, sendo cada NÓ um numero diferente (1A é 1).

Canal (frequência)- Canal 10 que corresponde à frequência 865,20 MHz;

```

sx1272.setChannel(CH_10_868);
  
```

Potencia (dBm) - Letra 'M' corresponde 14 dBm;

```

sx1272.setPower('M');
  
```

Identificação na Rede;

```

sx1272.setNodeAddress(1);
  
```

5.3.1.1 Ciclo de envio

Foi implementado um ciclo de cinco (5) repetições de TX e demonstra no Led RGB na cor azul. Neste ciclo são efetuadas as funções no processo de reconhecimento

e de fim da mensagem (*strtok*), a comparação dos dados recebidos do SLAVE (*strcmp*) e da junção da mensagem (*strcpy* e *strcat*).

Também houve a necessidade de construir a mensagem com separadores (#), visto que esta vem do SLAVE sem separadores. Esta limitação do tamanho da mensagem entre os dois dispositivos é condicionada pela biblioteca "wire".

- Função do reconhecimento do fim da mensagem;

A mensagem vinda do SLAVE pelo I2C, pode conter erros. Estes ocorrem quando existe uma variação do tamanho da mensagem. Este tamanho é definido para o máximo permitido pela comunicação Wire, que suporta 32 Bytes, o equivalente a 30 caracteres (char). Isto acontece pela variação de caracteres na variável, que varia em função da altitude registada (ex: baixa altitude 250 ft e grande altitude 30000 ft).

Assim, foi necessário implementar uma função para reconhecer o fim da mensagem, de modo a que se possa desprezar o que não pertence à mensagem de dados. Ao ser encontrado o símbolo & nos dados, estes são guardados na variável e o restante é desprezado.

Msg I2C: 1A9925101320181020&

```
limpa = strtok(msg1, "&");
```

- A comparação dos dados recebidos do SLAVE;

A comparação dos dados recebidos do SLAVE, é realizada recorrendo à comparação de variáveis padrão. Esta compara se o SLAVE enviou a mensagem "ERRO", "SOS" e "dados dos sensores".

Mensagem de "ERRO";

Nesse estado o MASTER envia ao Gateway os valores dos sensores a zero e completa a mensagem com as coordenadas, Data e Hora vindas do GPS. Se as coordenadas não estiverem válidas, envia zeros até ter posição válida.

Msg Enviada: 1A#0#0#0#0#15/10/2018#20:00:00#39.496707#-8.350728#100#BD

```
strcmp(limpa, "ERRO") == 0)
```

```
"Envia as Coordenadas do NÓ, Data, Hora e altitude GPS"
```

Se a mensagem for "SOS", envia os caracteres do alerta com "SOS" com o ID do NÓ.

```
strcmp(limpa, "SOS") == 0)
```

Este envio é feito em modo broadcast, utilizando o endereço de envio zero(0). Assim, todos os nós recebem e interpretam o endereço do nó que lançou o alerta.

A confirmação da receção das mensagens foi realizada através da biblioteca do LoRa, com a função `ACKRetries` e `ACK`.

O `sendPacketTimeoutACKRetries`, é uma função que fica em espera da resposta do recetor, esta recebe o estado (0 - Recebido / 1 - Não recebido). Como foi explicado anteriormente, se receber o estado recebido, continua o algoritmo, caso contrário, vai enviar novamente a mensagem.

```
sx1272.sendPacketTimeoutACKRetries(0, TXSos);
```

O `receivePacketTimeoutACK`, Também é uma função do LoRa que responde à solicitação do `ACKRetries` ou simplesmente `ACK`. A nossa escolha pelo `ACKRetries`, foi devido ao facto de que esta tenta novamente o envio / receção, enquanto `ACK` só informa do estado (0/1). O tempo de escuta do rádio (timeout) é de dez segundos.

```
sx1272.receivePacketTimeoutACK(10000);
```

```
- "Envia dados de SOS"
```

Na comparação da mensagem vinda do I2C, se não reconhecer estes dois estados anteriores, então envia a mensagem normal dos dados. Esta é complexa, devido a receção dos caracteres virem juntos do SLAVE.

Como a comunicação está restringida ao tamanho da mensagem entre o SLAVE e o MASTER, é necessário reorganizar a mensagem para que o Gateway possa compreender e a possa inserir na base de dados.

A mensagem é composta em duas partes distintas. A primeira corresponde aos valores da variável para a junção de cada dado, específica a cada sensor. A segunda é a construção da mensagem com o separador (#).

1º tarefa: Extração dos valores (extrato do código).

// Resolução de ser 95 e 100 de 2 caracteres a 2 em SpO2:

```
spo2slave = msg1[2];  
spo2slave += msg1[3];  
temperat = msg1[4];  
temperat += msg1[5];
```

// Resolução de ser 1013 e 9xx de 4 caracteres a 3 então:

```
pressao = msg1[6];  
if (pressao = msg1[6] > 1){  
    pressao = msg1[6];  
    pressao += msg1[7];  
    pressao += msg1[8];
```

2º Tarefa: junção dos valores e do separador (#) (extrato do código). A função *strcpy* e *strcat*, faz a transferência dos dados na variável com os valores de cada sensor.

```
strcpy (TXMsg, idNoConv);           // 1A
strcat (TXMsg, sep);                 // #
strcat (TXMsg, spo2slaveConv);       // SpO2
strcat (TXMsg, sep);                 // #
strcat (TXMsg, temperatConv);        // Temperatura
```

Após estas duas tarefas, é efetuado o envio da mensagem para o Gateway.

```
sx1272.sendPacketTimeout(8, TXMsg);

"Envia dados depois da junção"
```

5.3.1.2 Ciclo de Receção (SOS)

A receção das mensagem de alerta (SOS), tem implementado um ciclo de duas (2) repetições, em que cada ciclo tem um *Timeout* de oito mil milissegundos (8.000 ms = 8 segundos). Este é o tempo que o recetor (RX) está no ar à escuta de mensagens. Assim, o RX faz dois ciclos com o total de 16 segundos. Se obtiver algum alerta, atua no Led vermelho.

- **Receção de alerta (SOS)**; A receção do alerta é feito com a função de comparação de caracteres, que é feito com recurso à manipulação de variáveis.

```
strcmp(my_packet, "SOS") == 0);
LedONOFF = 1;
```

5.3.1.3 Pedido de dados ao SLAVE

O pedido de dados ao SLAVE é feito com uma comunicação *wire* (I2C), que procura o endereço do escravo (SLAVE) e faz o pedido, lendo a mensagens disponíveis dos valores de sensores. Neste pedido é indicado o endereço que foi definido no SALVE (2), sendo também manipulado o tamanho (30) de caracteres (32 bytes), para que os dados possam estar de acordo com a variável definida.

```
Wire.requestFrom(2, 32);
```

Após a receção dos dados existe a necessidade de **manipular a mensagem**, que é transferida do SLAVE para o MASTER com a conversão da variável *char* para *String*. Só assim se pode juntar os caracteres recebidos. Após essa junção, a mensagem é novamente convertida em variável *char*.

```
a += c;           //acumular a String
a.toCharArray(msg1, 70); //Converte String em Char
a = "";          // Limpa a String
```

5.3.1.4 GPS

O *GPS* necessita de vários ciclos para detetar os satélites e mostrar os dados. Neste sentido, foi necessário implementar cinquenta ciclos (50), dando assim a possibilidade de obter a informação. Foi implementado o pedido das coordenadas, dias, horas e altitude. O seu código é extenso, mas recorre às suas bibliotecas para facilitar o código dos dados, devendo realçar a conversão de **valores inteiros** para *String*. Nos valores das coordenadas, a conversão foi utilizada a função *dtostrf*, que converte valores **float** longos para **String**;

```
dtostrf(lati,8,6,latit);
dtostrf(longt,8,6,longti);
```

No GPS e por uma questão de poupança de energia, foi implementado um botão ON/OFF. Esta opção foi tomada pelo motivo de que dentro do avião não é possível apanhar o sinal dos satélites. Outra das funções do GPS é quando este sintonizar o sinal dos satélites, desativa a receção de mensagem no NÓ de sensor. Considera-se que o utilizador já saiu do avião, pelo que já não é viável a leitura do sensor SpO2.

5.3.1.5 Led Vermelho e Botão

Dada a necessidade de mostrar os alertas vindos nas mensagens dos outros NÓS ou GateWay, foi criado um mecanismo com um *Led* e um *botão*. Este led acende quando chega alguma mensagem de *SOS* e só apaga quando o utilizador carregar no botão, desativando assim o Led. Foi feita uma função para ser chamada no ciclo principal, permitindo assim a desativação do Led.

5.3.2 Slave (Sensores)

A implementação no SLAVE recai sobre os sensores e a salvaguarda dos dados obtidos, sendo a sua principal função. No entanto, está disponível para o MASTER, enviando os valores dos sensores quando este solicitar.

5.3.2.1 TCA Expander

O TCA Expander é um módulo que aumenta a capacidade das comunicações em I2C, de fácil de utilização e manuseamento. Na implementação usada foi fundamental

para uma comunicação entre os sensores, tornando mais estável a leitura dos valores. No TCA liga os sensores de SpO2, Pressão Atmosférica e Temperatura, Relógio e ainda liga o display Oled.

O TCA utiliza a biblioteca *Wire*, é definida a porta 0x70 para sua comunicação. O TCA tem disponível sete (7) portas de comunicação, para os sensores, em que se estabelece a comunicação com uma chamada a essa porta (*TCA_channel(7)*), e interrompe no fim do código do sensor ou noutra chamada a outro sensor.

Chamada à porta I2C;

```
TCA_channel(2);
```

5.3.2.2 Leitura do Sensor Oximetria

O sensor de Oxímetria (MAX30105), tem vários tipos de leituras, mas no projeto só foi utilizado a leitura para o SpO2, e o algoritmo para o calculo dos valores de SpO2. Para se calcular o SpO2, é necessário cinquenta (50) ciclos de armazenamento de valores para apresentação dos dados de SpO2. Utiliza uma janela deslizante de cinquenta de valores novos (leitura constante) fazendo um total de cem (100) registos (*bufferLength = 100;*).

- Assim considera-se valores normais de SpO2 entre 90 a 102%, sendo que os 102% correspondem a uma margem para alguma hiperventilação pontual.

- Para valores de alerta (SOS) consideram-se valores mínimos SpO2 entre 80% a 89% e máximos entre 103 a 110%. Estes intervalos consideram-se críticos para o ser humano, tanto na falta de oxigénio (Hipóxia), como na hiperventilação (excesso de oxigénio).

- Os valores que não se enquadrem nestes parâmetros, são ERROS de leitura do sensor, devido a uma má colocação do sensor ou uma anomalia no próprio sensor.

Conversão de inteiros para String;

```
itoa(SPO2Msg, SpO2_2, 10);
```

Composição da mensagem;

```
strcpy(MsgLora_T, idNo);  
strcat(MsgLora_T, sep);  
strcat(MsgLora_T, fim);
```

Fim da mensagem de dados;

```
char fim[] = "&";
```

5.3.2.3 Leitura do Sensor de pressão

O sensor MPL3115A2, permite-nos recolher vários parâmetros. Neste projeto os dados importantes de recolher foram a *pressão atmosférica e temperatura*. O valor da pressão atmosférica (peso da massa de ar que existe sobre um determinado local) permite-nos verificar a que pressões está sujeito o saltador (utilizador), tanto na subida como na descida rápida que efetua em queda livre. Os valores da temperatura, também são um fator importante, para se saber as temperaturas existentes no delta tempo.

