



Universidade do Minho
Escola de Engenharia

Pedro Filipe Mendes de Castro

Monitorização e Controlo em
Ambientes Industriais Através
de Rede de Sensores Sem Fios



Universidade do Minho
Escola de Engenharia

Pedro Filipe Mendes de Castro

Monitorização e Controlo em
Ambientes Industriais Através
de Rede de Sensores Sem Fios

Dissertação de Mestrado
Ciclo de Estudos Integrados Conducentes ao Grau de
Mestre em Engenharia de Telecomunicações e Informática

Trabalho efectuado sob a orientação do
Professor Doutor José Augusto Afonso

DECLARAÇÃO

Nome: Pedro Filipe Mendes de Castro

Correio electrónico: filmendes.castro@gmail.com

Tlm.: 915276483

Número do Bilhete de Identidade:12610111

Título da dissertação:

Monitorização e Controlo em Ambientes Industriais Através de Redes de Sensores Sem Fios

Ano de conclusão:2015

Orientador:

José Augusto Afonso

Designação do Mestrado:

Ciclo de Estudos Integrados Conducentes ao Grau de Mestre em Engenharia de Telecomunicações e Informática

Área de Especialização: Telecomunicações e Informática

Escola: Engenharia

Departamento: Eletrónica e Informática

1. É AUTORIZADA A REPRODUÇÃO INTEGRAL DESTA DISSERTAÇÃO APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE.
2. É AUTORIZADA A REPRODUÇÃO PARCIAL DESTA DISSERTAÇÃO (indicar, caso tal seja necessário, nº máximo de páginas, ilustrações, gráficos, etc.), APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE.
3. De acordo com a legislação em vigor, não é permitida a reprodução de qualquer parte desta dissertação

Guimarães, ___/___/_____

Assinatura: _____

AGRADECIMENTOS

Agradeço ao professor José Afonso, pela orientação, dedicação e paciência no decorrer deste trabalho.

Agradeço aos meus amigos António, Bruno, João, Philippe e ao Nuno pelo convívio e pela ajuda que me deram durante estes anos.

Agradeço à Cláudia e à minha mãe por tudo.

RESUMO

Ao longo dos últimos anos tem existido um elevado crescimento na utilização de dispositivos de comunicação sem fios, em que se destacam as redes de sensores sem fios, com elevado número de possíveis aplicações, nas mais diversas áreas, sendo cada vez mais uma realidade no quotidiano das pessoas e da indústria. As redes de sensores sem fios têm como principal objetivo o encaminhamento dos dados sensoriais recolhidos por determinados nós da rede para outros nós da rede (e.g., dados recolhidos pelos sensores de um dispositivo terminal e enviados para uma estação base).

Nesta dissertação, foi realizado um projeto que consiste no estudo, desenvolvimento e testes de uma rede sem fios ZigBee, para controlar e monitorizar equipamentos em ambiente industrial. A implementação desta rede surge como um processo de substituição da rede cablada existente e dos dispositivos que controlam os equipamentos. Esta implementação foi feita utilizando plataformas de hardware e software da Texas Instruments e o IDE NetBeans. Numa fase inicial foram realizados testes para avaliação da taxa de erros na transmissão num ambiente industrial, de forma a poder viabilizar o projeto. Foi desenvolvido código para dispositivos do tipo *router* ZigBee com a capacidade de controlar equipamentos de refrigeração, recolher os dados dos sensores que lhe estão associados, e enviá-los pela rede até ao seu destino, diretamente ou por intermédio de outros *routers*. Foi desenvolvido também um coordenador ZigBee ligado a um PC que, através de uma interface gráfica com o utilizador, tem a capacidade de monitorizar e controlar todos os *routers* existentes na rede.

ABSTRACT

Over the past few years there has been a substantial increase in the use of wireless communication devices, in particular wireless sensor networks, with a wide range of possible applications in the most diversified areas, becoming a constant reality in people's life and industry. Wireless sensor networks are meant to forward the collected data from some network nodes to other nodes (e.g., data collected from the sensors of a terminal device and sent to a base station).

In this dissertation, a project has been developed aiming at studying, develop and testing a ZigBee wireless network, to control and monitor equipment in an industrial environment, replacing the existing cables and devices that control those equipments. The implementation was made using hardware and software platforms of Texas Instruments and the NetBeans IDE. In an early stage, evaluation tests of the transmission error rate were conducted in an industrial environment. ZigBee router devices were developed with the capacity to control cooling equipments, collect data from the sensors associated with it, and send them through the network to their destination, directly or through other routers. A ZigBee coordinator connected to a PC was also developed, enabling the monitoring and control of all existing routers in the network through a graphical user interface.

ÍNDICE DE CONTEÚDOS

AGRADECIMENTOS	iii
RESUMO	v
ABSTRACT.....	vii
ÍNDICE DE CONTEÚDOS.....	ix
LISTA DE FIGURAS	xiii
LISTA DE TABELAS	xv
LISTA DE ABREVIATURAS.....	xvii
1. INTRODUÇÃO.....	1
1.1 Enquadramento e motivação	1
1.2 Objetivos	2
1.3 Estrutura da dissertação	2
2. ESTADO DA ARTE	5
2.1 Redes de sensores sem fios	5
2.1.1 Introdução	5
2.1.2 Aplicações	6
2.1.3 Protocolos MAC	7
2.1.3.1 Protocolo S-MAC	7
2.1.3.2 Protocolo LEACH.....	8
2.2 IEEE 802.15.4.....	10
2.2.1 Introdução	10
2.2.2 Camada física	12

2.2.3	Camada MAC	15
2.3	ZigBee.....	16
2.3.1	Introdução	16
2.3.2	Protocolo ZigBee.....	17
2.3.2.1	Camada de rede.....	18
2.3.2.2	Camada de aplicação	19
2.3.3	Tipos de dispositivos.....	20
2.3.4	Formato das tramas.....	20
2.4	LonWorks	21
3.	DESCRIÇÃO DO SISTEMA ATUAL.....	25
3.1	Introdução.....	25
3.2	Equipamento de refrigeração	25
3.3	Quadro elétrico	26
3.4	Sondas.....	27
3.5	Micro de porta	28
3.6	Válvula de expansão	28
3.7	Ventilador.....	29
3.8	Controlador de gestão central	30
3.9	Repetidor	30
3.10	Topologia da rede	30
4.	DESENVOLVIMENTO DO SISTEMA	33
4.1	Introdução.....	33
4.2	Plataforma de desenvolvimento.....	33
4.3	Descrição geral.....	36

4.4	<i>Router</i> ZigBee	38
4.4.1	Sensores e atuadores	38
4.4.2	Modo de funcionamento.....	42
4.5	Coordenador ZigBee	43
4.6	Rede ZigBee.....	45
4.7	Software	48
4.7.1	Primeira fase	50
4.7.2	Segunda fase.....	54
5.	RESULTADOS E DISCUSSÃO	59
5.1	Introdução.....	59
5.2	Primeira fase de testes	59
5.3	Segunda fase de testes	63
6.	CONCLUSÕES	67
	REFERÊNCIAS	71

LISTA DE FIGURAS

Figura 1. Arquitetura básica de um nó de uma rede de sensores sem fios.	6
Figura 2. Princípio de funcionamento do protocolo S-MAC.....	8
Figura 3. Exemplo de configuração de uma ronda do protocolo LEACH.....	10
Figura 4. Exemplo de topologias em estrela e peer-to-peer do IEEE 802.15.4.	11
Figura 5. Arquitetura LR-WPAN do IEEE 802.15.4.	12
Figura 6. Camadas detalhadas da arquitetura do protocolo ZigBee [18].....	17
Figura 7. Terminologia básica do protocolo LonTalk [21].....	22
Figura 8. Camadas do protocolo LonTalk.....	23
Figura 9. Ilustração de um equipamento de refrigeração genérico.	26
Figura 10. Controladores AK-PC 530 (esquerda) e AK-CC 550 (direita).....	27
Figura 11. Sonda EKS 111.....	27
Figura 12. Curva característica da sonda EKS 111.	28
Figura 13. Válvula de expansão.	29
Figura 14. Exemplo de ventilador.	29
Figura 15. Topologia da rede.	31
Figura 16. Módulo CC2530EM a), módulo CC2531 b), SmartRF05EB c), antena d). ...	34
Figura 17. Descrição geral de funcionamento.	37
Figura 18. Divisor de tensão.....	39
Figura 19. Valor da resistência medido e teórico.	40
Figura 20. Interface de seleção de posto.....	44
Figura 21. Interface de visualização de dados de um posto.....	45
Figura 22. Formato da trama de <i>beacon request</i>	45

Figura 23. Formato da trama de pedido de associação.....	45
Figura 24. Formato da trama de resposta ao pedido de associação.....	46
Figura 25. Formato da trama de dados do ZC.	47
Figura 26. Sequência do fluxo de dados entre ZC e ZR.....	48
Figura 27. Fluxograma do algoritmo de cálculo de taxa de erros.	53
Figura 28. Fluxograma de algoritmo de funcionamento de equipamento de refrigeração.....	57
Figura 29. Planta da fábrica com a disposição dos dispositivos.	60
Figura 30. Ocupação do canal 11 da banda de frequência de 2.4 GHz.	62
Figura 31. Taxa de erros nos canais 11 e 26.	63
Figura 32. Planta da empresa com disposição dos dispositivos na 2ª fase.	64
Figura 33. Taxa de erros do canal 26, com o envio de pacotes de 20 e 90 bytes.	66

LISTA DE TABELAS

Tabela 1. Características da camada física do IEEE 802.15.4.....	15
Tabela 2. Valores lidos pelo ADC.	41
Tabela 3. Testes no canal 26.....	61
Tabela 4. Testes no canal 11.....	62
Tabela 5. Testes no canal 26.....	65

LISTA DE ABREVIATURAS

ACK	Acknowledgement
ADC	Analog-to-Digital converter
AES	Advanced Encryption Standard
AF	Application Framework
APDU	Application Protocol Data Unit
API	Application Programming Interface
APL	Application Layer
APS	Application Support Sublayer
APSME-SAP	Application Service Management Entity Access Point
APSD-ESAP	Application Service Data Entity Access Point
ASK	Amplitude-Shift Keying
BPSK	Binary Phase-Shift Keying
CCA	Clear Channel Assessment
CH	Cluster Head
CPU	Central Processing Unit
CDMA	Code Division Multiple Access
CSMA	Carrier Sense Multiple Access
CSMA-CA	Carrier Sense Multiple Access Collision Avoidance
CTS	Clear to Send
DSSS	Direct Sequence Spread Spectrum
ED	End Device

FFD	Full Function Device
GTS	Guaranteed Time Slot
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
LAN	Local Area Network
LEACH	Low-Energy Adaptive Clustering Hierarchy
LQI	Link Quality Indication
LR-WPAN	Low-Rate Wireless Personal Area Network
MAC	Medium Access Control
NLDE	NWK Layer Data Entity
NLME	NWK Layer Management Entity
NPDU	NWK Packet Data Unity
NWK	Network
O-QPSK	Offset Quadrature Phase shift Keying
OSI	Open System Interconnection
PAN	Personal Area Network
PHY	Physical
PSSS	Parallel Sequence Spread Spectrum
RFD	Reduced Function Device
RTS	Request to Send
SAP	Service Access Point
SoC	System-on-Chip
S-MAC	Sensor Medium Access Control
SPI	Serial Peripheral Bus

TDMA	Time Division Multiple Access
USART	Universal Synchronous Asynchronous Receiver Transmitter
USB	Universal Serial Bus
ZC	ZigBee Coordinator
ZDO	ZigBee Device Object
ZED	ZigBee End Device
ZR	ZigBee Router

1. INTRODUÇÃO

1.1 Enquadramento e motivação

O crescente desenvolvimento na área das telecomunicações proporcionou uma evolução muito grande nas redes sem fios, e o exemplo disso é que estas estão por toda a parte. A inexistência de ligações físicas entre os seus diversos dispositivos permitiu uma forte expansão deste mercado nas últimas décadas. Várias tecnologias deste género foram implementadas, sendo grande parte com o objetivo de proporcionar elevadas taxas de transmissão, como é o caso do Wi-Fi [1]. Outras, como o ZigBee [2], foram desenvolvidas para cenários em que elevadas taxas de transmissão não são relevantes, mas o baixo custo, baixo consumo de energia e pequeno tamanho dos nós têm importância acrescida, tendo aplicação no contexto das redes de sensores sem fios (RSSF).

A enorme evolução das RSSF na última década, associada ao surgimento de dispositivos sensores e atuadores de elevada precisão e baixo custo, permitiu o desenvolvimento de múltiplas aplicações em diversas áreas, como a medicina, agricultura, desporto, indústria, edifícios inteligentes, segurança, etc.

As RSSF adequam-se a ambientes cuja baixa taxa de transferência de dados é adequada e a pequena dimensão dos nós é fundamental. É o caso de ambientes industriais, onde é feito o controlo e monitorização de vários equipamentos, distribuídos numa determinada área. Neste tipo de ambiente, os nós de uma RSSF são responsáveis por encaminhar a informação adquirida pelos sensores até uma estação central, de forma a poderem ser monitorizados e controlados. Em redes ZigBee, os dispositivos terminais são normalmente responsáveis pela recolha de dados provenientes de sensores e pelo controlo de atuadores, enquanto os *routers* são responsáveis pelo encaminhamento dos dados até o coordenador da rede.

A aplicação de redes de sensores sem fios na indústria, como é o caso da rede ZigBee, permite a eliminação das cablagens, o que proporciona uma redução na mão-

de-obra e material necessário em relação às redes cabladas. As redes ZigBee permitem ainda o crescimento da rede, mesmo depois desta implementada e em funcionamento, sem que seja necessária a alteração manual da topologia da rede. Devido à sua simplicidade, as redes ZigBee, têm normalmente um menor custo associado, em comparação com as redes cabladas.

1.2 Objetivos

Pretende-se com esta dissertação criar um sistema de monitorização e controlo através de uma rede sem fios ZigBee para operar num ambiente industrial/comercial, nos quais se destacam os seguintes objetivos:

- Desenvolver o software e hardware necessários para monitorizar e controlar equipamentos de refrigeração espalhados num ambiente industrial/comercial.
- Substituir a rede cablada existente, bem como os dispositivos de controlo do equipamento e da rede, pela rede sem fios ZigBee, com os seus respetivos dispositivos.
- Efetuar testes e análises da rede sem fios implementada, de forma a retirar conclusões da sua viabilidade.

1.3 Estrutura da dissertação

Este capítulo descreve o enquadramento e a motivação que levaram ao desenvolvimento da dissertação, o conjunto dos objetivos a alcançar neste projeto e a estrutura da dissertação.

O capítulo 2 apresenta o estado da arte referente a protocolos de redes de sensores sem fios. É feita uma descrição geral deste tipo de redes, seguida da descrição da norma IEEE 802.15.4 e da especificação ZigBee. Este capítulo apresenta ainda uma introdução à tecnologia LonWorks e uma breve descrição do protocolo utilizado.

No capítulo 3 é feita uma descrição do sistema atual que se deseja alterar. Nesta descrição são identificados todos os dispositivos do equipamento de refrigeração, bem como as suas características e o seu modo de funcionamento.

O capítulo 4 apresenta uma descrição da plataforma de desenvolvimento utilizada. Apresenta também uma descrição do funcionamento geral do sistema, bem como uma descrição do *router* ZigBee desenvolvido, da interface com o utilizador associada ao coordenador ZigBee, e da rede ZigBee. Neste capítulo, é também descrito como foi implementado o código nos dispositivos ZigBee.

O capítulo 5 está reservado às duas fases de testes realizados. Na primeira fase estão descritos os testes realizados de forma a perceber se a rede implementada oferecia viabilidade ao projeto. Na segunda fase realizaram-se testes com o sistema implementado.

Por fim, no capítulo 6 são apresentadas as conclusões referentes ao projeto realizado e perspectiva de trabalho futuro.

2. ESTADO DA ARTE

2.1 Redes de sensores sem fios

2.1.1 Introdução

Os avanços ocorridos recentemente na área da microeletrónica permitem a possibilidade de agrupar componentes como microcontrolador, memória, sensores e transceptor de rádio em um ou mais circuitos integrados. As redes de sensores sem fios (RSSF) aproveitaram esta evolução, tornando-se assim acessíveis e de larga aplicação em diversas áreas [3], como a medicina [4], [5], agricultura [6], desporto [7], indústria [8], edifícios inteligentes [9], segurança, etc.[1].

Estas redes de sensores diferem das redes de computadores tradicionais em vários aspetos. As redes tradicionais são normalmente utilizadas em aplicações que necessitam de uma elevada taxa de transmissão, enquanto as RSSF são usadas normalmente para interagir com sensores e atuadores utilizando baixas taxas de transmissão. Nestas redes existem restrições como o consumo de energia, uma vez que os nós da rede são normalmente alimentados por baterias, um fator que condiciona aspetos como o processamento, memória e potência de transmissão, sendo que a última limita a distância entre os nós da rede. Uma forma de reduzir o consumo de energia dos nós consiste em comutar o nó para um muito baixo consumo de energia (*sleep*) sendo reativado sempre que pretenda iniciar alguma tarefa. As RSSF possuem normalmente um número elevado de nós distribuídos, que devem possuir mecanismos de autoconfiguração, para o caso de falhas nos nós constituintes da rede.

Os nós de uma rede de sensores sem fios são constituídos por uma unidade de processamento e uma memória limitada, uma unidade de comunicação (normalmente um transceptor de rádio), sensores e atuadores e uma fonte de energia como mostra a Figura 1.

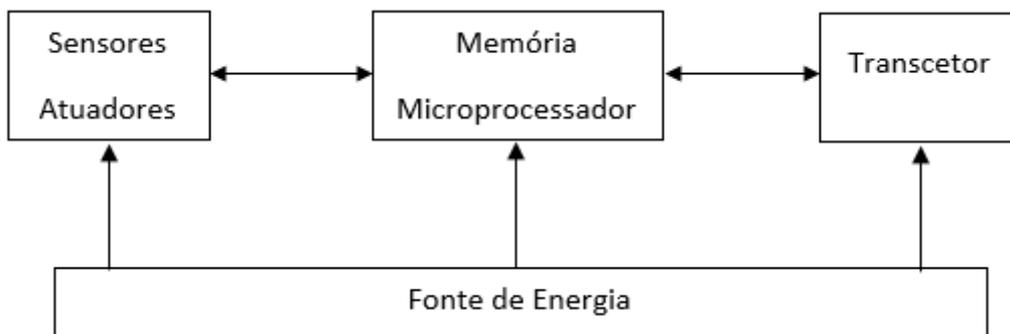


Figura 1. Arquitetura básica de um nó de uma rede de sensores sem fios.

