

UNIVERSIDADE DO EXTREMO SUL CATARINENSE - UNESC

CURSO DE CIÊNCIA DA COMPUTAÇÃO

THALES FERNANDES MAGGI

MONITORAMENTO DE SINAIS VITAIS UTILIZANDO RSSF

CRICIÚMA

2015

THALES FERNANDES MAGGI

MONITORAMENTO DE SINAIS VITAIS UTILIZANDO RSSF

Trabalho de Conclusão de Curso, apresentado para obtenção do grau de bacharel no curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC.

Orientador: Prof. MSc. Rogério Antônio Casagrande

CRICIÚMA

2015

THALES FERNANDES MAGGI

MONITORAMENTO DE SINAIS VITAIS UTILIZANDO RSSF

Trabalho de Conclusão de Curso aprovado pela Banca Examinadora para obtenção do Grau de Bacharel, no Curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC, com Linha de Pesquisa em redes de computadores.

Criciúma, 22 de junho de 2015

BANCA EXAMINADORA

Prof. Rogério Antônio Casagrande - MSc - (UNESC) - Orientador




Prof. Paulo João Martins - MSc - (UNESC)



Prof. Sérgio Corral - Especialista - (UNESC)

**A todos que influenciaram positivamente na
conclusão deste trabalho.**

AGRADECIMENTOS

A meus pais primeiramente, Ivan e Elizete, que sempre me apoiaram, ajudaram e acreditaram em mim.

A minha irmã, que é uma das pessoas mais fortes que eu conheço e que sempre conversamos sobre tudo, ajudando a ambos. E, a minha sobrinha, que só de lembrar o sorriso dela já me dá forças.

Aos meus amigos que me ajudaram de inúmeras formas, desde conversas sérias, até as mais banais, mas sempre transmitindo força e fé.

E mais importante ainda aos professores Paulo Martins e Rogério Casagrande e demais colaboradores do projeto de pesquisa e extensão cedido pela universidade que abriu a porta para a conclusão deste trabalho.

Agradeço a todos do fundo do coração!

**“Os seus problemas você deve esquecer,
isso é viver, é aprender, Hakuna Matata”**

Hakuna Matata – Rei Leão

RESUMO

Redes de sensores sem fio estão evoluindo muito ultimamente, amadurecendo suficientemente para ser utilizado eficientemente em projetos e trabalhos que podem gerar resultados muito bons no futuro em qualquer área, tanto na área da saúde como no militar, setor agrícola, escolar ou setor de segurança civil. Ela está sendo utilizada em projetos muito interessantes, como monitoramento de umidade, temperatura do solo de áreas extensas, plantas em uma estufa ou controlar uma produção de algum alimento em cada parte do processo. Este trabalho apresenta um protótipo de uma rede de sensores sem fio para monitorar sinais vitais utilizando a pilha de protocolos de comunicação ZigBee. Para visualização dos dados foi desenvolvida uma aplicação no sistema operacional móvel Android e uma versão desktop. Foi escolhido o ZigBee pois este foi criado exatamente para situações onde há restrições de consumo de energia, sendo neste caso um requisito para os nós que ficaram dispostos em uma pessoa, enviando dados dos sensores para a aplicação no computador e depois reenviado para um smartphone via Bluetooth. O resultado deste trabalho atingiu seu objetivo, que era construir uma solução genérica para diversas aplicações, como monitorar idosos em um asilo, ou pacientes em hospitais ou ambulatórios.

Palavras-chave: Redes de Sensores Sem Fio. RSSF. ZigBee. Wireless Sensor Network. WSN. Sinais Vitais.

ABSTRACT

Wireless sensor networks are evolving a lot lately, sufficiently mature to be used efficiently in projects and jobs that can generate very good results in the future in any area, both in health and in the military, agriculture, academic or civil security sector . It is being used in very interesting projects, such as the moisture monitoring, soil temperature of large areas, plants in a greenhouse or control a production of some food in every part of the process. This research presents a prototype of a wireless sensor network for monitoring vital signs using the ZigBee protocol stack communication. For visualization of data, an application on Android mobile and a desktop version was developed. ZigBee was chosen because it is designed exactly to situations where power consumption need to be really low, in which case a requirement for nodes that were arranged in a person, sending data from the sensors to the application on the computer and then returned to a smartphone via Bluetooth. The result of this work has reached its goal, which was to build a generic solution for various applications, such as monitoring the elderly in a nursing home, or patients in hospitals or clinics.

Keywords: Wireless Sensor Networks. WSN. ZigBee. Physiological parameters.

LISTA DE ILUSTRAÇÕES

Figura 1 - O espectro eletromagnético e a maneira como ele é usado na comunicação	17
Figura 2 - Padrões globais de redes sem fio.....	18
Figura 3 - Funcionamento da transferência em FHSS.....	20
Figura 4 - Cenários com um coordenador (esquerda) e com vários coordenadores (direita).....	23
Figura 5 – Sensor de oxímetro de pulso.....	27
Figura 6 - Arduino com sensores captando valores e mostrando na tela de cristal líquido.....	30
Figura 7 - Estrutura da plataforma arduino UNO.....	31
Figura 8 - Pilha de protocolos ZigBee.....	33
Figura 9 - Topologias e distribuição dos dispositivos da rede ZigBee.....	36
Figura 10 - Arquitetura do Sistema Operacional Android	41
Figura 11 – Cenário com todos os hardwares e comunicação utilizados.....	48
Figura 12 – <i>Shields</i> montados.....	48
Figura 13 – Shield E-Health com os locais a serem medidos	51
Figura 14 – Função de conversão para temperatura, variáveis a serem alteradas após medição.....	52
Figura 15 – Recepção dos dados da RSSF	53
Figura 16 – Conexão Bluetooth Desktop x Dispositivo móvel	54
Figura 17 – Aplicação protótipo Android lendo as informações da rede Bluetooth ..	54
Figura 18 – Posição dos nós comparadas para testar os pacotes até o nó coordenador.....	55

LISTA DE TABELAS

Tabela 1 – Especificações das licenças ISM utilizadas pelo protocolo IEEE 802.15.4	34
Tabela 2 – Configuração das antenas Xbee	50
Tabela 3 – Posição do corpo e respectivos códigos	52
Tabela 4 – Valores obtidos antes da calibração	56
Tabela 5 – Valores obtidos depois da calibração	56

LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>
BAN	<i>Body Area Network</i>
BPSK	<i>Binary Phase Shift Keying</i>
CPU	<i>Central Processing Unit</i>
CSMA/CD	<i>Carrier Sense Multiple Access with Collision Detection</i>
DSSS	<i>Direct Sequence Spread Spectrum</i>
ECG	<i>Eletrocardiograma</i>
EEG	<i>Eletroencefalograma</i>
EMG	<i>Eletromiografia</i>
EPROM	<i>Erasable Programmable Read Only Memory</i>
FBSN	<i>Floating Base Sensor Network</i>
FFD	<i>Full Function Device</i>
FHSS	<i>Frequency Hopping Spread Spectrum</i>
GPS	<i>Global Positioning System</i>
IDE	<i>Integrated Development Enviroment</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
ISM	<i>Industrial, Scientific & Medical</i>
LCD	<i>Liquid Crystal Display</i>
LR-WPAN	<i>Low-rate Wireless Personal Area Network</i>
MAC	<i>Medium Access Control</i>
MANET	<i>Mobile Ad hoc Network</i>
MEMS	<i>Micro Electro-Mecanical Systems</i>
OHA	<i>Open Handset Alliance</i>
O-QPSK	<i>Offset Quadrature Phase Shift Keying</i>
OSI	<i>Open Systems Interconnection</i>
PHY	<i>Physical Layer</i>
PIC	<i>Peripheral Interface Controller</i>
RFD	<i>Reduced Function Device</i>
RISC	<i>Reduced Instruction Set Computers</i>
RSSF	<i>Redes de Sensores Sem Fio</i>
SDK	<i>Software Development Kit</i>
SQL	<i>Structured Query Language</i>

USB	<i>Universal Serial Bus</i>
VM	<i>Virtual Machine</i>
WBAN	<i>Wireless Body Area Network</i>
WLAN	<i>Wireless Local Area Network</i>
WMAN	<i>Wireless Metropolitan Area Network</i>
WPAN	<i>Wireless Personal Area Network</i>
WWAN	<i>Wireless Wide Area Network</i>

SUMÁRIO

1 INTRODUÇÃO	11
1.1 OBJETIVO GERAL	13
1.2 OBJETIVOS ESPECÍFICOS	13
1.3 JUSTIFICATIVA	13
1.4 ESTRUTURA DO TRABALHO	14
2 CONEXÕES SEM FIO	16
2.1 FUNDAMENTOS SOBRE FREQUÊNCIA	16
2.2 TIPOS DE REDES	17
2.2.1 WPAN	18
2.2.2 WBAN	21
3 REDES DE SENSORES SEM FIO	22
3.1 FUNDAMENTOS DE UMA RSSF	22
3.2 CARACTERÍSTICAS DE UMA RSSF	23
3.3 SENSORES	24
3.3.1 Sensores Fisiológicos	25
3.4 MICROCONTROLADOR	27
3.5 ARDUINO	29
3.5.1 Shields	31
3.6 RESTRIÇÕES	32
3.7 CAMADAS DO PROTOCOLO IEEE 802.15.4 E ZIGBEE	33
3.7.1 Camada Física	33
3.7.2 Camada de Enlace	34
3.7.3 Camadas ZigBee Alliance	35
4 SISTEMA OPERACIONAL MÓVEL ANDROID	39
4.1 HISTÓRIA	39
4.2 CARACTERÍSTICAS	40
4.3 ARQUITETURA	41
4.3.1 Núcleo	42
4.3.2 Bibliotecas	42
4.3.3 Android RunTime	42
4.3.4 Framework de Aplicação	43
4.3.5 Camada de Aplicação	43
5 TRABALHOS CORRELATOS	44

5.1 RASTREABILIDADE BOVINA: UTILIZANDO PLATAFORMA ZIGBEE	44
5.2 DESENVOLVIMENTO DE UMA REDE DE SENSORES SEM FIO UTILIZANDO <i>ZIGBEE</i> PARA APLICAÇÕES DIVERSAS	44
5.3 REDES DE SENSORES SEM FIO NA COLETA DE DADOS FISIOLÓGICOS DE BOVINOS PARA APLICAÇÕES NA ZOOTECNIA DE PRECISÃO	45
5.4 <i>WIRELESS SENSOR NETWORKS FOR BIOMEDICAL APPLICATIONS</i>	45
6 UTILIZANDO A RSSF PARA MONITORAR SINAIS VITAIS	46
6.1 Metodologia	46
6.1.1 Escolha do Hardware	47
6.1.2 Softwares utilizados	49
6.1.3 Integração Hardware e Software	50
6.1.4 Recepção dos dados	53
6.2 RESULTADOS OBTIDOS	55
7 CONCLUSÃO	58
REFERÊNCIAS	60

1 INTRODUÇÃO

Hoje em dia o mundo passa por vários problemas ligados diretamente a uma sociedade mais velha, com predominância maior de doenças crônicas e dificuldades associadas a mobilidade. O conceito da computação ubíqua (onipresença da informática no cotidiano das pessoas) traz inovações para aplicar a computação no dia-a-dia das pessoas de forma imperceptível. Para isso, utilizando da união de diversas tecnologias já existentes como, por exemplo, a tecnologia móvel e os sensores (MOREIRA, 2011).

Com o avanço do estudo e uso na área de microprocessadores, descoberta de novos materiais de sensoriamento, micro sistemas eletromecânicos, do inglês *Micro Electro-Mechanical Systems* (MEMS), e comunicação sem fio entre dispositivos pequenos tem estimulado muitos pesquisadores e empresas no desenvolvimento e uso destes sensores “inteligentes” em áreas ligadas a processos físicos, químicos, biológicos, dentre outros (LOUREIRO et al, 2003).

Segundo Honorato (2011) os sensores são compostos normalmente de processador, modem wireless, memória e fonte de alimentação. Os nós ou sensores de uma rede são limitados em relação ao consumo de energia, capacidade de processamento computacional e memória, enquanto que os nós das redes ad hoc (é uma rede sem fio em que todos dispositivos comunicam entre si, sem um ponto para se conectarem como uma rede wireless convencional onde todos dispositivos se conectam a um roteador) normalmente são mais robustos. Estes sensores quando utilizados em uma rede, formam uma rede de sensores sem fio, ou RSSF, e, normalmente elas possuem um grande número de nós. Cada nó é equipado com uma variedade de sensores, tais como acústico, sísmico, infravermelho, câmera de vídeo, temperatura e pressão. Esses nós podem ser organizados em grupos onde pelo menos um dos sensores deve ser capaz de detectar um evento na região, processá-lo e tomar uma decisão se deve fazer ou não uma difusão (*broadcast*) do resultado para outros nós.

As RSSFs podem ser vistas como um tipo especial de Rede Móvel ad hoc, do inglês *Mobile Ad hoc Network* (MANET), No entanto, as redes MANET tem como função básica prover um suporte a comunicação entre esses elementos computacionais, que individualmente, podem estar executando tarefas distintas. Por

outro lado, RSSF tendem a executar uma função colaborativa onde os elementos (sensores) provêm dados, que são processados (ou consumidos) por nós especiais chamados de coordenadores (*sink nodes*) (LOUREIRO et al, 2003).

RSSF utilizam dispositivos com protocolos de comunicação, que, foi definido pelo Instituto de Engenheiros Eletricistas e Eletrônicos, conhecido como IEEE. Eles definiram um padrão para este tipo de rede, o IEEE 802.15.4, um padrão que define um conjunto de protocolos de comunicação para redes de sensor sem fio, que são de custo energético muito baixo e baixa taxa de transmissão. E com esta característica, uma empresa trabalhou em cima de alguns protocolos deste padrão, e criaram o ZigBee, que, tem sido amplamente utilizado desde então em várias aplicações que necessitam de redes de sensores sem fio (HONORATO, 2011).

Nesta arquitetura ZigBee, o processamento dos dados primário é feito por nós sensores, incluindo o processamento do sinal fisiológico no microcontrolador dos nós. O processamento de dados secundários é executado no *smartphone*. Isso inclui a representação de dados, filtragem de dados, interface gráfica e sincronização de dados (WAGNER et al, 2012, tradução nossa).

Os dados que os nós sensores captarem são passados pela rede de sensores até um nó coordenador, e dele para um computador, e deste para um dispositivo Android. Os dados são transmitidos para o Smartphone via comunicação Bluetooth, que, conforme Loureiro (2003), é uma tecnologia de baixo custo para conectividade sem fio de dispositivos eletrônicos mundialmente utilizado. Inicialmente pensado como um padrão de substituição de cabos para comunicação entre dispositivos eletrônicos, ele se tornou um consenso na indústria como uma forma de interligar dispositivos como telefones celulares, notebooks, computadores de mesa e impressoras.

Conforme mostrado acima, a ideia seria fazer a aquisição de sinais vitais como temperatura e batimentos cardíacos de um ser humano via *smartphone*, utilizando redes de sensores sem fio transmitindo estes dados para um dispositivo móvel. A comunicação entre os sensores até o nó coordenador é realizada por um protocolo de comunicação especializado em RSSF, e depois transmitido para o dispositivo via Bluetooth em um ambiente local de curto alcance.

1.1 OBJETIVO GERAL

Desenvolver uma aplicação móvel para monitorar os sinais vitais de pessoas utilizando uma rede de sensores sem fio em ambiente fechado.

1.2 OBJETIVOS ESPECÍFICOS

Como objetivos específicos desta pesquisa, pretende-se:

- a) compreender o funcionamento de redes de sensores sem fio;
- b) compreender os protocolos de comunicação *ZigBee*;
- c) compreender a comunicação Bluetooth;
- d) estudar e aplicar a linguagem de programação Android;
- e) pesquisar por melhores pontos do corpo humano para obtenção de parâmetros fisiológicos com maior precisão;
- f) implementar uma *Wireless Body Area Network* (WBAN) utilizando os conhecimentos obtidos sobre redes de sensores sem fio e protocolos de comunicação ZigBee;
- g) desenvolver uma aplicação para monitoramento de sinais vitais na plataforma móvel Android que utilize conexão Bluetooth;
- h) aplicar testes de validação de dados na aplicação.

1.3 JUSTIFICATIVA

Tecnologias de rede de sensores sem fio tem o potencial de mudar a maneira de viver com muitas aplicações em entretenimento, viagens, varejo, indústria, medicina, cuidados de pessoas dependentes, e gestão de emergência e muitas outras áreas. Um dos maiores desafios do mundo nas últimas décadas tem sido o aumento da população idosa contínua nos países desenvolvidos. O site *Population Reference Bureau* prevê que nos próximos 20 anos, a população com mais 65 anos serão nos países desenvolvidos quase 20% da população total. (ALEMDAR; ERSOY; 2010, tradução nossa).

Conforme Akyildiz et al (2003), as tecnologias de comunicação sem fio e aparelhos eletrônicos digitais permitiram o desenvolvimento de baixo custo, baixo consumo de energia, nós sensores multifuncionais que são pequenos em tamanho e se comunicam sem fio em curtas distâncias. Estes minúsculos nós sensores, que consistem em sensoriamento, processamento de dados e componentes de comunicação, alavancam a ideia de redes de sensores com base na colaboração de um grande número de nós. Uma RSSF adaptada para uma tarefa específica. Neste caso, como é uma rede de sensores em seres humanos, que vem do inglês *Body Area Network (BAN)*, é geralmente chamado de *wireless BAN* ou *WBAN*.

O ZigBee tem suporte de transmissão via Bluetooth, que também opera em transmissões de dados em curto alcance para dispositivos eletrônicos que tem suporte de conexão Bluetooth. Os dados dos sensores são transmitidos para um nó central, o nó coordenador e depois são retransmitidos para um dispositivo móvel com Sistema Operacional (SO) móvel Android.