A pressão atmosférica permite também ter uma leitura da altitude, que pode ser usada em qualquer parte do mundo, mesmo não conhecendo o relevo (terreno) e ou o nível médio das águas do mar, de um determinado local.

Existem vários tipos de cálculo de altitude. Este recai na necessidade do voo das aeronaves e o nível médio das águas do mar, mas a variação em redor do mundo é um fator preocupante para os pilotos. Assim, existem três (3) tipos de acerto altimétricos.

- Altitude Absoluta (altura) - **QFE** (Query: Field Elevation) - Considera-se o zero ao nível do solo e coloca-se o altímetro a zero na pista, não faz uma relação com o nível médio das águas do mar;

- Altitude Indicada - **QNH** (Query: Nautical Height) - Considera-se o nível médio das águas do mar, e a altitude é definida a partir dessa linha;

- Altitude Pressão - **QNE** (Query: Nautical Equivalent)- É a altitude definida a partir da pressão atmosférica (1013.25 hPa), na norma *International Standard Atmosphere (ISA)* que é utilizada em todo o mundo na navegação aérea, considerando-se a mais correta em redor do mundo.

A sua fórmula de cálculo foi definida pela *National Oceanic and Atmospheric Administration (NOAA)* [91] sendo representada por:

pressão do sensor = P_{sta}

Altitude Pressão = h_{alt}

$$h_{alt} = \left[1 - \left(\frac{P_{sta}}{1013,25} \right)^{0,190284} \right] 145366.45$$

- Configuração do sensor MPL;

Na utilização do TCA Expander é necessário que todos sensores estejam com a mesma frequência de relógio I2C (400MHz). Para que as leituras fossem corretas acrescentou-se uma linha na sua biblioteca. A biblioteca (SparkFunMPL3115A2.cpp) do sensor foi definido, *Wire.setClock* e definiu-se para **400MHz**.

5.3.2.4 Display Oled

Na implementação houve a necessidade de observar os valores do SpO2, e nesse sentido foi implementado um display Oled. Neste display foi considerado o tipo de ligação. Optou-se pela I2C que já estava implementada (TCA) para os sensores. O display apresenta os valores de SpO2, emite flash quando os valores estão fora dos parâmetros (SOS) e mostra quando existe ERRO na leitura do sensor.

```
display.print(spo2, DEC);
```

5.3.2.5 SLAVE -to- MASTER

Quando o MASTER faz o pedido da mensagem, este envia a mensagem já composta. Esta chamada é definida no *setup*;, que por sua vez chama uma função desenvolvida *MsgLora*.

Definição no setup;

```
Wire.onRequest(MsgLora);
```

Função desenvolvida;

```
Wire.write(MsgLora_T);
```

5.3.3 microSD

A necessidade de guardar os dados dos sensores, por eventual falha ou falta de alcance do radio (RF), implementou-se um módulo de microSD. Esta implementação verifica se já existem ficheiros criados. Se existir cria outro, com numeração seguida, ficando assim a salvaguarda da repetição dos dados. Após a leitura dos sensores e da mensagem ser composta (junta), é guardada no cartão SD, com a chamada da função desenvolvida e colocada no ciclo principal (*gravaSD*). O módulo microSD foi implementado no SLAVE, devido à incompatibilidade com a placa Multiprotocol do LoRa Libelium, que impede a utilização do porto SPI do módulo MASTER para outro fim.

Verificação se existe ficheiro:

```
SD.exists(charFileName)
```

Abrir o ficheiro;

```
myFile = SD.open(message, FILE_WRITE);
```

Fechar o ficheiro

```
myFile.close();
```

Os dados guardados são exclusivamente do SLAVE, visto que não existe uma comunicação bidirecional entre o MASTER e SLAVE. No entanto a mensagem é composta de igual forma à correspondente a um envio, para que o gateway possa inseri-las na base de dados. Isto só será possível se os campos que compõem a base de dados forem idênticos, com um identificador que é o existente no cartão SD.

Mensagem guardada: 1A#99#10#617#2018/10/18#13:7:31#13094#0#0#0#SD

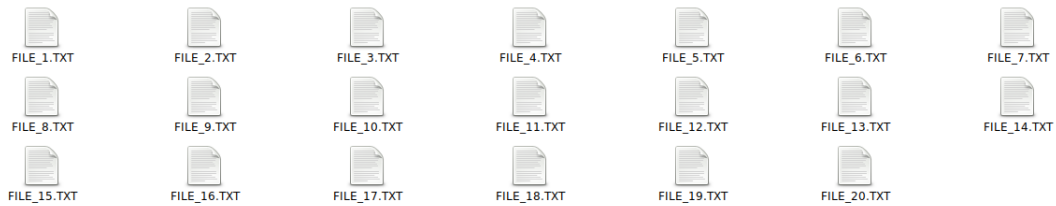


Figura 5.17: Ficheiros criados no cartão SD

Capítulo 6

Testes e Resultados

Este capítulo apresenta os testes efetuados ao sistema e os respectivos resultados. Estes incluem testes em laboratório e testes reais efetuados em aeronave e respectivos saltos. Os testes incluem a comunicação dos nós ao gateway em terra e comunicações entre nós em ambiente controlado, assim como testes de comunicações entre aeronave e gateway em terra e comunicações entre nós em ambiente real.

6.1 Teste no Desenvolvimento

Foram efetuados vários testes de implementação, com vista a verificar todo o sistema. Assim iniciou-se os testes com as leituras dos sensores, a comunicação entre os dois arduinos, entre o NÓ de sensores e o Gateway, o alcance das comunicações e um simples teste abordo de uma aeronave (C-295) com geolocalização.

6.1.1 Teste na leitura dos sensores

Os testes com os sensores, foram efetuados para se verificar se as bibliotecas e o código no arduino, estava nas expectativas desejadas ou se ainda se teria de fazer ajustes nos mesmos.

Assim foram obtidos os seguintes resultados (figura 6.1):

1 - Sensor de Oximetria (MAX30105)

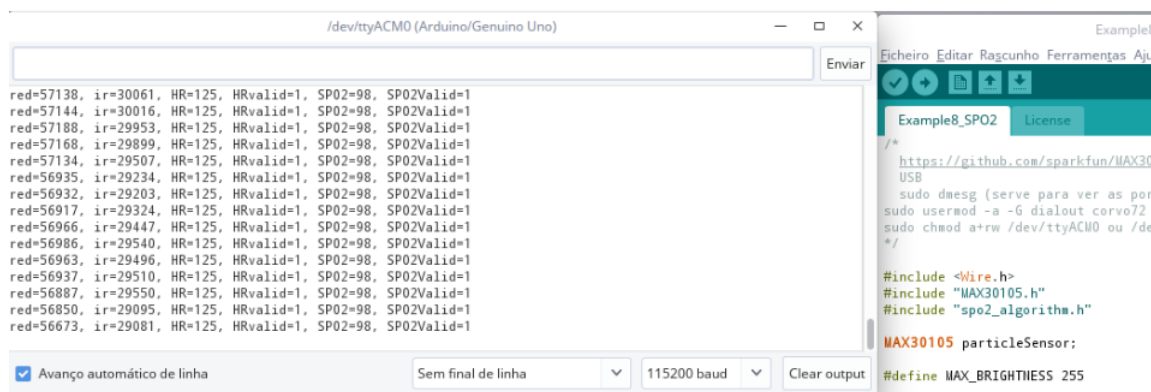


Figura 6.1: Teste Sensor MAX30105

6.1. TESTE NO DESENVOLVIMENTO

2 - Verificar se o sensor Oximetria está calibrado.

A figura 6.1 apresenta o resultado do teste. Para comparação, foi usado um aparelho de Oximetria de dedo, existente nas ambulâncias do INEM e ambos marcavam 98 de SpO2.



Figura 6.2: Teste Sensor MAX30105 - Aparelho do INEM

3 - Sensor de pressão atmosférica e temperatura (MPL3115A2)

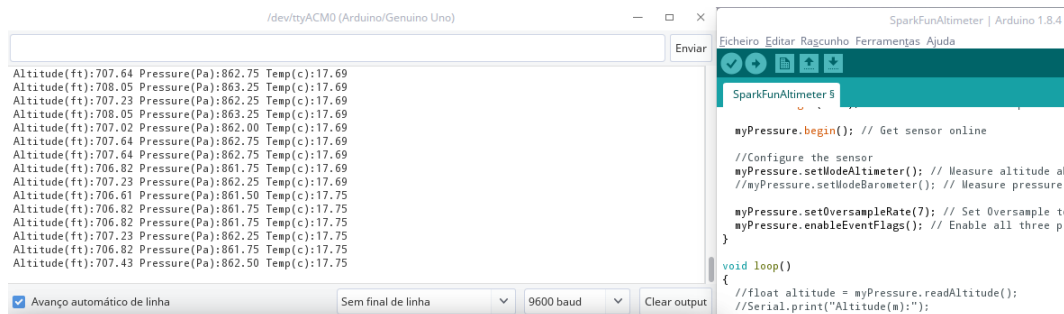


Figura 6.3: Teste Sensor MPL3115A2

4 - Sensor de GPS NEO-6M

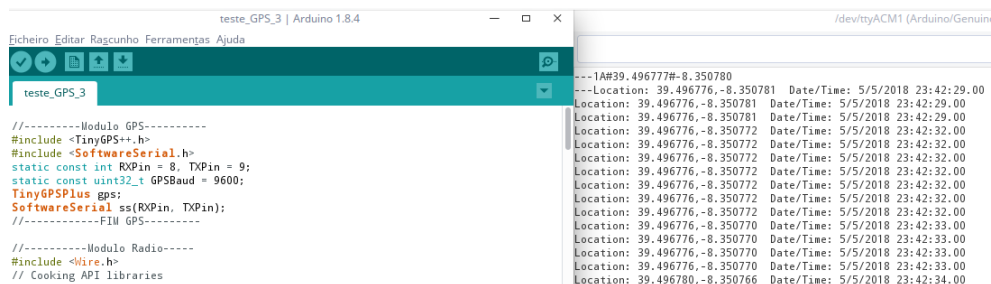


Figura 6.4: Teste GPS

5 - Comunicação rádio de envio e recepção de mensagens

A figura 6.5 apresenta o resultados dos testes de comunicação. Do lado esquerdo

encontra-se o código no arduino, do lado direito em cima, esta o *Serial.print* via USB com os valores a serem transmitidos e do lado direito em baixo, o Gateway a receber a comunicação.