2.1.2 Aplicações

As redes de sensores sem fios têm o potencial de serem aplicadas nas mais diversas áreas, tais como:

Indústria – Existem já muitos exemplos práticos de aplicações deste tipo: monitorização em indústrias petroquímicas, siderúrgicas, refinarias e fábricas. Um desses casos é o da produção industrial, que tem como funções a monitorização ou controlo de processos industriais como uma linha de montagem, monitorização de zonas de difícil acesso ou perigosas, controlo de temperaturas, pressões, fluxos e níveis, entre muitos outros. Podemos, ainda, salientar a indústria da aviação, onde cada vez mais se utiliza transdutores, e onde se opta por utilizar sensores sem fios por forma a reduzir a enorme quantidade de cabos que seriam necessários ao funcionamento destes sensores e atuadores.

Edifícios inteligentes – Utilizando redes de sensores sem fios conseguimos monitorizar e controlar facilmente parâmetros como a temperatura, humidade e qualidade do ar de uma determinada habitação, aumentando, assim, o nível de conforto dos seus ocupantes e podendo reduzir significativamente o consumo de energia, evitando gastos desnecessários.

Agricultura e pecuária – A monitorização de aspetos como a temperatura e humidade podem ajudar numa boa prática de produção agrícola, alertando para as necessidades de rega ou como forma de as evitar. Estes métodos são amplamente

usados em ambientes de estufa, pois permitem o controlo automático do ambiente desejado. Estas redes de sensores podem também ser utilizadas para monitorizar animais para obter, por exemplo, os dados fisiológicos do seu corpo.

Medicina/Biologia – A utilização de sensores no corpo de pacientes é capaz de fornecer dados como a temperatura do organismo, a frequência cardíaca e a pressão arterial, e enviar alertas para um profissional de saúde na existência de anomalias. Utilização de sensores capazes de detetar a presença de substâncias que indiquem o surgimento de um problema biológico tanto no corpo humano como em animais.

2.1.3 Protocolos MAC

Nas redes de sensores sem fios existem dois tipos básicos de protocolos MAC (*Medium Access Control*): os que são baseados em escalonamento, que correspondem aos protocolos centralizados (de reserva fixa ou dinâmica) e os que são baseados em contenção. No primeiro, o controlador central coordena o acesso ao meio através da reserva de recursos como frequências de transmissão e *slots* temporais; no segundo, os nós controlam o acesso ao meio de forma distribuída [11]. Nesta secção será apresentado um exemplo de protocolo de cada tipo descrito anteriormente.

2.1.3.1 Protocolo S-MAC

O protocolo S-MAC (*Sensor Medium Access Control*) [12] é baseado em contenção. Este protocolo parte do princípio que todos os dispositivos envolvidos na comunicação têm limitações energéticas e possuem uma baixa taxa de transmissão de dados. Como mostra a Figura 2, o método utilizado por este protocolo para minimizar o *idle listening* consiste em sincronizar as transmissões e receções de dados entre os vizinhos de modo a que ocorram de forma periódica durante um curto espaço de tempo.

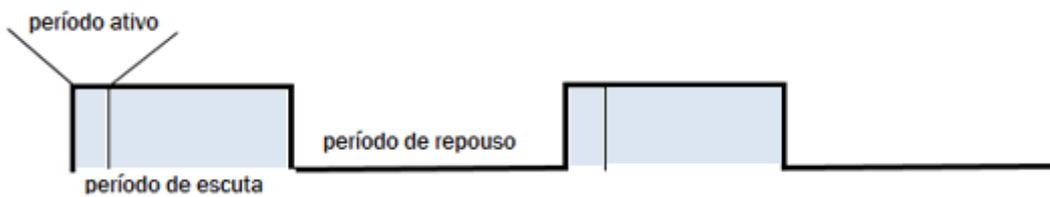


Figura 2. Princípio de funcionamento do protocolo S-MAC.

Os nós procedem ao envio de um pacote de sincronização (SYNC) periodicamente, para que os nós vizinhos tomem conhecimento do escalonamento adotado. É durante o período ativo que os pacotes de dados, precedidos de pacotes SYNC, RTS e CTS, são enviados. A duração do período de escuta e repouso é selecionada de acordo com a aplicação.

As áreas de abrangência das redes de sensores sem fios podem ser extensas, o que dificultaria a existência de um único escalonamento de período ativo para toda a rede. Assim, podem existir vários escalonamentos disjuntos seguidos por diferentes nós da rede. Desta forma, quando inicia a operação da rede, o processo de escalonamento começa pela escuta do meio por um nó, que, se escutar um pacote de sincronização no tempo previsto, adota esse escalonamento. Estes nós são denominados de seguidores. Caso contrário, o nó define o seu próprio escalonamento e passa a difundi-lo para outros nós durante o período ativo. Estes nós são denominados de sincronizadores.

2.1.3.2 Protocolo LEACH

O protocolo LEACH (*Low Energy Adaptive Clustering Hierarchy*) encaixa no conjunto de protocolos baseados em escalonamento. O princípio de funcionamento deste protocolo baseia-se na construção de vários *clusters* pelos nós da rede, em que cada *cluster* possui um nó denominado de *cluster-head* (CH), que tem o papel de controlador central nesse *cluster*. O CH cria um modo de escalonamento TDMA (*Time Division Multiple Access*), de forma a ser capaz de receber os dados dos nós associados, comprimi-los e enviar o pacote resultante para a estação base. Este processo, associado ao tempo que o CH tem de permanecer ativo, é bastante

dispendioso comparado com o consumo requerido aos outros nós do *cluster*, e rapidamente o deixaria sem energia. Para resolver esse problema, o protocolo prevê uma distribuição do papel de CH pelos vários nós da rede ao longo do tempo.

O protocolo LEACH é organizado por rondas, e cada ronda é subdividida em duas fases. A primeira é a fase de configuração, onde os *clusters* são formados, e esta subdivide-se em várias etapas. Primeiro todos os nós da rede executam um algoritmo para determinar se são ou não CH naquela ronda. De seguida os CHs eleitos difundem essa informação pelos seus vizinhos, e estes, através da intensidade do sinal recebido dos CHs mais próximos, decidem a que *cluster* se pretendem associar. Na etapa seguinte cada nó se associa ao CH escolhido. Depois de todos se terem associado, este define um escalonamento TDMA fixo e informa todos os nós pertencentes a esse *cluster*. A segunda é a fase estacionária, onde ocorrem as transferências de dados desde os nós até à estação base. Nesta fase os nós enviam os seus dados periodicamente nos respetivos *slots* alocados pelo CH, e só durante esse tempo é que os nós necessitam de ter o seu transceptor de rádio ligado, poupando assim energia. O mesmo não acontece com o CH, que necessita do seu transceptor ligado para receber os dados de todos os nós e, depois de tratados, enviá-los para a estação base. Uma vez que o meio de comunicação utilizado pelos diferentes *clusters* é o mesmo, este protocolo utiliza um código CDMA (*Code Division Multiple Access*) diferente para cada um, de modo a minimizar as interferências. Na Figura 3 podemos ver o exemplo da configuração de uma rede a funcionar segundo o protocolo LEACH. A figura representa apenas uma ronda, sendo que a cada ronda os *clusters* mudam de acordo com os CHs eleitos.

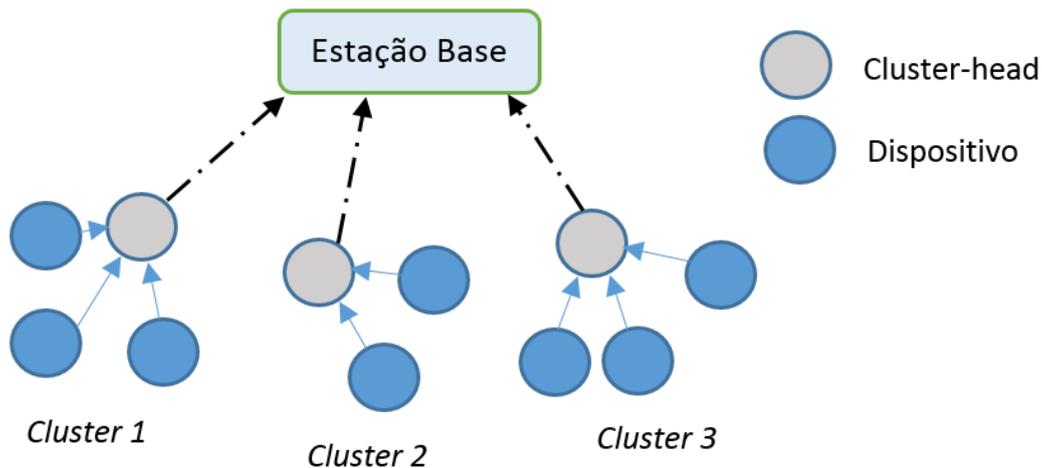


Figura 3. Exemplo de configuração de uma ronda do protocolo LEACH.

2.2 IEEE 802.15.4

2.2.1 Introdução

As redes de área pessoal de baixo débito (LR-WPAN – *Low-Rate Wireless Personal Area Network*) [13] são redes simples que permitem a conectividade entre dispositivos que não requerem elevadas taxas de transmissão e normalmente têm limitações no consumo energético. Este tipo de rede tem como vantagens a facilidade de implementação da infraestrutura da rede, a fiabilidade na entrega de dados ao destinatário, baixo custo, baixa complexidade e elevada autonomia dos dispositivos.

A norma IEEE 802.15.4 define dois tipos de dispositivos: FFD (*Full Function Device*) e RFD (*Reduced Function Device*). Este último é um dispositivo com funcionalidades reduzidas, que assume apenas o papel de dispositivo. Por outro lado, um dispositivo FFD pode assumir o papel de coordenador da PAN (*Personal Area Network*), de coordenador ou de dispositivo. Um dispositivo FFD pode comunicar com outros dispositivos FFDs ou com RFDs, enquanto um RFD só pode comunicar com dispositivos FFDs.

São duas as topologias de rede suportadas pela norma IEEE 802.15.4: topologia em estrela (*star*) e topologia *peer-to-peer*, representadas na Figura 4.

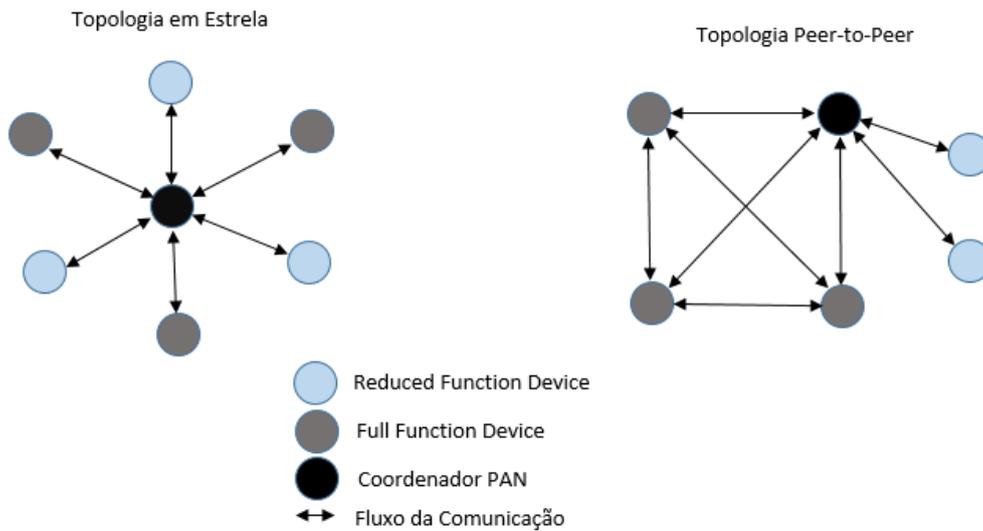


Figura 4. Exemplo de topologias em estrela e peer-to-peer do IEEE 802.15.4.

Na topologia em estrela existe um único controlador central, designado de coordenador da PAN. Todos os restantes dispositivos que compõem a rede comunicam única e exclusivamente com o coordenador da PAN, quer sejam RFD ou FFD. O coordenador da PAN pode comunicar com todos os dispositivos da sua rede. Qualquer dispositivo FFD pode criar a sua própria rede, desde que escolha um identificador da PAN que não esteja a ser usado por nenhuma rede próxima de si. Apenas os dispositivos FFD podem dar início à formação de uma rede IEEE 802.15.4, tornando-se assim coordenadores da PAN. Durante a fase de planeamento da rede, deve-se ter em conta o consumo dos dispositivos. O coordenador da PAN, devido ao seu elevado número de tarefas, consome mais energia, por isso deve ser alimentado diretamente pela rede elétrica, enquanto os restantes dispositivos, devido à sua simplicidade de tarefas, podem ser alimentados por pequenas baterias.

Na topologia *peer-to-peer* também existe um coordenador da PAN, mas esta difere da topologia em estrela pelo facto de os dispositivos FFD poderem comunicar todos uns com os outros, desde que estejam ao seu alcance, ou, indiretamente, utilizando outros dispositivos para esse fim. Este tipo de topologia permite a formação de redes mais complexas, de uma escala mais alargada, que podem conter centenas de dispositivos, como por exemplo as redes em malha (*mesh*) [14]. Uma das grandes vantagens da topologia *peer-to-peer* é o facto de permitir a implementação de rotas

alternativas. Estas rotas oferecem à rede, geralmente, uma maior robustez, quando comparadas com a topologia em estrela. Este tipo de topologias é aplicável em áreas como a monitorização e controlo industrial, agricultura inteligente e segurança.

A arquitetura das redes IEEE 802.15.4 está dividida em camadas, sendo que cada camada é responsável por uma parte das funcionalidades especificadas na norma, e as camadas inferiores oferecem serviços às camadas superiores. As interfaces entre as camadas servem para definir as interligações lógicas conforme descrito nesta norma. Um dispositivo LR-WPAN especifica pelo menos uma camada física (PHY) e uma camada de controlo de acesso ao meio (MAC), como mostra a Figura 5.

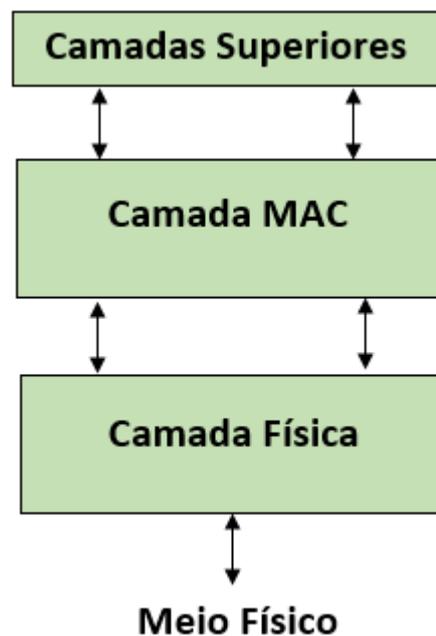


Figura 5. Arquitetura LR-WPAN do IEEE 802.15.4.

2.2.2 Camada física

A camada física proporciona a interface entre a camada MAC e os canais de rádio da camada física. É responsável pela transmissão e receção de dados utilizando um dos canais de rádio disponíveis. A camada física é responsável ainda pelas seguintes tarefas:

- Ativação e desativação do transceptor (*transceiver*). O transceptor de rádio pode operar em três estados: transmissão, receção e modo de poupança de energia

(*sleep*), de acordo com o pedido da camada MAC. O tempo necessário para o rádio comutar do estado de transmissão para o estado de recepção, ou vice-versa, não deve ser superior a doze períodos de símbolo (*symbol periods*).

- Detecção de energia (ED – *Energy Detection*). É uma estimativa da potência do sinal recebido dentro de um canal da norma IEEE 802.15.4. Com este procedimento não existe nenhuma tentativa de identificar o sinal ou decodificar o canal. A detecção da energia deve ser feita com uma duração igual a oito períodos de símbolo. O resultado da detecção de energia é normalmente utilizado pela camada superior como parte de um algoritmo de seleção de canal ou com a finalidade de avaliação de canal livre (CCA – *Clear Channel Assessment*), por si só ou combinado com a detecção da portadora.
- Indicação de qualidade de ligação (LQI - *Link Quality Indication*). A medição do LQI é realizada para cada pacote recebido. A camada física usa o resultado da detecção de energia, a relação sinal-ruído ou uma combinação entre estes dois valores para estimar a qualidade da ligação. O resultado do LQI é utilizado pelas camadas superiores.
- Avaliação de canal livre (CCA - *Clear Channel Assessment*). É utilizado para fornecer a informação sobre o estado do meio, se está livre ou ocupado. Um dos modos de operação é a detecção de energia, onde o meio é considerado ocupado se houver uma detecção de energia superior a um determinado limiar (ED *threshold*). Um outro modo de operação é através da detecção da portadora (*carrier sense*) onde o meio é dado como ocupado se o CCA detetar um sinal com as características de modulação e de espalhamento espectral do IEEE 802.15.4. A combinação destes dois modos de operação forma um outro modo de operação, em que o CCA indica que o meio está ocupado se detetar um sinal com as características de modulação e espalhamento espectral do IEEE 802.15.4 e com energia superior ao ED *threshold*.
- Seleção do canal de frequência. A norma IEEE802.15.4 permite a operação em 27 canais diferentes. A camada física deve ser capaz de sintonizar o seu transceptor no canal solicitado pela camada superior.

- Transmissão e recepção de dados. A norma IEEE 802.15.4 especifica quatro bandas de operação [13]:
 - Na banda 868/915 MHz, utilizando a técnica de espalhamento espectral por sequência direta (DSSS – *Direct Sequence Spread Spectrum*) e modulação BPSK (*Binary Phase-Shift Keying*).
 - Na mesma banda de frequências e utilizando a mesma técnica de espalhamento espectral DSSS, passando a usar modulação O-QPSK (*Orthogonal Quadrature Phase-Shift Keying*).
 - Ainda para a banda de 868/915 MHz, mas com a técnica de espalhamento espectral PSSS (*Parallel Sequence Spread Spectrum*) modulação BPSK e ASK (*Amplitude-Shift Keying*).
 - Na banda de 2450 MHz, utilizando a técnica de espalhamento espectral DSSS empregando modulação O-QPSK.

Podemos ver estas bandas mais detalhadamente na Tabela 1.

Tabela 1. Características da camada física do IEEE 802.15.4.