Esta tecnologia agregada a saúde trás muitos pontos positivos como conforto, mobilidade, baixo custo, monitoramento de vários pacientes em um dispositivo único e prever futuros problemas de saúde momentâneos, pois caso a leitura dos sinais vitais caia ou aumente subitamente será alertado o paciente com uma notificação no *smartphone* para que vá a algum centro médico. O sistema operacional móvel Android usa linguagem Java que é mundialmente conhecida e utilizada, e, sua plataforma de desenvolvimento tem uma interface intuitiva e fácil de programar o que torna o desenvolvimento de uma aplicação móvel mais ágil e simples.

1.4 ESTRUTURA DO TRABALHO

O presente trabalho é composto por 6 capítulos. O primeiro capítulo é apresentado uma introdução ao tema e o problema que ele pretende solucionar, os objetivos gerais e específicos.

O segundo capítulo aborda o assunto de conexões sem fio, explicando seus fundamentos, história e de onde começou a comunicação sem fio, os tipos de redes sem fio, suas diferenças e suas funções.

O terceiro capítulo é sobre as redes de sensores sem fio, apontando seus fundamentos, características, topologias, explicando camada por camada do protocolo de comunicação IEEE 802.15.4 e quais camadas foram adaptadas para criação do ZigBee.

O quarto capítulo é apresentado o sistema operacional móvel Android, falando sobre sua história, suas características e principalmente sua arquitetura.

No quinto capítulo expõe-se trabalhos correlatos ao trabalho, proporcionando visões diferenciadas e diversificadas de se implementar uma rede de sensor sem fio.

Já no sexto capítulo se encontra o trabalho desenvolvido, onde mostra como e com o que foi feito este trabalho, os testes e resultados obtidos.

E por fim, tem-se a conclusão deste trabalho, onde é feita as considerações finais e propostas para trabalhos futuros.

2 CONEXÕES SEM FIO

Com a evolução da tecnologia *wireless* (sem fio), das redes e dos *smartphones*, viram-se obrigados a criar padrões e especificações para redes pessoais locais para se adaptar a mobilidade e necessidades que os smartphones necessitavam. Segundo Tanenbaum (2001), a comunicação digital sem fios começou em 1901 pelo físico italiano Guglielmo Marconi que demonstrou como funcionava um telégrafo sem fio que transmitia informações de um navio para o litoral por meio de ondas eletromagnéticas, fazendo bipes e então se comunicando por código morse.

A tecnologia wireless descreve os sistemas de telecomunicação em que as ondas eletromagnéticas carregam o sinal sobre parte ou todo o trajeto de comunicação sem a utilização de cabos (JINDAL; JINDAL; GUPTA, 2005, tradução nossa).

São dispositivos interconectados sem a utilização de fios ou cabos, somente pelo “ar”, utilizando-se da rádio frequência, e, para isso foram estudados e adaptados padrões de transmissão para garantir uma conexão forte e estável o suficiente para que não possuísse uma alta perda de pacotes na hora da comunicação dos dados (TANEMBAUM, 2001).

2.1 FUNDAMENTOS SOBRE FREQUÊNCIA

Sinais de rádio que são utilizados hoje por muitos periféricos como televisão, rádio e Internet e trazem muitas informações são passadas via este sinal. Conforme Coleman e Westcott (2011) a frequência é descrita como uma forma de onda de um sinal, e a potência dele define o quão longe ele irá, lembrando que ele é lançado para todos os lados a partir da fonte, ou seja, é omnidirecional. Quanto maior a frequência menor será a distância que o sinal cobrirá. A fórmula a seguir exemplifica esta afirmação:

$$PS = 32.4 + (20 \log D) + (20 \log F)$$

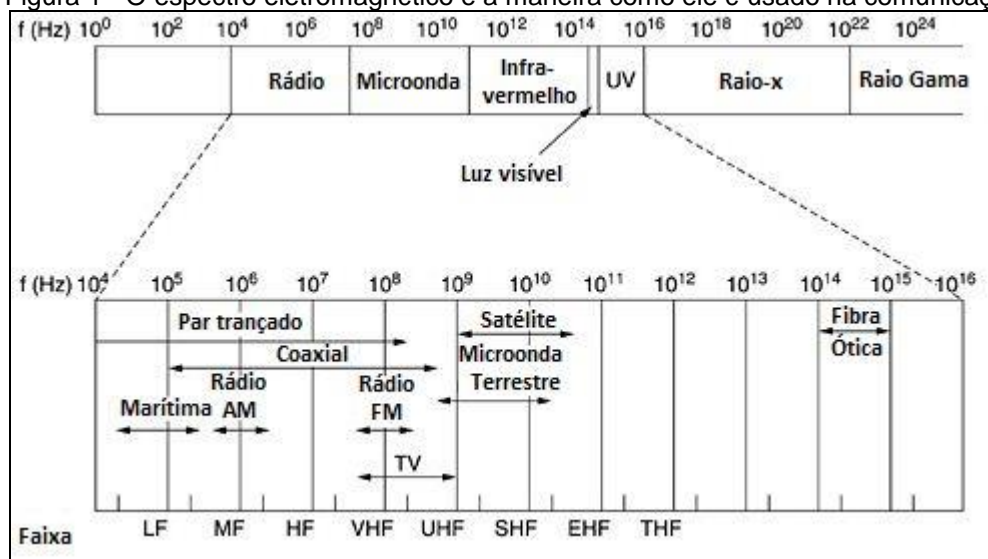
Onde:

- a) PS = perda do sinal;
- b) D = distância em quilômetros;
- c) F = frequência em MHz.

Para organizar e identificar uma radiofrequência ela é dividida em faixas que são espaços reservados para algum determinado serviço, definido por convenções internacionais e órgãos reguladores como o *Institute of Electrical and Electronics Engineers*, IEEE, e as faixas mais utilizadas em redes sem fio são 2.4GHz e 5.8GHz, que se subdividem em faixas menores. Estas subdivisões de faixas são chamadas de canais e permitem uma transmissão paralela em cada uma delas, que é algo muito conhecido na sociedade que são os canais de rádio e de televisão (RUFINO, 2011).

Na figura 1 pode-se ver como o espectro de frequências eletromagnéticas é dividido e utilizado nas comunicações.

Figura 1 - O espectro eletromagnético e a maneira como ele é usado na comunicação



Fonte: Adaptado de Tanenbaum (2001).

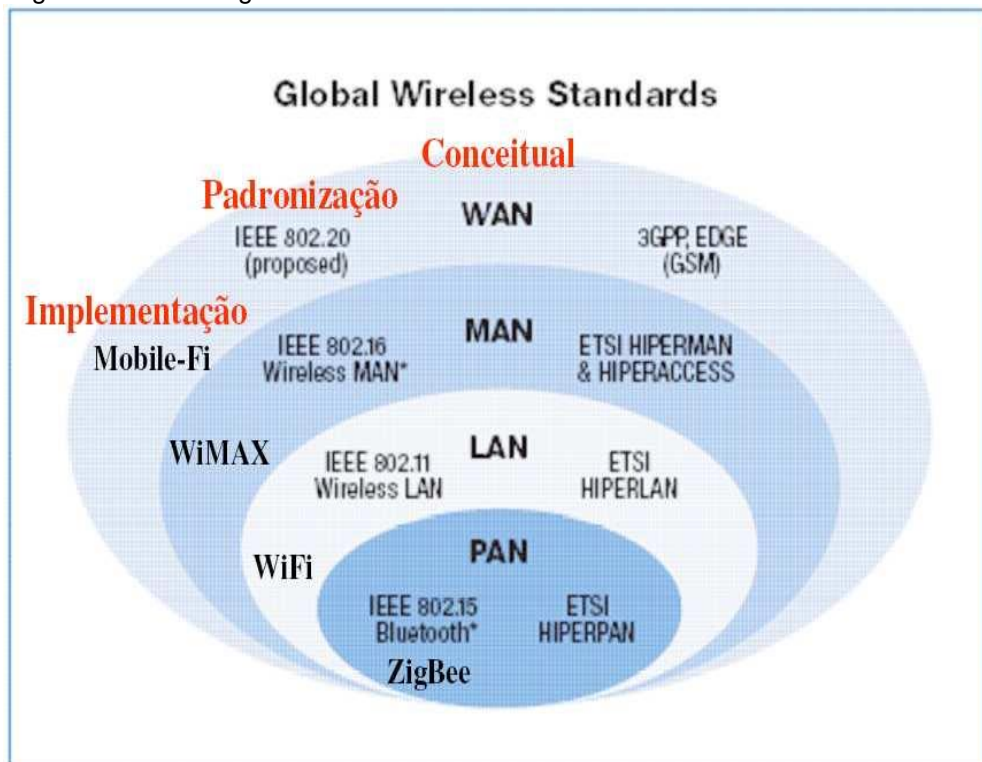
2.2 TIPOS DE REDES

A IEEE padronizou quatro tipos de redes de comunicação sem fio mais abrangentes, e delas existem algumas subdivisões para algumas situações muito específicas onde uma subdivisão da *Wireless Personal Area Network* (WPAN) fará

parte deste trabalho e será descrito posteriormente.

Tanenbaum (2001) diz que os quatro tipos de redes são WPAN, que são redes pessoais de curto alcance, utilizando padrão IEEE 802.15.x, sendo que o valor x define a qual grupo de trabalho se encontra, e é o padrão que será abordado neste trabalho. *Wireless Local Area Network (WLAN)*, uma rede local sem fio, que utiliza o padrão IEEE 802.11x, que são aquelas convencionais, de fácil configuração e conectividade. *Wireless Metropolitan Area Network (WMAN)*, que é uma rede sem fio metropolitana, utilizando padrão IEEE 802.16x, utilizado em metrópoles onde necessita de que a mesma rede esteja em muitos locais não muito distantes. *Wireless Wide Area Network (WWAN)* fornecem redes wireless de grande alcance, utilizado normalmente por empresas de telecomunicações. A figura 2 exemplifica a estrutura dos padrões globais de redes sem fio.

Figura 2 - Padrões globais de redes sem fio.



Fonte: Branquinho et al (2005).

2.2.1 WPAN

Redes de curto alcance sem fio, como dito anteriormente, estão sendo amplamente utilizados e melhoram a qualidade de vida das pessoas já que os

dispositivos controlam e ajudam a ter uma vida melhor, facilitando tarefas que necessitavam de aparelhos enormes e que te prendiam há um local.

Uma publicação da IEEE de 2006, específica sobre o protocolo 802.15, uma rede sem fio pessoal de baixa potência, em inglês *Low-rate Wireless Personal Area Network (LR-WPAN* ou só *WPAN*), seria uma forma de se comunicar facilmente entre aplicações a curta distância com pouco custo de energia, sendo de fácil instalação e configuração, funcionando na banda ISM (*Industrial, Scientific & Medical*), que opera nas faixas de 2,4GHz até 2,5, que é uma banda padronizada mundialmente para estes dispositivos que fazem conexões de curtas distâncias, e tem uma transferência de dados segura.

Exemplos de protocolos de comunicação LR-WPAN são o Bluetooth e o ZigBee, que serão detalhados a seguir.

2.2.1.1 Bluetooth – IEEE 802.15.1

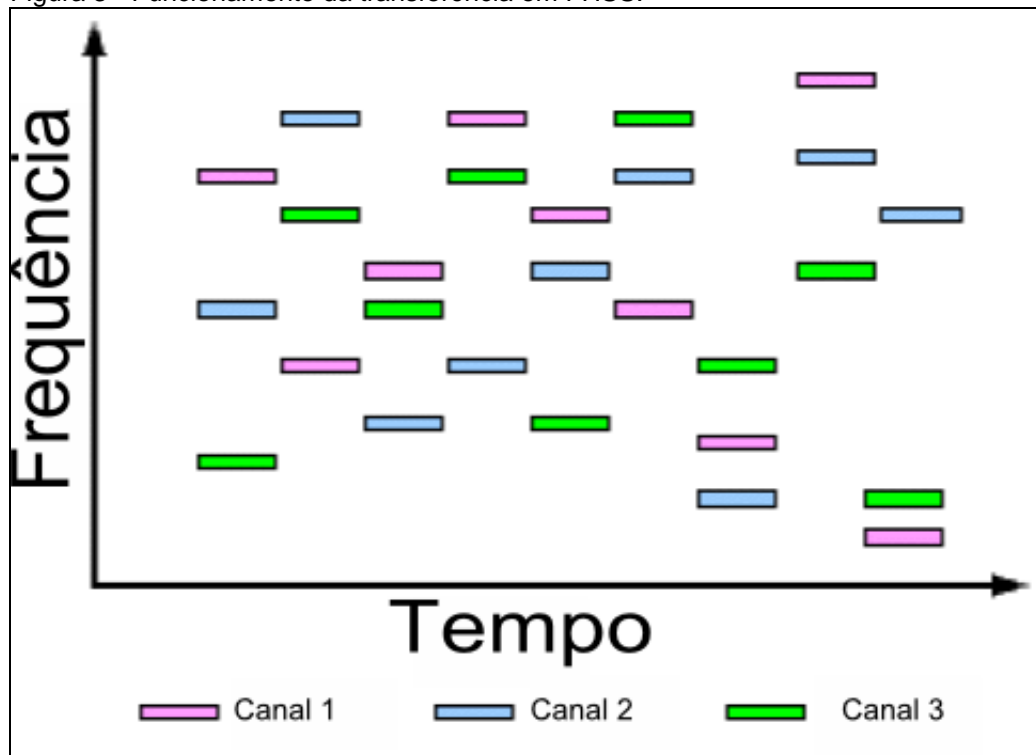
O Bluetooth começou em 1998, desenvolvido por um grupo com o objetivo de interligar aparelhos eletrônicos pessoais de baixo custo e com baixo consumo de energia. E se tornou muito popular pelo fato de ser fácil e rápida a conexão feita entre eles para transferência ou manipulação de outros dispositivos (RAPPAPORT, 2009).

O Bluetooth utiliza-se do padrão mestre-escravo nas comunicações entre os dispositivos, sendo que um mestre pode ter até sete escravos, o que forma uma rede *piconet*, onde a partir de 2 dispositivos conectados na hora de trocar informações elegem dinamicamente um dispositivo para ser o mestre e coordenar a sincronização com os escravos e se um escravo estiver conectado a outra rede ao mesmo tempo, forma então uma *scatternet* (TANEMBAUM, 2001).

Conforme Vicente (2010), para a comunicação entre dispositivos, utiliza-se a faixa de banda ISM a seu modo. No protocolo Bluetooth, as mensagens a serem enviadas dividem-se em vários pacotes e são transmitidas. O primeiro pacote é enviado em uma determinada frequência, então, o protocolo seleciona um novo canal para o envio do próximo pacote. Desse modo, todos os dados são enviados utilizando um espalhamento uniforme pelo espectro, conhecido como *Frequency Hopping Spread Spectrum (FHSS)*, que é a diminuição de interferência com outros

tipos de comunicação, utilizando-se da troca de canais rapidamente, diminuindo-se a chance de colisão na hora da comunicação Bluetooth como mostra na figura 3.

Figura 3 - Funcionamento da transferência em FHSS.



Fonte: Adaptado de Rahamnah, Qaisi e Alhousban (2013, tradução nossa).

2.2.1.2 ZigBee - IEEE 802.15.4

ZigBee é um padrão que define um conjunto de protocolos de comunicação sem fio onde sua taxa de transmissão máxima é de 250 Kbps e é de curto alcance. O padrão opera na frequência de 2,4 GHz e seu nome ZigBee vem do comportamento das abelhas que voam em zigue zague trocando informações sobre distância, direção e localização de onde encontrar alimentos (SILVA, 2011).

Ele foi desenvolvido por um grupo de empresas que se juntaram e formaram a *ZigBee Alliance*. O propósito destas empresas era descrever um padrão da terceira camada do modelo *Open Systems Interconnection* (OSI), a camada de rede do protocolo IEEE 802.15.4 de maneira que os fabricantes possam produzir

dispositivos compatíveis entre si. Ou seja, produtos de fabricantes diferentes que implementem o padrão *ZigBee* são compatíveis. Ele é implementado sobre o padrão IEEE 802.15.4, o qual define as camadas física *Physical Layer (PHY)* e de enlace *Medium Access Control (MAC)*, que será falado mais profundamente no próximo capítulo, sobre redes de sensores sem fio (AHAMED, 2009, tradução nossa).

2.2.2 WBAN

Ela é uma subdivisão da WPAN, a *Wireless Body Area Network (WBAN)*, significa uma rede corporal sem fio, a WBAN fornece uma interligação mais estreita (entre 2-5 metros), com requisitos técnicos mais rigorosos, como a alta confiabilidade, eficiência de energia extrema e segurança, especialmente a segurança para o corpo humano utilizando uma tecnologia vestível (LI; LI; KOHNO, 2009, tradução nossa). Ele utiliza o protocolo IEEE 802.15 *Task Group 6*, e foi desenvolvida e adaptada para monitorar o corpo internamente e/ou externamente. Formada por sensores pequenos que não possuem muita energia, e este padrão não se atém somente a seres humanos, qualquer animal pode ser adaptado uma WBAN (IEEE *Task Group 6*, 2010, tradução nossa).