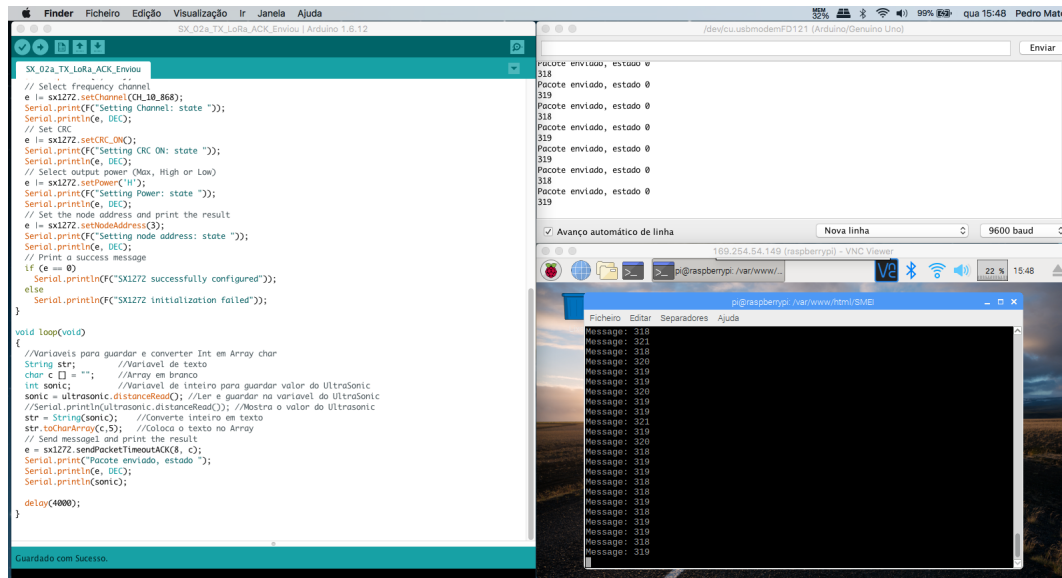


Figura 6.5: Teste de comunicações TX

6.1.2 Teste de comunicação entre Master-Slave

O arduino MESTRE faz o pedido ao SLAVE e este responde em conformidade com o ponto de situação que se encontra (ERRO, SOS, "1A#...valores.."), podendo observar na imagem.

```
ERRO&
1A#100#20#253#1004&
1A#99#20#253#1004&
1A#99#20#253#1004&
1A#96#20#253#1004&
1A#97#20#253#1004&
1A#99#20#253#1004&
1A#98#20#253#1004&
SOS&
ERRO&
```

Figura 6.6: Receção dos Pedidos MASTER-SLAVE

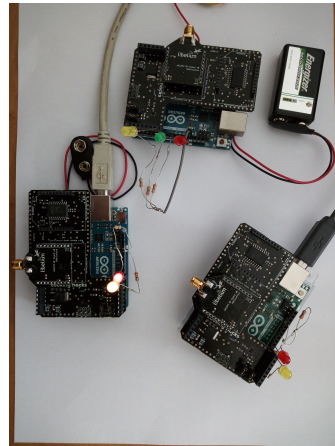
6.1.3 Teste de envio (TX) e receção (RX) de SOS

Foram executados testes com três (3) NÓ de sensores, o seu código era idêntico nos Três (3), enviava um alerta de SOS, podendo assim observar que o enviou de mensagens e a receção dos alertas:

6.1. TESTE NO DESENVOLVIMENTO

```
Setting CRC ON: state 0
Setting Power: state 0
Setting node address: state 0
SX1272 successfully configured
Msg Enviada: SOS
Msg Enviada: SOS
Msg Enviada: SOS
Msg Enviada: SOS
Msg Enviada: SOS
-----Message Recebida: SOS
-----Msg Recebida 0 (SIM) se 1 (NÃO) = 0
Msg Enviada: SOS
Msg Enviada: SOS
Msg Enviada: SOS
Msg Enviada: SOS
Msg Enviada: SOS
-----Message Recebida: SOS
-----Message Recebida: SOS
Msg Enviada: SOS
Msg Enviada: SOS
Msg Enviada: SOS
Msg Enviada: SOS
Msg Enviada: SOS
-----Msg Recebida 0 (SIM) se 1 (NÃO) = 0
-----Message Recebida: SOS
-----Message Recebida: SOS
Msg Enviada: SOS
Msg Enviada: SOS
Msg Enviada: SOS
-----Message Recebida: SOS
-----Message Recebida: SOS
Msg Enviada: SOS
Msg Enviada: SOS
```

(a) Teste de TX e RX com SOS



(b) Hardware dos 3 NÓS de Sensores

6.1.4 Teste de envio NÓ de sensor - Geteway

Envio dos valores do NÓ para o Gateway (Imagem mais à direita), Recepção do Gateway (imagem à esquerda).

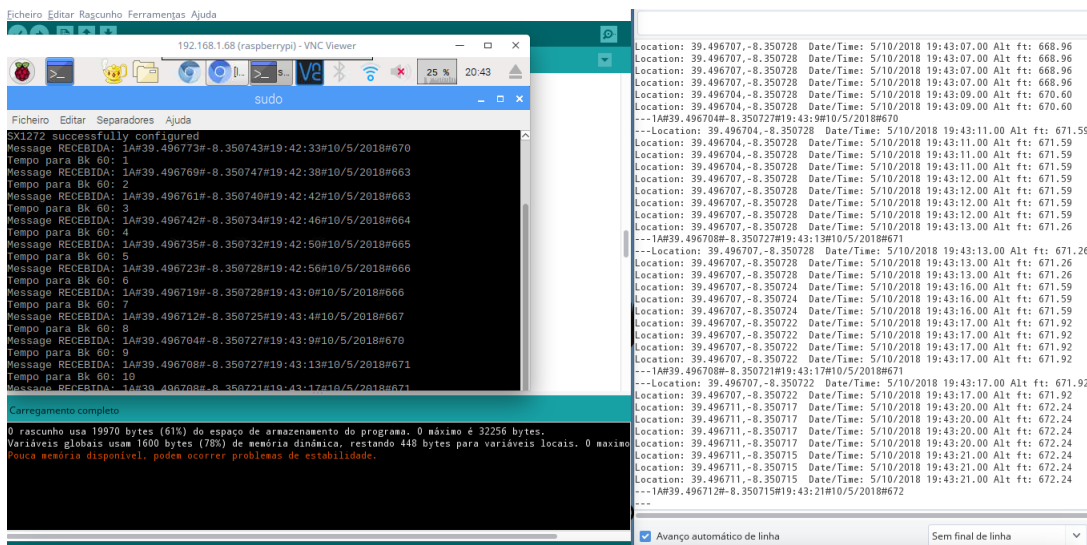


Figura 6.8: Teste TX e RX NÓ e Gateway

6.1.5 Testes de Alcance em LoRa

Teste das comunicações em Terra Foram efetuados testes de alcance das comunicações. O Gateway foi colocado no telhado do Corpo de Bombeiros Voluntários de Vila Nova da Barquinha e definiu-se uma linha até 20Km. Foram obtidas as mensagens com posicionamento GPS. Esses dados foram inseridos em Base de dados. Os dados foram

exportados da base de dados com a extensão e carregados no google Earth. O resultado do teste está descrito na figura 6.9. A realçar que alguns pacotes foram perdidos (não recebidos), devido ao relevo e aos obstáculos encontrados ao longo do caminho. Ainda assim, verificou-se que foram conseguidas distâncias próximas de 20 Km.

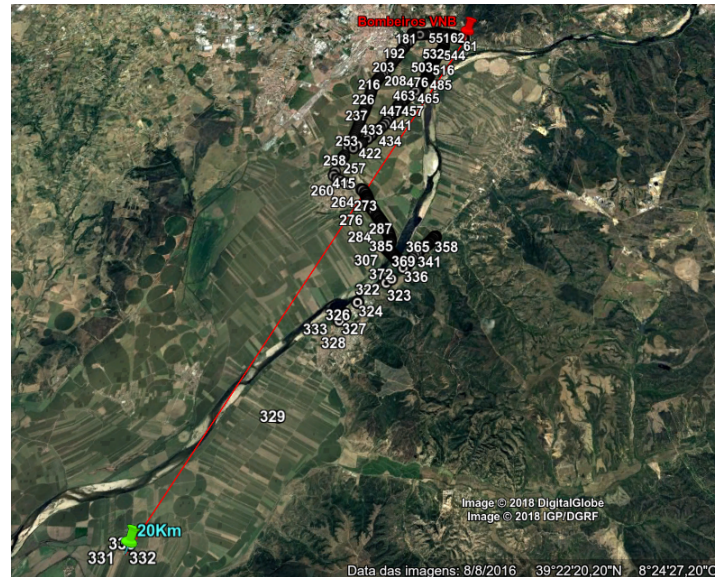


Figura 6.9: Pontos da emissão da mensagem (GPS)

Coordenada mais longe registada.



Figura 6.10: Mensagem enviada 20Km

Perfil do relevo do Gateway e os 20Km

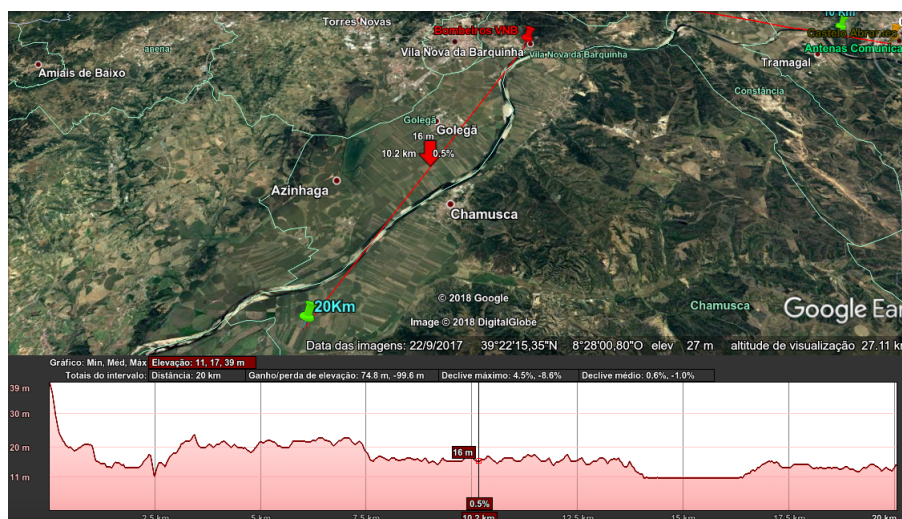


Figura 6.11: Perfil do Relevo

A figura 6.12a mostra a colocação do Gateway no Telhado dos BV de VNB



(a) Colocação do Gateway



(b) Colocação do NÓ

Figura 6.12: Representação NÓ de sensor e Gateway

6.2 Teste em Laboratório

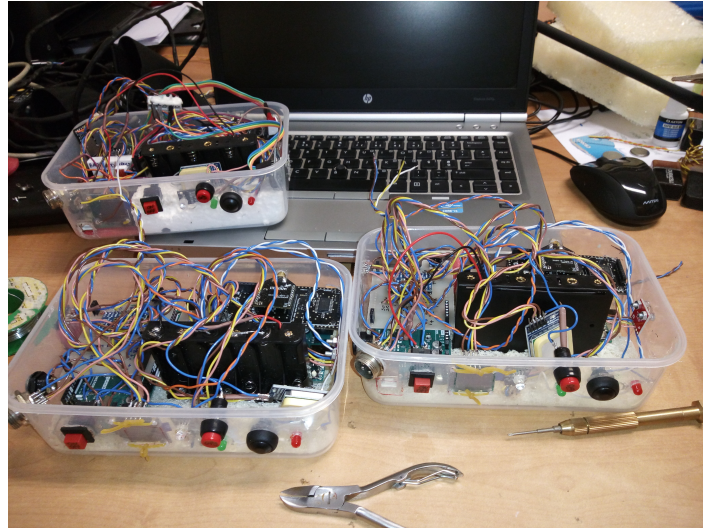
Foram efetuados testes com os três NÓS de Sensores. Pretendia-se consolidar o funcionamento dos NÓS, com o proposto referido no desenvolvimento e ambição do projeto.