Bandas de Operação	Frequência (MHz)	Modulação	Taxa de Transmissão (kbit/s)	Taxa de Símbolos (Ksímbolos/s)	Símbolos
868/915	868-868.6	BPSK	20	20	Binário
	902-928	BPSK	40	40	Binário
868/915 (opcional)	868-868.6	ASK	250	12.5	20-bit PSSS
	902-928	ASK	250	50	5-bit PSSS
868/915 (opcional)	868-868.6	O-QPSK	100	25	16-ary <i>Orthogonal</i>
	902-928	O-QPSK	250	62.5	16-ary <i>Orthogonal</i>
2450	2400-2483.5	O-QPSK	250	62.5	16-ary <i>Orthogonal</i>

2.2.3 Camada MAC

A camada MAC gere o acesso ao canal de rádio e é responsável pelas seguintes tarefas:

- Gerar os *beacons*, caso o dispositivo seja o coordenador da PAN.
- Sincronização com os *beacons* da rede.
- Suporte à associação e à saída de dispositivos na PAN.
- Suporte à segurança no dispositivo.
- Implementar o mecanismo CSMA-CA (*Carrier Sense Multiple Access - Collision Avoidance*) para acesso ao canal de comunicação.
- Gestão do mecanismo GTS (*Guaranteed Time Slot*).
- Garantir uma ligação fiável entre as camadas MAC de dois dispositivos.

As redes baseadas no IEEE 802.15.4 podem operar de dois modos diferentes, com ou sem o uso de *beacons* de comunicação:

- *Beacon Enabled*: O coordenador gera e envia *beacons* periodicamente, de modo a sincronizar os restantes dispositivos ligados à PAN e fornecer indicações acerca

do funcionamento da rede. Os dispositivos sincronizam-se com a recepção do *beacon* para saberem quando podem transmitir. O tempo é estruturado em supertramas e é utilizado o mecanismo *slotted* CSMA-CA para troca de informação entre dispositivos (em alternativa, pode-se utilizar o mecanismo GTS). Neste modo podem existir períodos inativos para poupança de energia.

- *Non Beacon Enabled*: Neste modo não existe qualquer tipo de sincronização entre os dispositivos e o coordenador, assim como não existe estrutura de supertrama. Os dispositivos utilizam simplesmente o mecanismo *unslotted* CSMA-CA para enviarem pacotes.

2.3 ZigBee

2.3.1 Introdução

A tecnologia ZigBee foi criada pela ZigBee Alliance, e descreve um protocolo de comunicações sem fios de baixo consumo. As redes ZigBee são vocacionadas para aplicações que não necessitem de elevadas taxas de transmissão de dados, de baixo custo, que possuam nós de pequeno tamanho, e que proporcionem consumos de energia reduzidos. De modo a satisfazer estes requisitos os dispositivos da rede devem utilizar componentes de baixo consumo e de baixa complexidade, que se traduzem em limitadas capacidades de processamento, mas que, por outro lado, proporcionam a possibilidade de serem alimentados por pequenas baterias, tornando estes equipamentos autónomos por um período de tempo bastante alargado.

O ZigBee é uma tecnologia que utiliza potências de transmissão e taxas de transmissão de dados bastante inferiores às das redes de área local sem fios. Devido a estas potências de transmissão reduzidas, a área de alcance da rede é reduzida; no entanto, esta situação é minimizada pela capacidade de suporte, pelo ZigBee, de comunicações *multihop*.

A generalidade das aplicações utilizadas neste tipo de redes não necessita de elevadas taxas de transmissão de dados. Ao contrário da comunicação de dados entre

utilizadores, predominam aplicações de monitorização e controlo, onde grande parte dos dados transmitidos pela rede são recolhidos, através de sensores, de algum tipo de ambiente específico (e.g., meio ambiente [15], indústria [16], etc.).

2.3.2 Protocolo ZigBee

O protocolo ZigBee está especificado sob a forma de camadas, em que as inferiores fornecem serviços às superiores [17]. A Figura 6 apresenta o modelo da pilha protocolar ZigBee. A norma define duas camadas, sendo elas NWK (*Network Layer*) e APL (*Application Layer*). As restantes camadas, MAC (*Medium Access Control*) e PHY (*Physical Layer*), são definidas pela norma IEEE 802.15.4.

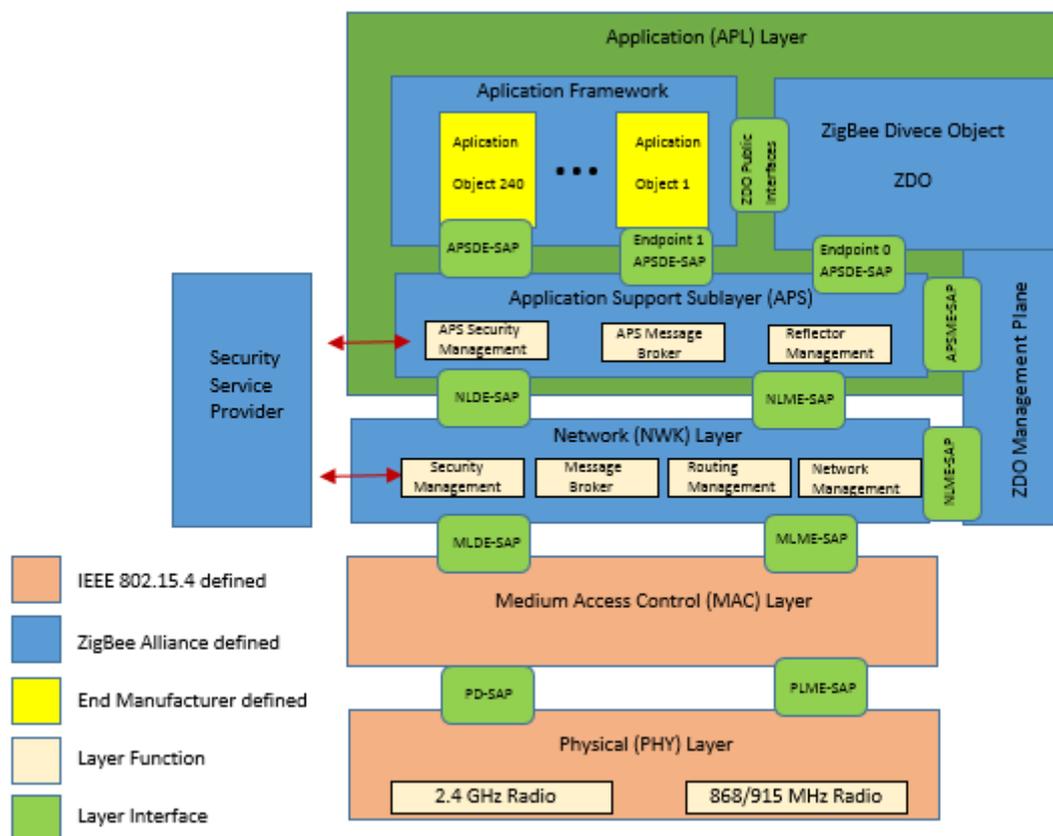


Figura 6. Camadas detalhadas da arquitetura do protocolo ZigBee [18].

2.3.2.1 Camada de rede

A camada de rede (NWK) do protocolo ZigBee tem várias funções: a descoberta de novos dispositivos que possam passar a integrar a rede; a monitorização de dispositivos que integram e abandonam a rede; armazenar informações relativas a todos os dispositivos; e atribuir endereços (caso seja o coordenador da rede).

A configuração de dispositivos é feita através desta camada, e é aqui que são implementados os mecanismos de descoberta de rotas e encaminhamento da informação (*routing*).

Esta camada é responsável por fornecer serviços que permitam o correto funcionamento da camada de controlo de acesso ao meio (MAC) do 802.15.4, e, da mesma forma, serviços adequados para a interface com a camada de aplicação. Para esta camada, existem duas entidades: o serviço de dados e o serviço de gestão.

O serviço de gestão contém uma entidade (NLME – *Network Layer Management Entity*) que através do seu ponto de acesso ao serviço (SAP – *Service Access Point*) fornece serviços responsáveis por criar recursos que permitam à aplicação interagir com a pilha. O NLME tem ainda outros serviços disponíveis como:

- Configuração de novos dispositivos. As opções de configuração incluem o início de operação como coordenador ZigBee ou então o processo de se integrar numa rede existente.
- Criação de uma rede.
- Junção ou abandono a uma rede.
- Endereçamento.
- Descoberta de vizinhança.
- Descoberta de rotas.

A NLDE (*Network Layer Data Entity*) é a entidade que fornece o serviço de transmissão de dados através do seu SAP, e está capacitada a fornecer serviços como:

- Criação de NPDUs (*Network Protocol Data Units*).
- Topologia específica de roteamento.

- Segurança.

2.3.2.2 Camada de aplicação

A camada de aplicação é a camada superior da pilha protocolar do ZigBee, e contém a subcamada *Application Support Sublayer (APS)*, o *ZigBee Device Object (ZDO)* e a *Application Framework (AF)*. Esta camada tem a função de garantir um suporte fiável e uma correta gestão para as diversas aplicações.

A APS fornece uma *interface* entre a camada de rede e as aplicações fornecidas, através de um conjunto de serviços usados pela ZDO. As entidades que fornecem os serviços a esta camada são a *Application Service Data Entity Access Point (APSDE-SAP)* e a *Application Service Management Entity Access Point (APSME-SAP)*.

A APSDE-SAP torna possível a transmissão de dados para o transporte de PDUs de aplicação entre dois ou mais dispositivos localizados na mesma rede, suporta a fragmentação e reconstrução de pacotes e garante um transporte de dados viável, enquanto a APSME-SAP fornece serviços de segurança, registo e remoção de endereços de grupo.

A *Application Framework (AF)* fornece funções que permitem enviar e receber dados. Nesta subcamada o utilizador pode definir até 240 objetos de aplicação, em que cada um é identificado pelo seu *endpoint*, tornando possível o desenvolvimento e identificação de aplicações no mesmo nó.

Os *ZigBee Device Objects (ZDOs)* têm o objetivo de satisfazer os requisitos de todas as aplicações que estejam a ser executadas na pilha protocolar do ZigBee. Os ZDOs são responsáveis por inicializar a APS e a camada de rede, bem como juntar informações de configuração das aplicações finais para determinar e implementar a descoberta e gestão da rede. Desta forma, o ZDO é usado pelas aplicações como interface para a camada de rede.

2.3.3 Tipos de dispositivos

Para criar uma rede fiável composta por vários dispositivos, estão definidos três tipos de dispositivos ZigBee: coordenador (ZC – *ZigBee Coordinator*), *router* (ZR – *ZigBee Router*) e *end device* (ZED – *ZigBee End Device*).

O ZC é um dispositivo FFD, e é o responsável por criar e gerir toda a rede ZigBee, sendo equivalente ao coordenador da PAN do 802.15.4. Este dispositivo pode comportar-se também como um ZR e associar-se a uma rede já existente. Estes comportamentos são definidos na configuração do dispositivo.

O ZR é um dispositivo FFD que permite o encaminhamento de mensagens entre dispositivos que estejam fora do alcance entre si, tem a capacidade de atribuir endereços a dispositivos que se liguem a ele e calcula a melhor rota entre dispositivos que pretendam comunicar entre si. Este dispositivo pode funcionar como coordenador, da mesma forma que o ZC pode funcionar como ZR.

O ZED é um dispositivo RFD, o que o torna um dispositivo terminal da rede, sem capacidade de retransmissão de informação de outros nós, não tendo, desta forma, qualquer influência na manutenção da rede. Este dispositivo tem capacidade de entrar em modo de poupança de energia e acordar apenas por curtos espaços de tempo para efetuar as suas tarefas. Esta capacidade permite-lhe uma enorme redução de consumo de energia, o que possibilita a utilização de pequenas baterias como a sua fonte de alimentação.

2.3.4 Formato das tramas

A norma IEEE 802.15.4 define quatro tipos de tramas [19]:

- Tramas de *beacon* – são utilizadas pelo coordenador e pelos *routers* para efetuar a transmissão de *beacons*.
- Tramas de dados – são as tramas que transportam os dados úteis, ou seja, os dados que o utilizador pretende transmitir.

- Tramas de reconhecimento (ACK – *Acknowledgement*) – são utilizadas para confirmar a receção de pacotes. Os ACKs são transmitidos no tempo livre entre o envio das tramas.
- Tramas de comando MAC – são usadas para tratar do controlo de transferências das entidades MAC.

2.4 LonWorks

A tecnologia LonWorks (*Local Operating Network*) foi criada pela Echelon Corporation no início da década de noventa, com o objetivo de implementar redes de controlo distribuído e automatizado [20]. A mesma corporação, no final da década de noventa, realizou a abertura da especificação do seu protocolo, permitindo que diversos fabricantes desenvolvessem produtos baseados na tecnologia LonWorks. Esse protocolo, que recebeu o nome de LonTalk [21], é totalmente aberto e passível de se implementar em qualquer microprocessador ou microcontrolador.

A plataforma LonWorks funciona sobre uma grande variedade de meios físicos de comunicação, como rede elétrica, par trançado, radiofrequência, infravermelhos cabo coaxial e fibra ótica. Devido à sua arquitetura descentralizada, é passível de ser implementada em diversas áreas como a automação comercial e industrial, sistemas de transportes, automatização de edifícios e controlo de processos.

O protocolo LonTalk possui uma notação e terminologia bastante próprias, como podemos ver na Figura 7. O *Store & Forward Repeater* pode interligar dois canais, a *Bridge* conecta os dois canais (x e y) e encaminha os pacotes de x para y e vice-versa, enquanto o *Router* conecta sempre duas *subnets*. Este protocolo limita a capacidade de cada *subnet* um máximo de 127 nós. O *Gateway* pode conectar dois domínios LonTalk, e cada domínio pode ter até 255 *subnets*.

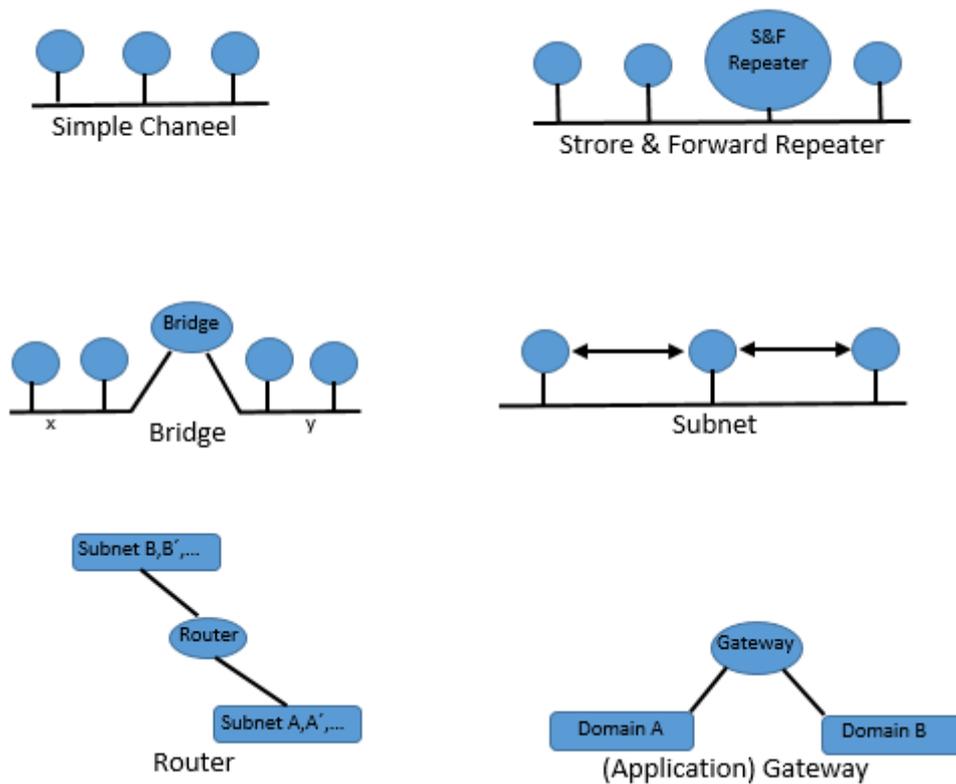


Figura 7. Terminologia básica do protocolo LonTalk [21].

O protocolo LonTalk foi desenvolvido de acordo com o modelo OSI (*Open Systems Interconnection*), como mostra a Figura 8. Os serviços oferecidos pela pilha protocolar são os seguintes:

- Camada física – o bit enviado é reconhecido como o mesmo bit em todos os dispositivos recetores. O protocolo é independente do meio de comunicação.
- Camada de ligação – esta camada, para além da camada de ligação de dados, que define os modos de acesso ao meio de transmissão e da codificação de dados, é também constituída por uma subcamada MAC, que implementa um algoritmo chamado Predictive p-persistent CSMA (*Carrier Sense Multiple Access*), para evitar colisões.
- Camada de rede – Trata do endereçamento, da forma de nomeação dos dispositivos, e define as rotas quando a origem e o destino se encontram em canais de comunicação diferentes. Este serviço não suporta a segmentação de mensagens.

- Camada de transporte – esta camada é responsável pela entrega dos pacotes que contêm informação, e tem a capacidade de autenticar mensagens como recurso opcional. Esta camada 4 é também composta pela subcamada de controlo de transação, responsável pela deteção de mensagens duplicadas.
- Camada de sessão – implementa o mecanismo de pergunta-resposta, oferecendo suporte a ações remotas que permitem que um cliente faça uma requisição a um servidor.
- Camadas de apresentação e aplicação – o protocolo define estas camadas como sendo a camada 6 e 7, mas muito próximas uma da outra. Estas camadas juntas fornecem a interoperabilidade entre os nós LonTalk. A camada de aplicação fornece serviços de envio e receção de mensagens, e tem o conhecimento das variáveis da rede. A camada de apresentação fornece ainda informações do cabeçalho APDU (*Application Protocol Data Unit*), de forma que este seja interpretado pelas variáveis da rede.



Figura 8. Camadas do protocolo LonTalk.

Este protocolo suporta diversos tipos de topologias, de forma a satisfazer os mais diversos tipos de aplicações das diferentes áreas. Na construção de uma rede podem ser usadas topologias como *star*, *bus*, ou *ring*, ou então uma topologia livre que pode ser uma combinação destas três últimas. Esta topologia livre pode ser criada sem restrições no que à sua construção diz respeito, pois tanto pode ser uma mistura de topologias *star* e *bus* como outra qualquer, sendo apenas sujeita a um limite físico do comprimento do cabo da rede.