3 REDES DE SENSORES SEM FIO

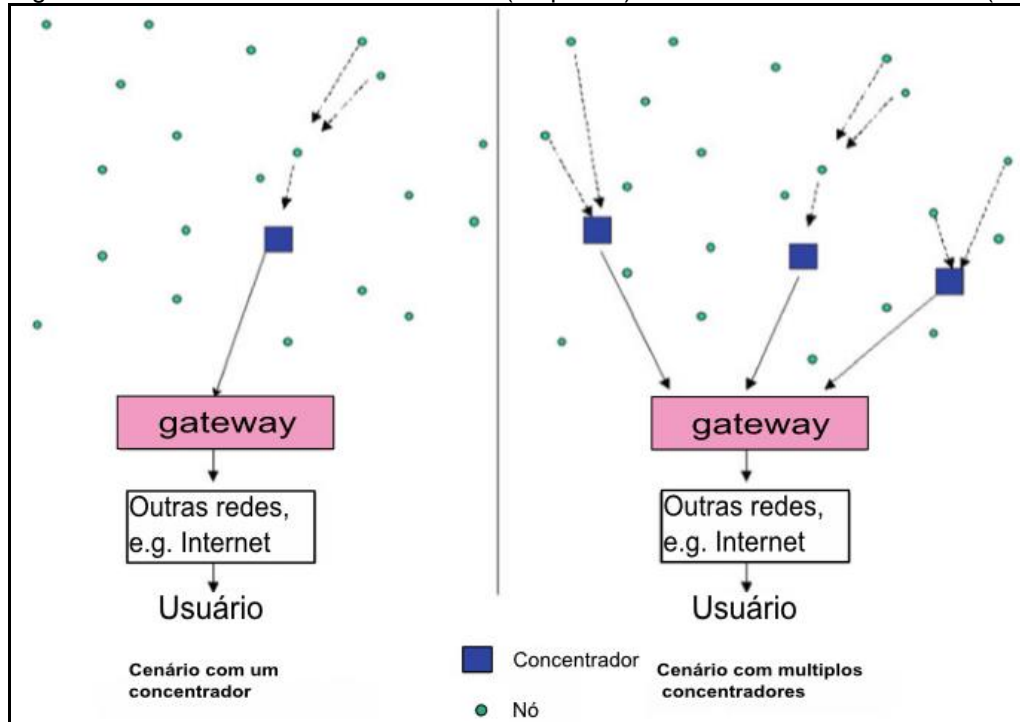
Redes de Sensores Sem Fio (RSSF) estão crescendo muito por causa da evolução dos microprocessadores e micro sistemas eletromecânicos que facilitam no desenvolvimento de sensores mais inteligentes e menores que ajudam pesquisadores e empresas a monitorar setores, pessoas, áreas, processos químicos, entre outros. Estes sensores apresentam características como precisão, baixo consumo de energia a capacidade de transmitir tudo pela radiofrequência, ou seja, sem fio, facilitando o tratamento e dependendo da situação, dando segurança pois pode coletar informações monitorando tudo à distância (YICK; MUKHERJEE; GHOSAL, 2008, tradução nossa).

3.1 FUNDAMENTOS DE UMA RSSF

Uma RSSF pode ser definida como um conjunto de nós capazes de comunicar a informação obtida de uma área monitorada através de um conjunto de nós sensores interconectados sem fio. Essa informação é transmitida de nó em nó até chegar a um nó coordenador (também conhecido em inglês como *sink node*, em português, nó sorvedouro ou concentrador) que pode localmente se utilizar das informações coletadas, ou, estar conectada a outra rede, tal como Internet, Bluetooth, Satélite, entre outros através de um *gateway* para oferecer acesso às informações coletadas para diversos usuários (BURATTI et al, 2011, tradução nossa).

A comunicação e monitoramento são feitas cooperativamente através dos nós. Há casos onde se necessita de mais coordenadores, pois, quando há muitos nós, existem problemas de falta de escalabilidade onde um coordenador só, estaria se sobrecarregando de informações. A figura 4 exemplifica estes cenários.

Figura 4 - Cenários com um coordenador (esquerda) e com vários coordenadores (direita).



Fonte: Adaptado de Buratti et al (2011, tradução nossa).

Buratti et al (2011, tradução nossa) e Honorato (2011) e afirmam que este cenário de múltiplos coordenadores é necessário em escalas maiores e um protocolo mais complexo para tratamento de métricas de desempenho, como variação no atraso de entrega de pacotes, taxa de transferência, que são conhecidos em inglês como *jitter* e *throughput*, e número mínimo de saltos, pois corre-se o risco de os nós não conseguirem entregar a informação ao coordenador, devido as condições do método de propagação do sinal.

Como o nosso foco é em uma rede de sensores sem fio corporal, haverá poucos nós, tornando-se suficiente o uso do cenário com um coordenador.

3.2 CARACTERÍSTICAS DE UMA RSSF

As principais características da RSSF são: escalabilidade com relação ao número de nós na rede, a auto-organização, eficiência energética, conectividade suficiente entre os nós, de baixa complexidade, baixo custo e tamanho pequeno dos nós sensores (BURATTI et al, 2011, tradução nossa).

Algumas características importantes citadas por Buratti (2011):

- a) alta escalabilidade, pois são mais flexíveis que as redes de sensores que utilizam uma estrutura cabeada, já que é mais fácil inserir novos nós e não existe um custo tão alto para remover um determinado nó ou replanejar a topologia da rede;
- b) auto-organizável, deve se recuperar mesmo se um dos nós falhar, mantendo o funcionamento;
- c) conectividade entre os nós, todos os nós devem conseguir mandar seus pacotes até o coordenador;
- d) baixa complexidade dos nós na rede;
- e) baixo custo, pois se for comparar o que ele pode fazer com outros produtos existentes ele se torna mais barato, afirma Akyildiz et al (2003, tradução nossa);
- f) sensores pequenos e discretos para fácil manipulação em diversos ambientes;
- g) possibilidade de ter até 65,535 nós sensores em uma rede;
- h) conservação de energia é a característica mais importante e relevante na hora de se montar uma RSSF, poupar o máximo de energia para maior duração dos dispositivos em funcionamento;

3.3 SENSORES

Segundo Buratti et al (2011, tradução nossa) o surgimento do primeiro termostato foi em 1883 e é considerado por alguns como o primeiro sensor moderno. Inúmeras formas de sensores surgiram desde então. Nesta última década, áreas da tecnologia foram aprimoradas e permitiram evolução dos sensores. Algumas destas áreas foram, comunicação pervasiva (sem fio), conectividade com a web, dispositivos móveis inteligentes e integração com a nuvem, adicionaram imensamente para as capacidades dos sensores.

O avanço que tem ocorrido na área de microprocessadores, novos materiais de sensoriamento, micro sistemas eletromecânicos e comunicação sem fio têm estimulado o desenvolvimento e o uso de sensores inteligentes em áreas

ligadas a processos físicos, químicos, biológicos, entre outros (AKYILDIZ et al, 2003, tradução nossa).

Yick et al (2008, tradução nossa) diz que um sensor sem fio é caracterizado por seu pequeno tamanho e a sua capacidade de perceber os fenômenos ambientais através de um conjunto de transdutores (transformam o que o sensor está captando em informação elétrica, como corrente, resistência ou tensão), uma unidade de processamento e um rádio transceptor (um dispositivo que transmite e recebe informações) para comunicação, com alimentação autônoma.

Sensores atuais empregam uma arquitetura conhecida como, *Reduced Instruction Set Computers* (RISC), em português, computador com um conjunto reduzido de instruções, que são microcontroladores com um programa e capacidade de memória de dados muito pequena (cerca de 100 kb). Uma memória flash externa pode ser adicionado para fornecer armazenamento secundário e para aliviar as restrições de tamanho da aplicação impostas pela memória (BARONTI, 2007, tradução nossa).

Os sensores então percebem um fenômeno, mudando seu comportamento, traduzindo sinais físicos obtidos de um ambiente externo para grandezas elétricas, onde necessitam serem manipuladas e tratadas de acordo com sua finalidade a fim de monitorar e/ou controlar algo, o que, neste trabalho, será o corpo humano, utilizando-se sensores que captam parâmetros fisiológicos do corpo humano, como temperatura corporal, pressão arterial, pulso e oxigenação do sangue, glicose, acelerômetro, Eletrocardiograma (ECG), entre outros.

3.3.1 Sensores Fisiológicos

Monitoramento e armazenamento de parâmetros fisiológicos de pacientes fora do ambiente clínico estão se tornando cada vez mais importante na pesquisa, bem como em fisiologia aplicada e medicina em geral (WAGNER et al, 2012, tradução nossa).

Conforme Mcgrath, Scanail e Nafus (2013), quando necessário cuidado especial de um paciente, são monitoradas medidas fisiológicas como ECG, Eletromiografia (EMG) e Eletroencefalograma (EEG). O tempo de resposta destes

sensores é determinado pela velocidade à qual a quantidade e tipo de estímulos necessários são transportados para o local de medição. Por exemplo, os sensores como o ECG, que medem um sinal elétrico tem um tempo de resposta muito rápido. Já o tempo de resposta da condutância galvânica da pele é menor porque este sensor requer que suor mova-se até um eletrodo, um processo muito mais lento.

Conforto e biocompatibilidade são considerações importantes na hora de se monitorar dados corporais, por exemplo, os sensores podem causar problemas como irritação da pele quando em contato com a pele durante períodos prolongados de tempo, ou ruído sinal se mal conectado, por isso devem-se ter restrições quanto ao tamanho e seu design (MILENKOVIC; OTTO; JOVANOVIĆ, 2006, tradução nossa).

Três exemplos de sensores fisiológicos que Mcgrath, Scanail e Nafus (2013) mostram são:

- a) sensor de temperatura corporal: há vários métodos de se medir a temperatura corporal, pois, cada parte do corpo possui uma temperatura diferente, podendo haver variâncias. Alguns lugares que podem ter maior precisão, porém tem maior desconforto, como nas axilas ou oral, e alguns de mais fácil acesso como na própria mão ou peito, tem menor precisão e precisam de algoritmos para compensar esta diferença. A temperatura corporal normal de um ser humano varia em cada parte do corpo, nas axilas é entre 35,5°C a 37°C, já na oral é considerado normal estar entre 36°C e 37,4°C;
- b) sensor de oxímetro de pulso: é um método não invasivo para monitorar a saturação do oxigênio no sangue e o pulso de um paciente, pois pode ser usado no pé ou na mão. Ele é usado na saúde, em especial para avaliar pacientes com problemas respiratórios. Leituras de oxímetro de pulso normais variam de 95 a 100 por cento. Valores abaixo de 90 por cento são considerados baixos. Ele é medido utilizando duas luzes, cada uma com um comprimento de onda diferente, que saem de um lado do prendedor e atinge um fotodetector do outro lado, as luzes passam por dentro do dedo, medindo assim a mudança de absorção alterando mais ainda o comprimento das ondas que passaram pelo fluxo arterial. Pigmentos na pele, unhas polidas,

luz excessiva no ambiente, má posição do sensor pode causar coleta de dados errônea. Exemplo de um oxímetro de pulso na figura 5.

- c) sensor de movimento(Acelerômetro): como o nome já diz, o acelerômetro mede a aceleração, que é a taxa de variação da velocidade de um objeto, com esses valores, é possível tirar conclusões de diversas situações. Por exemplo, em muitos casos, é necessário monitorar as posições do corpo humano e os movimentos feitos por causa de doenças específicas como apneia do sono ou síndrome das pernas inquietas. Analisando os movimentos durante o sono ajuda a determinar a qualidade do sono e padrões de sono irregulares. O sensor de posição do corpo também pode detectar desmaios ou quedas de idosos ou pessoas com deficiência.

Figura 5 – Sensor de oxímetro de pulso.



Fonte: Rama (2009).

O uso destes sensores pode ser por meio de placas microcontroladoras que interpretam os dados que o sensor coleta e pode ser manipulado posteriormente.

3.4 MICROCONTROLADOR

O microcontrolador é definido em Souza (2005) como um pequeno componente eletrônico, como se fosse um microprocessador, mas mais fácil de ser

manipulado e programado para quem entende o mínimo de lógica de programação e um pouco de eletrônica. Com uma inteligência programável, utilizado para controlar processos lógicos, onde irá dar objetivo ao sistema.

Componentes importantes de um microcontrolador:

- a) unidade central de processamento (*Central Processing Unit* - CPU);
- b) sistema de clock para dar sequência as atividades da CPU;
- c) memória para armazenamento de instruções e para manipulação de dados;
- d) entradas para introduzir informações no CPU do mundo externo;
- e) saídas para transmitir informações processadas pelo CPU para o mundo externo;
- f) programa para definir um objetivo ao sistema.

Hoje em dia o microcontrolador se encontra em todos dispositivos eletrônicos muito conhecidos, como micro-ondas, relógio digital, carros, telefones, plataformas de prototipagem, entre outros.

Os microcontroladores que mais ficaram conhecidos foram o *Peripheral Interface Controller*, comumente conhecido como PIC, que se originou em Harvard em 1975 e depois foi produzido pelo Microchip em 1985 utilizando a arquitetura Harvard, e modelo operacional RISC e usava memórias do tipo *Erasable Programmable Read Only Memory* (EPROM), que eram memórias que continham dados não-voláteis, ou seja, quando desligado o dispositivo, as instruções não apagavam, mas quando necessitasse alguma alteração, era preciso deixar a placa exposta a luz ultravioleta um tempo para que apagasse as instruções, pois os fótons do raio ultravioleta ionizavam a placa de silício, inflando e dissipando os dados guardados (DESHMUKH, 2005, tradução nossa).

E o Atmel 8-bit AVR, os criadores não deram nenhuma resposta definitiva sobre o porque do AVR, mas acredita-se que seja do nome dos criadores do AVR, que são Alf-Egil Bogen e Vegard Wollan's, e o R da tecnologia RISC que é utilizado no microcontrolador. Este microcontrolador ficou muito conhecido, pois, dentre os outros além de ter a arquitetura Harvard e RISC implementada, a memória era flash,

roda as instruções, armazena e apaga a programação mais rapidamente, pois era tudo feito por sinais elétricos (AVR TV, 2008, tradução nossa).

O microcontrolador da Atmel, suporta operações totalmente estáticas de 0 a 24MHz, esta baixa frequência é muito importante na hora de rodar aplicações onde o consumo de energia deve ser manter o mais baixo possível (DESHMUKH, 2005, tradução nossa).

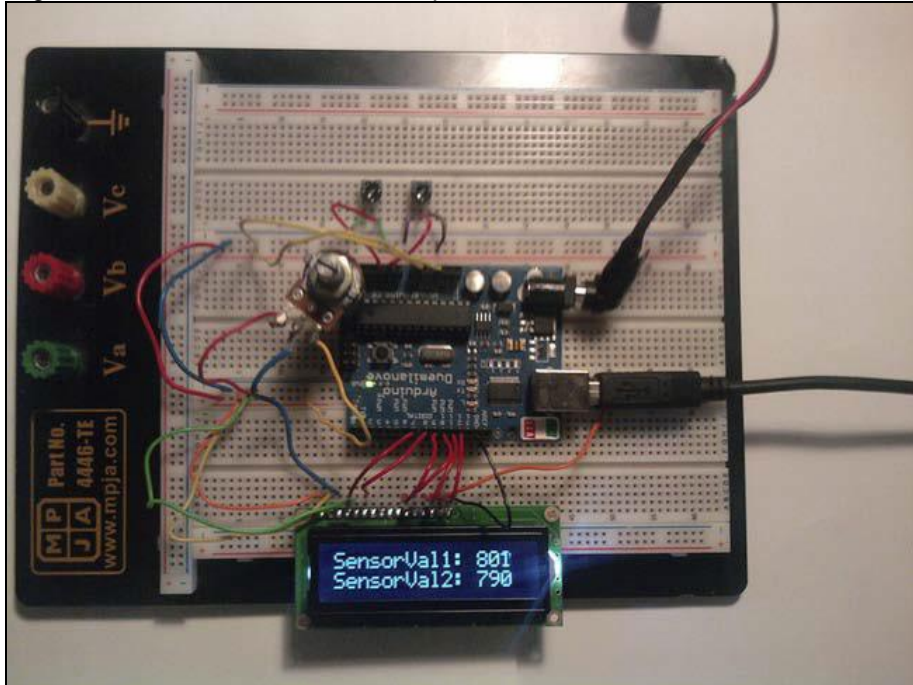
E, devido as suas capacidades, o microcontrolador da Atmel é utilizado em escala mundial, em todos os objetos eletrônicos em que se possa programar, sendo ainda mais conhecido por estar nas plataformas de prototipagem Arduino, que é uma placa onde se pode conectar sensores e desenvolver aplicações sem ter muito conhecimento na área.

3.5 ARDUINO

Segundo Monk (2010), o arduino é uma pequena placa com um microcontrolador Atmel, com um plug *Universal Serial Bus*, USB, para se conectar ao seu computador e uma série de entradas de ligação que podem ser conectados em dispositivos eletrônicos externos, tais como motores, relês, sensores de luz, sensores de batimentos cardíacos, ECG, leds, alto-falantes, microfones, entre outros, conforme na figura 6 exemplifica um Arduino utilizando dois sensores e mostrando na tela de cristal líquido.

Eles podem ser tanto alimentado através da conexão USB do computador quanto a partir de uma bateria de 9V. Eles podem ser controlados a partir do computador ou programados pelo computador e trabalhar de forma independente.

Figura 6 - Arduino com sensores captando valores e mostrando na tela de cristal líquido.



Fonte: Timmis (2011).

Em 2005 teve o início do Projeto Arduino, que começou em Ivrea, Itália, que, utilizando-se de microcontroladores Atmel, criaram uma plataforma de prototipagem totalmente aberta, *hardware* e *software open source*, que facilita o desenvolvimento de dispositivos eletrônicos. É conhecido como uma placa de computação física ou incorporada, o que significa que se trata de um sistema que pode interagir com o seu meio ambiente através da utilização de sensores, dispositivos eletrônicos, como smartphones ou notebooks, e *softwares* (MCROBERTS, 2010, tradução nossa).

Arduino veio com uma linguagem de programação fácil de aprender (derivada da linguagem de programação C++), que incorpora várias funções complexas de programação em comandos simples que são muito mais fáceis para um iniciante para aprender (WARREN; ADAMS; MOLLE, 2011, tradução nossa).