(a) Visão lateral



(b) Visão Geral



(c) Montagens dos NÓS

Figura 6.13: Representação dos três NÓS de sensores prontos.

Os resultados obtidos da comunicação entre NÓS de Sensores, foram notórios, dando a possibilidade de uma comunicação de NÓ para NÓ ou de um NÓ para todos os NÓS. Estes teste foram alargados a uma bateria de testes, estes testes apresentados vai ao encontro da nossa proposta.

6.2.1 Teste de energia do NÓ de sensores

A energia é um ponto fundamental para o NÓ de Sensores. Considerando-se que cada NÓ vai estar ligado entre uma a duas horas. A escolha das baterias de características de uma pilha AA, a 1.2 volts e 1900 mAh.

Cada NÓ de sensores, foi alimentado por seis baterias em série, atingindo a tensão de 7.2V. Submetidas a carga, conseguem armazenar 8V, utilizando o carregador próprio para as referidas baterias. O modelo utilizado foi o Ni-Cd/Ni-MH MW6168V, que permite carregar até dez Cells, configurado para a corrente de 500mA. O NÓ com as baterias carregadas em transmissão para a Gateway, não esgotou a energia com duração de 6 horas. Quando foi o conjunto de baterias foi submetido a carga, estes

ainda apresentavam 7.65V. Conclui-se que o consumo do NÓ de sensor é relativamente baixo, permitindo operar durante o dia inteiro.

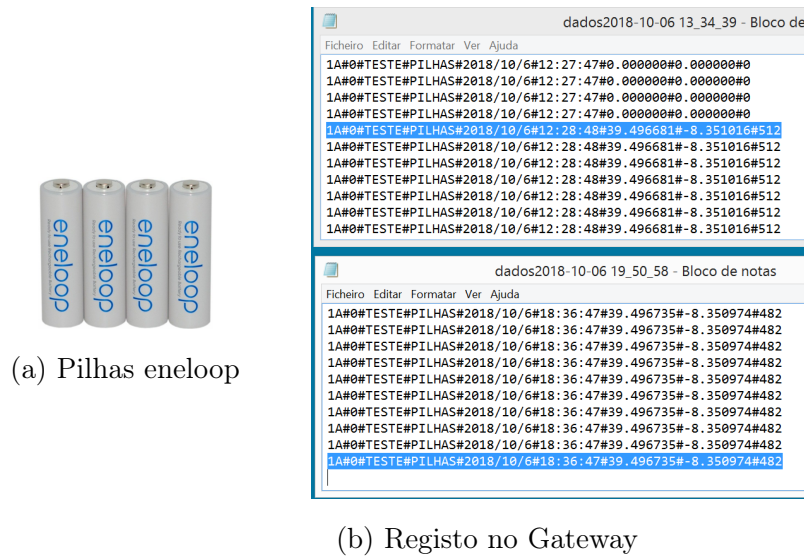


Figura 6.14: Teste da capacidade das baterias.

6.2.2 Comunicação entre NÓS com SOS

Foram realizados testes de SOS entre os NÓS de Sensores. Nestes testes foi utilizado o ACK como referido na secção 5.3.1.1. O seu princípio de funcionamento vai ao encontro das necessidades de um sistema de alerta, sem que o Gateway funcione como base de distribuição das mensagens. O NÓ de sensor alerta o seu utilizador com o sistema de flash no display, e envia mensagens de alerta para a rede. O envio de mensagens é feito com base na necessidade dos outros utilizadores saberem que se passa com um determinado NÓ. Os outros NÓS de Sensores recebem a mensagem de SOS que acendem o seu LED vermelho. O ACK permite que o emissor receba a confirmação da mensagem.

O teste consistiu em implementar um alerta no NÓ de Sensor 2A (imagem 6.16 o NÓ do meio), com a imposição da variável de SpO2 com um valor abaixo de noventa com os outros NÓS de Sensores a funcionarem o seu algoritmo normalmente.

Como pode ser visto na figura 6.15, o envio e a confirmação da receção, com o Estado a zero (0). Os NÓS de Sensores receberam a mensagem e acenderam o Led vermelho do lado direito (figura 6.16).

```

Location: INVALID
Location: INVALID
Location: INVALID
Location: INVALID
Location: INVALID
----- SOS Alerta --- Msg Enviada SOS: SOS#2A Estado: 0
----- SOS Alerta --- Msg Enviada SOS: SOS#2A Estado: 0
----- SOS Alerta --- Msg Enviada SOS: SOS#2A Estado: 0
-----Msg Recebida 0 (SIM) se 1 (NÃO) = 0
-----Msg Recebida 0 (SIM) se 1 (NÃO) = 0

```



Figura 6.15: Emissor ACK

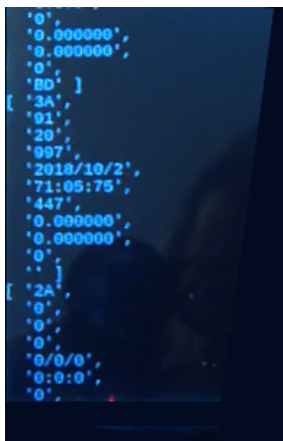


Figura 6.16: Três NÓS de Sensores em Laboratório (SOS)

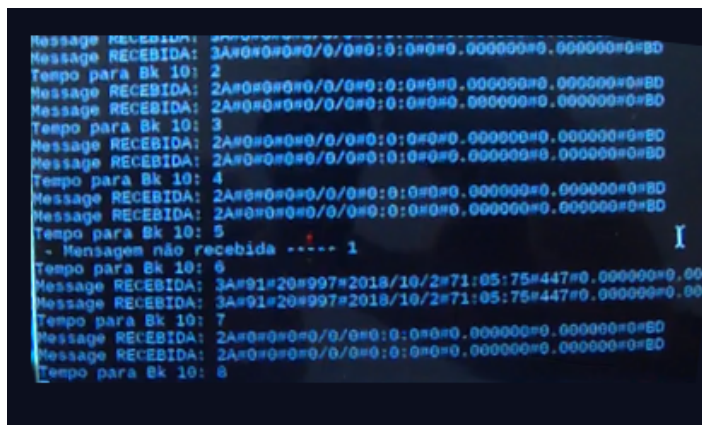
6.2.3 Comunicação entre NÓS e GateWay

A comunicação entre NÓS de Sensores é realizada com *Unicast* e *Broadcast*. A comunicação em *Unicast* é efetuada quando o NÓ de sensor faz a comunicação com o Gateway. Na comunicação *Broadcast* é efetuada quando o NÓ emite um SOS. O SOS é enviado em simultâneo a todos os NÓS e Gateway, que utilizem a mesmo canal/frequência. O mesmo acontece com o Gateway, que utiliza *Unicast*, quando recebe individualmente as mensagens dos NÓS de Sensores e *Broadcast* quando emite o SOS para toda a rede.

Teste comunicação e envio de mensagens ao Gateway: Foram efetuados teste da receção do Gateway das mensagens enviadas dos NÓS de sensores. O Gateway recebe as mensagens e após os trinta segundos guarda-os na base de dados.



(a) Backup MySQL.



(b) RX no Gateway.

Figura 6.17: Teste de Comunicação NÓS e Gateway

Teste envio de SpO2 dos NÓS de Sensores:

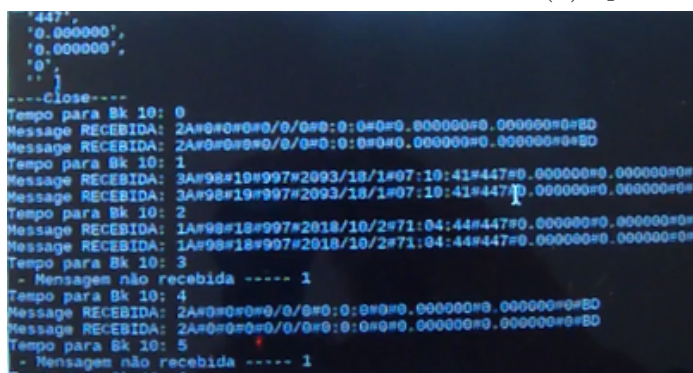
Foram excetuado testes de leitura e envio dos valores de SpO2 para o Gateway. Como os outros sensores estão dependentes do sensor de SpO2, também foram verificados e enviados na mesma mensagem.



(a) Conjunto.



(b) SpO2 Display.



(c) Resultados no Gateway.

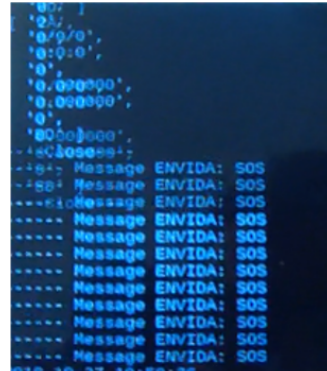
Figura 6.18: Representação do Teste

Teste envio de SOS do Gateway aos NÓS de Sensores:

Foi efetuado o teste de retorno da recepção do SOS vindo dos NÓS de Sensores. O seu funcionamento é baseado no envio do SOS do Gateway e a recepção do NÓ, que aciona o Led vermelho. Este só apaga, quando o utilizador pressionar no botão.



(a) Sinal de aviso.



(b) TX no Gateway

Figura 6.19: Teste de envio de SOS.

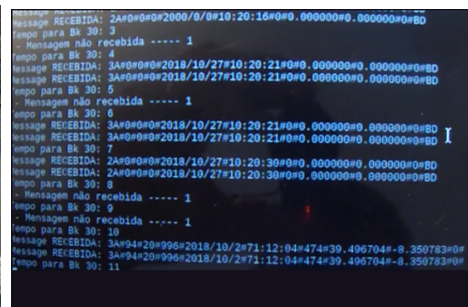
6.2.4 Envio de coordenadas de GPS para Gateway

Foram executados testes do tempo que demora a captação de satélites. Verificou-se uma eficácia entre trinta (30) segundos a dois (2) minutos. Outro teste efetuado, recaiu sobre o interruptor da alimentação do modulo do GPS, dando a possibilidade de o ligar só quando necessário. As coordenadas de GPS só são registadas no Gateway, devido as limitações do hardware.

Teste de envio em Laboratório: Colocaram-se os três NÓS de Sensores a captar o posicionamento, adquirindo e enviando as coordenadas ao Gateway.



(a) Funionamento dos três nós



(b) RX no Gateway

Figura 6.20: Teste de envio das coordenadas.

Teste de envio em campo: Foi executado um percurso em terreno montanhoso, enviando as coordenadas dos GPS para o Gateway.