3. DESCRIÇÃO DO SISTEMA ATUAL

3.1 Introdução

A monitorização e controlo de sistemas de refrigeração tem como base uma rede interna capaz de proporcionar a comunicação entre os diferentes dispositivos dessa rede. Este tipo de sistemas de refrigeração é implementado em grandes áreas como supermercados e hipermercados, entrepostos comerciais e fábricas onde se exige um elevado número de equipamentos ou locais refrigerados.

A rede existente tem como objetivo controlar e monitorizar todos os equipamentos de refrigeração existentes na superfície em causa.

Existem diferentes tipos de equipamentos de refrigeração cujas temperaturas são determinadas de acordo com a função que lhe é destinada. Apesar destes equipamentos variarem no tipo de função, têm os mesmos elementos constituintes, e, à exceção do ventilador, todos eles são da marca Danfoss.

3.2 Equipamento de refrigeração

Como referido anteriormente, existem vários tipos de equipamentos de refrigeração, desde câmaras frigoríficas, armários expositores, vitrinas, etc. A Figura 9 ilustra um equipamento de refrigeração genérico, de forma a exemplificar os seus elementos constituintes, que podem variar (essencialmente sensores e atuadores) dependendo da função do equipamento de refrigeração. Este equipamento de refrigeração é constituído por uma válvula de expansão, um ventilador, duas sondas de temperatura, um micro de porta, um evaporador e um quadro elétrico. A válvula de expansão, o ventilador e as sondas de temperatura são atuadores e sensores que são controlados pelo quadro elétrico, enquanto o evaporador é um elemento estático onde simplesmente se dá o processo de refrigeração.

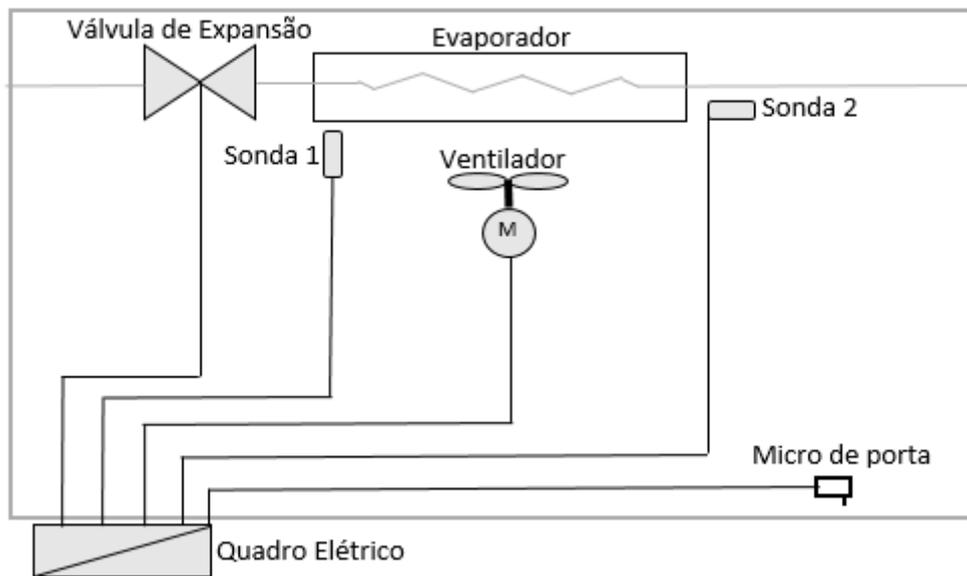


Figura 9. Ilustração de um equipamento de refrigeração genérico.

3.3 Quadro elétrico

O quadro elétrico tem como função servir de infraestrutura de apoio ao funcionamento dos sensores e atuadores do sistema de refrigeração. Para esse efeito, é constituído por um controlador e alguns relés eletromecânicos e térmicos. Como podemos ver na Figura 10, os controladores utilizados, o AK-CC 550 [22] ou o AK-PC 530 [23], são responsáveis por monitorizar e controlar todo o equipamento de refrigeração e ainda por enviar informações, como valores de leitura dos sensores e estado dos atuadores, para um controlador de gestão central. Estes controladores estão preparados para se ligarem diretamente às sondas de temperatura e ao micro de porta, sem a necessidade de hardware adicional. Os relés eletromecânicos são utilizados no circuito de potência, onde as correntes são mais elevadas, e o controlador não é capaz de fornecer tais potências, como é o caso do ventilador ou da válvula de expansão. Já os relés térmicos são usados para proteger estes circuitos de potência.



Figura 10. Controladores AK-PC 530 (esquerda) e AK-CC 550 (direita).

A principal diferença entre estes dois controladores está no facto de o AK-PC 530 ser dotado de um maior número de ligações para sensores e atuadores. Outra diferença é que a sua alimentação é de 24 V AC enquanto a alimentação do AK-CC 550 é de 230 V AC. Dependendo das características do equipamento de refrigeração pode-se seleccionar o mais adequado, sendo que o AK-PC530 é mais utilizado na casa das máquinas.

3.4 Sondas

Em cada equipamento de refrigeração são colocadas duas sondas de temperatura (sonda1 e sonda2) em posições específicas, a fim de obter um melhor controlo do equipamento. Na Figura 11 podemos ver as sondas utilizadas, do tipo EKS 111 [24]. São sondas PTC (*Positive Temperature Coefficient*), o que significa que a sua resistência elétrica aumenta com o aumento da temperatura.



Figura 11. Sonda EKS 111.

Na Figura 12 podemos observar que a variação da sua resistência elétrica com a temperatura não é linear. Verifica-se uma resistência de aproximadamente 1000 Ω à temperatura de 25 °C. Estas sondas estão ligadas diretamente ao controlador, onde o seu sinal analógico irá ser transformado em digital, mais precisamente num valor de temperatura.

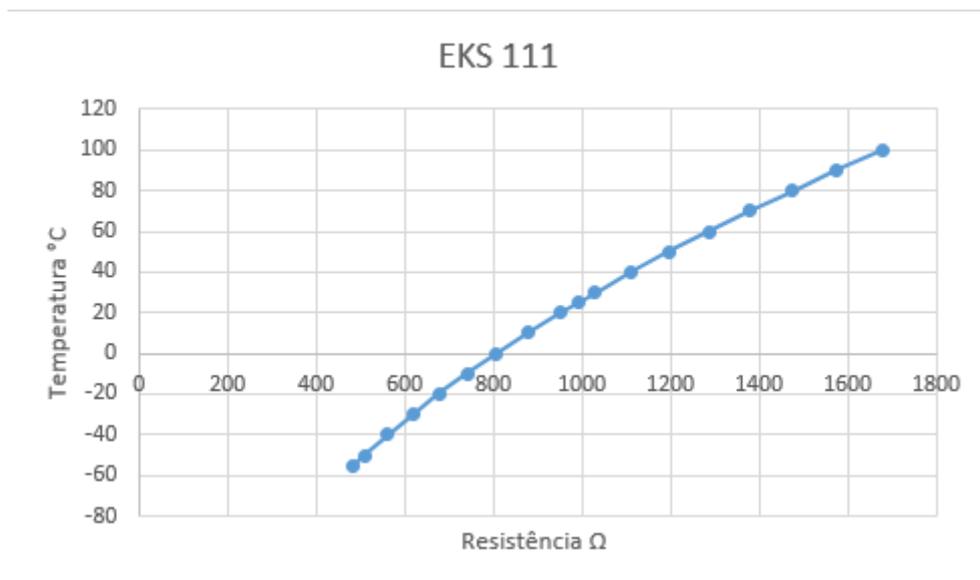


Figura 12. Curva característica da sonda EKS 111.

3.5 Micro de porta

O micro de porta é um sensor fim de curso, instalado junto à porta do equipamento de refrigeração. A sua função é fornecer informação ao controlador sobre o estado da porta, que pode estar aberta ou fechada.

3.6 Válvula de expansão

A válvula de expansão controla o fluxo de líquido refrigerante, que permite que o evaporador (serpentina em tubo de cobre) refrigere através de efeitos termodinâmicos. A válvula é um atuador que, ligada ao quadro elétrico, é comandada pelo controlador, este faz ligar ou desligar a válvula de acordo com os valores de leitura das sondas em relação aos valores desejados pelo equipamento de

refrigeração. As válvulas utilizadas, do tipo AKV 10-1 [25] são alimentadas com uma tensão de 230 V AC e são compatíveis com os controladores descritos anteriormente.



Figura 13. Válvula de expansão.

3.7 Ventilador

Os ventiladores são atuadores e controlam o fluxo de ar que circula através do evaporador. Estão ligados ao circuito de potência do quadro elétrico mas são comandados pelo controlador. São alimentados a 230 V AC e funcionam no estado ligado ou desligado.



Figura 14. Exemplo de ventilador.

3.8 Controlador de gestão central

Para monitorizar e controlar todos os equipamentos de refrigeração, é usado um controlador de gestão central. Este recebe a informação proveniente de todos os controladores dos equipamentos de refrigeração, através de cabos de rede, e proporciona a interação com o utilizador. O controlador de gestão central utilizado, denominado AK-SM 850 [26], é constituído por um monitor e botões de acesso para que se possa monitorizar e configurar sem utilizar qualquer tipo de equipamento externo. No monitor podem-se ver o estado dos atuadores e os valores recolhidos pelos sensores de todos os equipamentos que fazem parte da rede. Este equipamento encontra-se normalmente no quadro elétrico geral da casa das máquinas.

3.9 Repetidor

O repetidor é utilizado quando a extensão da rede é superior ao comprimento máximo definido pela topologia. Neste caso, se o comprimento do cabo exceder os 1200 metros, é necessário a introdução de um repetidor. O repetidor utilizado é o AKA 223.

3.10 Topologia da rede

A comunicação entre dispositivos baseia-se na tecnologia LonWorks, utilizando a norma RS-485, adotada como especificação da camada física do protocolo LonTalk. Nesta topologia todos os equipamentos estão ligados em série, o cabo utilizado é o LAPP UNITRONIC Li2YCY (TP), constituído por dois pares de cabos com diâmetro de 0.65 mm². Este tipo de topologia permite uma taxa de transmissão máxima de 39.06 kbps, um número máximo de 60 controladores e um comprimento máximo de cabo de 1200 m. Caso esse comprimento seja ultrapassado é necessário inserir um repetidor. A Figura 15 apresenta a topologia da rede, formada pelos dispositivos AK-SM 850, AKA 223 e cinco dispositivos AK-CC550, interligados pelo cabo LAPP UNITRONIC Li2YCY (TP).

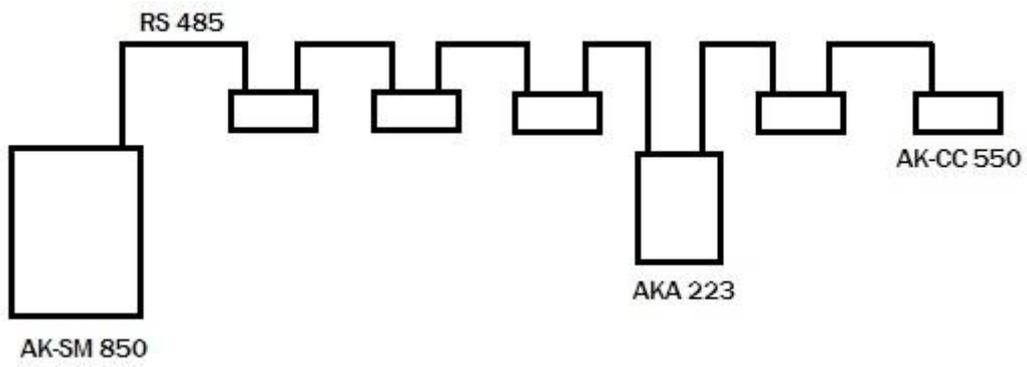


Figura 15. Topologia da rede.

4. DESENVOLVIMENTO DO SISTEMA

4.1 Introdução

Este projeto tem como proposta a implementação de um sistema de monitorização e controlo de uma rede sem fios para operar num ambiente industrial/comercial. Desta forma será utilizada uma rede sem fios ZigBee com o objetivo de monitorizar e controlar módulos de refrigeração espalhados pelo ambiente.

Este capítulo descreve a plataforma utilizada para o desenvolvimento deste sistema, os objetivos funcionais do sistema e as características de cada módulo de comunicação instalado em cada equipamento de refrigeração.

4.2 Plataforma de desenvolvimento

A plataforma utilizada baseia-se no kit de desenvolvimento CC2530 (CC2530DK) [27] fornecido pela Texas Instruments. Este kit contempla um SoC (*System on Chip*) que integra um microcontrolador e um transceptor de rádio no mesmo chip, e inclui hardware e software que tornam capaz a implementação de protótipos e elaboração de testes relacionados com projetos de radiofrequência. Para a realização deste projeto foi necessário a utilização de um mínimo de dois kits. A Figura 16 mostra os componentes de cada kit de desenvolvimento: dois módulos CC2530EM, um *dongle* CC2531, duas placas SmartRF05EB [28] e duas antenas.

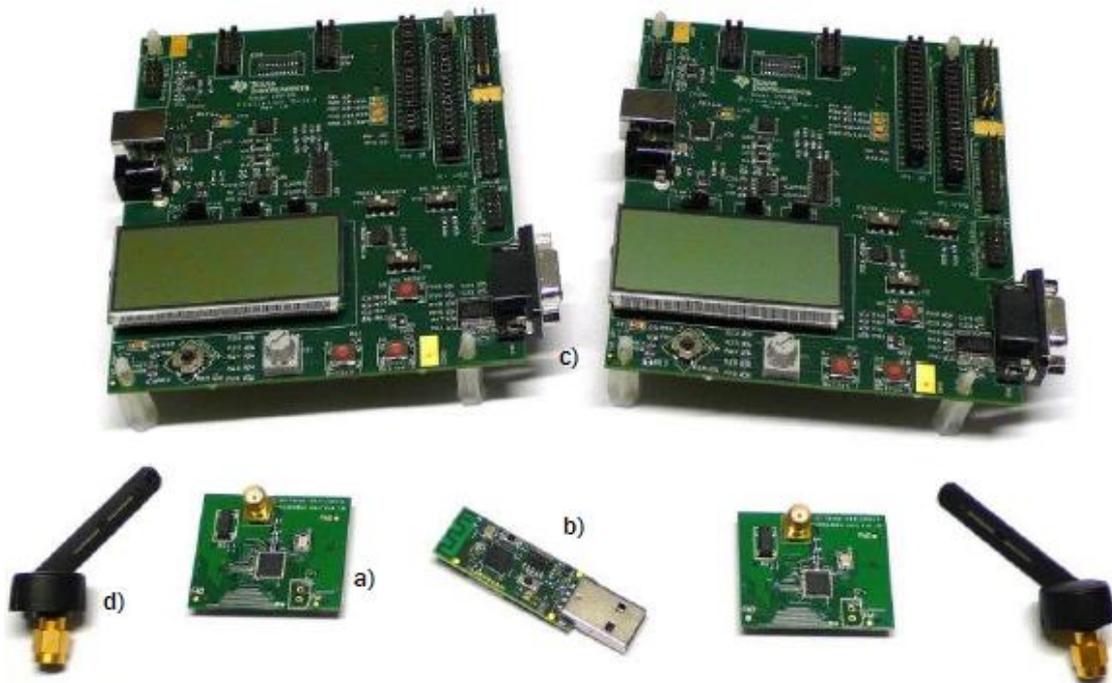


Figura 16. Módulo CC2530EM a), módulo CC2531 b), SmartRF05EB c), antena d).

O CC2530 opera na banda dos 2.4 GHz e tem um débito máximo de 250 kbps, conforme definido pela norma IEEE 802.15.4. Funcionalidades como criptografia de dados, avaliação do canal, indicação da qualidade da ligação, autenticação de dados, geração automática de preâmbulo da camada física, e geração e verificação do CRC, estão implementadas neste kit. O CC2530 inclui: uma unidade de processamento central (CPU) de elevado desempenho, baseada no microcontrolador 8051; e cinco modos de operação diferentes (ativo, inativo e três modos de poupança de energia), que lhe permitem fazer uma gestão eficaz do consumo de energia.

O serviço de controlo às interrupções fornece dezasseis fontes de interrupção, que podem acontecer mesmo quando o dispositivo está no modo inativo, e algumas delas quando o serviço está em modo de baixo consumo, permitindo assim colocá-lo novamente no modo ativo.

O CC2530 fornece, ainda, um conjunto de periféricos:

- Interface série (USART 0 e USART 1) – Usada para depuração do circuito e acesso à memória. Configuráveis individualmente no modo SPI (*Serial Peripheral Bus*) ou no modo UART.

- Controlador I/O – Responsável por controlar todos os pinos de uso geral. A CPU pode controlar os pinos como entradas e saídas através de software, e as interrupções da CPU podem ser habilitadas nos pinos individualmente.
- Temporizadores – Possui três temporizadores de uso geral, um de 16 bits (*timer 1*) e dois de 8 bits (*timer 3* e *timer 4*). Possui também um temporizador dedicado para ser utilizado pela camada MAC do IEEE 802.15.4 (*timer 2*). Contém, ainda, um temporizador de baixo consumo de energia com um oscilador de 32 kHz (*sleep timer*) que é utilizado, essencialmente, para o despertar de um modo de poupança de energia.
- ADC (*Analog to Digital Converter*) – Com resoluções de 7 a 12 bits e 8 canais possíveis, está preparado para receber sinais analógicos, e especificamente, num dos canais de entrada, um sinal de um sensor de temperatura.
- Coprocessador AES (*Advanced Encryption Standard*) – Este coprocessador criptográfico permite cifrar e decifrar dados usando o algoritmo AES com chave de 28 bits. Deste modo, suporta as operações de segurança requeridas pela camada MAC do IEEE 802.15.4 e pelas camadas de rede e de aplicação do ZigBee.
- Amplificador operacional – Ideal para operar em combinação com o ADC, em aplicações de sensores, de modo a amplificar os sinais provenientes desses sensores. As suas entradas podem ser acedidas através dos pinos de I/O.

O CC2530 fornece um transceptor de rádio compatível com a norma IEEE 802.15.4. Este contém um núcleo de processamento que o torna capaz de efetuar os seus procedimentos em paralelo com o microcontrolador. Desta forma, é capaz de emitir comandos e automatizar sequências de eventos de rádio ao mesmo tempo que o microcontrolador desempenha outra tarefa. O transceptor de rádio contém um módulo de reconhecimento de endereços e filtragem de pacotes, o que torna possível rejeitar pacotes que não estejam destinados àquele dispositivo, acrescentando eficiência ao sistema.

O *dongle* CC2531 é um módulo que funciona como *packet sniffer*, sendo de fácil conectividade com um computador devido à sua ligação USB (Universal Serial Bus).