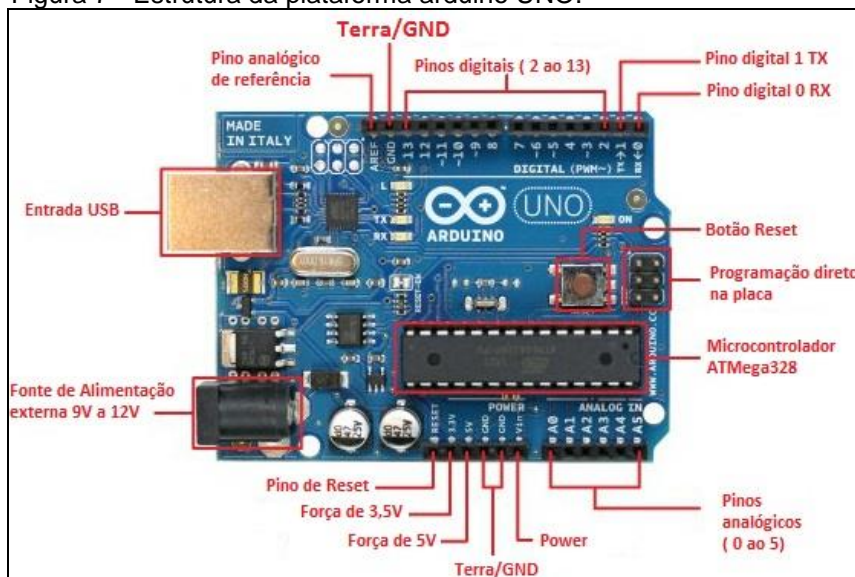
Concluindo sobre o arduino, ele pode ser utilizado para por em prática muitas ideias que antigamente eram impossíveis, e que agora podem atingir objetivos maiores com ajuda de muitos outros objetos, como os *shields*, que serão comentados no próximo capítulo. Pois como neste trabalho, foi utilizado *shields* para

conexão sem fio com os dispositivos eletrônicos e tratamento de dados recebidos pelos sensores corporais sem fio para monitoramento de sinais vitais e base.

3.5.1 Shields

No arduino existem duas linhas de conectores nas bordas da placa. A linha na parte superior da placa são as saídas/entradas digitais (ligado / desligado) dos pinos, embora quaisquer entradas marcado com "~" podem ser usado como saídas analógicas. A linha inferior de conectores tem conexões de energia à esquerda e entradas analógicas à direita, conforme na figura 7 (MONK, 2010, tradução nossa).

Figura 7 - Estrutura da plataforma arduino UNO.



Fonte: Adaptado de Arduino ([20--], tradução nossa).

Esses conectores são organizados como este para que os chamados *shields* possam ser conectados no arduino. É possível comprar *shields* prontos para muitas finalidades diferentes, incluindo:

- conexão com redes Ethernet;
- telas de *Liquid Crystal Display* (LCD), e telas sensíveis ao toque;
- comunicação de dados sem fio XBee ;
- adaptar equipamentos de monitoração de saúde;
- controle de servo motores;
- rastreamento por GPS;

g) entre outros.

Neste trabalho foram utilizados dois *shields* para adaptar aparelhos de medição de sinais vitais e serem transmitido sem fio.

3.6 RESTRIÇÕES

Akyildiz et al (2003, tradução nossa) diz que uma das restrições mais importantes sobre nós sensores é a exigência de ter baixo consumo de energia. Nós sensores carregam limitadas, geralmente insubstituíveis, fontes de energia. Logo, enquanto as redes tradicionais visam atingir alta qualidade de serviço (entrega e recebimento de pacotes mais rápidos), protocolos de rede de sensores deve se concentrar principalmente na conservação de energia. Eles devem ter mecanismos de *trade-off* embutido, e dão ao usuário final a opção de prolongar a vida útil da rede à custa de menor rendimento ou superior atraso de transmissão.

O problema da energia é uma questão muito importante em uma RSSF, pois em muitas situações não se tem acesso aos nós para repor baterias novas, se fosse uma RSSF para fim militar, por exemplo, você poderia monitorar os movimentos dos inimigos, presença de material perigoso como gases tóxicos, radiação, entre outros.

A vida da rede é profundamente dependente da duração da bateria, e a necessidade de novos esquemas para melhorar a eficiência energética que conduz a observar todas as camadas do nó sensor em consideração e vem sendo um desafio muito importante em uma RSSF.

Conforme Patil e Szygenda (2013, tradução nossa), conhecer o nível de energia necessário para ter controle de transmissão dos pacotes e de dados é bastante complexo. O nível de potência do sensor afeta diretamente o alcance da transmissão, alterando assim, o caminho de roteamento percorrido.

Por exemplo, se um nó vizinho está danificado, o nó sensor deve transmitir a um nível de potência maior, reduzindo assim a vida útil da bateria e as suas chances de participar em quaisquer outras operações de roteamento intermediário. Ou seja, em níveis mais baixos de energia pode levar a interferências e causa congestionamento no canal e na rede.

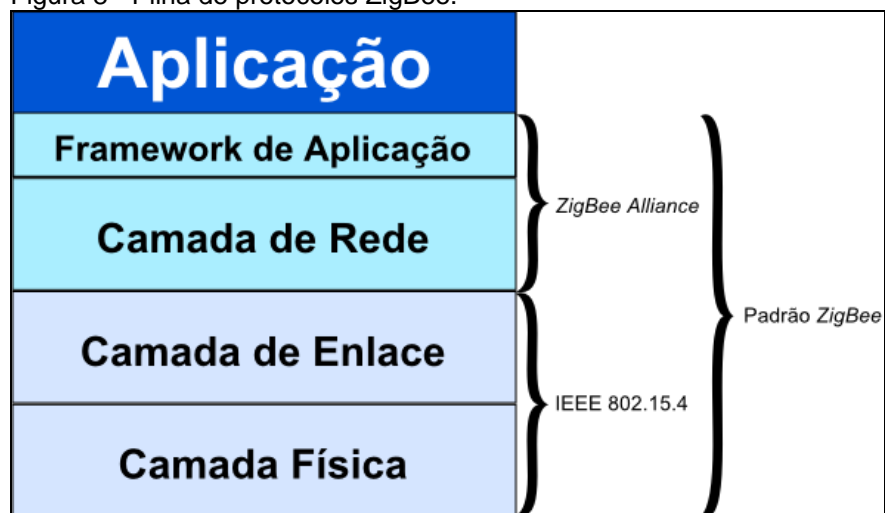
Portanto, o controle de potência gasta ao transmitir pode ser classificado como um problema de design da comunicação entre as camadas (conhecido em inglês como *cross-layer*), como ela afeta todos nós e cada camada da pilha de protocolos (PATIL; SZYGENDA, 2013, tradução nossa).

3.7 CAMADAS DO PROTOCOLO IEEE 802.15.4 E ZIGBEE

Redes trabalham com pilhas de camada, cada camada é responsável por uma ação e oferece serviço a camada superior. O layout destas camadas é baseado no modelo OSI (AKYILDIZ, 2001, tradução nossa).

O protocolo IEEE 802.15.4 trabalha somente nas duas primeiras camadas, a física e a de enlace, já o ZigBee nas camadas de rede e no *framework* da camada de aplicação, conforme mostrado na figura 8.

Figura 8 - Pilha de protocolos ZigBee.



Fonte: Adaptado de Baronti et al (2006).

3.7.1 Camada Física

Conforme na publicação da IEEE de 2006 sobre a camada física do protocolo IEEE 802.15.4, ele é responsável pelas seguintes tarefas:

- a) ativação e desativação do rádio transceptor;
- b) detectar o restante de energia dentro do canal atual;
- c) indicar link de qualidade para pacotes recebidos;

- d) avaliar canal livre para usar o *Carrier Sense Multiple Access with Collision Detection* (CSMA/CD) na camada de enlace MAC;
- e) seleção da frequência do canal;
- f) transmissão de dados e recepção.

A camada física utiliza três faixas de frequências não licenciadas dependendo de sua localização, pois em alguns países utilizam outras bandas por já estarem sendo utilizadas para outras finalidades, totalizando no total de 27 canais *half-duplex*, ou seja, ou transmitem ou recebem dados, conforme a tabela 1.

Tabela 1 – Especificações das licenças ISM utilizadas pelo protocolo IEEE 802.15.4

Faixas	Frequência(MHz)	Local Disponível	Largura de Banda	Quantidade de Canais
2,4 GHz (ISM)	2400 – 2483,5	Mundial	250 Kbps	16
915 MHz (ISM)	902 – 928	EUA	40 Kbps	10
868 MHz	868 – 868,6	Europa	20Kbps	1

Fonte: IEEE (2006).

Sua modulação, método de propagação e a distância entre os canais mudam de acordo com a banda utilizada. As faixas de 868 MHz e 915 MHz implementam uma modulação *Binary Phase Shift Keying* (BPSK) e utilizam o método de propagação *Direct Sequence Spread Spectrum* (DSSS). Já a banda de 2,4 GHz utiliza modulação *Offset Quadrature Phase Shift Keying* (O-QPSK) com DSSS em 2 *microchips* (uma sequência pseudo-aleatória de 32 chips é transmitida num período base de 16 μ s). Ou seja, dezesseis canais com taxa de dados de 250 kbit/s cada, estão disponíveis com -85 dBm de sensibilidade necessário. O alcance de transmissão é de aproximadamente 200m (BURATTI et al, 2011, tradução nossa).

3.7.2 Camada de Enlace

A camada de enlace cuida de todo o acesso ao canal e é responsável pelas seguintes tarefas:

- a) gerar *beacon* na rede se o dispositivo for coordenador;
- b) sincronização de *beacons* na rede;
- c) suporte a associação e dissociação PAN;
- d) controle de segurança;
- e) aplica o mecanismo CSMA/CA para acessar um canal;

f) funções de topologias opcionais se for do tipo estrela.

Beacons são *frames* curtos enviados pelos roteadores a um nó qualquer com o propósito de sincronizar a comunicação em uma rede (TANENBAUM, 2001).

Conforme buratti et al (2011) a camada de enlace fornece controle de acesso a um canal compartilhado e entrega de dados confiável. O IEEE 802.15.4 utiliza um método de transmissão CSMA/CA, o que requer a ouvir o canal antes de transmitir para reduzir a probabilidade de colisões com outras transmissões em curso, o que a camada física já diz se tem algo ou não no canal.

3.7.3 Camadas *ZigBee Alliance*

O ZigBee implementa em cima das camadas do protocolo IEEE 802.15.4, e nela existem três tipos de classificações de dispositivos, os de aplicação (sensores), os lógicos (coordenador, roteador e dispositivos finais) e os físicos (*Full Function Device* e *Reduced Function Device*) (SOMANI; PATEL, 2012, tradução nossa).

Full Function Device (FFD), Dispositivos de Funções Completas, são dispositivos mais complexos e precisam de um hardware mais potente para a implantação da pilha de protocolos, conseqüentemente, consomem mais energia. Numa topologia de rede ZigBee eles podem assumir o papel de coordenador, roteador ou mesmo de um dispositivo final. FFD podem se comunicar com quaisquer membros da Rede. São implementados em microcontroladores com no mínimo 32KB de memória de programa e ter uma certa quantidade de memória RAM, para implementações de tabelas de rotas e configurações de parâmetros.

Reduced Function Device (RFD), Dispositivos de Funções Reduzidas, são dispositivos mais simples, onde sua pilha de protocolo pode ser implementada usando os mínimos recursos possíveis de hardware, como por exemplo, em microcontroladores de 8 bits com memória de programa próxima a 6KB, mas só podem se comunicar com dispositivos FFDs (Coordenador ou Roteador). Numa topologia de Rede ZigBee eles assumem o papel do end device (dispositivo final). Na prática podem ser: interruptores de iluminação, controle de relês, sensores, entre outros.

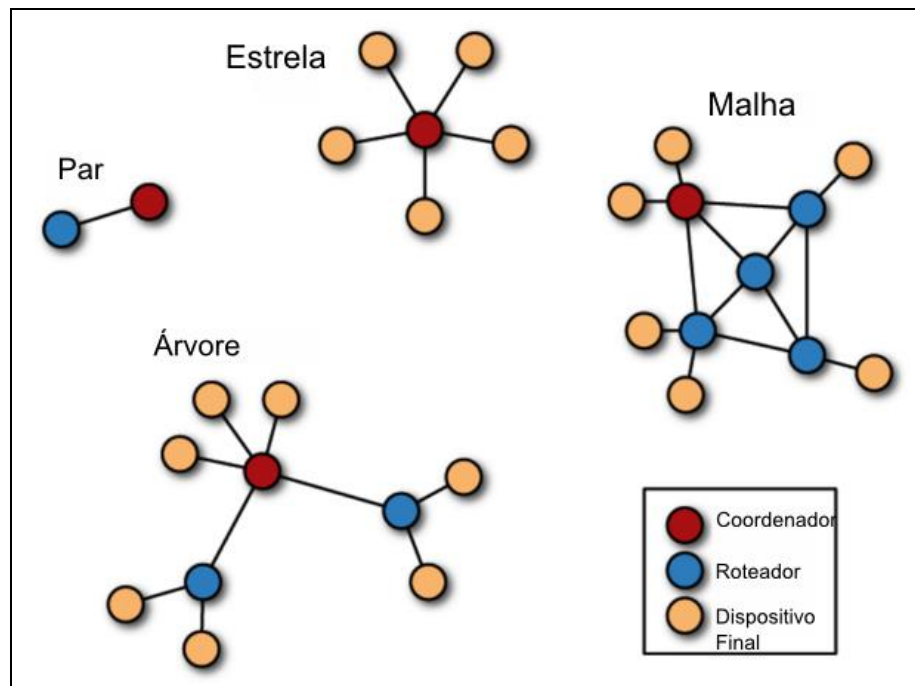
Cada dispositivo na IEEE 802.15.4 tem um único comprimento de endereço de 64 bits. Após associar a um coordenador, um dispositivo será atribuído um endereço curto de 16 bits. Em seguida, as trocas de pacotes entre o coordenador e os dispositivos utilizam os endereços curtos (PAN; TSENG, 2007, tradução nossa).

Dispositivos lógicos:

- a) coordenador: constitui a raiz da árvore de rede e pode construir uma ponte para outras redes. Há exatamente um coordenador em cada rede. Ele é responsável por iniciar a rede e configurar os parâmetros de rede, tais como canal de frequência de rádio, identificador de rede único e definir outros parâmetros operacionais. Ele também pode armazenar as informações sobre rede, chaves de segurança;
- b) roteador: roteador age como nós intermediários, transmitindo dados de outros dispositivos. Roteador pode se conectar a uma rede já existente, também capaz de aceitar conexões de outros dispositivos e ser uma espécie de retransmissor para aumentar o alcance da rede;
- c) dispositivos finais: dispositivo final pode ser alimentado por bateria de baixa potência. Eles podem coletar diversas informações a partir de sensores e interruptores. Eles têm funcionalidade suficiente para conversar com seus pais (coordenador ou um roteador) e não podem transmitir dados a partir de outros dispositivos. Esta funcionalidade permite diminuir o potencial para reduzir o seu custo energético. Quanto menor a potência melhor.

A figura 9 mostra como são distribuídos os nós e os tipos de topologias utilizadas.

Figura 9 - Topologias e distribuição dos dispositivos da rede ZigBee.



Fonte: Adaptado de Faludi (2010, tradução nossa).

3.7.3.1 Camada de Rede

A camada de rede fornece transmissões seguras e confiáveis entre os dispositivos, ela é responsável pela organização e prestação de roteamento em uma rede de múltiplos saltos, enquanto a camada de aplicação tem a intenção de fornecer um quadro para o desenvolvimento de aplicações distribuídas e comunicação (BARONTI et al, 2006, tradução nossa).

As responsabilidades da camada de rede ZigBee incluem: mecanismos para entrar e sair da rede, segurança, roteamento, descoberta do caminho, descoberta de vizinhos de um salto e armazenamento de informações sobre os nós vizinhos (BURATTI, 2011, tradução nossa).

3.7.3.2 Camada de Aplicação

A camada de aplicação consiste no suporte da sua própria subcamada, o *framework* da aplicação.

As responsabilidades da subcamada de aplicação incluem: a armazenar tabelas de roteamento para ligação (definida como a capacidade de combinar dois dispositivos em conjunto com base em seus serviços e as suas necessidades),

encaminhamento de mensagens entre dispositivos, gerenciar endereços, mapear o endereço de 64 bits no endereço de rede (16 bits) e dar suporte a realização do transporte de dados (BURATTI, 2011, tradução nossa).

A camada de aplicação também é afetada pelo fabricante, pois ele que faz a personalização para que o dispositivo seja capaz de se adaptar a várias situações, ou seja, o mesmo dispositivo com protocolo ZigBee pode ser utilizado tanto para fins militares quanto para fins industriais.

Conforme as topologias do ZigBee encontrada na Figura 9, um coordenador pode redirecionar o conteúdo que foi transmitido até ele para um *smartphone* com plataforma Android que usa linguagem de programação Java que é mundialmente conhecida e utilizada, e, sua plataforma de desenvolvimento tem uma interface intuitiva e fácil de programar o que torna o desenvolvimento de uma aplicação móvel mais ágil e simples. O sistema operacional Android é um dos sistemas móveis mais utilizado no mundo segundo o site *Net Applications*, que mede tráfego de dispositivos em alguns sites.

4 SISTEMA OPERACIONAL MÓVEL ANDROID

O Android é uma plataforma *open source*, projetada para dispositivos móveis. É propriedade da Google e *Open Handset Alliance* (OHA), mas quem desenvolveu foi a OHA. O objetivo da aliança é "acelerar a inovação para clientes de dispositivos móveis e oferecer uma experiência móvel mais rica, barata e melhor" (GARGENTA, 2011, tradução nossa).

Ele tem uma grande base de usuários e fãs, é a plataforma líder mundial de smartphones, e não há fim à vista para o seu crescimento. Android pode ser encontrado em execução em centenas de diferentes e diversas categorias de dispositivos diferentes (LEHTIMÄKI, 2013, tradução nossa).