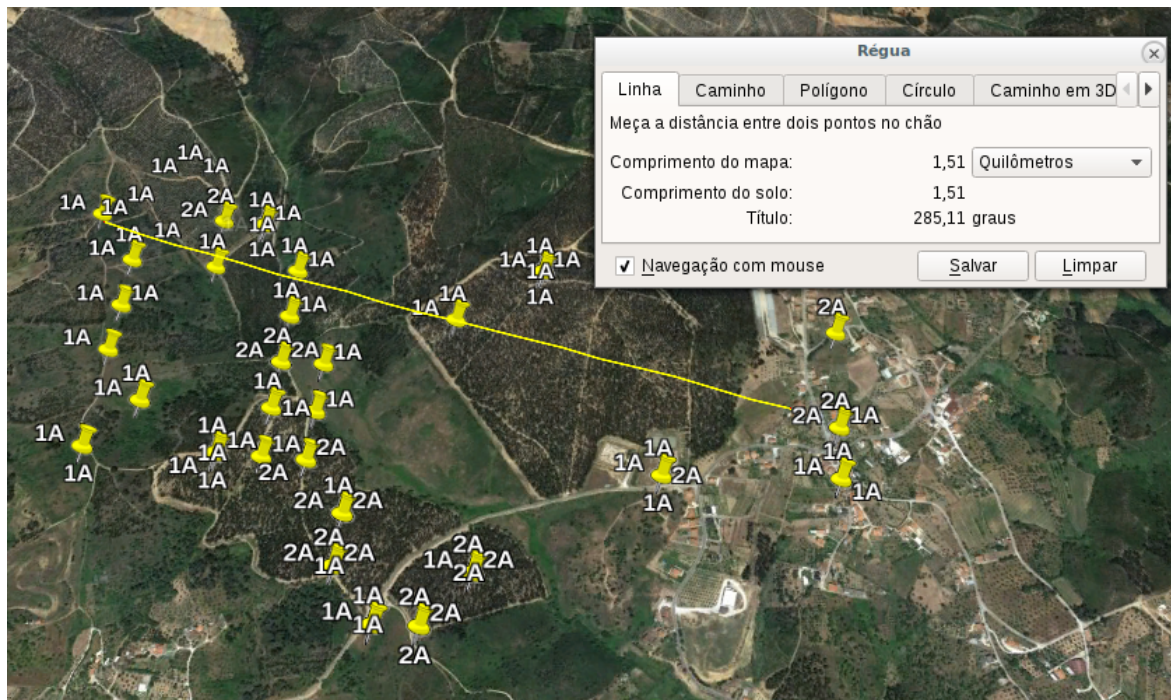


Figura 6.21: Marcação em mapa das coordenadas registadas.

Dados adquiridos no Gateway em TXT:

```
1A#0#0#0#2018/10/27#13:22:14#0#39.499741#-8.357388#533#BD
2A#0#0#0#2018/10/27#13:34:11#0#39.496655#-8.350975#490#BD
```

Dados registados na base de dados:

```
(1007,'1A','0','0','0','2018/10/27','13:22:14','0','39.499741','-8.357388','533','BD')
(1008,'2A','0','0','0','2018/10/27','13:34:46','0','39.496666','-8.350986','489','BD')
```

6.3 Teste em Ambiente real

Os NÓS de Sensores foram desenvolvidos com o objetivo de fazerem testes em ambiente real, que compreendia a realização de saltos em paraquedas. O meio tem muitas condicionantes considerando um hardware em protótipo, no entanto tentou-se acondicionar os NÓS em bolsas de transporte realizadas para o efeito.

6.3.1 Resultados obtidos como um NÓ de Sensor - 1ºSALTO

Teste das comunicações no avião (paraquedas) para Gateway Foi efetuado um primeiro teste a bordo da aeronave C-130 (militar), onde foram registados alguns eventos das mensagens. O Gateway foi colocado no solo e o NÓ LoRa foi transportado pelo paraquedista. A figura 6.22 mostra a localização das mensagens recebidas. Este

teste, apesar de incompleto devido às limitações do hardware de teste, permitiu tirar as primeiras conclusões. Foram recebidas mensagens com o NÓ. Considerando-se relevante o ponto 930, localizado dentro da aeronave assim como os pontos obtidos durante o salto. Por exemplo, uma mensagem a 4.167 ft recebida do NÓ de sensores.

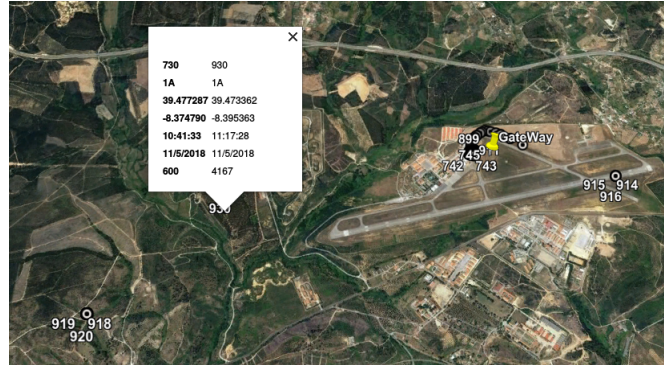


Figura 6.22: Teste Salto com NÓ de Sensor (GPS).

6.3.2 Resultados obtidos com um NÓ de Sensores - 2ºSALTO

O segundo teste real foi abordo da aeronave C-295 da Força Aérea Portuguesa. O saltador (utilizador) utilizou um NÓ de sensor com o sensor de SpO2, ligado ao pulso. Este teste realizou-se no dia 27 de outubro de 2018.



(a) Colocação do sensor SpO2.



(b) Dentro do Avião C-295.



(c) Display SpO2 dentro do Avião.

Figura 6.23: Resultados obtidos com um NÓ de Sensores 2ºSALTO

O Gateway foi colocado na zona de aterragem, recebendo as mensagens enviadas do NÓ de Sensor.



(a) Colocação do Gateway na ZA



(b) Caixa do Gateway

Figura 6.24: Colocação do Gateway na Zona de Aterragem

Foram obtidos dados no Gateway, onde se registou em ficheiros TXT e na Base de dados. Dados extraídos da base de dados, desde a descolagem até aterragem.

```
('1A','96','30','1019','2018/9/27','13:55','-157','0.000000','0.000000','')
('1A','98','31','962','2018/9/27','13:56','1429','0.000000','0.000000','')
('1A','94','31','935','2018/9/27','13:57','2206','0.000000','0.000000','')
('1A','94','31','935','2018/9/27','13:57','2206','0.000000','0.000000','')
('1A','93','30','915','2018/9/27','13:57','2794','0.000000','0.000000','')
('1A','97','30','907','2018/9/27','13:57','3032','0.000000','0.000000','')
('1A','94','30','893','2018/9/27','13:58','3453','0.000000','0.000000','')
('1A','94','30','893','2018/9/27','13:58','3453','0.000000','0.000000','')
('1A','100','28','729','2018/9/27','14:3','8828','0.000000','0.000000','')
('1A','100','28','713','2018/9/27','14:4','9403','0.000000','0.000000','')
('1A','100','28','713','2018/9/27','14:4','9403','0.000000','0.000000','')
('1A','96','27','696','2018/9/27','14:7','10026','0.000000','0.000000','')
('1A','96','27','696','2018/9/27','14:7','10026','0.000000','0.000000','')
('1A','92','27','693','2018/9/27','14:8','10137','0.000000','0.000000','')
('1A','100','27','693','2018/9/27','14:9','10137','0.000000','0.000000','')
('1A','100','27','693','2018/9/27','14:9','10137','0.000000','0.000000','')
('1A','97','27','693','2018/9/27','14:9','10137','0.000000','0.000000','')
('1A','92','26','694','2018/9/27','14:11','10100','0.000000','0.000000','')
('1A','92','26','694','2018/9/27','14:11','10100','0.000000','0.000000','')
('1A','98','17','778','2093/18/9','2:14','7127','0.000000','0.000000','')
('1A','99','18','864','2018/9/27','14:13','4341','0.000000','0.000000','')
```



```
( '1A', '91', '19', '906', '2018/9/27', '14:14', '3062', '0.000000', '0.000000', '' )
( '1A', '99', '26', '1018', '2018/9/27', '14:18', '-129', '39.477486', '-8.373415', '' )
```

Resultados obtidos: Podemos observar na amostra dos dados recebidos, que desde a descolagem até aterragem, o GPS não conseguiu adquirir sinal dos satélites. Outro fator foi o tempo que o NÓ esteve em transmissão, vinte e três (23) minutos e o Gateway só recebeu vinte e três (23) mensagens. Considerando que o NÓ de sensor envia mensagem em menos de cinco (5) segundos. Podendo assim concluir com este teste que a distância em que andou o avião, a estrutura do avião e a perda de pacotes (mensagens) no ar, são fatores a ter em conta.

6.3.3 Resultados obtidos como três NÓS de Sensores - 3ºSALTO

No terceiro salto pretendeu-se combinar os três NÓS de Sensores. Os três utilizadores (*Matos, Santos e Modesto*), colocaram os aparelhos e a braçadeira como no segundo salto. A intenção do teste seria observar os resultados do SpO2, a quantidade de mensagens enviadas ao Gateway e testar a ligação GPS apenas à saída do Avião, com um interruptor de alimentação ao módulo.



Figura 6.25: 3ºSalto com os três NÓS de Sensores.

Extrato dos ficheiros obtidos dos cartões SD:

NÓ de Sensor - 1A

1A#99#15#0#2018/10/18#13:7:31#14294#0#0#0#SD
1A#99#15#718#2018/10/18#13:7:45#9222#0#0#0#SD
1A#48#15#718#2018/10/18#13:8:0#9222#0#0#0#SD
1A#100#15#718#2018/10/18#13:8:15#9222#0#0#0#SD
1A#99#15#718#2018/10/18#13:8:29#9222#0#0#0#SD
1A#97#15#718#2093/18/10#2:13:8#9222#0#0#0#SD
1A#20#15#718#2018/10/18#13:8:58#9222#0#0#0#SD
1A#25#15#718#2018/10/18#13:9:13#9222#0#0#0#SD

NÓ de Sensor - 2A

2A#98#17#692#2018/10/18#13:22:43#10174#0#0#0#SD
2A#25#17#692#2018/10/18#13:22:58#10174#0#0#0#SD
2A#100#17#693#2018/10/18#13:23:12#10137#0#0#0#SD
2A#93#17#693#2018/10/18#13:23:26#10137#0#0#0#SD
2A#100#17#693#2018/10/18#13:23:41#10137#0#0#0#SD
2A#100#17#693#2018/10/18#13:23:55#10137#0#0#0#SD
2A#93#17#693#2018/10/18#13:24:9#10137#0#0#0#SD
2A#25#17#693#2018/10/18#13:24:24#10137#0#0#0#SD

NÓ de Sensor - 3A

3A#74#22#615#/#:#:#13175#0#0#0#SD
3A#99#22#615#/#:#:#13175#0#0#0#SD
3A#100#22#615#/#:#:#13175#0#0#0#SD
3A#25#22#621#2113/0/1#1:0:0#12931#0#0#0#SD
3A#98#21#629#2113/0/1#1:0:0#12608#0#0#0#SD
3A#50#21#627#2113/0/1#1:0:0#12688#0#0#0#SD
3A#95#21#626#2113/0/1#1:0:0#12729#0#0#0#SD
3A#25#21#629#2113/0/1#1:0:0#12608#0#0#0#SD

Extrato de dados da Base de Dados:

"1A";"0";"0";"0";"2018/10/18";"11:43:34";"0";"0.000000";"0.000000";"0";"BD"
"1A";"0";"0";"0";"2018/10/18";"11:50:49";"0";"0.000000";"0.000000";"0";"BD"

Resultados obtidos: O teste dos três NÓS de Sensores, demonstrou algumas falhas no leitor de SpO2 (20, 25, 48, 50, 74) estas falhas de leitura são provocadas pelo utilizador, com a necessidade de se mexer, onde sensor não consegue fazer as leituras todas para o calculo do SpO2. Em particular o NÓ 3A, teve problemas com o módulo do

relógio. Este desprogramou-se. No Gateway, não houve registro, devido a deficiente endereçamento dos NÓS. Posteriormente esse problema foi corrigido em laboratório.