Utilizando o software Packet Sniffer, fornecido pela Texas Instruments, com este dispositivo é possível captar os pacotes que circulam na rede entre os vários dispositivos e, desta forma, obter uma visualização gráfica e temporal dos mesmos.

A placa SmartRF05EB (*Evaluation Board*) disponibiliza uma vasta gama de interfaces com o utilizador. Fornece também o acesso aos pinos do CC2530EM, tornando mais fácil a conexão com os seus periféricos. Esta placa tem disponíveis para uso: LEDs, botões, *joystick*, potenciómetro, UART, LCD, USB 2.0 e porta série RS-232. A placa SmartRF05EB pode ser conectada a um PC através da porta USB para programar o módulo CC2530EM.

Em termos de software, a plataforma de desenvolvimento utilizada foi a Z-Stack, na versão Z-Stack-CC2530-2.5.0, fornecida pela Texas Instruments. Esta versão suporta os perfis ZigBee e ZigBee Pro da norma ZigBee 2007. O software utilizado para o desenvolvimento de código foi o IAR Embedded Workbench for 8051, para fazer ou ajustar a programação (utilizando a linguagem de programação C++) de forma a criar o ZED, ZR e ZC, e o NetBeans IDE 8.0.2, para, por sua vez, criar um programa (linguagem de programação Java) que proporciona a interação entre o utilizador e o sistema.

4.3 Descrição geral

Neste projeto propõe-se substituir a rede cablada LonWorks por uma rede sem fios ZigBee, com requisitos de funcionamento idênticos. Os controladores são o suporte da topologia da rede LonWorks, pelo que, quanto ao procedimento, substituiu-se os controladores usados nos quadros elétricos dos equipamentos de refrigeração e o controlador de gestão central, descritos no capítulo 3, por módulos CC2530EM. Os restantes elementos constituintes dos equipamentos de refrigeração mantêm-se. Na Figura 17 podemos ver a representação do sistema criado, em que a distribuição dos equipamentos de refrigeração é feita segundo a planta do local a instalar.

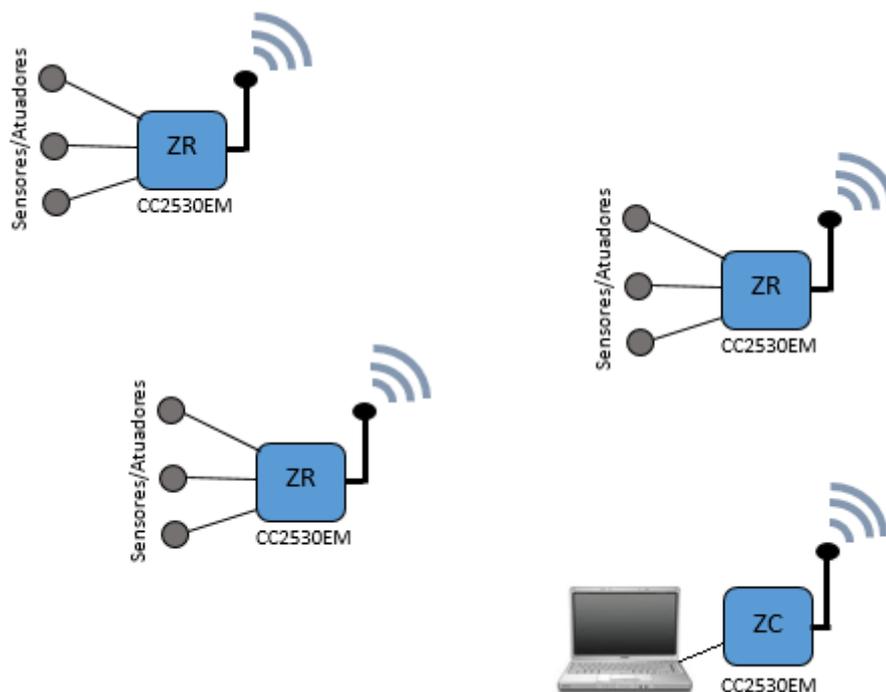


Figura 17. Descrição geral de funcionamento.

Um dos modos de funcionamento dos módulos CC2530EM é como *Routers* ZigBee (ZRs). Estes encontram-se no quadro elétrico de cada equipamento de refrigeração. Aos ZRs estão ligados fisicamente os sensores e atuadores de cada equipamento de refrigeração, como descritos no capítulo 3. O outro modo de funcionamento do módulo CC2530EM é como Coordenador ZigBee (ZC), único em todo o sistema, designado de estação central e, que normalmente se encontra na casa das máquinas, ou em local próprio designado para o efeito.

Cada ZR, para além de controlar os sensores e atuadores ligados a ele, tem também a função de enviar os dados recolhidos para o destinatário. Caso o destinatário não esteja ao seu alcance, o ZR envia os dados para outro ZR ao seu alcance, e assim sucessivamente, até os dados chegarem ao destinatário. Pretende-se que os dados recolhidos nos equipamentos de refrigeração cheguem até ao ZC, para que possam ser visualizados em tempo real. Na generalidade dos casos, o controlador do equipamento de refrigeração, neste caso um dispositivo ZR, funciona de modo independente, ou seja, lê os valores dos seus sensores e, de acordo com a sua função, controla os seus atuadores, enviando sempre todos os dados pela rede para o ZC. Isto

não impede que um sensor de um equipamento de refrigeração seja usado para controlar um atuador de um outro equipamento de refrigeração, utilizando a rede de comunicação para esse efeito.

4.4 Router ZigBee

Neste projeto é possível a utilização de dispositivos ZR, em detrimento de ZEDs, pois estes são instalados nos quadros elétricos de todos os equipamentos de refrigeração. Sendo assim, uma vez que o quadro é alimentado pela rede elétrica, a utilização destes dispositivos não é condicionada pelo consumo de energia dos dispositivos. Devido à pequena dimensão dos equipamentos de refrigeração (dificilmente chegam à dezena de m²), a ligação entre os sensores/atuadores e o ZR é feita através de cabos. Os ZRs têm de ser configurados um a um, porque dependem da função do equipamento de refrigeração: número de sensores e atuadores; temperaturas de funcionamento. O ZR é responsável por controlar todo o equipamento de refrigeração, bem como proporcionar interação com todos os dispositivos da rede ZigBee utilizada. O ZR tem a capacidade de enviar ou receber dados de, ou para qualquer outro ZR dentro da rede e também para o ZC.

4.4.1 Sensores e atuadores

Cada dispositivo ZR é responsável por recolher e tratar os dados de duas sondas de temperatura (sonda1 e sonda2), descritas no capítulo 3. A sonda1 controla a temperatura do evaporador e a sonda2 controla a temperatura ambiente do equipamento de refrigeração. A configuração do porto P0 como entrada para o ADC permite a leitura dos valores da sonda1 no pino P0.7 e da sonda2 no pino P0.6, com uma resolução de 12 bits. Ao contrário do controlador AK-PC 530, descrito na secção 3.3, foi necessário a criação de um divisor de tensão, como mostra a Figura 18, de forma a fornecer uma alimentação às sondas.

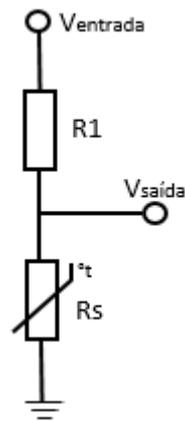


Figura 18. Divisor de tensão.

Para o cálculo do divisor de tensão usou-se o valor de referência da sonda R_s (1000 Ω a 25 $^{\circ}\text{C}$ [22]) e uma resistência R_1 com o mesmo valor resistivo (1000 Ω). A alimentação do divisor de tensão ($V_{entrada}$) é de 3.25 V DC, e a tensão de saída para uma temperatura de 25 $^{\circ}\text{C}$ ($V_{saída}$), é de 1.625 V DC, como podemos ver na equação 1.

$$V_{saída} = \frac{R_s}{R_s + R_1} * V_{entrada} \quad (1)$$

$$V_{saída} = \frac{1000}{1000 + 1000} * 3.25 = 1.625 \text{ V}$$

A tensão de referência da placa SmartRF05EB é de 3.25 V DC, o que significa que, se a entrada do valor da sonda (corresponde ao $V_{saída}$ do divisor de tensão) for de 3.25 V DC, o valor obtido através ADC é o valor máximo que este nos fornece. A escolha de $V_{saída}$ surgiu de forma a obter um valor de resistência intermédio que pudesse aumentar ou diminuir de igual forma.

Na Figura 19, podemos ver que existe uma pequena diferença entre o valor teórico (fornecido pelo *datasheet* da sonda) e o valor medido (valor obtido numa medição prática).

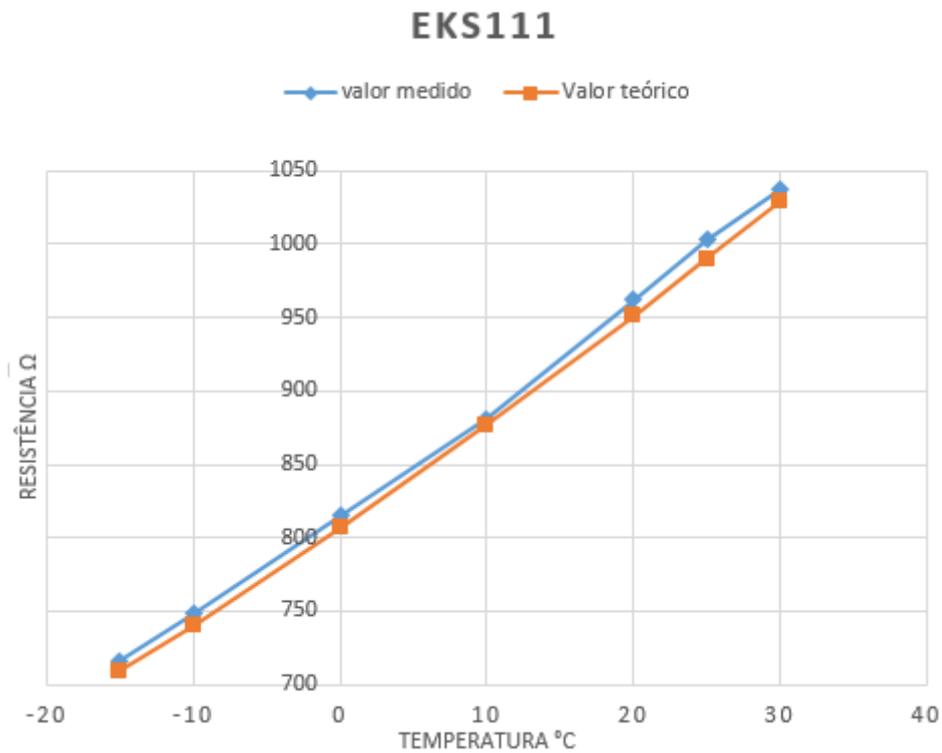


Figura 19. Valor da resistência medido e teórico.

A variação do valor lido pelo ADC com a temperatura não é linear, o que obrigou a uma medição pormenorizada dos valores. O valor digital fornecido pelo ADC, corresponde à sua resolução e varia de 0 até 2048. Estes valores são obtidos pela leitura dos valores analógicos de tensão da sonda, que variam entre 0 V DC e 3.25 V DC. Como podemos observar na Tabela 2, quando o valor da tensão de entrada no ADC (*Vsaída* do divisor de tensão) é metade do valor máximo (1.625 V DC, a 25 °C), o valor fornecido pelo ADC é aproximadamente metade do seu valor máximo. Os valores da tabela 2 foram obtidos através de um teste prático.

Tabela 2. Valores lidos pelo ADC.

Temperatura °C	Valor lido pelo ADC
30	1034
25	1018
20	1000
15	978
10	958
5	933
0	908
-5	880
-10	852
-15	842

Para além das sondas de temperatura, o ZR recebe informação de um outro sensor (interruptor fim de curso), designado de “micro de porta”. Como o próprio nome indica, este sensor é instalado junto à porta, a fim de obter informação sobre o seu estado. O micro de porta tem apenas dois estados possíveis: ligado e desligado. O micro de porta tem uma alimentação de 3.25 V DC, enquanto o ZR tem o pino P2.1 configurado como entrada digital. Quando a porta se encontrar aberta, o micro de porta vai estar no estado ligado (interruptor fechado) e por consequência a entrada P2.1 recebe 3.25 V DC, o que a coloca a nível lógico “1”. Quando a porta se encontra fechada o micro de porta tem o contacto fechado e a entrada P2.1 está no nível lógico *low*.

Os atuadores do equipamento de refrigeração controlados pelo ZR são a válvula de expansão e o ventilador. O porto P1 do módulo do ZR disponibiliza duas saídas, P1.0 e P1.1, capazes de fornecer uma corrente de 20 mA. No projeto do sistema, estas saídas alimentam dois relés, uma vez que não fornecem corrente suficiente para

alimentar diretamente os atuadores. A válvula de expansão e o ventilador são alimentados a uma tensão de 230 V AC, fornecida pelo quadro elétrico e controlada através dos contactos dos relés pelo ZR.

4.4.2 Modo de funcionamento

É através do software desenvolvido que o ZR controla todos os dispositivos de um equipamento de refrigeração e a sua interação com a rede ZigBee. O ZR é registado com um número de equipamento fixo que o identifica, o qual é atribuído pelo gestor do sistema e inserido durante a configuração do dispositivo. Caso exista alguma avaria com o módulo do ZR e este necessite de ser trocado, o seu substituto tem de permanecer com o mesmo número de equipamento.

No ZR é feita a conversão dos valores das sondas, lidos pelo ADC, em valores de temperatura. O equipamento de refrigeração trabalha entre duas temperaturas pré-definidas. A temperatura de paragem corresponde à temperatura mínima que se pretende que o equipamento atinja. Uma vez atingida essa temperatura, o sistema deve parar de refrigerar. Em contraste, a temperatura de arranque corresponde à temperatura mais elevada que o equipamento pode atingir. Quando alcançada essa temperatura, é espectável que o sistema inicie a refrigeração. O ZR controla a temperatura do equipamento de refrigeração através da leitura da sonda de temperatura ambiente. Quando o valor for igual ou superior ao valor da temperatura de arranque, a válvula de expansão é ligada de modo a refrigerar o equipamento, até este atingir a temperatura de paragem. A partir do momento que se liga o equipamento de refrigeração (o ZR é ligado por consequência) o ventilador também é ligado, exceto nos seguintes casos:

- O micro de porta se encontre ligado, o que significa que a porta está aberta;
- O sistema esteja em descongelação.

O ZR controla as descongelações através da sonda de temperatura do evaporador. Caso a temperatura atinja um valor mínimo pré-definido, o ZR faz com que o ventilador pare e a válvula de expansão se desligue, de modo a não refrigerar. Dessa

forma, a temperatura vai subir, e quando chegar a uma temperatura pré-definida o sistema volta ao funcionamento normal.

O micro de porta funciona como uma interrupção ao sistema. No momento em que se abre a porta de um equipamento de refrigeração, o micro de porta é ligado, e, tal como o ventilador, a válvula de expansão é desligada.

As temperaturas das sondas e os estados dos atuadores são enviados por mensagens através da rede ZigBee para o ZC, de modo a ser possível a visualização do funcionamento do equipamento de refrigeração, ou caso se pretenda interagir com ele.

4.5 Coordenador ZigBee

O ZC é o dispositivo responsável por disponibilizar a visualização dos dados de todos os equipamentos de refrigeração. Os dados referidos podem ser visualizados em tempo real num computador, que recebe os dados do ZC por comunicação série, através da porta USART. Os dados recebidos que se pretendem disponíveis em tempo real são os seguintes:

- Número de equipamento;
- Temperatura ambiente;
- Temperatura do evaporador;
- Estado do micro de porta;
- Estado do ventilador;
- Estado da válvula de expansão;
- Temperatura máxima (temperatura de arranque);
- Temperatura mínima (temperatura de paragem).

Cada pacote de dados que o ZC recebe vem distinguido com o número que identifica o equipamento de refrigeração, designado número de equipamento. Através deste número, o ZC agrupa todos os dados relativos a esse equipamento de refrigeração.

A interação entre o utilizador e o sistema foi desenvolvida na linguagem de programação Java. Podemos ver na Figura 20 e na Figura 21 a interface gráfica do utilizador.



Figura 20. Interface de seleção de posto.

Desde o momento que se conecta o computador ao ZC, o utilizador tem a possibilidade de selecionar o posto a partir do qual pretende monitorizar. A figura 20 mostra a interface que permite a seleção do posto desejado. Como mostra a figura 21, depois de selecionado o posto são apresentados todos os dados relativos a esse posto, e ainda o botão “Sair” que retorna à janela de seleção de posto. A interface de seleção de posto e o botão “Definir valor” servem para alterar valores como:

- O estado do ventilador e o estado da válvula de expansão, definidos na figura 21 como “Estado Ventilador” e “Estado Válvula”, respetivamente, podem ser alterados de “0” para “1”, desligado e ligado, respetivamente.
- A temperatura máxima pretendida, definida como “Temp_Max” na figura 21, pode ser definida pelo utilizador, assim como a temperatura mínima, definida como “Tem_Min”.

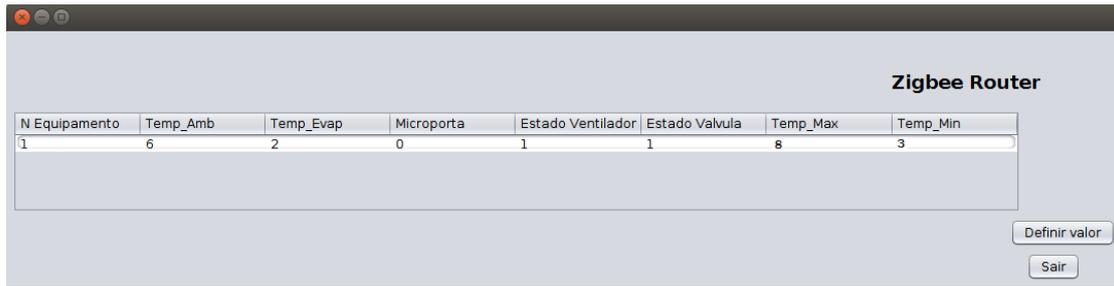


Figura 21. Interface de visualização de dados de um posto.