É uma plataforma que, pela primeira vez, é totalmente aberta que separa o hardware do software que roda sobre ele. Isto permite um número muito maior de dispositivos para executar as mesmas aplicações e cria um ecossistema muito mais rico e firme para os desenvolvedores e consumidores (GARGENTA, 2011, tradução nossa).

4.1 HISTÓRIA

O Google, observando o grande crescimento no uso da Internet e em dispositivos móveis, adquiriu o Android Inc., que era a proprietária do SO móvel Android em 2005, para concentrar seu desenvolvimento em uma plataforma para smartphones. A Apple introduziu o iPhone em 2007, com algumas ideias inovadoras, como o *multi touch* e um mercado aberto para aplicativos (STEELE; TO, 2011, tradução nossa).

Diversidade de dispositivos e adaptação rápida ajudou o Android a crescer sua base de usuários, mas ele vem com alguns desafios para desenvolvedores. Eles precisam suportar vários tamanhos de tela, taxa de resolução, tipos de teclados, sensores de hardware diferentes, versões do SO, taxas de dados sem fio e configurações do sistema. Cada um pode levar a um comportamento diferente e imprevisível, mas testes de aplicações em todos os ambientes é uma tarefa impossível (STEELE; TO, 2011, tradução nossa).

Ao ser anunciado oficialmente, o sistema rapidamente atraiu a atenção de desenvolvedores e do mercado em geral. Isso aconteceu graças ao grupo de empresas que a Google reuniu em sua criação, empresas de ramos diversificados, mas líderes. Ramos do mercado de telefonia, que fortificou a ideia de desenvolver uma plataforma *open source* padronizada e que acatasse as expectativas e tendências do mercado (LECHETA, 2010).

4.2 CARACTERÍSTICAS

Segundo o livro de Burnett (2010), os principais motivos e características que diferenciam o Android dos outros SO's são os seguintes:

- a) uma plataforma de desenvolvimento realmente aberta e de graça baseada em Linux: os fabricantes de celulares gostam disso pois podem usar e personalizar a plataforma sem pagar royalties;
- b) a arquitetura baseada em componentes inspirada em misturas da Internet: utiliza-se partes de um aplicativo para ser usado de outras formas não inicialmente previstas pelo desenvolvedor. Você pode até mesmo substituir componentes internos com suas próprias versões melhoradas. Incentivando a criatividade e investimentos na área de dispositivos móveis;
- c) alta gama de serviços embutidos fora da caixa: sistema de localização global, conhecido como *Global Positioning System* (GPS) para você personalizar a experiência do usuário, dependendo de onde ele está. Um banco de dados relacional *Structured Query Language* (SQL) permite aproveitar o poder do armazenamento local, sem necessidade de ter uma aplicação rodando o tempo todo, salvando e buscando informações somente quando o necessário. Mapa e navegadores podem ser incorporados diretamente em seus aplicativos. Todas essas funcionalidades melhorando a usabilidade das aplicações enquanto reduz o trabalho dos desenvolvedores ao mesmo tempo;
- d) na plataforma Android, não há distinção entre aplicações nativas e de terceiros, ajudando, assim, manter a concorrência saudável entre os desenvolvedores de aplicativos pois, todos os aplicativos do Android

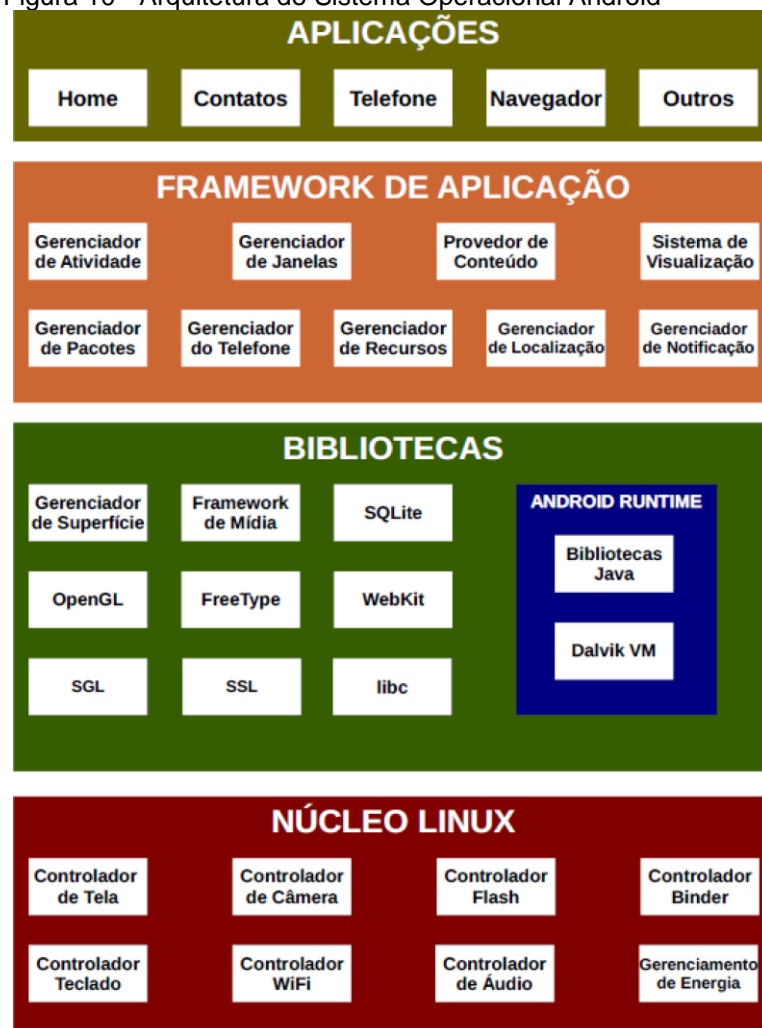
usam a mesma *Application Programming Interface* (API), que é o conjunto de padrões de programação que permite a construção de aplicativos e a sua utilização de forma não tão evidente para os usuários;

- e) linguagem de programação é uma das mais conhecidas e utilizadas no mundo, o Java.

4.3 ARQUITETURA

Conforme Gargenta (2011), o Android tem uma arquitetura separada por camadas. Estas camadas não estão claramente separadas, pois algumas delas infiltram-se umas nas outras, conforme na figura 10.

Figura 10 - Arquitetura do Sistema Operacional Android



Fonte: Adaptado de Gargenta (2011, tradução nossa).

4.3.1 Núcleo

Android é construído em cima de uma sólida e comprovada base: o *kernel* do Linux, que, faz a gerência dos recursos do sistema para que os *softwares* possam utilizá-lo. Criado por Linus Torvalds em 1991, pode ser encontrado hoje em tudo, desde relógios de pulso a supercomputadores. O Linux fornece a camada de abstração de hardware para o Android, permitindo ser portado para uma ampla variedade de plataformas no futuro (GARGENTA, 2011, tradução nossa).

4.3.2 Bibliotecas

Em cima do *kernel* estão as bibliotecas nativas. Estes são módulos de código que são compilados em código de máquina nativo para o dispositivo e fornecer alguns dos serviços comuns que estão disponíveis para aplicativos e outros programas. Eles incluem o Gerenciador de Superfície (responsável pelos gráficos na tela do dispositivo), bibliotecas gráficas 2D e 3D, WebKit (o motor de renderização navegador web padrão) e o SQLite (a tecnologia de armazenamento de dados para a plataforma Android). Estas bibliotecas nativas são executadas como processos dentro do *kernel* do Linux (SIX, 2012, tradução nossa).

4.3.3 Android *RunTime*

Também rodando como processos dentro do kernel do Linux é o Android *RunTime*. Cada aplicação Android roda em um processo separado, com sua própria instância da Dalvik *Virtual Machine* (VM). Com base no Java VM, o projeto foi otimizado para dispositivos móveis. O Dalvik VM tem um baixo uso de memória e o carregamento da aplicação é otimizado. Várias instâncias desta máquina virtual podem ser executadas simultaneamente em um dispositivo (ANUZZI; DARCEY; CONDER, 2014, tradução nossa).

4.3.4 Framework de Aplicação

Segundo Gargenta (2011), o framework de aplicação é onde o desenvolvedor pode abusar com as ferramentas que o próprio dispositivo oferece, aplicando suas técnicas e habilidades para trazer ao mercado bons aplicativos. Nesta camada encontram-se várias bibliotecas Java construída especificamente para o Android e serviços que fornecem as capacidades que uma aplicação pode ter, tais como localização, sensores, Wi-Fi, telefonia, entre outros.

4.3.5 Camada de Aplicação

A camada de aplicação é onde rodam os programas, e, da mesma forma e com as mesmas possibilidades, as aplicações feitas por terceiros e pela própria fabricante tem os mesmos recursos e possibilidades (GARGENTA, 2011, tradução nossa).

5 TRABALHOS CORRELATOS

Os trabalhos apresentados a seguir estão relacionados com os objetivos e/ou temas apresentados neste trabalho.

5.1 RASTREABILIDADE BOVINA: UTILIZANDO PLATAFORMA ZIGBEE

Este trabalho apresenta uma RSSF com o objetivo de monitorar e rastrear os rebanhos bovinos, utilizando linguagem de programação Java e banco de dados MySQL (SOUZA, 2014).

As antenas Xbee passam a informação geográfica utilizando protocolo ZigBee entre os nós até chegar no nó coordenador, que está em uma torre que envia as informações para uma base de dados na Internet, que pode ser acessada por um navegador *web*, auxiliando os fazendeiros a controlar seu rebanho com *dashboards* e mapas (SOUZA, 2014).

O resultado apresentado atendeu as especificações exigidas, abrindo precedentes para novos projetos na área de rastreabilidade de bens fazendo uso do hardware utilizado neste protótipo.

5.2 DESENVOLVIMENTO DE UMA REDE DE SENSORES SEM FIO UTILIZANDO ZIGBEE PARA APLICAÇÕES DIVERSAS

Este trabalho apresenta uma RSSF utilizando a tecnologia ZigBee com a finalidade de servir para diversas aplicações, reduzindo o tempo de montar uma RSSF, se preocupando somente com os sensores e a forma de análise dos dados (HONORATO, 2011).

Com a união de um sistema gerenciador de banco de dados (SGBD) MySQL, Honorato (2011) implementou um protocolo para que seja possível capturar os dados e enviar comandos para os nós sensores, podendo ser adaptada para diversas finalidades e aplicações.

Este trabalho obteve resultados satisfatórios, podendo ser utilizado em aplicações que necessitem monitoramento remoto, diminuindo o tempo de desenvolvimento, necessitando apenas selecionar os sensores e analisar os dados.

5.3 REDES DE SENSORES SEM FIO NA COLETA DE DADOS FISIOLÓGICOS DE BOVINOS PARA APLICAÇÕES NA ZOOTECNIA DE PRECISÃO

O trabalho apresenta o desenvolvimento de uma infraestrutura de rede e uma aplicação para monitoramento das variáveis fisiológicas de bovinos por meio de uma rede de sensores sem fio. Para isso, foi desenvolvida uma infraestrutura de rede *ad hoc* com o protocolo *Floating Base Sensor Network* (FBSN) (ACRE, 2008).

Foram desenvolvidos protótipos para coletar sinais cerebrais e temperaturas dos bovinos, juntamente com softwares embarcados, de supervisão e um simulador de deslocamento bovino (ACRE, 2008).

Os resultados obtidos neste trabalho mostraram que é possível utilizar com sucesso uma rede de sensores sem fio na coleta e monitoramento dos sinais fisiológicos de bovinos (ACRE, 2008).

5.4 WIRELESS SENSOR NETWORKS FOR BIOMEDICAL APPLICATIONS

Este artigo expõe uma forma de implementar uma rede de sensores sem fio corporal, incorporando em uma rede normal para que possa ter acesso aos dados de forma mais abrangente, utilizando o protocolo 6LoWPAN e analisa sua qualidade de serviço, com entrega e recebimento de dados, integridade e durabilidade da rede na área biomédica (ABREU; MENDES, 2013).

Desta forma, este trabalho apresenta a implementação e avaliação de uma estratégia de implantação com pouco gasto energético e com base em uma função objetiva com reconhecimento para o protocolo de roteamento de baixo consumo de energia e com poucas perdas de pacotes e nós relé (ABREU; MENDES, 2013).

6 UTILIZANDO A RSSF PARA MONITORAR SINAIS VITAIS

Este trabalho tem o propósito de monitorar sinais vitais de seres humanos utilizando uma rede de sensores sem fio para captação dos dados e um smartphone para visualização do mesmo.

Com a evolução da tecnologia, monitorar e controlar setores, processos, pessoas, áreas se tornou cada vez mais fácil com a utilização de redes de sensores sem fio, a escolha do protocolo de comunicação utilizada na rede foi feita a partir de artigos e pesquisas que levaram ao protocolo ZigBee, pois foi a tecnologia que atendeu ao propósito do trabalho de forma completa, utilizando o padrão do IEEE 802.15.4 e mantendo a interoperabilidade entre equipamentos, escalabilidade e versatilidade na escolha de fabricantes.

Como explicado antes, é um padrão de comunicação que pode ser utilizado para diversas funções, o nosso objetivo é experimentar sua utilização na área da saúde fora de ambientes clínicos, monitorando várias pessoas que necessitam cuidados especiais ou não, como idosos, onde o ser humano não se encontre em estado crítico, que necessite de cuidados emergenciais.

Os parâmetros fisiológicos a serem monitorados serão temperatura e posição em que o corpo humano se encontra, se está de pé ou sentado, posição de supino, de bruços ou de lado.

6.1 METODOLOGIA

Para a realização deste trabalho, foram efetuadas pesquisas sobre formas de comunicação sem fio que pudessem ser utilizadas para transferir sinais vitais de seres humanos até um computador e uma forma de enviar estes dados recebidos para um smartphone, que atendesse o critério de utilizar pouca energia, pois cada sensor não teria uma fonte de energia fixa, já que as pessoas podem estar em movimento ou não, e que trabalhasse com vários nós.

Após pesquisas sobre algumas redes e reuniões com o orientador, foi decidido o uso do protocolo ZigBee, devido ao baixo consumo de energia e era de rápida implementação. Com a ajuda do laboratório de projeto de pesquisa do curso, que financiou todo o projeto, foram então escolhidas as antenas e demais equipamentos necessários para manipular os dados adquiridos do corpo humano e que seriam enviados de antena a antena até chegar ao nó coordenador que está conectado ao computador. Através de leitura serial, este iria ler os dados e após o computador estar pareado com o smartphone via Bluetooth, iria repassar estes dados ao dispositivo para se ter uma forma de visualização e acesso aos dados coletados.

O Bluetooth foi escolhido devido o seu uso mundialmente conhecido, atualmente todos os smartphones o tem integrado, e como se trata de monitoração de dados fisiológicos, a visualização destes dados em tempo real e a transmissão de forma consistente foram a chave para a escolha do uso do Bluetooth.

6.1.1 Escolha do Hardware

Durante as pesquisas foram encontradas diversas ferramentas e outras possibilidades de se realizar, mas com objetivos e necessidades diferentes, e, como este projeto é somente um protótipo, subentende-se que não há necessidade de equipamentos de primeira linha, foi utilizado o necessário para criar um cenário que mostre o potencial das redes de sensores sem fio e seu protocolo de comunicação, como esta aplicação pode ajudar a sociedade e pesquisadores. Foram escolhidas estas ferramentas:

- a) 3 Antenas Xbee Series 2;
- b) 2 Arduinos Uno R3;
- c) 1 Sparkfun Xbee *Shield*;
- d) 1 Arduino Xbee Shield;
- e) 1 Xbee Explorer p/ Antena Xbee;
- f) 2 *Shields E-Health* da *Cooking Hacks*;
- g) 1 sensor de temperatura;
- h) 1 sensor de posição MMA8452Q;
- i) 1 smartphone com SO Android e com Bluetooth;

- j) 1 Notebook com Bluetooth e java;
- k) 2 cabos adaptados de alimentação do arduino;
- l) 2 baterias 9v.

As antenas Xbee Series 2 foram escolhidas pela sua facilidade e viabilidade da aplicação, importante notificar que as versões Series 1 e 2 não se comunicam, a rede necessita ser composta apenas de um tipo, basta configurar o tipo que ele vai ser, o ID da rede e já pode ser incluído na rede.

Cada antena tem uma potência de saída de 1mW (0dBm), tendo uma área de alcance de 30 metros em lugares fechados, podendo se comunicar com cada nó dentro desta distância, ou seja, se cada nós manter dentro de um raio média de 15m, podem se comunicar e transmitir entre distâncias muito grandes, com uma taxa de dados de 250Kbps.

As antenas que transmitem os dados captados pelos sensores são conectadas a um *shield* Xbee, que serve de suporte para que o Arduino após receber os dados dos sensores que estão conectados ao *shield e-Health*, faça um tratamento, pois ele recebe dados elétricos, então é necessário fazer uma conversão e então é transmitido, sendo que a fonte de energia vem da bateria de 9v que está conectada ao Arduino, ela que alimenta todos os *shields* conectados a ele.

Conforme segue a figura 11, ela demonstra os *hardwares* utilizados, a ordem dos *Shields* montados e a comunicação.

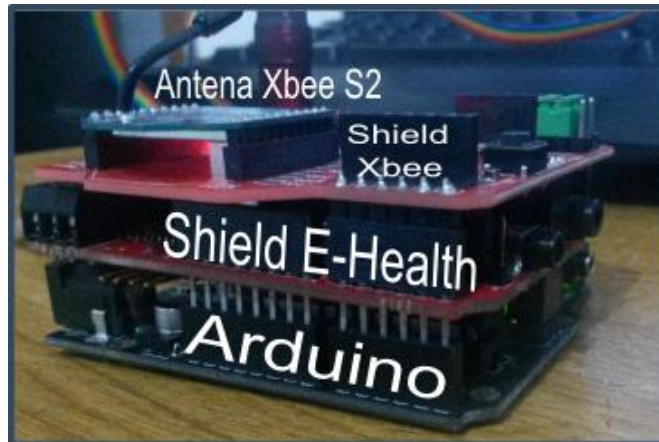
Figura 11 – Cenário com todos os hardwares e comunicação utilizados.