6.3.4 Resultados obtidos como três NÓS de Sensores - 4ºSALTO

O teste do quarto salto foi definido uma infiltração em calote, podendo os utilizadores (Modesto, Ramalho e Pereira) estar mais tempo com o paraquedas aberto. Este tempo compreendia estar entre 15 a 10 minutos, podendo assim o GPS adquirir o posicionamento. Os NÓS de Sensores foram colocados como o salto anterior.



Figura 6.26: NÓ de Sensor - 1A.



Figura 6.27: NÓ de Sensor - 2A.



Figura 6.28: NÓ de Sensor - 3A.

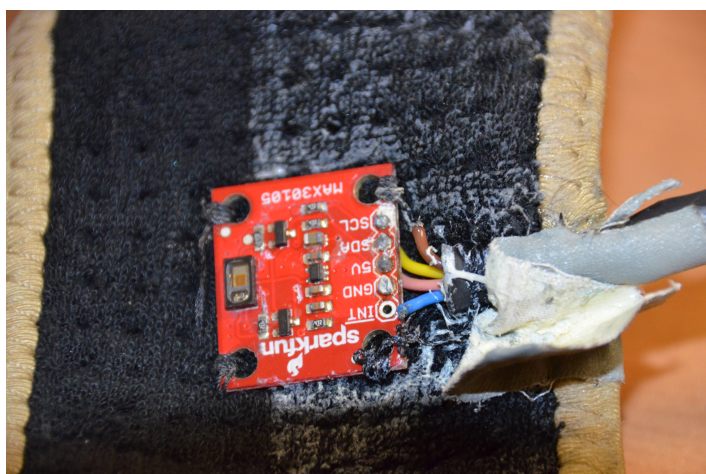


Figura 6.29: SpO2 3A - 4 Salto.

Extrato dos ficheiros obtidos dos cartões SD:

NÓ de Sensor - 1A

Não registou dados

NÓ de Sensor - 2A

2A#99#19#690#2018/10/24#16:8:58#10249#0#0#0#SD

2A#92#19#690#2018/10/24#16:9:0#10249#0#0#0#SD

2A#34#19#690#2018/10/24#16:9:3#10249#0#0#0#SD

2A#25#19#691#2018/10/24#16:9:6#10212#0#0#0#SD

2A#25#19#690#2018/10/24#16:9:9#10249#0#0#0#SD

NÓ de Sensor - 3A

3A#25#26#919#2018/10/24#15:18:29#2676#0#0#0#SD
3A#99#26#917#2018/10/24#15:18:32#2735#0#0#0#SD
3A#98#26#915#2093/18/10#4:15:18#2794#0#0#0#SD
3A#91#26#913#2093/18/10#4:15:18#2853#0#0#0#SD
3A#100#26#912#2018/10/24#15:18:40#2883#0#0#0#SD

Extrato de dados da Base de Dados:

Não registou dados.

Resultados obtidos: No teste do quarto salto os testes observados não atingiram o que se tinha proposto para este teste. O NÓ de Sensor 1A, teve problemas na ligação do sensor SpO2, como podemos observar na imagem 6.29. Dois fios partiram-se e o NÓ não se liga enquanto este estiver sem ligação, como se pode observar na imagem 6.26. O NÓ de Sensor 2A, teve muitas falhas de leitura (img.6.27). Estas falhas foram devidas a erros de leitura do sensor. No NÓ de Sensor 3A, este teve um comportamento normal de leituras e registou no SD (img.6.28). A receção das mensagens no Gateway, não se verificou, desde o início até ao fim do teste. Esta anomalia ocorreu devido ao facto de nas mensagens de SOS, o Gateway receber os SOS dos NÓS de Sensores sem que estivessem salvas as mensagens de dados, antes do Gateway enviar os seus SOS para a rede. Outra anomalia aconteceu com NÓ 3A. O seu módulo de rádio desconectou-se. Foi corrigido este processo no ficheiro do rádio (dados.cpp_exe).

6.3.5 Resultados obtidos como três NÓS de Sensores - 5ºSALTO

Foi efetuado o quinto teste, com o objetivo de testar as comunicações e o GPS. Os utilizadores (Modesto, Ramalho e Pereira), ligaram os aparelhos e o GPS antes do embarque no avião. O avião foi o C-130 da Força Aérea Portuguesa, que efetuou dois lançamentos a 10.000ft e o último a 8.000ft. Este ultimo foi onde se encontravam os NÓS de Sensores.



(a) NÓ 1A



(b) NÓ 1A



(c) NÓ 3A

Figura 6.30: NÓS de Sensores dentro do avião



(a) Caixa do Gateway



(b) Paraquedista



(c) Gateway na ZA

Figura 6.31: Colocação do Gateway na Zona de Aterragem

Extrato dos ficheiros obtidos dos cartões SD:

NÓ de Sensor - 1A

Não registou dados - perda do cartão SD durante o salto.

NÓ de Sensor - 2A

2A#25#16#1008#2018/11/7#12:51:8#144#0#0#0#SD

2A#25#16#1008#2018/11/7#12:51:11#144#0#0#0#SD

2A#25#16#1008#2018/11/7#12:51:14#144#0#0#0#SD

2A#25#16#1008#2018/11/7#12:51:16#144#0#0#0#SD

2A#25#16#1008#2018/11/7#12:51:19#144#0#0#0#SD

NÓ de Sensor - 3A

3A#54#25#966#2018/11/7#11:53:11#1315#0#0#0#SD

{*1} 3A#54#25#967#2093/18/11#3:11:53#1287#0#0#0#SD

{*2} 3A#69#24#966#2093/18/11#3:11:53#1315#0#0#0#SD

3A#25#25#966#2018/11/7#11:53:20#1315#0#0#0#SD

3A#62#25#967#2018/11/7#11:53:23#1287#0#0#0#SD

3A#90#24#966#2018/11/7#11:53:26#1315#0#0#0#SD

Extrato de dados da Base de Dados:

```
{*3} "1A","92","18","100","7201/81/17","11:33:34",-13730,"0.000","0.000","0","BD"
"2A","0","0","0","2018/11/7","11:33:33","0","0.000000","0.000000","0","BD"
"2A","0","0","0","2018/11/7","11:33:33","0","0.000000","0.000000","0","BD"
{*4} "1A","0","0","0","2000/0/0","0:0:28","0","0.000000","0.000000","0","BD"
"2A","0","0","0","2018/11/7","11:33:33","0","0.000000","0.000000","0","BD"
"1A","0","0","0","2000/0/0","0:0:28","0","0.000000","0.000000","0","BD"
"2A","0","0","0","2018/11/7","11:33:33","0","0.000000","0.000000","0","BD"
"2A","0","0","0","2018/11/7","11:34:34","0","0.000000","0.000000","0","BD"
"2A","0","0","0","2018/11/7","11:34:34","0","0.000000","0.000000","0","BD"
"2A","0","0","0","2018/11/7","11:34:54","0","0.000000","0.000000","0","BD"
{*5} "1A","95","18","100","7201/81/17","11:35:21",-13730,"0.000","0.000","0","BD"
"1A","94","16","697","2018/11/71","21:34:9","9989","39.477772",-8.373653,"288"
"1A","94","16","697","2018/11/71","21:34:9","9989","39.478191",-8.373899,"285"
```

Extrato ficheiro TXT do Gateway:

3A#0#0

3A#0#0

3A#0#0

3A#0#0

Resultados obtidos: Neste teste observou-se que a estabilidade dos NÓS de Sensores foram melhores, no entanto, ainda existiram falhas de hardware e software.

No NÓ 1A as comunicações com o Gateway foram relativamente boas. O seu microSD foi perdido durante os testes, não podendo assim observar os seus dados. No entanto, o Gateway registou algumas transmissões. Nestas transmissões foi observado que ainda existem alguns erros do software (alíneas: Base de dados *3, *4 e *5). A junção da mensagem extraída do array, por vezes não corre como previsto. Esta junção só é necessária porque o SLAVE tem de enviar a sua mensagem no I2C sem espaços por limitações da comunicação, contudo foi corrigida no código do MASTER.

No NÓ 2A, apresentou o mesmo problema que no quarto salto. Assim tivemos muitas leituras de vinte e cinco (25) em SpO2. As outras funções correram normalmente.

No NÓ 3A, aconteceu o mesmo problema do envio das mensagens, estas não são recebidas pelo Gateway. Neste NÓ de Sensor também existiu o problema da extração dos

valores da Array (*1 e *2), também já corrigido no código do MASTER.

O Gateway teve um comportamento como previsto, embora ainda não esteja salvaguardado a recepção de mensagens incompletas, como aconteceu com as mensagens enviadas do NÓ 3A.

Com o GPS ligado antes do embarque no avião, não foi possível adquirir sinal de satélite. O mesmo aconteceu com os testes em que se ligou o GPS na saída do avião. Conclui-se que com o módulo existente, com o movimento do avião, não consegue adquirir o sinal.

Capítulo 7

Conclusões e trabalho futuro

Com este trabalho e consequente investigação foram abordados vários aspetos relevantes considerando todo o sistema, desde a aquisição de sinais até à arquitetura da rede de sensores enquadrada no âmbito das aplicações em IoT.

7.1 Conclusões

O trabalho desenvolvido teve vários âmbitos de atuação, desde o desenvolvimento de hardware para aquisição de dados, dos protocolos de comunicações de toda a rede de sensores e também do desenvolvimento do gateway e do sistema de informação.

Relativamente ao desenvolvimento do hardware do NÓ de Sensores, este é um desafio constante, não só pelo estado de maturação das tecnologias de rede assim como pelas limitações do hardware considerando o cenário de utilização. Tendo em conta os resultados obtidos neste projeto, é possível afirmar que o desempenho dos Nós de sensores foi bastante satisfatório, uma vez que para além dos testes de laboratório, foi possível efectuar testes em ambiente real.

Deste modo, foi possível verificar as potencialidades do sistema e identificar alguns constrangimentos a ter em conta numa no desenvolvimento de uma versão 2.0 com um grau de robustez melhorada.

A título de exemplo, conclui-se que o módulo GPS não se mostrou adequado para uso com velocidades de deslocação elevadas, assim como, o tipo de sensor de SpO2 e a sua colocação necessita de maior estudo e testes para esta aplicação específica, para que as suas medidas sejam mais robustas e fiáveis.

Nesta investigação concluiu-se que a tecnologia LoRa, é uma tecnologia interessante para os NÓS de Sensores, podendo ser usada para transmitir a longas distâncias. Existem protocolos nas suas bibliotecas bastante estáveis e amigáveis.

Relativamente ao módulo usado, *Shield Multiprotocol*, concluiu-se que esta opção não permitiu a optimização dos portos de comunicação disponíveis nos módulos do nó, não permitindo a ligação de outros dispositivos.