4.6 Rede ZigBee

A rede ZigBee criada é constituída pelo ZC e por múltiplos ZRs, sendo que os dois tipos de dispositivos são FFD, como definido na norma IEEE 802.15.4. A partir do momento em que o ZC se liga é formada uma PAN, sendo-lhe atribuída um identificador gerado aleatoriamente pelo ZC. O processo de associação de um ZR começa com a transmissão de uma trama *beacon request*, em modo *broadcast*, como podemos ver na Figura 22, obtendo como resposta uma trama *beacon*. De seguida, o ZR envia uma trama de pedido de associação, como podemos ver na Figura 23.

P.nbr.	Time (ms)	Length	Frame control field	Sequence number	Dest. PAN	Dest. Address	Beacon request	LQI	FCS
RX	+720	10	Type Sec Pnd Ack.req PAN_compr	0x3E	0xFFFF	0xFFFF		196	OK
8	=5830		CMD 0 0 0 0						

Figura 22. Formato da trama de *beacon request*.

P.nbr.	Time (ms)	Length	Frame control field	Sequence number					LQI	FCS
RX	+110	21	Type Sec Pnd Ack.req PAN_compr	0xCB						
9	=2632		CMD 0 0 1 0							

Dest. PAN	Dest. Address	Source PAN	Source Address	Association request	LQI	FCS
0xAF38	0x0000	0xFFFF	0x00124B000117E9E8	Alt.coord FFD Power Idle.RX Sec Alloc.addr	232	OK
				0 0 0 0 0 1		

Figura 23. Formato da trama de pedido de associação.

No pedido de associação, o ZR envia o endereço da rede ao qual se pretende associar e envia o seu endereço MAC. O ZC responde a este pedido com uma trama que contém o Extended PAN ID, o endereço MAC do ZR e o endereço de rede que lhe é atribuído. Na Figura 24 podemos ver o formato da trama de resposta do ZC, ao pedido de associação do ZR.

P.nbr.	Time (ms)	Length	Frame control field				Sequence number	
RX	+1		Type	Sec	Pnd	Ack.req	PAN_compr	
13	=3130	27	CMD	0	0	1	1	0xFF

Dest. PAN	Dest. Address	Source Address	Short_addr	Assoc. status	LQI	FCS
0xAF38	0x00124B000117E9E8	0x00124B00012F74DE	0x42A8	Successful	240	OK

Figura 24. Formato da trama de resposta ao pedido de associação.

Depois do processo de associação, os ZRs podem iniciar a transferência de dados para o ZC. Os primeiros dados a serem enviados são relativos ao número de equipamento e a quantidade de sensores e de atuadores que o equipamento de refrigeração controlado por esse ZR possui. Esta primeira trama de dados tem um tamanho de 33 bytes, com 6 bytes de *payload*. No envio da primeira trama de dados, bem como durante todo o processo de associação, todas as tramas enviadas são precedidas de ACKs, por parte do ZR e do ZC, com a exceção do *beacon request* que é precedido de uma trama *beacon*. De seguida, as mensagens enviadas passam a conter a informação proveniente de cada ZR, conforme referido na secção 4.5. Estas mensagens são enviadas com periodicidade de 1 segundo. Do ZC para o ZR o envio de mensagens acontece da mesma forma e com a mesma periodicidade, e o seu conteúdo são os dados do ZR, passíveis de serem alterados pelo utilizador. O tamanho das tramas trocadas na rede é variável, sendo que as tramas enviadas durante o processo de associação têm 10, 21 e 27 bytes, como podemos ver nas Figura 22, Figura 23 e Figura 24, respetivamente, enquanto as tramas de *data request* e ACK têm 12 e 5 bytes, respetivamente. O ZR envia uma trama de dados com 20 bytes de *payload*

para o ZC e, contando com os restantes campos, a trama tem um tamanho de 47 bytes. Já a trama do ZC é de 37 bytes, em que apenas 10 bytes são de *payload*, uma vez que este não necessita de enviar tanta informação para o ZR. Na Figura 25 podemos ver o formato das tramas de dados, enviadas do ZR para o ZC ou vice-versa, uma vez que só diferem no tamanho e endereços.

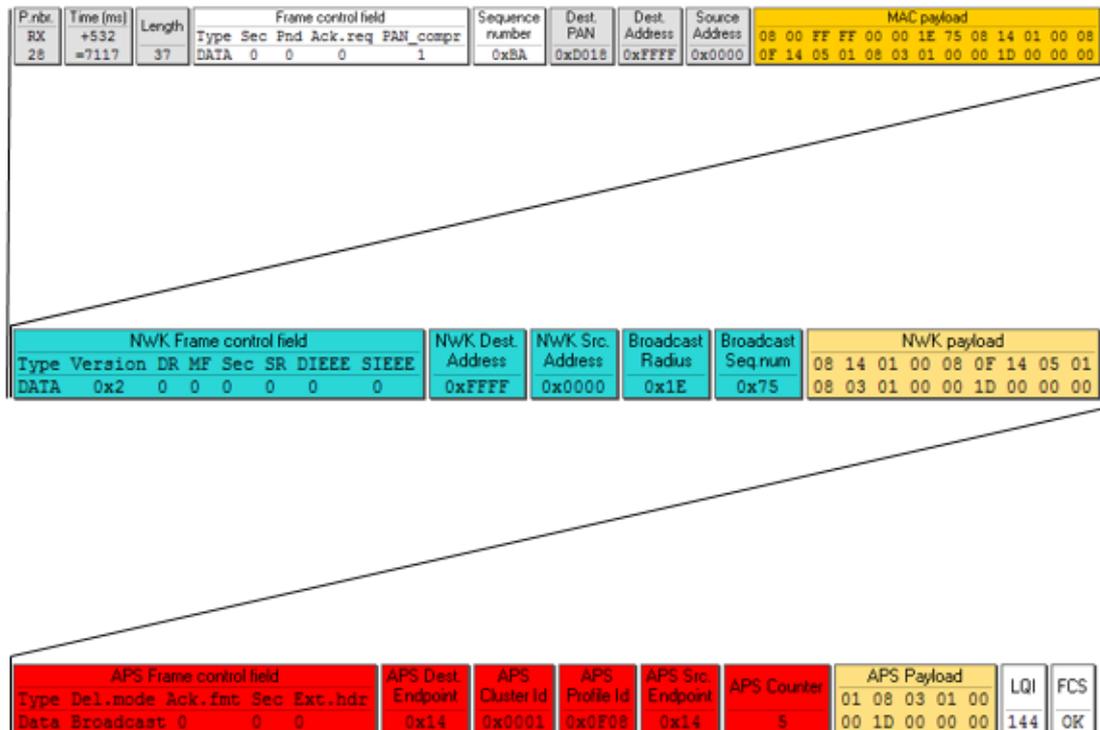


Figura 25. Formato da trama de dados do ZC.

Na Figura 26 podemos ver a sequência do fluxo de comunicação entre os dispositivos da rede ZigBee implementada.

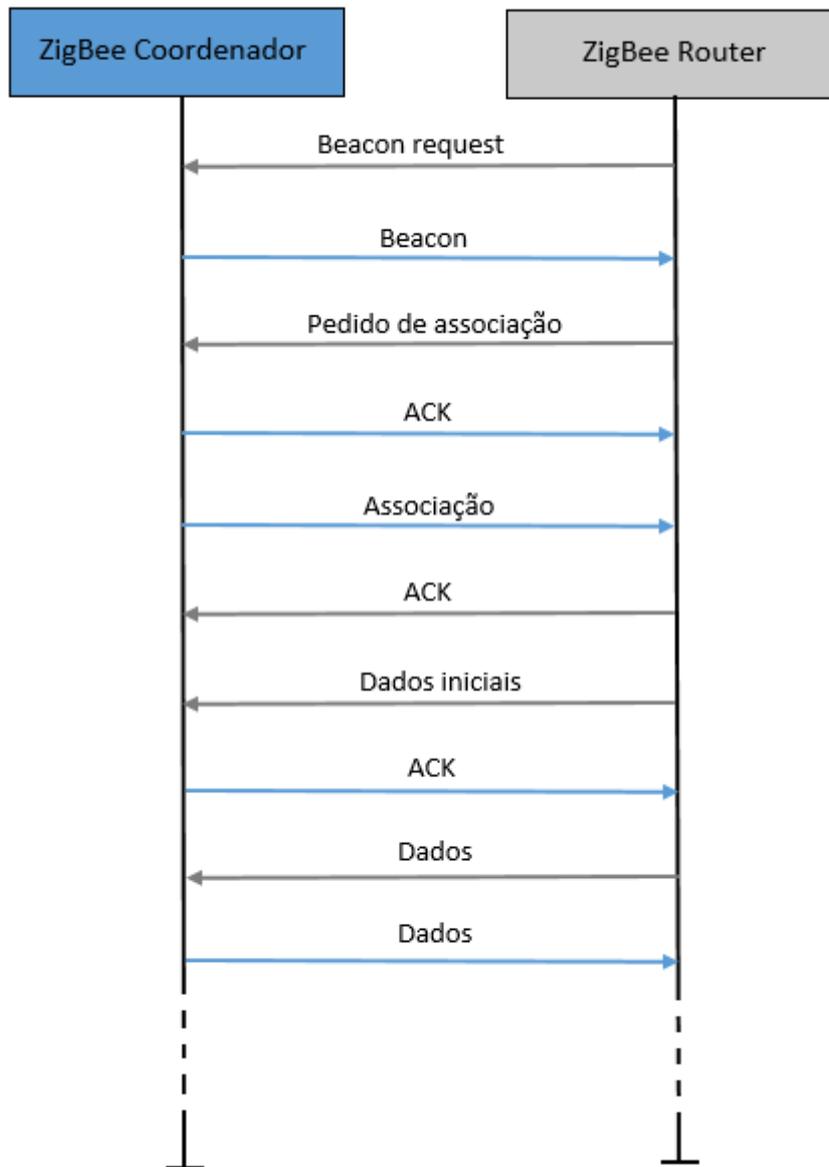


Figura 26. Sequência do fluxo de dados entre ZC e ZR.

4.7 Software

A plataforma de desenvolvimento Z-Stack da Texas Instruments fornece uma aplicação, chamada SampleApp, que serve como base para implementação deste sistema. As redes ZigBee são formadas por três tipos de dispositivos: coordenador, *router* e *end device*. Cada dispositivo executa um conjunto específico de funções, em que correm aplicações diferentes em cada um. As principais funções fornecidas pela

aplicação `SampleApp` da `Z-Stack`, que serviram de base aos dispositivos implementados, são as seguintes:

- `SampleApp_Init()` - Esta função é chamada na inicialização da rede e pode conter algumas aplicações específicas da inicialização do dispositivo (e.g., tabelas de inicialização, configurações da `UART`, etc.). Nesta função é atribuído o endereço *broadcast* como destino das mensagens periódicas geradas pelo dispositivo, como podemos ver no seguinte excerto de código:

```
SampleApp_Periodic_DstAddr.addrMode=(afAddrMode_t)AddrBroadcast;
SampleApp_Periodic_DstAddr.addr.shortAddr = 0xFFFF;
```

- `SampleApp_ProcessEvent()` - Esta função é chamada para processar todos os eventos recebidos pela aplicação. Um dos eventos possíveis é a mudança do estado do dispositivo na rede, que proporciona a ativação de um temporizador que, sempre que expira, chama uma função para o envio periódico de um pacote por RF.

- `SampleApp_MessageMSGCB()` - Esta função é executada sempre que o dispositivo recebe um pacote por RF. Tem como argumento um apontador para o conteúdo do pacote recebido (`afIncomingMSGPacket_t *pkt`).

- `SampleApp_SendPeriodicMessage()` - Esta função é chamada sempre que o temporizador expira. Quando chamada, envia os dados para o endereço de rede configurado.

Para o funcionamento de cada dispositivo, foram realizadas diversas alterações e acrescentos no código da aplicação base `SampleApp`. O desenvolvimento do código dos dispositivos realizou-se em duas fases. A primeira consistiu nas alterações à `SampleApp` para envio dos dados dos ZEDs para o ZC e deste para o PC, tendo como base um tutorial redigido pelo orientador, bem como na implementação de um algoritmo para testar a taxa de erros na transmissão de dados na rede. Na segunda fase foi desenvolvido o código para o sistema implementado.

4.7.1 Primeira fase

A aplicação `SampleApp` continha apenas um ficheiro `SampleApp.c`, e a partir deste foram criados três ficheiros diferentes, um para o *end device*, um para o *router* e um para o coordenador, para que cada tipo de dispositivo pudesse executar um código diferente a nível da aplicação. Os dispositivos *end device* foram utilizados apenas na primeira fase do projeto e as alterações de código foram efetuadas na sua aplicação, no ficheiro `SampleApp_ED.c`. A primeira alteração efetuada no código deste dispositivo foi a mudança do endereço do destinatário dos pacotes periódicos, que se encontra na função `SampleApp_Init()`. Estes pacotes eram enviados para o endereço *broadcast* e passaram a ser enviados para o coordenador, que tem um endereço fixo (é sempre 0). O código alterado fica da seguinte forma:

```
SampleApp_Periodic_DstAddr.addrMode = (afAddrMode_t)Addr16Bit;
SampleApp_Periodic_DstAddr.addr.shortAddr = 0x00;
```

De seguida, dentro da função `SampleApp_ProcessEvent()`, procedeu-se à configuração do temporizador para a leitura das amostras periódicas. O código que permite a reinicialização do temporizador para a próxima leitura foi modificado de forma a retirar a componente aleatória. Podemos ver o código da aplicação original no seguinte excerto de código:

```
osal_start_timerEx( SampleApp_TaskID,
SAMPLEAPP_SEND_PERIODIC_MSG_EVT,
(SAMPLEAPP_SEND_PERIODIC_MSG_TIMEOUT + (osal_rand() & 0x00FF)));
```

A este código foi retirada a parte que gera a componente aleatória: `(osal_rand() & 0x00FF)`.

A configuração do período desejado entre leituras, definido para 200 ms nesta fase, foi efetuada no ficheiro `SampleApp.h`, alterando a seguinte linha de código:

```
#define SAMPLEAPP_SEND_PERIODIC_MSG_TIMEOUT 200
```

Nesta primeira fase, foi utilizado como sensor o potenciómetro da SmartRF05EB. A leitura dos dados deste sensor foi feita no canal 7 do ADC, com uma resolução de 12 bits. Para usar a função `HalAdcRead` foi necessário fazer `#include "hal_adc.h"` no

início do ficheiro. O código seguinte foi utilizado para o efeito e colocado dentro da função `SampleApp_SendPeriodicMessage()`, chamada sempre que o temporizador expira:

```
uint16 leitura;

leitura = HalAdcRead(HAL_ADC_CHANNEL_7, HAL_ADC_RESOLUTION_12);

buffer[1] = HI_UINT16( leitura );

buffer[2] = LO_UINT16( leitura );
```

Para obter informação sobre a taxa de erros na transmissão de dados na rede, foi criada a variável `numero_de_sequencia`, de 16 bits. Esta variável era incrementada sempre que a função era chamada, sendo o seu valor enviado no *payload* do pacote de dados para o coordenador por RF, utilizando o comando `AF_DataRequest`. Nos parâmetros deste comando, foi alterado o apontador para o *payload*, de `(uint8*)&SampleAppPeriodicCounter` para uma variável chamada `buffer`, do tipo *array*, e o tamanho do *payload* foi alterado na função `AF_DataRequest` de 1 para 90 bytes. Estas alterações foram feitas utilizando o seguinte código dentro da função `SampleApp_SendPeriodicMessage()`:

```
uint8 buffer[90];

buffer[3] = HI_UINT16( numero_de_sequencia );

buffer[4] = LO_UINT16( numero_de_sequencia );

numero_de_sequencia++;

HalLedSet(HAL_LED_1, HAL_LED_MODE_TOGGLE);

if ( AF_DataRequest(

    &SampleApp_Periodic_DstAddr, &SampleApp_epDesc,

    SAMPLEAPP_PERIODIC_CLUSTERID,

    90,

    buffer,

    &SampleApp_TransID,

    AF_DISCV_ROUTE,

    AF_DEFAULT_RADIUS ) == afStatus_SUCCESS ) {}

else {
```

```
// Error occurred in request to send.
}
```

Da mesma forma que nos *end devices*, as alterações na aplicação coordenador foram efetuadas no ficheiro `SampleApp_coord.c`. De forma a poder estabelecer uma comunicação via UART, foi necessário acrescentar as seguintes configurações:

- Fazer `#include "hal_uart.h"` no ficheiro.
- Definir parâmetros do UART na secção `CONSTANTS` do ficheiro:

```
/// Parâmetros do UART
#define SAMPLEAPP_PORT 0
#define SAMPLEAPP_BAUD HAL_UART_BR_38400
```

- Definir a função de callback do UART na secção `LOCAL FUNCTIONS` do ficheiro:

```
static void SampleApp_Callback(uint8 port, uint8 event);
```

- Inicializar o UART e abrir a porta série no final da função `SampleApp_Init()`:

```
/// Configuração do UART
halUARTCfg_t uartConfig;
uartConfig.baudRate = SAMPLEAPP_BAUD;
uartConfig.flowControl = FALSE;
uartConfig.callBackFunc = SampleApp_Callback;
/// Abre a porta série
HalUARTOpen (SAMPLEAPP_PORT, &uartConfig);
```

- Inserção do código para o envio do *payload* pelo UART no final da função

```
SampleApp_MessageMSGCB() :
HalUARTWrite( SAMPLEAPP_PORT, pkt->cmd.Data, 90 );
```

- Inserir a função de *callback* do UART no final do ficheiro:

```
static void SampleApp_Callback(uint8 port, uint8 event){ }
```

Nesta fase do projeto, o objetivo do coordenador, para além de permitir estabelecer uma comunicação série com um PC, era calcular a taxa de erros, com base nos pacotes de dados recebidos, e mostrá-la no LCD da SmartRF05EB. Para usar funções do LCD foi necessário incluir a biblioteca `"hal_lcd.h"` no ficheiro. Sempre que um pacote era recebido pelo coordenador, a função `SampleApp_MessageMSGCB()` era executada. Esta função tinha como argumento um apontador para o conteúdo do pacote recebido (`afIncomingMSGPacket_t *pkt`), sendo que a variável `pkt->cmd.data` continha os dados enviados pelo utilizador.

Para poder calcular a taxa de erros, passou-se os dados relativos ao número de sequência para a variável `num_seq`, utilizando o seguinte código:

```
num_seq = pkt->cmd.Data[3]*256 + pkt->cmd.Data[4];
```

A Figura 27 apresenta o algoritmo de cálculo de taxa de erros sob a forma de fluxograma funcional. Quando a função `SampleApp_MessageMSGCB()` era executada, recebia o `num_seq`, que era comparado com um número de sequência esperado (`num_seq_esperado`). Se o número recebido fosse diferente do número esperado significava que se tinham perdido pacotes de dados durante a transmissão e, desta forma, procedia-se ao incremento do valor da variável de contabilização do número de pacotes perdidos, seguido do cálculo da taxa de erros.