Fonte: Do autor

Já na figura 12 observa-se como ficou a montagem dos *shields* e a antena.

Figura 12 – *Shields* montados.



Fonte: Do autor

6.1.2 Softwares utilizados

Os softwares utilizados para implementação deste trabalho foram:

- a) IDE Dev-C++;
- b) Arduino IDE;
- c) NetBeans IDE 8.0.2;
- d) Android Studio IDE;
- e) XCTU 6.2.0;

Para manipular a plataforma Arduino e os sensores, foram utilizados dois ambientes de desenvolvimento e uma biblioteca da empresa *Cooking Hacks*, empresa da qual foi efetuada as compras dos *shields E-Health* e sensores, esta biblioteca é livre de custos e de código fonte aberto, disponível no site. Os ambientes são, o IDE Dev-C++, que utiliza a linguagem de programação C++, para manipular a biblioteca quando necessário, e, o Arduino IDE, onde é programada a placa Arduino utilizando as funções da biblioteca *E-Health*.

O NetBeans 8.0.2 e o Android Studio, utilizam uma das linguagens de programação mais conhecida do mundo segundo o site *RedMonk*, o Java. Para a criação da aplicação desktop foi utilizado o NetBeans IDE, utilizando as bibliotecas RXTXcomm para leitura da porta serial virtual criada pela antena Xbee que está conectado no computador via USB, e a bluecove-2.1.1 para conexão Bluetooth com o dispositivo e passar as informações coletadas.

O X-CTU configura o *firmware* utilizado pela antena Xbee, dizendo se é um *end-device*, roteador ou coordenador e outras propriedades como a rede, os endereços, segurança, modo *sleep* e configuração de entradas e saídas.

6.1.3 Integração Hardware e Software

Após obter todos os equipamentos, foi necessário primeiramente instalar o *software* X-CTU da *Digi International*, que faz a leitura e alteração do *firmware* das antenas, e o driver FTDI que é encontrado dentro da pasta do Arduino IDE. Utilizando o *Xbee Explorer USB* com a antena Xbee S2, conecta-se ao computador, e, utilizando o *software* X-CTU, no botão “*Discover Devices*” se consegue procurar utilizando a porta COM conectada e com as configurações padrão adicionar a antena na lista.

Depois de adicionada foi atualizado o *firmware* de acordo com cada antena, conforme segue a tabela 2. Na coluna “Função” a letra “C” é para *coordinator* e “R” para *Router*, esta configuração que determina o modelo de sua rede, se será modo API ou AT. O modo API é baseado em *frames* de dados, mais complexo para se trabalhar, normalmente utilizados quando se há muitos nós. Como se trata de uma aplicação protótipo foi utilizado o modo transparente, nó roteador manda pacotes para o próximo nó, até achar o coordenador. Neste modo a configuração das antenas é ágil e atende a necessidade deste trabalho.

Tabela 2 – Configuração das antenas Xbee

Nó					
identificador	Modelo Xbee	Função	Versão	<i>PAN ID</i>	<i>Baudrate</i>
ZB_COORD	XB24-ZB	C. AT	20A7	2001	115200
ZB_TEMP	XB24-ZB	R. AT	22A7	2001	115200
ZB_POS	XB24-ZB	R. AT	22A7	2001	115200

Fonte: Do autor.

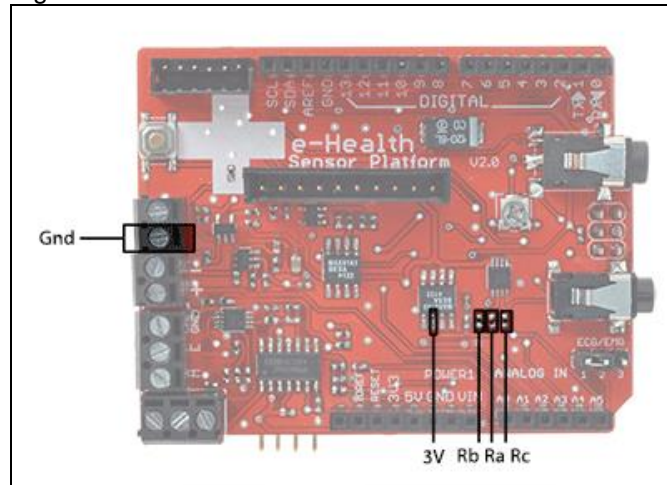
Não há antenas *End Device* pois todos os nós necessitam se comunicar com todos pois como eles estarão em humanos, eles podem se mover, e, um *End Device* não pode interferir no caminho de um nó roteador até o nó coordenador.

Então para a leitura dos sensores foram programadas as plataformas Arduino, utilizando as funções disponíveis pela biblioteca *E-Health*, estas funções fazem a conversão dos dados elétricos para dados fisiológicos e são enviadas pela

porta serial, o *shield* Xbee projeta estes dados de acordo com a configuração do *firmware*.

A função da temperatura sofreu alteração no código fonte dela, devido o sensor vir com uma calibração padrão, que normalmente é o suficiente para a maioria dos casos. Pode-se ter maior precisão e diminuir as inconsistências durante a medição de temperatura, efetuando uma nova calibração, onde é feito um processo de medição dos valores reais da tensão e resistência do *shield*, utilizando um multímetro, e alterando as variáveis na biblioteca, como mostra as figuras 13 e 14 abaixo.

Figura 13 – Shield E-Health com os locais a serem medidos



Fonte: Cooking Hacks (2013).

Já na figura 14, após a medição, é feita a alteração das variáveis com os valores obtidos pelo multímetro. Os valores obtidos foram R_a com 4420, R_c com 4740 e R_b .643. A tensão deu 3.00, não foi modificado.

Figura 14 – Função de conversão para temperatura, variáveis a serem alteradas após medição.

```
float eHealthClass::getTemperature(void)
{
    //Variáveis Locais
    float Temperature; //Temperatura corporal.
    float Resistance; //Resistência do sensor.
    float ganancia=5.0;
    float Vcc=3.3;
    float RefTension=3.0; // Voltagem Referente ao Wheatstone bridge.
    float Ra=4700.0; //Resistencia Wheatstone bridge.
    float Rc=4700.0; //Resistencia Wheatstone bridge.
    float Rb=821.0; //Resistencia Wheatstone bridge.
    int sensorValue = analogRead(A3); // Leitura do sensor

    float voltage2=((float)sensorValue*Vcc)/1023; // Conversão de binário para tensão

    // Tensão de saída da Wheatstone bridge.
    voltage2=voltage2/ganancia;
    // Cálculo da resistência do sensor
    float aux=(voltage2/RefTension)+Rb/(Rb+Ra);
    Resistance=Rc*aux/(1-aux);
    if (Resistance >=1822.8) {
        // Se a temperatura estiver entre 25Â°C e 29.9Â°C. R(tÂ°)=6638.20457*(0.95768)^t
        Temperature=log(Resistance/6638.20457)/log(0.95768);
    }
    else {
        if (Resistance >=1477.1){
            // Se a temperatura estiver entre 30Â°C e 34.9Â°C. R(tÂ°)=6403.49306*(0.95883)^t
            Temperature=log(Resistance/6403.49306)/log(0.95883);
        }
        else {
            if (Resistance >=1204.8){
                // Se a temperatura estiver entre 35Â°C e 39.9Â°C. R(tÂ°)=6118.01620*(0.96008)^t
                Temperature=log(Resistance/6118.01620)/log(0.96008);
            }
            else{
                if (Resistance >=988.1){
                    // Se a temperatura estiver entre 40Â°C e 44.9Â°C. R(tÂ°)=5859.06368*(0.96112)^t
                    Temperature=log(Resistance/5859.06368)/log(0.96112);
                }
                else {
                    if (Resistance >=811.7){
                        // Se a temperatura estiver entre 45Â°C e 50Â°C. R(tÂ°)=5575.94572*(0.96218)^t
                        Temperature=log(Resistance/5575.94572)/log(0.96218);
                    }
                }
            }
        }
    }
}
```

Fonte: Cooking Hacks (2013, tradução Nossa).

Para a aplicação do dispositivo móvel saber qual dado era temperatura e qual dado era posição, foi modificada a função que imprime os dados da posição e da temperatura para valores identificadores. A temperatura foi adicionado a letra T antes de cada valor, (e.g. T36,50). E da posição, como podem ser cinco posições possíveis, foi adicionado a letra P e um número indicador da posição.

Tabela 3 – Posição do corpo e respectivos códigos

Posição do Corpo Humano	Código
Supino	P1
Deitado pra esquerda	P2
Deitado para direita	P3
De braços	P4
De pé ou sentado	P5

Fonte: Do autor.

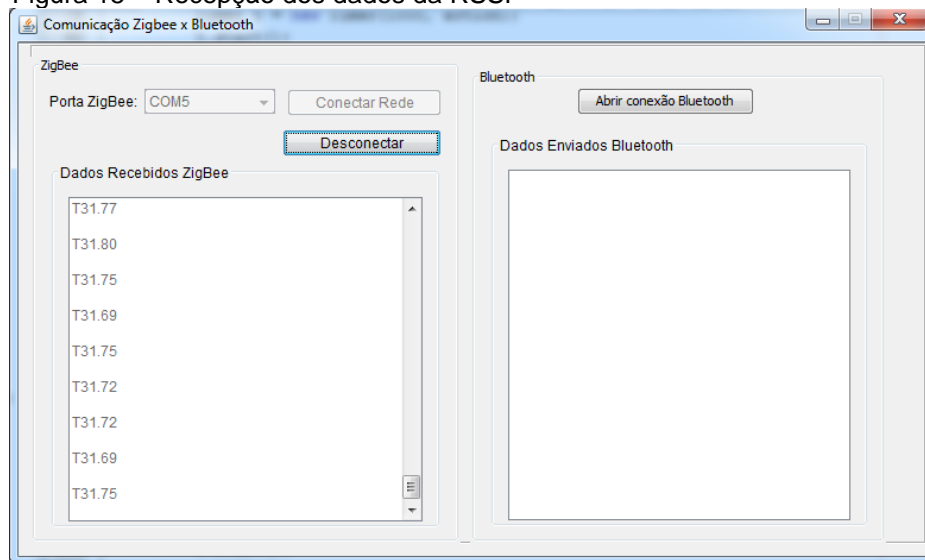
6.1.4 Recepção dos dados

Após estes dados serem distribuídos até o nó coordenador que fica no computador, foi então desenvolvido uma aplicação desktop que faz a leitura da porta serial e repassa para o dispositivo móvel.

Esta aplicação foi desenvolvida utilizando a linguagem Java no ambiente de desenvolvimento NetBeans 8.0.2, em conjunto com as bibliotecas RXTXcomm para comunicação serial RS232 e a bluecove-2.1.1 para comunicação Bluetooth.

Para a visualização dos dados é necessário somente indicar a porta COM utilizada pelo *Xbee Explorer* USB, como mostra a figura 15.

Figura 15 – Recepção dos dados da RSSF

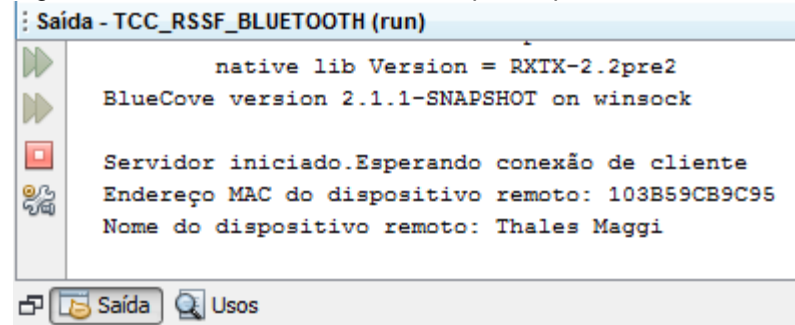


Fonte: Do autor.

A aplicação feita para o SO móvel Android foi utilizada a linguagem Java também, mas em outro ambiente de desenvolvimento, o Android Studio.

Para a comunicação de dados se estabelecer entre o computador e o dispositivo móvel, é necessário aplicar o conceito de cliente-servidor, muito semelhante à comunicação TCP/IP, onde o servidor é o computador Desktop e o cliente o smartphone. Clicando no botão de se comunicar via Bluetooth, a aplicação Desktop irá iniciar o serviço, esperando que algum dispositivo com o código de serviço idêntico se conecte, conforme a figura 16.

Figura 16 – Conexão Bluetooth Desktop x Dispositivo móvel



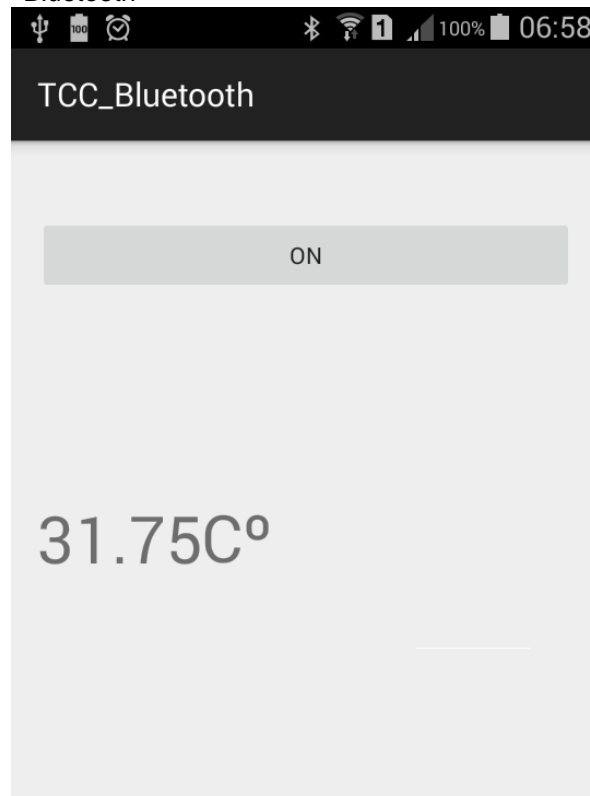
```
native lib Version = RXTX-2.2pre2
BlueCove version 2.1.1-SNAPSHOT on winsock

Servidor iniciado. Esperando conexão de cliente
Endereço MAC do dispositivo remoto: 103B59CB9C95
Nome do dispositivo remoto: Thales Maggi
```

Fonte: Do autor.

Para isto, o dispositivo móvel e o computador já devem estar pareados. A aplicação móvel então é iniciada e conectada a aplicação Desktop, já mostrando os dados obtidos da RSSF. Segue um exemplo na figura 17.

Figura 17 – Aplicação protótipo Android lendo as informações da rede Bluetooth



Fonte: Do autor.

6.2 RESULTADOS OBTIDOS

Os primeiros testes foram com os nós sensores, alterando suas localizações e testando seus limites.

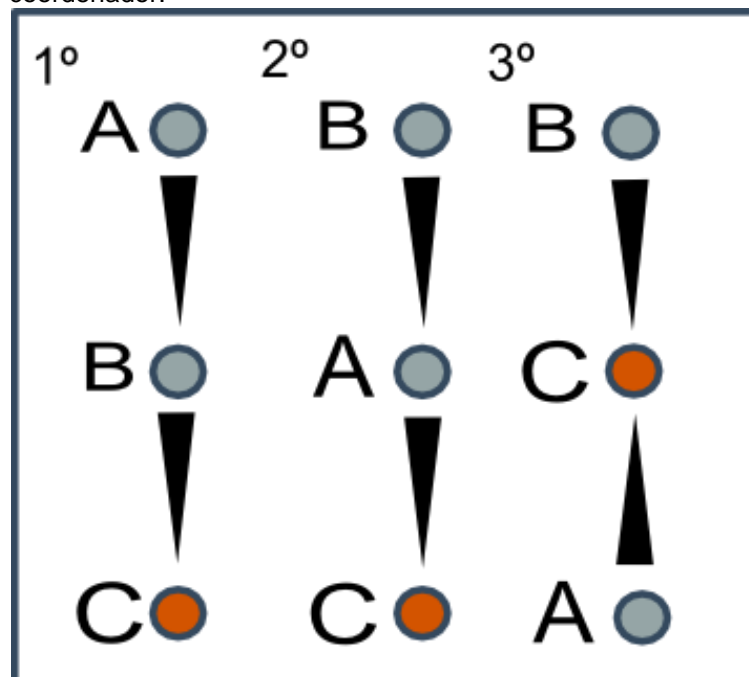
É importante lembrar que estas antenas Xbee *Series 2* são da versão padrão, com uma distância de sinal em lugares fechados em até 30 metros e 120 metros em campo aberto.

O primeiro teste foi relativo ao alcance e como eles se comportam em movimento. Em distância, lugares com muitas paredes interferindo perde-se o sinal, entre portas ou alguns objetos obstruindo o caminho afetava o sinal, demorando na entrega dos pacotes e as vezes com perdas.

Se movendo com as antenas em funcionamento, quando ele detecta qualquer mudança ele vai guardando as informações num *buffer* e após conseguir se conectar com o próximo nó envia o que estava guardado.

O segundo teste foi relativo à ordem das antenas em relação ao nó coordenador, que não influenciou nos resultados obtidos. Conforme a figura 18.

Figura 18 – Posição dos nós comparadas para testar os pacotes até o nó coordenador.



Fonte: Do autor.

O terceiro teste foi averiguado se os valores obtidos a partir do sensor de temperatura são suficientes para medir o corpo humano. Para este teste, os dados de temperatura que o sensor está captando, devem ser o mais próximo do valor base dito como normal em algumas partes do corpo humano. A temperatura central cutânea média do ser humano, medido normalmente a partir da axila, varia entre 35,5°C e 37.5°C, ou seja, cada parte do corpo tem suas variâncias de temperatura.