No que diz respeito aos sensores utilizados, o sensor de oximetria (*MAX30105*), teve um bom desempenho nas leituras na ponta dos dedos, enquanto que no pulso não se

mostrou tão eficaz. O sensor de pressão (*MPL3115A*), demonstrou grande estabilidade nas leituras.

O modulo de GPS (*NEO-6M - UART*), obteve leituras em testes de laboratório, mas em testes reais o seu desempenho não foi satisfatório.

O módulo de relógio (*RTC DS1307*) desprogramou-se algumas vezes. O módulo de comunicação rádio LoRa (*SX1272 Libelium*), foi bastante estável e comunicou a grandes distâncias.

O NÓ de Sensor foi desenvolvido usando uma estrutura modular com dois Arduinos MEGA, um para a aquisição outro para a transmissão. Foi a solução possível para o tempo e espaço, pelas limitações dos sensores e da placa de multi-funções, que não permitiram a utilização de uma arquitetura com apenas um módulo.

Os NÓS de Sensores testados em laboratório, superaram as expectativas e o objetivos propostos. Nos testes reais os NÓS de Sensores, tiveram anomalias, não só por ser um hardware de teste em protótipo, como também pelas características do meio ambiente em que estavam inseridos, devido a vários fatores perturbadores, tais como, a fuselagem do avião, a vibração, o ruído das comunicações da própria aeronave, a baixa temperatura, o salto em paraquedas e os violentos impactos sobre o aparelho, levaram a que os NÓS de Sensores muitas das vezes não tivessem um comportamento correspondente ao previsto.

A dimensão da mensagem na transmissão de cada nó para o Gateway, limitou o registo de dados.

Em suma, consideramos que com estes testes, o desempenho obtido na investigação e na implementação destas tecnologias é bastante satisfatório. Os resultados obtidos, foram esclarecedores para o desenvolvimento de um protótipo com hardware e software adequado ao ambiente real e zelar pela vida do seu utilizador, salvaguardando ou minimizando o aparecimento de patologias associadas e da própria morte.

Do ponto de vista dos saltadores que testaram o sistema, estes destacaram a importância da leitura dos valores de SpO2 em tempo real, assim como a utilização do sistema de GPS, principalmente em saltos noturnos.

Da análise, que resultaram do presente estudo, podemos concluir que é de suma importância, para precaver a perda de vidas humanas, a continuação e aperfeiçoamento do presente projeto.

7.2 Trabalho futuro

Futuramente, considera-se que são necessárias melhorias no sistema para um bom desempenho dos NÓS de Sensores em ambiente militar.

Ao nível do hardware, o desenvolvimento do nó de sensor passa por reduzir o seu tamanho, usando um único módulo eliminando a *Shield Multiprotocol*. Relativamente aos sensores, dever-se-á testar outros módulos de oximetria, que sejam mais fiáveis para leituras no pulso ou noutra parte do corpo, nomeadamente ao nível do crânio. O objetivo final seria desenvolver um NÓ de Sensores em tamanho *NANO*, a utilizar como um relógio de pulso.

Por outro lado, deve ser considerado integrar o módulo LoRa Libelium com outros sistemas de radio LoRa e testar-se outros módulos de GPS mais rápidos a adquirir o posicionamento dos satélites.

Ao nível da comunicação, devem ser implementados protocolos de segurança na rede LoRa.

Considerando o Gateway, é necessário desenvolver a integração de registos dos cartões SD com a base de dados, salvaguardando assim os registos duplos. Por outro lado, a mensagem deveria ser dividida, com uma parte menos importante para a comunicação ficar guardada no cartão SD e a outra ser transmitida (SpO2 e GPS), diminuindo assim o seu tamanho. Outra funcionalidade a desenvolver consiste no cálculo da distância entre os NÓS e o Gateway.

Outra componente do trabalho, a desenvolver, consiste na integração do gateway com um sistema de informação mais alargado, baseado em serviços cloud, que permita uma análise mais aprofundada dos dados recolhidos por parte de pessoal médico especializado, para o estudo aprofundado das causas associadas às doenças típicas resultantes deste tipo de atividade.

Bibliografia

- [1] “Wireless sensor networks.” <https://trends.google.pt>, 2018. (Accessed on 01/01/2018).
- [2] “Internet of things.” <https://trends.google.pt>, 2018. (Accessed on 12/13/2017).
- [3] S. Charbathia and S. Sharma, “A comparative study of rivest cipher algorithms,” *International Journal of Information and Computation Technology*, vol. 4, pp. 1831–1838, 2014.
- [4] “Comparação entre tecnologias.” <https://www.networkworld.com/article/3211390/internet-of-things/lorawan-key-to-building-full-stack-production-iot-networks.html>, 2018. (Accessed on 01/09/2018).
- [5] “Camadas lógicas de rede lorawan.” <https://www.leverage.com/research-papers/lora-lorawan-primer>, 2018. (Accessed on 01/09/2018).
- [6] “Camada de modulação.” <https://en.wikipedia.org/wiki/Chirp>, 2018. (Accessed on 01/09/2018).
- [7] “Quadro de frequências lora.” <http://www.3glteinfo.com/lora/lorawan-frequency-bands/>, 2018. (Accessed on 01/09/2018).
- [8] “Estrutura do canal 868 mhz.” <http://www.rfwireless-world.com/Tutorials/LoRa-channels-list.html>, 2018. (Accessed on 01/09/2018).
- [9] “Segurança lorawan.” <https://www.thethingsnetwork.org/wiki/LoRaWAN/Home>, 2018. (Accessed on 01/09/2018).
- [10] “Dispositivos - estrutura lora.” <https://www.wirelessdesignmag.com/article/2015/04/lora-alliance-poised-make-lpwan-reality-m2m>, 2018. (Accessed on 01/09/2018).
- [11] “Adafruit feather 32u4 rfm95 lora radio- 868 or 915 mhz [radiofruit] id: 3078 - \$34.95 : Adafruit industries, unique & fun diy electronics and kits.” <https://www.adafruit.com/product/3078>. (Accessed on 01/11/2018).

- [12] “Sigfox - the global communications service provider for the internet of things (iot).” <https://www.sigfox.com/en>. (Accessed on 01/11/2018).
- [13] “Wasmote - open source sensor node for the internet of things | zigbee, sigfox, lorawan, 3g / 4g compatible | libelium.” <http://www.libelium.com/products/wasmote/>. (Accessed on 01/11/2018).
- [14] “Extreme range links: Lora 868 / 900mhz sx1272 module for arduino, waspmote and raspberry pi.” <https://www.cooking-hacks.com/documentation/tutorials/extreme-range-lora-sx1272-module-shield-arduino-raspberry-pi-intel-galileo/>. (Accessed on 01/11/2018).
- [15] “Placa arduino mega.” <https://store.arduino.cc/arduino-mega-2560-rev3>, 2018. (Accessed on 01/09/2018).
- [16] “Sensor de oxímtria max30105.” <https://www.sparkfun.com/products/14045>, 2018. (Accessed on 01/12/2018).
- [17] “Sensor de pressão mpl3115a2.” <https://www.boxelectronica.com/pt/sensores-arduino/894-sparkfun-sensor-de-altitude-pressao-mpl3115a2.html>, 2018. (Accessed on 01/12/2018).
- [18] “Módulo gps neo-6m.” <http://www.botnroll.com/pt/gsm/2022-modulo-gps-neo-6m-uart-.html>, 2018. (Accessed on 01/12/2018).
- [19] “Módulo oled lcd 0.96” i2c iic.” <https://www.electrofun.pt/display/display-oled/modulo-display-oled-lcd-0-96-i2c>, 2018. (Accessed on 01/12/2018).
- [20] “Módulo rgb.” <https://www.electrofun.pt/modulo-led-rgb-5mm>, 2018. (Accessed on 01/12/2018).
- [21] “Módulo tca9548a i2c multiplexer.” <https://www.ptrobotics.com/varios/4385-tca9548a-i2c-multiplexer.html>, 2018. (Accessed on 01/12/2018).
- [22] “Módulo microsd.” <http://www.botnroll.com/pt/arduino-acessorios/1401-modulo-microsd-para-arduino.html>, 2018. (Accessed on 01/12/2018).
- [23] “Módulo libelium paraarduino.” <https://www.cooking-hacks.com/sx1272-lora-shield-for-raspberry-pi-868-mhz>, 2018. (Accessed on 01/12/2018).

-
- [24] M. Machado, “Efeitos fisiopatologicos das missoes sogas.” <http://www.operacional.pt/efeitos-fisiopatologicos-das-missoes-soga>, Ago 2016. (Accessed on 04/11/2018).
- [25] D. Smihula, “Wave of technological innovations and teen do the information revolution,” *Jornal of Economics and international Finance Vol.2(4)*, pp. 58–67, 2010.
- [26] K. Ashton, “That ‘internet of thing’ thing.” online on 28/12/2017: <http://rfdjournal.com/articles/view?4986>, jun 2009.
- [27] “Gartner’s 2016 hype cycle for emerging technologies identifies three key trends that organizations must track to gain competitive advantage.” <https://www.gartner.com/newsroom/id/3412017>. (Accessed on 03/05/2018).
- [28] euronews, “A vida privada na internet..” online on 28/12/2017: <http://pt.euronews.com/2012/02/08/a-vida-privada-na-internet>, fev 2012.
- [29] K. E. Nolan, W. Guibene, and M. Y. Kelly, “An evaluation of low power wide area network technologies for the internet of things,” *Wireless Communications and Mobile Computing Conference (IWCMC), 2016 International*, pp. 439–444, 2016.
- [30] A. P. Konstantinova, “a great contributor to the development of wireless communication,” *paper, IEEE Conference on the History of Electronics*, pp. 1–12, 2004.
- [31] Wikipedia, “sound surveillance system.” online on 29/12/2017: <https://en.wikipedia.org/wiki/SOSUS>, set 2017.
- [32] C.-Y. Chong and S. P. Kumar, “Sensor networks: Evolution, opportunities, and challenges,” *paper, Proceedings of the IEEE, VOL. 91, NO. 8*, pp. 1247–1256, 2003.
- [33] ariasystems, “The past, present and future of the internet of things.” online on 02/01/2018: <https://www.ariasystems.com/resources/past-present-and-future-internet-things>, set 2017.
- [34] B. Santos, L. Silva, Clayson, C. João, N. Bruna, P. Marcos, V. Luiz, V. Olga, Goussevskaia, and A. Loureir, “Internet das coisas: da teoria à prática,” in *Livro de Minicursos SBRC 2016*, pp. 1–329, jun 2016.
- [35] Intel.com, “Delivering impactful smart home experiences.” online on 04/01/2018: <https://www.intel.com/content/www/us/en/internet-of-things/smart-home.html>, Jan 2018.