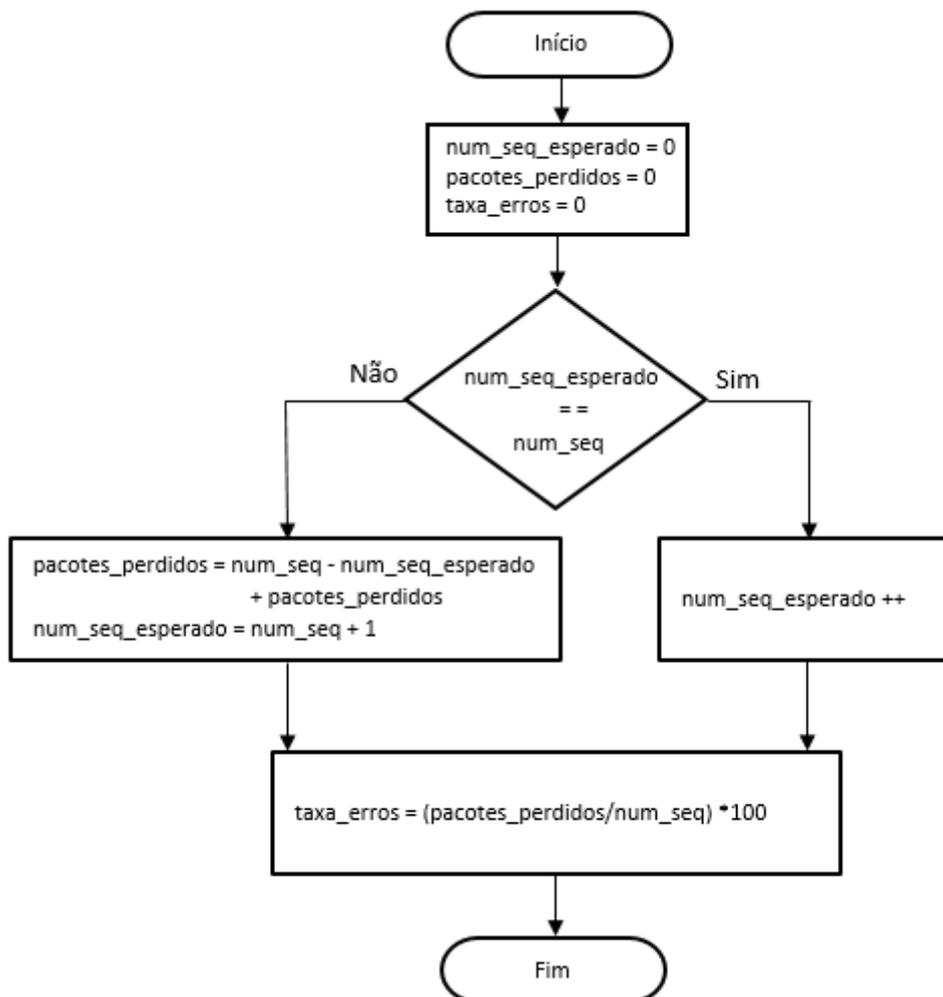


Figura 27. Fluxograma do algoritmo de cálculo de taxa de erros.

Depois dos cálculos efetuados, o coordenador mostrava numa das linhas do LCD a taxa de erros resultante e noutra linha o valor da leitura do sensor, enviado pelo *end device*, utilizando o seguinte código:

```
HalLcdWriteStringValue("Leitura : ", pkt->cmd.Data[1]*256 + pkt->cmd.Data[2], 10, HAL_LCD_LINE_1);

HalLcdWriteStringValue("% de erros :
", taxa_erros, 10, HAL_LCD_LINE_2);
```

A aplicação SampleApp original está sempre a enviar pacotes por RF. Nesta fase do projeto, era necessário apenas o envio de pacotes do *end device* para o coordenador. Sendo assim, desativou-se o início da temporização periódica por parte do coordenador (comando `osal_start_timerEx`), na rotina de atendimento `ZDO_STATE_CHANGE`, dentro da função `SampleApp_ProcessEvent()`.

O *router* nesta fase do projeto tinha apenas a função de reencaminhar pacotes, recebidos do *end device*, para o coordenador. Esta tarefa é gerida pela camada de rede, sem que seja necessária a intervenção da aplicação. Sendo assim, a modificação a ser feita no ficheiro `SampleApp_router.c` é a mesma que foi feita no ficheiro `SampleApp_coord.c`, à exceção das configurações da UART, que não são necessárias, e da função `SampleApp_MessageMSGCB()`, que não deve ter qualquer tipo de acrescento de código.

4.7.2 Segunda fase

A diferença mais significativa da primeira para a segunda fase foi a substituição dos dispositivos *end device* por *routers*. O código de origem e as alterações efetuadas na primeira fase no ficheiro `SampleApp_ED.c` passaram para o ficheiro `SampleApp_router.c`, às quais se soma o acrescento de algumas funções e código de forma a atingir os objetivos de funcionamento do sistema. Nesta fase os *routers* são responsáveis pela leitura dos sensores e controlar os atuadores dos equipamentos de refrigeração.

Como requisito de funcionamento do sistema, o primeiro pacote de dados a ser enviado contém informação sobre o equipamento de refrigeração (nº do

equipamento, nº de sensores e nº de atuadores). Essa mensagem é enviada quando o equipamento muda de estado (e.g., quando termina o estado de associação à rede), na rotina de atendimento `ZDO_STATE_CHANGE` inserida na função `SampleApp_ProcessEvent()`.

Para controlar o equipamento de refrigeração, foi necessário criar e pré-configurar as seguintes variáveis:

- `temperaturaMax`, `temperaturaMin` – Estas variáveis devem ser pré-configuradas com o valor de temperatura máxima e mínima, respetivamente, que se pretende atingir no equipamento de refrigeração. Esse valor pode ser alterado pelo utilizador do sistema durante o seu funcionamento.
- `estadoVentilador`, `estadoValvula` – Estas variáveis só assumem dois valores (“0” ou “1”, desligado e ligado, respetivamente), e devem ser pré-configuradas com o valor “0” e “1”, respetivamente. Em caso de testes, o utilizador pode alternar o valor do estado das variáveis entre “0” e “1”.

Sempre que o *router* recebe um pacote, a função `SampleApp_MessageMSGCB()` é executada. Os dados recebidos dizem respeito aos valores das variáveis `temperaturaMax`, `temperaturaMin`, `estadoVentilador`, `estadoValvula` que, como referido anteriormente, podem ser configurados no coordenador, através da interface com o utilizador.

Como forma de obter o valor da temperatura adquirido pelas sondas, foi acrescentada a função `SampleApp_converter()` no ficheiro `SampleApp_router.c`. Esta função recebe como parâmetro o valor da leitura do ADC (`uint16 leitura`), e procede à transformação desse valor num valor de temperatura, como foi explicado na secção 4.4.1.

A função `SampleApp_controlar()` foi também acrescentada no ficheiro `SampleApp_router.c`. Esta função tem como objetivo controlar o equipamento de refrigeração. Este controlo passa pela configuração dos pinos como entrada ou saída, e pelo valor de leitura das sondas através do ADC e respetiva conversão em valor de temperatura, como podemos ver no seguinte excerto de código:

```
//leitura da sonda de temperatura ambiente e respetiva conversão
leituraAmbiente=HalAdcRead(HAL_ADC_CHANNEL_7,HAL_ADC_RESOLUTION_1
2);

temperaturaAmbiente = SampleApp_converter(leituraAmbiente);

//leitura da sonda do evaporador e respetiva conversão
leituraEvap = HalAdcRead(HAL_ADC_CHANNEL_6,HAL_ADC_RESOLUTION_12);
temperaturaEvap = SampleApp_converter(leituraEvap);

//Colocar P0.1 como entrada

POSEL = 0x02;

P0_1 = 1;

//Colocar como saída os pinos 0 e 1, do porto 1

P1DIR = 0x03;
```

A Figura 28 apresenta o algoritmo de funcionamento de um equipamento de refrigeração sob a forma de fluxograma funcional, implementado na função `SampleApp_controlar()`. A sua descrição foi explicada na secção 4.4.2.

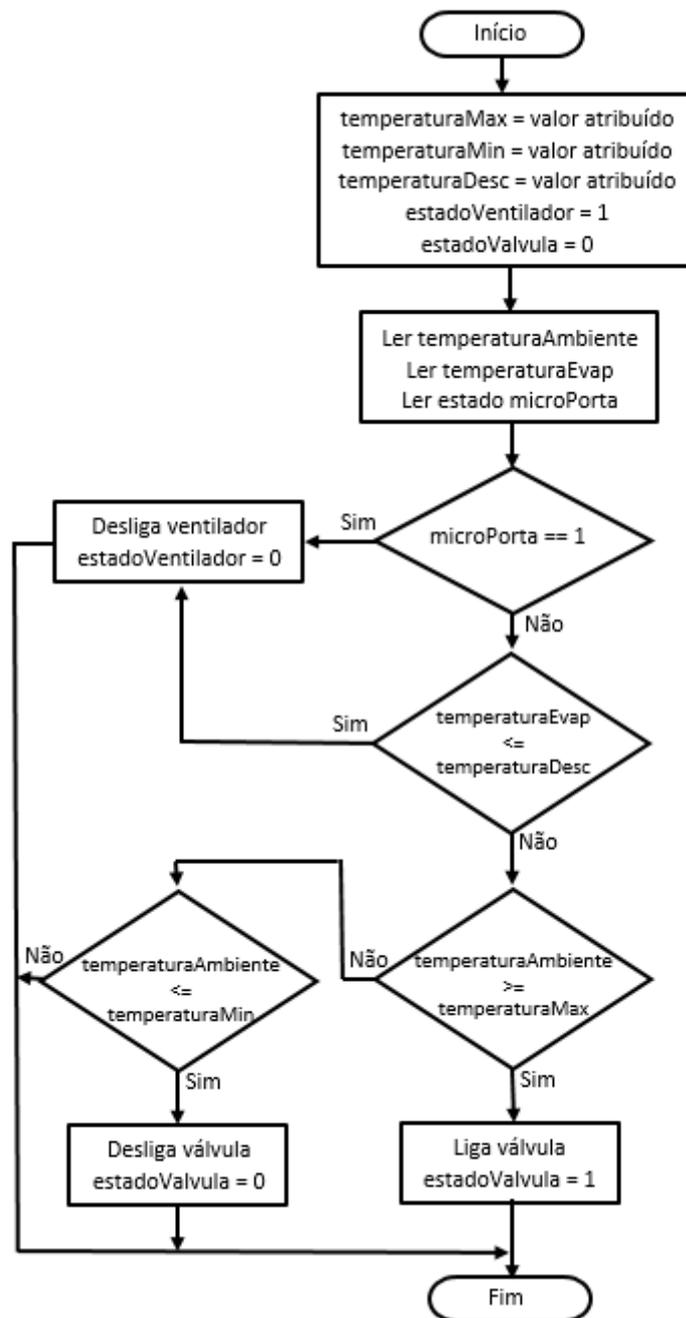


Figura 28. Fluxograma de algoritmo de funcionamento de equipamento de refrigeração.

O período desejado entre leituras foi alterado em relação à primeira fase, sendo agora configurado para 1000 ms, no ficheiro SampleApp.h, alterando a seguinte linha de código:

```
#define SAMPLEAPP_SEND_PERIODIC_MSG_TIMEOUT 1000
```

Sempre que o temporizador expira, a função `SampleApp_SendPeriodicMessage()` é executada. Dentro desta função é também chamada a executar a função `SampleApp_controlar()`, e de seguida é feito o envio do pacote de dados para o coordenador por RF. No primeiro byte dos dados é enviado o número do equipamento, que é configurado no *router* e nunca muda. De seguida são enviados todos os dados relativos ao equipamento de refrigeração, agora num pacote com um *payload* de 20 bytes.

No coordenador foram adicionadas algumas alterações no ficheiro `SampleApp_coord.c`, para além das efetuadas na primeira fase, que se mantiveram. Nesta fase, é requisito do coordenador ler o primeiro pacote de dados recebidos (após associação à rede por parte dos *routers*), que contém o nº de equipamento, nº de sensores e nº de atuadores, distintamente dos outros pacotes de dados. Para isso foi acrescentada a rotina de atendimento `SAMPLEAPP_INIT_CLUSTERID`, na função `SampleApp_MessageMSGCB()`.

Um outro requisito do coordenador é o envio de pacotes periódicos para os *routers*, sempre que o temporizador expira, com o valor das variáveis passíveis de alteração por parte do utilizador, através da interface gráfica. Desta forma, foi acrescentada à função `SampleApp_SendPeriodicMessage()` o código seguinte para leitura dos valores dessas variáveis pela porta série:

```
HalUARTRead( SAMPLEAPP_PORT, buffer, 10 );
```

Sempre que o coordenador recebe dados, a função `SampleApp_MessageMSGCB()` é executada e de seguida os dados são enviados através da porta série, para a aplicação do PC, onde se procede à visualização e possível alteração de valores de variáveis.

5.RESULTADOS E DISCUSSÃO

5.1 Introdução

Durante o desenvolvimento deste projeto foram realizadas duas fases distintas de testes. A segunda fase de testes foi realizada cerca de três meses após a primeira. O local dos testes, bem como o ambiente em que se realizaram foi sempre o mesmo. O objetivo destes testes é perceber se a rede ZigBee implementada satisfaz os requisitos necessários à realização deste projeto.

A realização de duas fases distintas de testes permitiu a observação e recolha de resultados da rede ZigBee a funcionar de diferentes formas, em que se distinguiu da primeira para a segunda fase, o tráfego de dados na rede e o modo de funcionamento dos equipamentos instalados.

5.2 Primeira fase de testes

Na primeira fase foram realizados testes de transferência de dados num ambiente industrial e cada kit (placa SmartRF05EB e módulo CC2530EM) foi colocado no local do equipamento a substituir, como podemos ver na Figura 29. Estes testes foram efetuados numa empresa de transformação de carnes onde existia um constante movimento de pessoas e materiais.

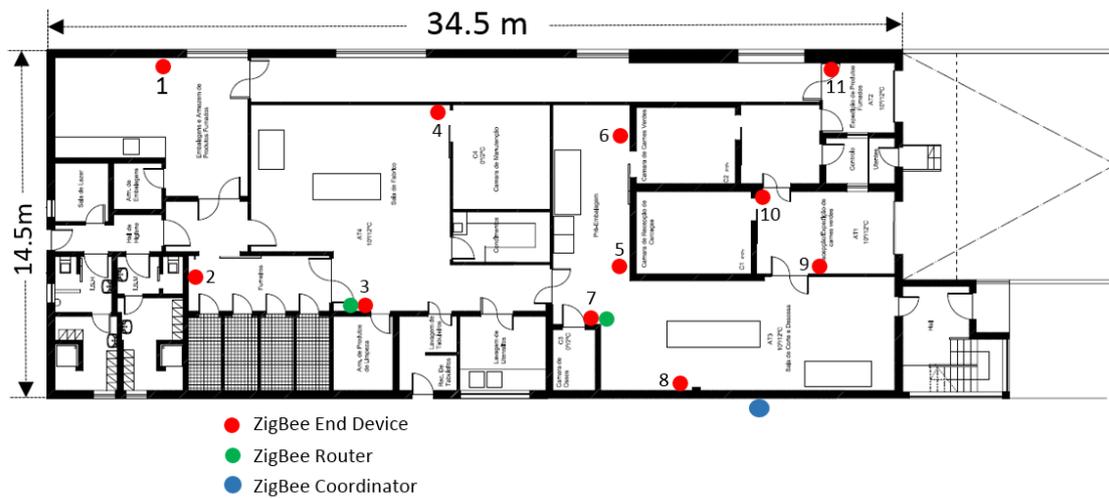


Figura 29. Planta da fábrica com a disposição dos dispositivos.

Para os testes realizados na primeira fase foram utilizados três dispositivos diferentes:

- ZigBee *End Device* – Gera uma determinada quantidade de dados, e envia-os para o coordenador direta ou indiretamente.
- *Router* ZigBee – Recebe os dados dos ZEDs mais distantes, que não conseguem alcançar o ZC diretamente e envia-os para o ZC.
- Coordenador ZigBee – Recebe os dados provenientes dos ZEDs.

Este teste foi realizado para garantir que a taxa de erros na transmissão era desprezável e, assim, poder continuar o projeto.

Foram recolhidas amostras do envio de 1000 pacotes de 90 bytes cada um, enviados a cada 200 ms, do ZED para o ZC. Quando um ZED não se conseguia conectar diretamente com o ZC, colocou-se um ZR a aproximadamente metade da distância entre o ZED e o ZC, e desta forma o ZR aumenta o alcance da rede recebendo os dados do ZED e encaminhando-os para o ZC. Na Tabela 3 podemos observar os valores relativos à transmissão dos dados no canal 26 da banda de frequência de 2.4 GHz. Esta zona da frequência estava livre quando analisada pelos analisadores de espectro.

Tabela 3. Testes no canal 26

Canal 26 da banda de frequências de 2.4 GHz				
Nº de equipamento	Nº de pacotes enviados	Nº de pacotes recebidos	Taxa de erro (em %)	Posição do ZR (nº de equipamento)
1	1000	1000	0	3
2	1000	994	0.6	3
3	1000	999	0.1	-
4	1000	999	0.1	7
5	1000	1000	0	-
6	1000	991	0.9	7
7	1000	1000	0	-
8	1000	1000	0	-
9	1000	1000	0	-
10	1000	1000	0	-
11	1000	999	0.1	-

De seguida foram realizados testes no canal 11 da mesma banda de frequências. De forma a perceber a ocupação do canal, foi utilizado um analisador de espectro (Chanalyzer 3.3 for Wi-Spy). Como mostra a Figura 30, o canal estava a ser utilizado por equipamentos de tecnologia Wi-Fi. Da mesma forma que o teste anterior, foram enviados 1000 pacotes de 90 bytes, espaçados 200 ms entre si. Os resultados obtidos podem ser visualizados na Tabela 4.