Então foram feitos 2 testes em 2 pontos do corpo humano, primeiramente medindo a temperatura durante 5 minutos em cada ponto, obtendo um valor por minuto e feito uma média aritmética para comparar se está dentro do valor médio, antes da calibração, e depois fazer os mesmos passos só com os valores da biblioteca calibrados.

Os dois pontos são o peito e pulso, foram escolhidos estes pontos pois oferecem maior facilidade na captação e na usabilidade do sensor. Os resultados estão nas tabelas 4 e 5 a seguir. Os valores foram obtidos do próprio autor.

Tabela 4 – Valores obtidos antes da calibração

Minuto	Pulso	Peito
1	36.85°C	37.31C ⁰
2	37.40°C	37.01C ⁰
3	37.40C ⁰	37.01C ⁰
4	37.04C ⁰	36.88C ⁰
5	37.59C ⁰	36.76C ⁰
Média	37.26°C	36.99°C

Fonte: Do autor.

Tabela 5 – Valores obtidos depois da calibração

Minuto	Pulso	Peito
1	39.88°C	40.22C ⁰
2	39.88°C	40.25C ⁰
3	40.73C ⁰	40.22C ⁰
4	40.45C ⁰	40.32C ⁰
5	40.42C ⁰	40.35C ⁰
Média	40.27°C	40.27°C

Fonte: Do autor.

Seguindo os resultados, percebe-se que não haveria necessidade de calibragem do *shield* no código fonte, pois o mesmo vem com valores bem maiores e que não está entre os valores da média normal e também não condiz com o estado do ser humano monitorado.

Já no teste com os valores *default*, o resultado foi satisfatório, pois a temperatura muda dependendo da hora do dia, se a pessoa recém fez atividades físicas ou até em estado emocional alterado.

Outra observação foi que, a diferença do valor do pulso para o peito, apresentou resultados curiosos, onde a temperatura do pulso, que fica longe do centro do corpo, apresentou valores maiores do que os recebidos do peito, que fica mais perto do centro.

O próximo teste foi sobre o sensor de aceleração, o acelerômetro, que reproduz a posição que o corpo se encontra no momento, de pé, deitado de bruços ou supino, deitado de lado esquerdo ou direito.

A resposta deste teste se deu muito satisfatória, obtendo resultados instantâneos e de altíssima precisão.

Os resultados adquiridos com implementação desta rede são muito satisfatórios, pois mesmo com algumas inconsistências, os dados são transmitidos e visualizados, sendo tratado na aplicação do *smartphone* os dados recebidos, caso receba algum pacote defeituoso, os dados na interface não são atualizados.

7 CONCLUSÃO

Ao longo deste projeto financiado pela instituição foram estudados e abordados vários tópicos e conceitos computacionais, como, redes de sensores sem fio, protocolos de comunicação sem fio, comunicação serial e linguagem de programação para dispositivos móveis com sistema operacional Android e programação desktop, que foram necessários para realização deste projeto.

Este trabalho atingiu os seus objetivos iniciais, onde o conhecimento adquirido durante a pesquisa foi utilizado para montar uma rede de sensores sem fio que monitora sinais vitais do corpo humano, visualizando os dados a partir de um *smartphone* podendo ser utilizado em diversas situações, como monitorar idosos em um asilo facilitando para os enfermeiros caso alguma situação de risco ocorra. Obviamente para uso médico outras questões precisam ser levadas em consideração na prática, como segurança, confiabilidade dos dados e qualidade de serviço. Este trabalho demonstrou a possibilidade de construir uma rede de sensores sem fio simples e de baixo custo para aplicação médica, o que proporcionou experimentação de diversos conceitos de software e hardware na prática, ampliando os conhecimentos adquiridos.

A visualização dos dados com o *smartphone* permite maior mobilidade e facilidade no monitoramento dos dados, sendo que muitas pessoas já possuem um dispositivo móvel com Bluetooth e acesso à Internet, melhorando com isso sua qualidade de vida.

A maior dificuldade encontrada para este trabalho foi a aquisição dos produtos, pois não há empresa no Brasil que venda os sensores utilizados, dificultando na hora da compra e na chegada dos produtos, devido à demora da entrega muitos problemas foram descobertos e com pouco tempo para resolvê-los.

A utilização de RSSF abre muitas portas para novas aplicações e soluções, como sugestão de trabalhos futuros pode-se empregar as seguintes extensões:

- a) adicionar novos sensores como ECG e EMG;
- b) possibilidade de não utilizar o arduino, somente os sensores e as antenas de forma que melhore a usabilidade e o conforto no seu uso;

- c) criar uma aplicação web para guardar os dados e poder consultá-los futuramente;

REFERÊNCIAS

- ABREU, Carlos; MENDES, Paulo. **WIRELESS SENSOR NETWORKS FOR BIOMEDICAL APPLICATIONS: Quality of Service, Admission Control and Lifetime-Extending Challenges**. Maio, 2013.
- AHAMED, S.s. Riaz. THE ROLE OF ZIGBEE TECHNOLOGY IN FUTURE DATA COMMUNICATION SYSTEM. **Journal Of Theoretical And Applied Information Technology**. Ramanathapuram, p. 129-135. abr. 2009.
- ANNUZZI, Joseph; DARCEY, Lauren; CONDER, Shane. **Introduction to Android™ Application Development**. 4. ed. New Jersey: Addison - Wesley, 2014. 627 p.
- AKYILDIZ, I. F. et al "A survey on sensor networks," *Communications Magazine, IEEE* , vol.40, no.8, pp. 102- 114, Agosto 2003.
- ARCE, Aldo Ivan Céspedes. **Redes de sensores sem fio na coleta de dados fisiológicos de bovinos para aplicações na zootecnia de precisão**. 2008. 86 f. Dissertação (Mestrado) - Curso de Zootecnia e Engenharia de Alimentos, Universidade de São Paulo, Pirassununga, 2008.
- Arduino Board Uno. Disponível em: <<http://arduino.cc/en/Main/ArduinoBoardUno> >. Acesso em: 23 nov. 2014.
- AVR TV. The Story of AVR. Disponível em: <<https://www.youtube.com/watch?v=HrydNwAxbcY>>. Acesso em: 23 nov. 2014.
- BARONTI, Paolo et al. Wireless Sensor Networks: A survey on the state of the art and the 802.15.4 and ZigBee standards. **Sciencedirect**. Pisa, p. 1655-1695. jan. 2007.
- BRANQUINHO, O. C.; REGGIANI, N.; ANDREOLLO, A. G. (2005). **Redes de comunicação de dados sem fio** - uma análise de desempenho. In: ISA SHOW SOUTH AMERICA Feira Sul-Americana e 5 Congresso Internacional de Automação, Sistemas e Instrumentação. São Paulo, Brasil: [s.n.].
- BURATTI, Chiara et al. **Sensor Networks with IEEE 802.15.4: Systems Distributed Processing, MAC, and Connectivity**. Springer, 2011. 269 p.
- BURNETTE, E. **Hello, Android: Introducing Goggle's mobile development platform**. 3. ed. Raleigh, N.C: Pragmatic Programmers, 2010.
- COLEMAN, David D.; WESTCOTT, David A. **CWNA – Certified Wireless Network Administrator: Study Guide**. Indianapolis: Wiley Publishing, 2006. 594 p.
- Cooking Hacks, e-Health Sensor Platform V2.0 for Arduino and Raspberry Pi. Disponível em: < <http://www.cooking-hacks.com/documentation/tutorials/ehealth-biometric-sensor-platform-arduino-raspberry-pi-medical>>. Acesso em: 10 abr. 2015.

CORREIA, L. et al. Arquiteturas para Redes de Sensores sem Fio. SBRC Tutorial 2004. Maio de 2004.

DESHMUKH, A. V. Microcontrollers – Theory And Applications. Noida, UP, India: Tata McGraw Hill, 2005, p4.

FALUDI, Robert. **Building Wireless Sensor Networks Download: A practical Guide to the ZigBee Mesh Networking Protocol**. O'Reilly, 2010. 321 p.

FREESCALE SEMICONDUCTOR (Org.). **MMA8452Q, 3-Axis, 12-bit/8-bit Digital Accelerometer**. 9.2 [s. l.]: Freescale Semiconductor, 2015. 49 p. Disponível em: <http://www.freescale.com/files/sensors/doc/data_sheet/MMA8452Q.pdf>. Acesso em: 07 jun. 2015.

GARGENTA, Marko. **Learning Android**. Gravenstein Highway North: O'reilly Media, 2011. 245 p.

JINDAL, S.; JINDAL, A.; GUPTA, N. (2005). Grouping wi-max, 3g and wi-fi for wireless broadband. In: The First IEEE and IFIP International Conference in Central Asia on Internet, 2005. Hyatt Regency Hotel, Bishkek, Kyrgyz Republic: [s.n.], p. 5.

JOGI, Sunil; CHOUDHARY, Manoj. **Ultra Wideband Demystified: Technologies, Applications, and System Design Considerations**. Aalborg: River Publisher, 2009. 205 p. Disponível em: < >. Acesso em: 18 set. 2014.

LECHETA, R. R. **Google Android: aprenda a criar aplicações para dispositivos móveis com o Android SDK**. 2. ed. São Paulo (SP): Novatec, 2010.

LEHTIMÄKI, Juhani. **SMASHING ANDROID UI: RESPONSIVE USER INTERFACES AND DESIGN PATTERNS FOR ANDROID PHONES AND TABLETS**. Chichester: John Wiley & Sons, 2013. 386 p.

LI, Changle; LI, Huan-bang; KOHNO, Ryuji. Performance Evaluation of IEEE 802.15.4 for Wireless Body Area Network (WBAN). In: COMMUNICATIONS WORKSHOPS, 2009. ICC WORKSHOPS 2009. IEEE INTERNATIONAL CONFERENCE ON, 1., 2009, Dresden. **Anais....** Dresden: IEEE, 2009. p. 1 - 5.

LOUREIRO, Antonio A.f. et al. Redes de Sensores Sem Fio. In: SIMPÓSIO BRASILEIRO DE REDES DE COMPUTADORES, 21., 2003, Minas Gerais. **Anais... .** Natal: Índice, 2003. p. 179 - 226.

MCGRATH, Michael J.; SCANAILL, Cliodhna Ní; NAFUS, Dawn. **Sensor Technologies: Healthcare, Wellness and Environmental Applications**. New York: Apress Media, 2013. 302 p.

MCROBERTS, Michael. **Beginning Arduino**. New York: Apress, 2010. 433 p.

MILENKOVIC, Aleksandar; OTTO, Chris; JOVANOVIĆ, Emil. Wireless sensor networks for personal health monitoring: Issues and an implementation. **Elsevier**. Huntsville, 6 March 2006. Computer Communications, p. 2521-2533.

MOREIRA, Jailson. Um Exemplo de Computação Ubíqua em Serviços de Saúde Orientados ao Utente. **2011. 77 f. Dissertação (Mestrado) - Curso de Engenharia Eletro Técnica de Computadores, Faculdade de Engenharia da Universidade do Porto, Porto, 2011.**

Net Applications. Disponível em: < <http://www.netmarketshare.com/operating-system-market-share.aspx?qprid=9&qpcustomb=1&qpct=4&qpsp=175&qpnp=12&qptimeframe=M&qpcustomd=br> >. Acesso em: 23 nov. 2014.

OLIVEIRA, Luís M. L.; RODRIGUES, Joel J. P. C.. Wireless Sensor Networks: a Survey on Environmental Monitoring. **Journal Of Communications**. [S.l.], p. 143-151. abr. 2011.

PAN, Meng-shiuan; TSENG, Yu-chee. **ZigBee Wireless Sensor Networks and Their Applications**. Hsin-chu: Springer Berlin Heidelberg, 2007. 20 p.

PATIL, Harsh Kupwade; SZYGENDA, Stephen A.. **Security for Wireless Sensor Networks using Identity-Based Cryptography**. Flórida: Crc Press, 2013. 204 p.

Rama. Disponível em: <http://pt.wikipedia.org/wiki/Ox%C3%ADmetro_de_pulso#mediaviewer/File:Saturometre_2.jpg>. Acesso em: 23 nov. 2014.

RAPPAPORT, Theodore S.. **Comunicação sem Fio: Princípios e práticas**. 2. ed. São Paulo: Pearson Education, 2009. 391 p. Daniel Vieira.

RUFINO, Nelson Murilo de Oliveira. **Segurança em Redes Sem Fio: Aprenda a proteger suas informações em ambientes Wi-Fi e Bluetooth**. São Paulo: Novatec, 2011.

SILVA, Leandro Honorato de Souza. Desenvolvimento de uma Rede de Sensores Sem Fio Utilizando ZigBee para Aplicações Diversas. **2011. 96 f. TCC (Graduação) - Curso de Engenharia da Computação, Universidade de Pernambuco, Recife, 2011.**

SIX, Jeff. **Application Security for the Android Platform**. Sebastopol: O'reilly Media, 2012. 97 p.

SOMANI, Nisha Ashok; PATEL, Yask. ZIGBEE: A LOW POWER WIRELESS TECHNOLOGY FOR INDUSTRIAL APPLICATIONS. **International Journal Of Control Theory And Computer Modelling**. Limda Whagodia, p. 27-33. mar. 2012.

SOUZA, David José. **Desbravando o PIC: Ampliado e Atualizado para PIC 16F628A**. 8ª ed. São Paulo, SP, Brasil: Érica, 2005.

SOUZA, Gabriel Scheffer de. **Rastreabilidade Bovina: Utilizando Plataforma ZIGBEE.** 2014. 82 f. TCC (Graduação) - Curso de Ciência da Computação, Universidade do Extremo Sul Catarinense, Criciúma, 2014. Disponível em: <<http://tcc.kironunesec.net.br/arquivos/trabalhos/394.pdf>>. Acesso em: 23 nov. 2014.

STEELE, J.; TO, N. **The Android developer's cookbook: building applications with the Android SDK.** 2. ed. Upper Saddle River, NJ: Addison-Wesley, 2011.

TANENBAUM, A. S. **Organização estruturada de computadores.** 4. ed. Rio de Janeiro, LTC, 2001.

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS. **ISBN 0-7381-4997-7 SH95552:** IEEE Standard for Information technology— Telecommunications and information exchange between systems— Local and metropolitan area networks— Specific requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs). New York: Ieee, 2006. 323 p.

WAGNER, Matthias et al. Android Based Body Area Network for the Evaluation of Medical Parameters. In: INTERNATIONAL WORKSHOP ON INTELLIGENT SOLUTIONS IN EMBEDDED SYSTEMS, 10., 2012, Klagenfurt. **Anais....** Klagenfurt: Wises, 2012. p. 33 - 38.

WARREN, John-david; ADAMS, Josh; MOLLE, Harald. **Arduino Robotics.** New York: Apress, 2011. 599 p.

Wi Fi Physical Layer. Disponível em: <<http://puspc.com/oamr/wp-content/uploads/2013/10/wireless-network-course-wifi-model.pdf>>. Acesso em: 06 novembro 2014.

YICK, Jennifer; MUKHERJEE, Biswanath; GHOSAL, Dipak. Wireless sensor network survey. **Elsevier.** Davis, 14 April 2008. Computer Networks, p. 2292-2330.

ZHAO, Feng; GUIBAS, Leonidas. **Wireless Sensor Networks: An Information Processing Approach.** Morgan Kaufmann, 2004. 376 p.

APÊNDICE(S)

Monitoramento de sinais de vitais utilizando RSSF

Thales Fernandes Maggi¹, Rogério Antônio Casagrande¹

¹Universidade do Extremo Sul Catarinense (UNESC)
Criciúma – SC – Brasil

thales_maggi@hotmail.com, roc@unesc.net

Abstract. *Wireless sensor networks is a technology with great potential because they can be used in any area, both in health and in the military, agriculture, school, etc. This paper presents a prototype of a wireless sensor network for monitoring vital signs using the ZigBee communication protocols stack. For data transmission ZigBee and Bluetooth has been used. Tests with the sensor nodes and sensors themselves were made, emphasizing the importance that was to build a generic solution for various applications, such as monitoring the elderly in a nursing home, or patients in hospitals or clinics.*

Resumo. *Redes de sensores sem fio é uma tecnologia com muito potencial, pois podem ser utilizados em qualquer área, tanto na saúde como no militar, setor agrícola, escolar, etc. Este trabalho apresenta um protótipo de uma rede de sensores sem fio para monitorar sinais vitais utilizando a pilha de protocolos de comunicação ZigBee. Para transmissão dos dados foi utilizado o ZigBee e o Bluetooth. Testes com os nós sensores e nos próprios sensores foram realizados, relevando a importância que era construir uma solução genérica para diversas aplicações, como monitorar idosos em um asilo, ou pacientes em hospitais ou ambulatórios.*

1. Introdução

Hoje em dia o mundo passa por vários problemas ligados diretamente a uma sociedade mais velha, com predominância maior de doenças crônicas e dificuldades associadas a mobilidade. O conceito da computação ubíqua (onipresença da informática no cotidiano das pessoas) traz inovações para aplicar a computação no dia-a-dia das pessoas de forma imperceptível. Para isso, utilizando da união de diversas tecnologias já existentes como, por exemplo, a tecnologia móvel e os sensores (MOREIRA, 2011).

Com o avanço do estudo e uso na área de microprocessadores, descoberta de novos materiais de sensoriamento, micro sistemas eletromecânicos, do inglês Micro Electro-Mechanical Systems (MEMS), e comunicação sem fio entre dispositivos pequenos tem estimulado muitos pesquisadores e empresas no desenvolvimento e uso destes sensores “inteligentes” em áreas ligadas a processos físicos, químicos, biológicos, dentre outros (LOUREIRO et al, 2003).