- [36] D. Doolina and N. Sitar, “Wireless sensors for wildfire monitorin,” *White Paper: University of California*.
- [37] C. Seidel, F. Macêdo, F. Etienne, and C. Oliveira, “AplicaÇÃo de redes de sensores sem fio (rssi) para engenharia ambiental,” *White Paper*.
- [38] M. Suzuki, S. Saruwatari, N. Kurata, and H. morikawa, “A high-density earthquake monitoring system using wireless sensor networks,” *White Paper: SenSys November 2007 Sydney Australia*, pp. 6–9, 2007.
- [39] N. Wang, N. Zhang, and M. Wang, “Wireless sensors in agriculture and food industry, recent development and future perspective,” *White Paper Review: ELSEVIER*, pp. 1–14, 2006.
- [40] G. Yang, L. Xie, M. Mäntysalo, X. Zhou, Z. Pang, L. Da Xu, S. Kao-Walter, Q. Chen, and L.-R. Zheng, “A health-iot platform based on the integration of intelligent packaging, unobtrusive bio-sensor, and intelligent medicine box,” *IEEE transactions on industrial informatics*, vol. 10, no. 4, pp. 2180–2191, 2014.
- [41] S. Shinde, T. Kadaskar, P. Patil, and R. Barathe, “A smart pill box with remind and consumption using iot,” 2017.
- [42] K. Dhariwal and A. Mehta, “Architecture and plan of smart hospital based on internet of things (iot),” 2017.
- [43] A. Sfar, Z. Chtourou, E. Natalizio, and Y. Challal, “A systemic and cognitive vision for iot security: a case study of military live simulation and security challenges,” in *International Conference on Smart, Monitored and Controlled Cities SM2C’17*, 2017.
- [44] S. Silfverskiöld, H. Liwång, G. Hult, Å. Sivertun, P. Bull, J. Sigholm, M. Lundmark, C. von Gerber, K. Andersson, and P. Stureson, “Technology forecast 2017-military utility of future technologies: A report from seminars at the swedish defence university’s (sedu) military-technology division,” 2017.
- [45] R. N. Akram, K. Markantonakis, K. Mayes, O. Habachi, D. Sauveron, A. Steyven, and S. Chaumette, “Security, privacy and safety evaluation of dynamic and static fleets of drones,” in *Digital Avionics Systems Conference (DASC), 2017 IEEE/AIAA 36th*, pp. 1–12, IEEE, 2017.
- [46] H. Davis, *Absolute Beginner’s Guide to Wi-Fi Wireless Networking*. Indianapolis, IN, USA: Que Corp., 2004.

-
- [47] N. R. Malankar and R. Shah, “Qos analysis over wimax network with varying modulation schemes and efficiency modes,”
 - [48] Z. C. Corp., “Lte is the fourth-generation technology with a great improvement on spectrum efficiency, higher user data rate, low latency, cost reduction, system capacity and coverage.” online on 05/01/2018: <https://www.zyxel.com/solutions/LTE-Solutions-20120224-050736.shtml>.
 - [49] Z. Alliance, “Zigbee 3.0 | zigbee alliance.” online on 05/01/2018: <http://www.zigbee.org/zigbee-for-developers/zigbee-3-0/>.
 - [50] “Bluetooth low energy - wikipedia.” https://en.wikipedia.org/wiki/Bluetooth_Low_Energy. (Accessed on 01/05/2018).
 - [51] “Radio-frequency identification - wikipedia.” https://en.wikipedia.org/wiki/Radio-frequency_identification. (Accessed on 01/05/2018).
 - [52] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, “Understanding the limits of lorawan,” *IEEE Communications Magazine*, vol. 55, no. 9, pp. 34–40, 2017.
 - [53] “Sigfox vs. lora: A comparison between technologies & business models.” <https://www.link-labs.com/blog/sigfox-vs-lora>. (Accessed on 01/05/2018).
 - [54] P. Barker and M. Hammoudeh, “A survey on low power network protocols for the internet of things and wireless sensor networks,” in *Proceedings of the International Conference on Future Networks and Distributed Systems*, p. 33, ACM, 2017.
 - [55] S. Hu, A. Berg, X. Li, and F. Rusek, “Improving the performance of otdoa based positioning in nb-iot system,” *arXiv preprint arXiv:1704.05350*, 2017.
 - [56] Y.-P. E. Wang, X. Lin, A. Adhikary, A. Grövlén, Y. Sui, Y. Blankenship, J. Bergman, and H. S. Razaghi, “A primer on 3gpp narrowband internet of things (nb-iot),” *arXiv preprint arXiv:1606.04171*, 2016.
 - [57] “Presentation title.” http://www.3gpp.org/images/presentations/3GPP_Standards_for_IoT.pdf. (Accessed on 01/05/2018).
 - [58] “Microsoft powerpoint - wireless for process industries_iet_3a_2007.ppt.” <https://library.e.abb.com/public/4e4d05f06a16191985257bc6005f7274/Wireless%20Instrumentation%20for%20Process%20Industries.pdf>. (Accessed on 01/06/2018).

- [59] M. Nixon, “A comparison of wirelesshart™ and isa100.11a,” *White Paper: Comparison of WirelessHART and ISA100.11a*, pp. 1–39, 2012.
- [60] “Telensa - smart city iot powered by connected street lighting.” <https://www.telensa.com/>. (Accessed on 01/06/2018).
- [61] “Lora integration & sx1272 & sx1276 modules - link labs.” <https://www.link-labs.com/lora>. (Accessed on 01/11/2018).
- [62] N. Keith, G. Wael, and K. Mark, “An evaluation of low power wide area network technologies for the internet of thing,” pp. 439–444, 2016.
- [63] “Lorawan mac layer.” <https://sirinsoftware.com/blog/lorawan-mac-layer/>. (Accessed on 01/11/2018).
- [64] “Loramac: Introduction.” <https://stackforce.github.io/LoRaMac-doc/>. (Accessed on 01/11/2018).
- [65] J. de Carvalho Silva, J. J. Rodrigues, A. M. Alberti, P. Solic, and A. L. Aquino, “Lorawan—a low power wan protocol for internet of things: A review and opportunities,” in *Computer and Energy Science (SpliTech), 2017 2nd International Multidisciplinary Conference on*, pp. 1–6, IEEE, 2017.
- [66] N. Harris and J. Curry, “Development and range testing of a lorawan system in an urban environment,” *World Academy of Science, Engineering and Technology, International Journal of Electronics and Communication Engineering*, vol. 5, no. 1, 2018.
- [67] “Rn2483 - wireless modules.” <https://www.microchip.com/wwwproducts/en/RN2483>. (Accessed on 01/11/2018).
- [68] “Long range, low power rf transceiver with lora technology | semtech.” <https://www.semtech.com/products/wireless-rf/lora-transceivers>. (Accessed on 01/11/2018).
- [69] B. M. Talha, E. Mehmet Ali, A. Muhammed Ali, and V. Luca, “Lorawan as an e-health communication technology,” *Computer Software and Applications Conference*, no. 41st, p. 4, 2017.
- [70] M. Afef, C. Tarak, L. Kaouthar, J. Mohamed, and F. Bernd, “Iot-based health monitoring via lorawan,” *EUROCON*, p. 6, 2017.

- [71] S. T. V. N. N. K. Danco Davcev, Kosta Mitreski, "Iot agriculture system based on lorawan," *UKIM, Faculty of Computer Science and Engineering Skopje, Macedonia*, p. 4, 2018.
- [72] N. Ruhaizan Fazrren Ashraff Mohd and Z. Fadhlan H. K., "Smart traffic light for congestion monitoring using lorawan," *Control and System Graduate Research Colloquium*, no. 8th, p. 6, 2017.
- [73] S. Hafiz Husnain Raza, G. Piro, G. Luigi Alfredo, and G. Boggia, "When renewable energy meets lora: A feasibility analysis on cable-less deployments," *INTERNET OF THINGS JOURNAL*, no. 10, p. 12, 2018.
- [74] L. Yan-Ting, L. Bo-Yi, Y. Xiao-Feng, C. Zong-Xuan, Y. Zi-Xian, L. Wei-Hong, H. Song-Yi, L. Jun-Lin, P. Jing-Wen, and C. Jen-Yeu, "A solar powered long range real-time water quality monitoring system by lorawan," *Wireless and Optical Communications Conference*, no. 27th, p. 2, 2018.
- [75] B. Jalaian, T. Gregory, N. Suri, S. Russell, L. Sadler, and M. Lee, "Evaluating lorawan-based iot devices for the tactical military environment," *U.S. Army Research Laboratory*, p. 5, 2018.
- [76] H.-C. Lee and K.-H. Ke, "Monitoring of large-area iot sensors using a lora wireless mesh network system: Design and evaluation," *TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT*, p. 11, 2018.
- [77] Y. Asif M, R. Edward M, and M. Ghaderi, "A low-cost lorawan testbed for iot: Implementation and measurements," *TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT*, p. 6, 2018.
- [78] X. Yang, E. Karampatzakis, C. Doerr, , and F. Kuipers, "Security vulnerabilities in lorawan," *Third International Conference on Internet-of-Things Design and Implementation*, p. 12, 2018.
- [79] "Raspberry pi 7"touchscreen display case - black." <https://www.ptrobotics.com/lcd-cameras-raspberry-pi/4633-raspberry-pi-7-touchscreen-display-case-black.html>. (Accessed on 06/09/2018).
- [80] "Cooking hacks - electronic and iot kits, tutorials and guides for makers and education." <https://www.cooking-hacks.com/>. (Accessed on 06/09/2018).

- [81] “Raspberry pi to arduino shields connection bridge.” <https://www.cooking-hacks.com/documentation/tutorials/raspberry-pi-to-arduino-shields-connection-bridge/>. (Accessed on 06/09/2018).
- [82] “Arduino - arduinomega2560.” <https://www.arduino.cc/en/Guide/ArduinoMega2560>. (Accessed on 06/09/2018).
- [83] “Sparkfun particle sensor breakout - max30105 - sen-14045 - sparkfun electronics.” <https://www.sparkfun.com/products/14045>. (Accessed on 06/08/2018).
- [84] “Mpl3115a2 pressure sensor hookup guide - learn.sparkfun.com.” <https://learn.sparkfun.com/tutorials/mp3115a2-pressure-sensor-hookup-guide>. (Accessed on 06/08/2018).
- [85] “Uart gps neo-6m uart gps module, u-blox neo-6m onboard, curved/horizontal pinheader.” <https://www.waveshare.com/product/UART-GPS-NEO-6M.htm>. (Accessed on 06/08/2018).
- [86] “Github - mikalhart/tinygpsplus: A new, customizable arduino nmea parsing library.” <https://github.com/mikalhart/TinyGPSPPlus>. (Accessed on 06/09/2018).
- [87] “Oled : Adafruit industries, unique & fun diy electronics and kits.” https://www.adafruit.com/category/63_98. (Accessed on 06/09/2018).
- [88] “Overview | arduino lesson 3. rgb leds | adafruit learning system.” <https://learn.adafruit.com/adafruit-arduino-lesson-3-rgb-leds/overview>. (Accessed on 06/09/2018).
- [89] “Overview | adafruit tca9548a 1-to-8 i2c multiplexer breakout | adafruit learning system.” <https://learn.adafruit.com/adafruit-tca9548a-1-to-8-i2c-multiplexer-breakout/overview>. (Accessed on 06/09/2018).
- [90] “Microsd card module for arduino (sku:dfrobot229) - dfrobot electronic product wiki and tutorial: Arduino and robot wiki-dfrobot.com.” [https://www.dfrobot.com/wiki/index.php/MicroSD_card_module_for_Arduino_\(SKU:DFR0229\)](https://www.dfrobot.com/wiki/index.php/MicroSD_card_module_for_Arduino_(SKU:DFR0229)). (Accessed on 06/09/2018).
- [91] “pressurealtitude.pdf.” <https://www.weather.gov/media/epz/wxcalc/pressureAltitude.pdf>. (Accessed on 06/21/2018).