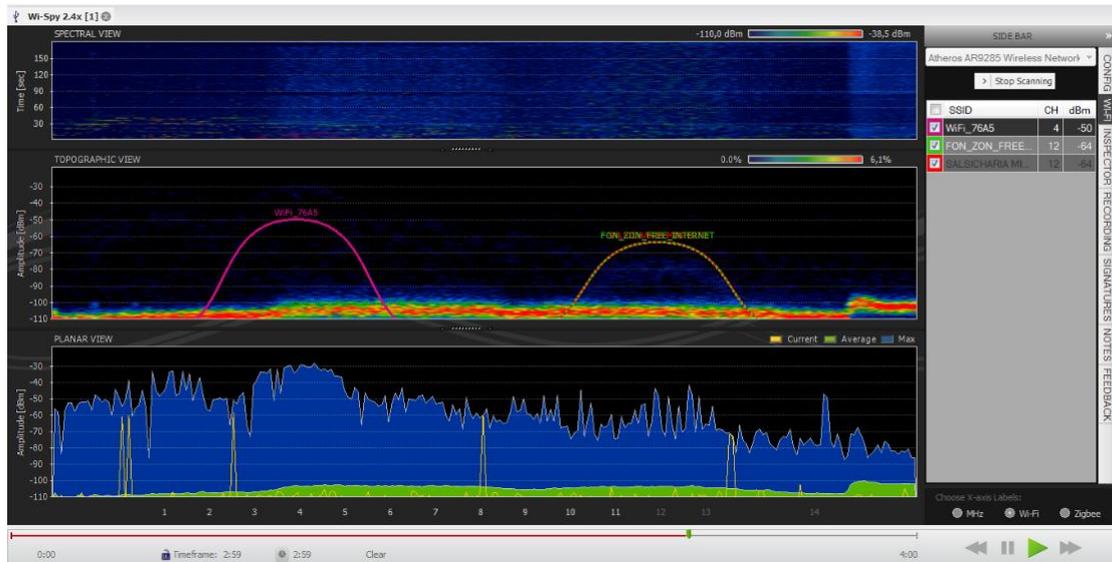


Figura 30. Ocupação do canal 11 da banda de frequência de 2.4 GHz.

Tabela 4. Testes no canal 11.

Canal 11 da banda de frequências de 2.4 GHz				
Nº de equipamento	Nº de pacotes enviados	Nº de pacotes recebidos	Taxa de erro (em %)	Posição do ZR (nº do equipamento)
1	1000	998	0.2	3
2	1000	992	0.8	3
3	1000	999	0.1	-
4	1000	998	0.2	7
5	1000	1000	0	-
6	1000	991	0.9	7
7	1000	1000	0	-
8	1000	999	0.1	-
9	1000	1000	0	-
10	1000	1000	0	-
11	1000	998	0.2	-

Apesar da utilização de outras tecnologias no canal 11, os resultados obtidos nas duas amostras são muito similares. Na Figura 31 podemos observar o pequeno aumento da taxa de erro no canal 11 em relação ao canal 26. No entanto, esta ligeira percentagem de erros, não traz qualquer impedimento à continuação do desenvolvimento do projeto, uma vez que sendo muito pequena, é difícil a sua visualização e não tem influência no desempenho dos equipamentos de refrigeração.

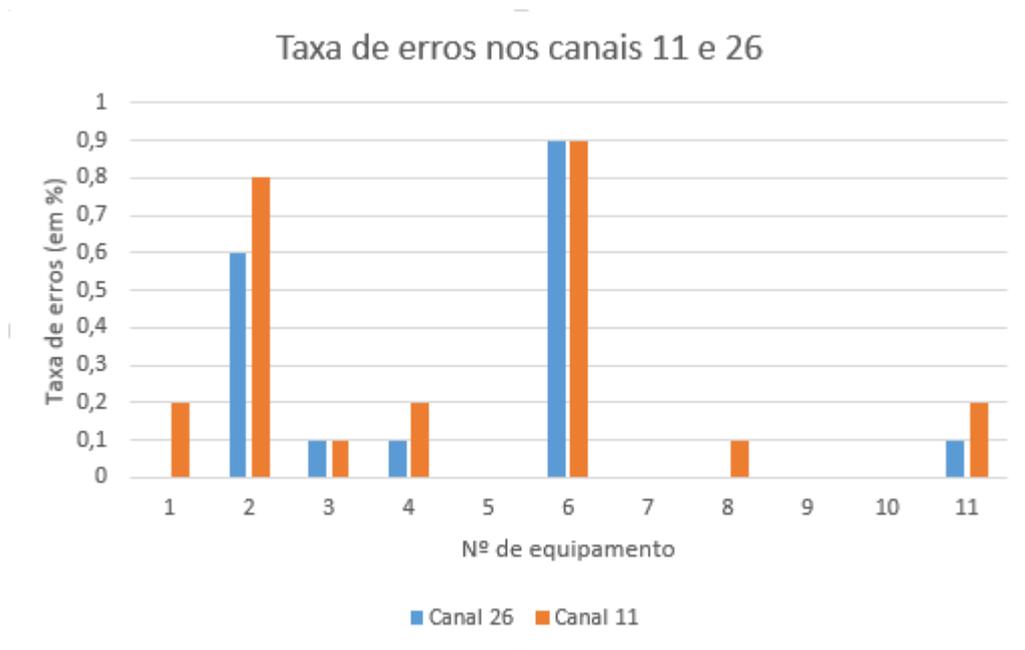


Figura 31. Taxa de erros nos canais 11 e 26.

5.3 Segunda fase de testes

Numa fase mais avançada do desenvolvimento do projeto, realizou-se a segunda fase de testes. Nesta fase, o objetivo era enviar, através da rede ZigBee, os dados relativos aos equipamentos de refrigeração dos ZRs para o ZC. Como o sistema desenvolvido não restringe o consumo de energia, foram utilizados apenas ZRs e um Zc, eliminando a utilização de ZEDs, o que era um dos pressupostos do projeto. De forma a obter resultados que se enquadrem com o funcionamento do sistema, nesta fase foram enviados pacotes de dados de 20 bytes, espaçados 1 segundo entre si, que levam os dados relativos ao funcionamento do equipamento de refrigeração e ainda

Tabela 5. Testes no canal 26.

Canal 26 da banda de frequência de 2.4 GHz			
Nº de equipamento	Nº de pacotes enviados	Nº de pacotes recebidos	Taxa de erro (em %)
1	1000	999	0.1
2	1000	996	0.4
3	1000	1000	0
4	1000	1000	0
5	1000	1000	0
6	1000	995	0.5
7	1000	1000	0
8	1000	1000	0
9	1000	1000	0
10	1000	1000	0
11	1000	999	0.1

Uma vez substituídos os ZEDs por ZRs, a necessidade de utilizar dispositivos que permitam o crescimento da rede deixa de existir, assim, um ZR que não alcance diretamente o ZC, alcança sempre outro ZR, que por sua vez recebe e envia os dados para o ZC.

Da primeira fase de testes para a segunda podemos observar uma pequena diminuição da taxa de erros, como se verifica na Figura 33. Esta taxa de erros diz respeito ao canal 26 da banda de frequência dos 2.4 GHz, no qual, na primeira fase de testes, eram enviados 1000 pacotes de dados de 90 bytes cada um, espaçados 200 ms entre si, e na segunda fase era enviado o mesmo número de pacotes de dados, de 20 bytes e espaçados 1 segundo entre si.

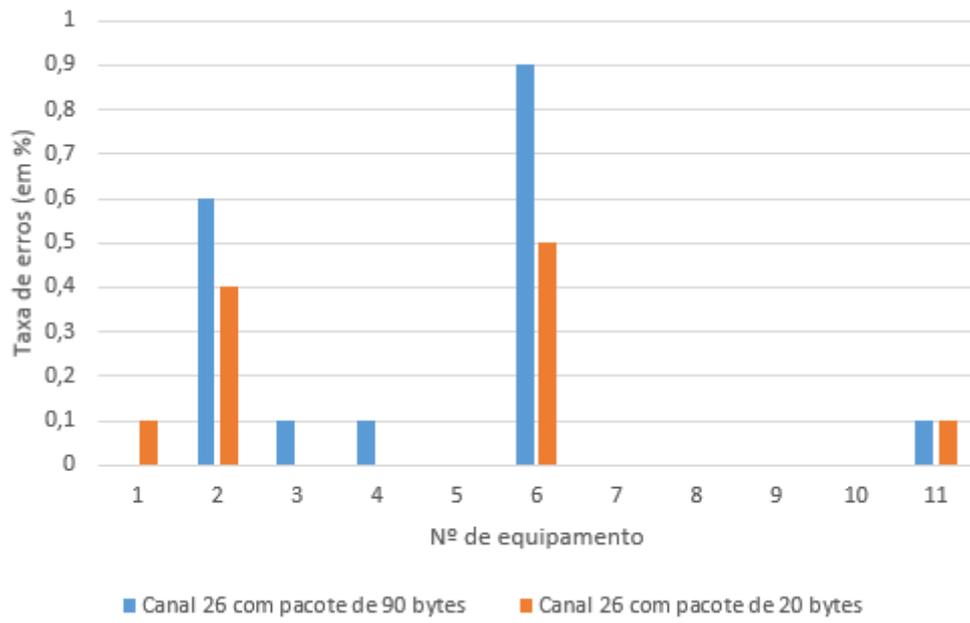


Figura 33. Taxa de erros do canal 26, com o envio de pacotes de 20 e 90 bytes.

6. CONCLUSÕES

O foco principal deste projeto visava o desenvolvimento de uma rede sem fios ZigBee para controlar e monitorizar um processo industrial, de forma a substituir a rede cablada existente. Os elementos a substituir eram os controladores dos equipamentos de refrigeração e todos os dispositivos que faziam parte da rede existente. Atualmente o tempo de vida dos equipamentos de refrigeração existentes é relativamente curto, entre 10 a 15 anos. Desta forma, e acrescentando o facto de o número de instalações industriais/comerciais que utilizam estes equipamentos ter crescido consideravelmente na última década, nomeadamente as médias e grandes superfícies comerciais, é possível proceder-se à substituição das redes cabladas existentes pela rede sem fios a curto/médio prazo.

Na análise prévia, foram recolhidas informações sobre o estado da arte, nomeadamente no que concerne às redes de sensores sem fios e à rede cablada que se pretendia substituir, de forma a consolidar o conhecimento sobre a matéria, e assim inicializar o projeto.

Inicialmente, implementou-se código para comunicação entre um dispositivo terminal ZigBee e um coordenador ZigBee, em que o coordenador ZigBee calculava e fornecia informações sobre a taxa de erros na transmissão de dados na rede.

Como forma de teste, os pacotes de dados enviados pela rede foram configurados para que as tramas tivessem um tamanho próximo do limite máximo suportado pela norma IEEE 802.15.4. Estes dispositivos foram colocados no ambiente industrial onde deveriam ser instalados, e foram realizados testes para determinar a taxa de erros. Apesar das redes ZigBee serem suscetíveis a interferências de outras redes sem fios em certos canais, o canal selecionado para a rede projetada revelou ser imune a estas interferências nos testes realizados.

Outro aspeto a ter em consideração era a distância entre o dispositivo terminal e o coordenador, que em alguns casos se mostrou ser demasiado elevada. Para resolver esse problema, implementou-se um *router* ZigBee, que foi colocado no caminho entre os dois dispositivos, sendo capaz, assim, de receber os dados do dispositivo terminal e encaminhá-los para o coordenador. Desta forma, dado que a taxa de erros na transmissão dos dados na rede foi baixa, o resultado destes testes permitiu concluir que o projeto tinha viabilidade para avançar.

O passo seguinte foi realizar um estudo acerca de todos os componentes que faziam parte do sistema existente, de forma a determinar quais os novos dispositivos a implementar. Os dispositivos substituídos foram os controladores do equipamento de refrigeração e o controlador central, entrando para os seus lugares a placa SmartRF05EB com o respetivo módulo CC2530EM. Todas as ligações dos sensores e atuadores existentes eram suportadas pelos novos dispositivos introduzidos. Como o sistema não apresentava restrições quanto ao consumo de energia, decidiu-se implementar apenas *routers* ZigBee, em detrimento de dispositivos terminais ZigBee, junto com um coordenador ZigBee ligado a um PC. Desta forma, conseguiu-se também resolver o problema da distância entre dispositivos terminais e coordenador, uma vez que diversos *routers* estão distribuídos no espaço industrial.

Como forma de monitorizar e controlar os equipamentos de refrigeração, foi desenvolvida uma interface gráfica com o utilizador. Esta interface é executada no PC, e para isso foi necessário criar uma ligação entre o coordenador ZigBee e o PC através da porta série RS-232 da placa SmartRF05EB, de forma a obter uma comunicação bidirecional de dados. A interface com o utilizador mostrou-se bastante útil, pois fornecia uma fácil visualização dos dados recolhidos de todos os equipamentos de refrigeração e proporcionava o seu controlo.

Depois do projeto devidamente estruturado, procedeu-se a uma segunda fase de testes. Nesta fase o número de pacotes de dados a enviar pela rede era menor que na primeira, e a cadência de envio mais lenta. Os *routers* ZigBee foram colocados junto aos quadros dos equipamentos de refrigeração, de forma a simular a substituição dos controladores. As distâncias entre os equipamentos de refrigeração e o coordenador

ZigBee já não tinham importância, uma vez que a rede era constituída apenas por *routers* e estes tinham sempre mais de que um *router* ao seu alcance. Por consequência, os dados chegavam sempre dos *routers* ao coordenador e vice-versa. Nesta segunda fase de testes os resultados foram ainda mais satisfatórios do que na primeira, pois verificou-se que a taxa de erros relativa à transferência de dados na rede ZigBee foi ainda menor.

Pode-se, assim, concluir que os objetivos requeridos pelo sistema são satisfeitos pela rede sem fios ZigBee e desta forma, a substituição da rede cablada pela rede sem fios ZigBee é viável.

Como trabalho futuro, poderiam ser acrescentadas mais funções aos dispositivos ZR, de forma a obter um melhor controlo e uma maior eficiência do equipamento de refrigeração. Uma função a acrescentar era o controlo mais eficiente da válvula de expansão. Esta válvula poderia ter várias posições de trabalho, desde fechada até aberta ao máximo, e desta forma, seria possível ter a válvula mais aberta quando a temperatura do equipamento de refrigeração fosse mais elevada, permitindo a passagem de uma maior quantidade de gás refrigerante, ou quase fechada quando o equipamento de refrigeração estivesse próximo de atingir a temperatura de paragem.

REFERÊNCIAS

- [1] IEEE Std 802.11 2012, "IEEE Standard for Information technology-- Telecommunications and information exchange between systems Local and metropolitan area networks--Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications".
- [2] Patrick Kinney, "ZigBee Technology: Wireless Control that Simply Works", Communications Design Conference, October 2003.
- [3] Luís M. Borges, Fernando J. Velez, António S. Lebres, "Survey on the Characterization and Classification of Wireless Sensor Network Applications", IEEE Communications Surveys & Tutorials, Vol. 16, No. 4, 2014.
- [4] H. F. López, J. A. Afonso, J. H. Correia, R. Simões, "HM4All: A Vital Signs Monitoring System based in Spatially Distributed ZigBee Networks", Proceedings of 4th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth 2010), Munique, Alemanha, March 2010.
- [5] O. Gama, P. Carvalho, J. A. Afonso, P. M. Mendes, "Quality of Service in Wireless e-Emergency: Main Issues and a Case-study", 3rd Symposium of Ubiquitous Computing and Ambient Intelligence, 2009.
- [6] Yiming, Zhou, et al. "A design of greenhouse monitoring & control system based on ZigBee wireless sensor network." IEEE International Conference on Wireless Communications, Networking and Mobile Computing (WiCom 2007), 2007.
- [7] Diogo S. Oliveira, José A. Afonso, "Mobile Sensing System for Georeferenced Performance Monitoring in Cycling", Proceedings of the World Congress on Engineering (WCE 2015), Londres, Reino Unido, July 2015.
- [8] Vehbi C. Gungor, Gerhard P. Hancke, "Industrial Wireless Sensor Networks: Challenges, Design Principles, and Technical Approaches", IEEE Transactions on Industrial Electronics, Vol. 56, No. 10, October 2009.

- [9] José A. Afonso, Filipe Rodrigues, Pedro Pereira, Henrique Gonçalves, João L. Afonso, "Wireless Monitoring and Management of Energy Consumption and Power Quality Events," Proceedings of the World Congress on Engineering (WCE 2015), Londres, Reino Unido, July de 2015.
- [10] Ian F. Akyildiz and Mehmet Can Vuran, "Wireless Sensor Networks", John Wiley & Sons, 2010
- [11] S. S. Oliveira and S. Motoyama, "Applications Oriented Medium Access Control Protocols for Wireless Sensor Networks", IEEE Latin America Transactions, Vol. 7, No. 5, September 2009.
- [12] H. Karl and A. Willig, "Protocols and Architectures for Wireless Sensor Networks", John Wiley & Sons, 2005.
- [13] IEEE Std 802.15.4-2011, "Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)", September 2011.
- [14] Ian F. Akyildiz et al., "A Survey on Wireless Mesh Networks", IEEE Radio Communications, September 2010.
- [15] Peter Corke et al., "Environmental Wireless Sensor Networks", Proceedings of the IEEE, Vol. 98, No. 11, November 2010.
- [16] P. Macedo, J. A. Afonso, "Simulation Analysis of IEEE 802.15.4 for Wireless Networked Control Systems", Proceedings of 35th Annual Conference of the IEEE Industrial Electronics Society (IECON 2009), Porto, Portugal, November 2009.
- [17] ZigBee Standards Organization, "ZigBee Specification", Technical Report, January 2008.
- [18] Drew Gislason, "ZigBee Wireless Networking", Newnes, 2008.
- [19] IEEE Std 802.15.4-2006 – Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs), September 2006.
- [20] Echelon, "Introduction to the LonWorks Platform", Revision 2, 2009.

- [21] Echelon, LonTalk Protocol Specification, version 3.0, 1994.
- [22] Danfoss, “Controller for Appliance Control AK-CC 550”, Manual, February 2010.
- [23] Danfoss, “Capacity Controller AK-PC 530”, Manual, May 2010.
- [24] Danfoss, “Temperature Sensor type EKS 111”, Technical Brochure, June 2012.
- [25] Danfoss, “Electrically Operated Expansion Valve Types AKV 10”, Datasheet, December 2012.
- [26] Danfoss, “System Manager AK-SM 850”, Quick Setup Guide, July 2013.
- [27] CC2530 Datasheet, “CC253x System-on-Chip Solution for 2.4-GHz IEEE 802.15.4 and ZigBee Applications”, Texas Instruments, September 2010.
- [28] SmartRF05EB Datasheet, “Z-Stack User’s Guide for SmartRF05EB and CC2530”, Texas Instruments, 2010.