Segundo Honorato (2011) os sensores são compostos normalmente de processador, modem wireless, memória e fonte de alimentação. Os nós ou sensores de uma rede são limitados em relação ao consumo de energia, capacidade de processamento computacional e memória, enquanto que os nós das redes ad hoc (é uma rede sem fio em que todos dispositivos comunicam entre si, sem um ponto para se conectarem como uma rede wireless convencional onde todos dispositivos se conectam a um roteador) normalmente são mais robustos. Estes sensores quando utilizados em uma rede, formam uma rede de sensores sem

sem fio, ou RSSF, e, normalmente elas possuem um grande número de nós. Cada nó é equipado com uma variedade de sensores, tais como acústico, sísmico, infravermelho, câmera de vídeo, temperatura e pressão. Esses nós podem ser organizados em grupos onde pelo menos um dos sensores deve ser capaz de detectar um evento na região, processá-lo e tomar uma decisão se deve fazer ou não uma difusão (broadcast) do resultado para outros nós.

As RSSFs podem ser vistas como um tipo especial de Rede Móvel ad hoc, do inglês Mobile Ad hoc Network (MANET). No entanto, as redes MANET tem como função básica prover um suporte a comunicação entre esses elementos computacionais, que individualmente, podem estar executando tarefas distintas. Por outro lado, RSSF tendem a executar uma função colaborativa onde os elementos (sensores) provêm dados, que são processados (ou consumidos) por nós especiais chamados de coordenadores (sink nodes) (LOUREIRO et al, 2003).

RSSF utilizam dispositivos com protocolos de comunicação, que, foi definido pelo Instituto de Engenheiros Eletricistas e Eletrônicos, conhecido como IEEE. Eles definiram um padrão para este tipo de rede, o IEEE 802.15.4, um padrão que define um conjunto de protocolos de comunicação para redes de sensor sem fio, que são de custo energético muito baixo e baixa taxa de transmissão. E com esta característica, uma empresa trabalhou em cima de alguns protocolos deste padrão, e criaram o ZigBee, que, tem sido amplamente utilizado desde então em várias aplicações que necessitam de redes de sensores sem fio (HONORATO, 2011).

Nesta arquitetura ZigBee, o processamento dos dados primário é feito por nós sensores, incluindo o processamento do sinal fisiológico no microcontrolador dos nós. O processamento de dados secundários é executado no smartphone. Isso inclui a representação de dados, filtragem de dados, interface gráfica e sincronização de dados (WAGNER et al, 2012, tradução nossa).

Os dados que os nós sensores captarem são passados pela rede de sensores até um nó coordenador, e dele para um computador, e deste para um dispositivo Android. Os dados são transmitidos para o Smartphone via comunicação Bluetooth, que, conforme Loureiro (2003), é uma tecnologia de baixo custo para conectividade sem fio de dispositivos eletrônicos mundialmente utilizado. Inicialmente pensado como um padrão de substituição de cabos para comunicação entre dispositivos eletrônicos, ele se tornou um consenso na indústria como uma forma de interligar dispositivos como telefones celulares, notebooks, computadores de mesa e impressoras.

Conforme mostrado acima, a ideia seria fazer a aquisição de sinais vitais como temperatura e batimentos cardíacos de um ser humano via smartphone, utilizando redes de sensores sem fio transmitindo estes dados para um dispositivo móvel. A comunicação entre os sensores até o nó coordenador é realizada por um protocolo de comunicação especializado em RSSF, e depois transmitido para o dispositivo via Bluetooth em um ambiente local de curto alcance.

2. Justificativa

Tecnologias de rede de sensores sem fio tem o potencial de mudar a maneira de viver com muitas aplicações em entretenimento, viagens, varejo, indústria, medicina, cuidados de pessoas dependentes, e gestão de emergência e muitas outras áreas. Um dos maiores desafios do mundo nas últimas décadas tem sido o aumento da população idosa contínua nos países desenvolvidos. O site Population Reference Bureau prevê que nos próximos 20 anos, a

população com mais 65 anos serão nos países desenvolvidos quase 20% da população total. (ALEMDAR; ERSOY; 2010, tradução nossa).

Conforme Akyildiz et al (2003), as tecnologias de comunicação sem fio e aparelhos eletrônicos digitais permitiram o desenvolvimento de baixo custo, baixo consumo de energia, nós sensores multifuncionais que são pequenos em tamanho e se comunicam sem fio em curtas distâncias. Estes minúsculos nós sensores, que consistem em sensoriamento, processamento de dados e componentes de comunicação, alavancam a ideia de redes de sensores com base na colaboração de um grande número de nós. Uma RSSF adaptada para uma tarefa específica. Neste caso, como é uma rede de sensores em seres humanos, que vem do inglês Body Area Network (BAN), é geralmente chamado de wireless BAN ou WBAN.

O ZigBee tem suporte de transmissão via Bluetooth, que também opera em transmissões de dados em curto alcance para dispositivos eletrônicos que tem suporte de conexão Bluetooth. Os dados dos sensores são transmitidos para um nó central, o nó coordenador e depois são retransmitidos para um dispositivo móvel com Sistema Operacional (SO) móvel Android.

Esta tecnologia agregada a saúde trás muitos pontos positivos como conforto, mobilidade, baixo custo, monitoramento de vários pacientes em um dispositivo único e prever futuros problemas de saúde momentâneos, pois caso a leitura dos sinais vitais caia ou aumente subitamente será alertado o paciente com uma notificação no smartphone para que vá a algum centro médico. O sistema operacional móvel Android usa linguagem Java que é mundialmente conhecida e utilizada, e, sua plataforma de desenvolvimento tem uma interface intuitiva e fácil de programar o que torna o desenvolvimento de uma aplicação móvel mais ágil e simples.

3. Metodologia

Para a realização deste trabalho, foram efetuadas pesquisas sobre formas de comunicação sem fio que pudessem ser utilizadas para transferir sinais vitais de seres humanos até um computador e uma forma de enviar estes dados recebidos para um smartphone, que atendesse o critério de utilizar pouca energia, pois cada sensor não teria uma fonte de energia fixa, já que as pessoas podem estar em movimento ou não, e que trabalhasse com vários nós.

3.1. Definição do Hardware

Após pesquisas sobre algumas redes e reuniões com o orientador, foi decidido o uso do protocolo zigbee, devido ao baixo consumo de energia e era de rápida implementação. Com a ajuda do laboratório de projeto de pesquisa do curso, que financiou todo o projeto, foram então escolhidas as antenas e demais equipamentos necessários para manipular os dados adquiridos do corpo humano e que seriam enviados de antena a antena até chegar ao nó coordenador que está conectado ao computador. Através de leitura serial, este iria ler os dados e após o computador estar pareado com o smartphone via bluetooth, iria repassar estes dados ao dispositivo para se ter uma forma de visualização e acesso aos dados coletados diferente.

Foram utilizadas para transmissão dos dados dos sensores, três antenas xBee Series 2, dois arduinos, dois shields e-Health, um sensor de temperatura e um acelerômetro.

Conforme a figura 1, podemos ver todos os hardwares utilizados e as formas de comunicação. Os nós sensores são nós roteadores e o nó conectado ao computador é o coordenador, que serve como gateway, onde poderá utilizar os dados para enviar á qualquer lugar.



Figura 1. Cenário com todos os hardwares e comunicação utilizados

Na figura 2 encontra-se a pilha de *shields* utilizados e placas utilizadas.

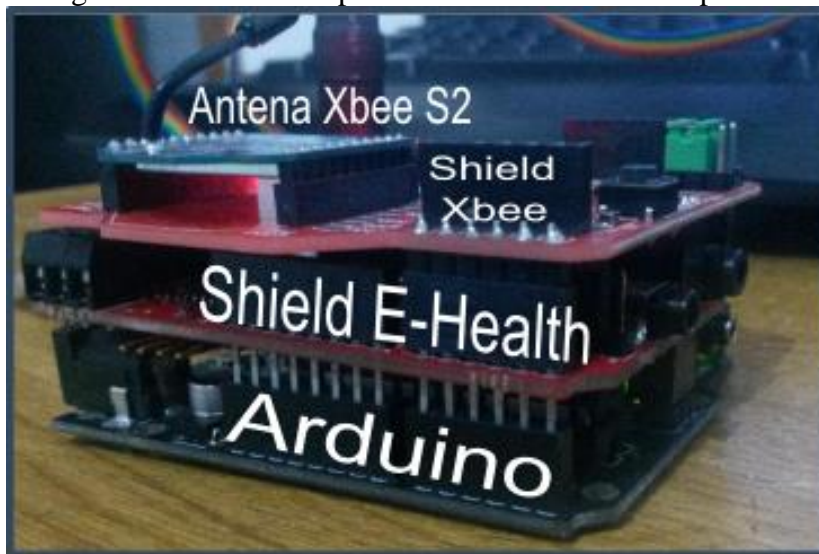


Figura 2. *Shields* montado

3.2. Integração Hardware x Software

Após obter todos os equipamentos, foi necessário primeiramente instalar o software x-ctu da digi international, que faz a leitura e alteração do firmware das antenas, e o driver ftdi que é encontrado dentro da pasta do arduino ide. Utilizando o xbee explorer usb com a antena xbee s2, conecta-se ao computador, e, utilizando o software x-ctu, no botão “*discover devices*” se consegue procurar utilizando a porta com conectada e com as configurações padrão adicionar a antena na lista.

Depois de adicionada foi atualizado o firmware de acordo com cada antena, conforme segue a tabela 1. Na coluna “função” a letra “c” é para *coordinator* e “r” para *router*, esta configuração que determina o modelo de sua rede, se será modo api ou at. O modo api é baseado em frames de dados, mais complexo para se trabalhar, normalmente utilizados quando se há muitos nós. Como se trata de uma aplicação protótipo foi utilizado o modo transparente, nó roteador manda pacotes para o próximo nó, até achar o coordenador. Neste modo a configuração das antenas é ágil e atende a necessidade deste trabalho.

Tabela 1 – Configuração das antenas Xbee

Nó identificador	Modelo Xbee	Função	Versão	PAN ID	Baudrate
ZB_COORD	XB24-ZB	C. AT	20A7	2001	115200
ZB_TEMP	XB24-ZB	R. AT	22A7	2001	115200
ZB_POS	XB24-ZB	R. AT	22A7	2001	115200

Não há antenas *end device* pois todos os nós necessitam se comunicar com todos pois como eles estarão em humanos, eles podem se mover, e, um *end device* não pode interferir no caminho de um nó roteador até o nó coordenador.

Então para a leitura dos sensores foram programadas as plataformas arduino, utilizando as funções disponíveis pela biblioteca e-health, estas funções fazem a conversão dos dados elétricos para dados fisiológicos e são enviadas pela porta serial, o shield xbee projeta estes dados de acordo com a configuração do seu *firmware*.

3.3 Aplicação

Após estes dados serem distribuídos até o nó coordenador que fica no computador, foi então desenvolvido uma aplicação desktop que faz a leitura da porta serial e repassa para o dispositivo móvel.

Esta aplicação foi desenvolvida utilizando a linguagem Java no ambiente de desenvolvimento NetBeans 8.0.2, em conjunto com as bibliotecas RXTXcomm para comunicação serial RS232 e a bluecove-2.1.1 para comunicação Bluetooth.

Para a visualização dos dados é necessário somente indicar a porta COM utilizada pelo *Xbee Explorer* USB, como mostra a figura 3.

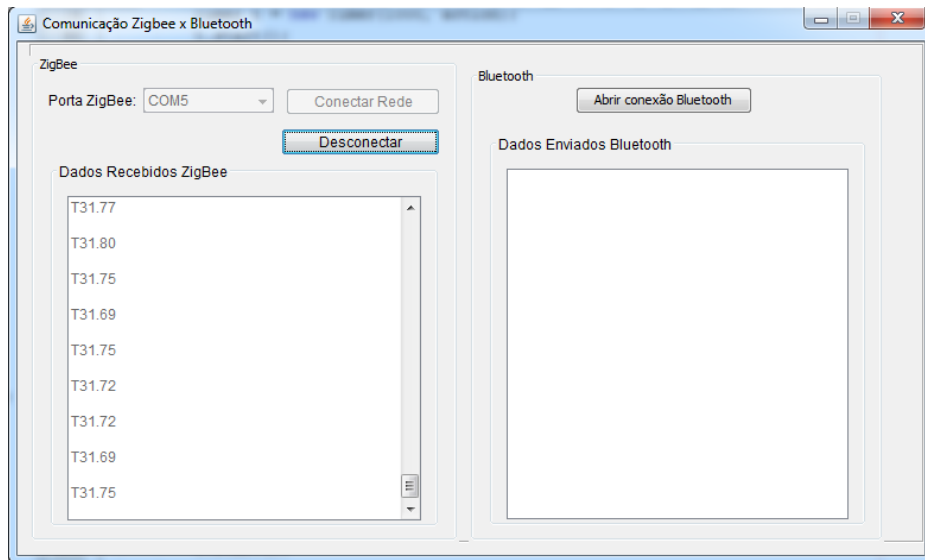
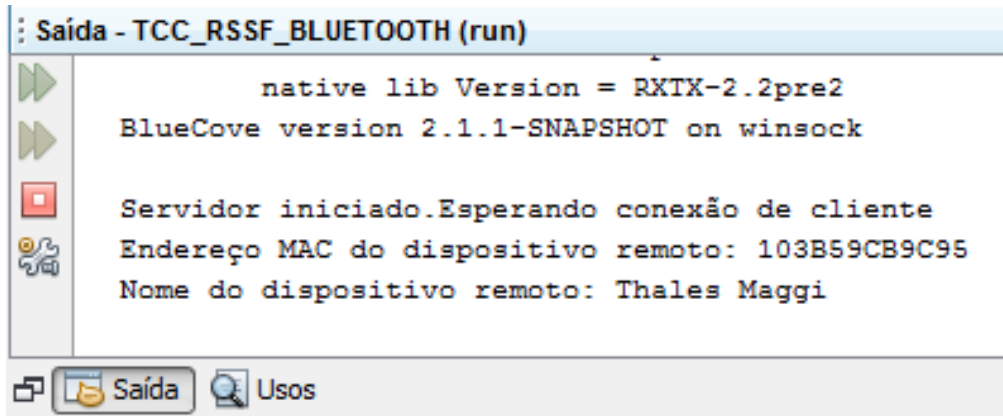


Figura 3 – Aplicação desktop para visualização da transmissão dos pacotes

A aplicação feita para o SO móvel Android foi utilizada a linguagem Java também, mas em outro ambiente de desenvolvimento, o Android Studio.

Para a comunicação de dados se estabelecer entre o computador e o dispositivo móvel, é necessário aplicar o conceito de cliente-servidor, muito semelhante à comunicação TCP/IP, onde o servidor é o computador Desktop e o cliente o smartphone. Clicando no botão de se

comunicar via Bluetooth, a aplicação Desktop irá iniciar o serviço, esperando que algum dispositivo com o código de serviço idêntico se conecte, conforme a figura 4.



```
native lib Version = RXTX-2.2pre2
BlueCove version 2.1.1-SNAPSHOT on winsock
Servidor iniciado. Esperando conexão de cliente
Endereço MAC do dispositivo remoto: 103B59CB9C95
Nome do dispositivo remoto: Thales Maggi
```

Figura 4 – Aplicação desktop para visualização da transmissão dos pacotes

Para isto, o dispositivo móvel e o computador já devem estar pareados, então a aplicação é iniciada e conectada a aplicação Desktop, já mostrando os dados obtidos da RSSF. Segue um exemplo na figura 5.

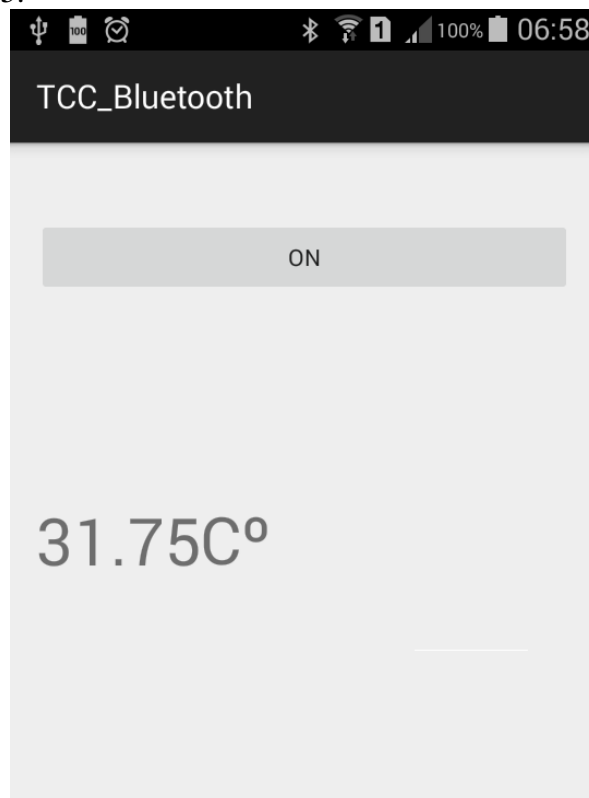


Figura 5 – Aplicação móvel para leitura dos dados via Bluetooth

4. Testes Realizados

Os primeiros testes foram com os nós sensores, alterando suas localizações e testando seus limites.

O primeiro teste foi relativo ao alcance e como eles se comportam em movimento. Em distância, lugares com muitas paredes interferindo perde-se o sinal, entre portas ou alguns objetos obstruindo o caminho afetava o sinal, demorando na entrega dos pacotes e as vezes com perdas.

Se movendo com as antenas em funcionamento, quando ele detecta qualquer mudança ele vai guardando as informações num *buffer* e após conseguir se conectar com o próximo nó envia o que estava guardado.

O terceiro teste foi averiguado se os valores obtidos a partir do sensor de temperatura são suficientes para medir o corpo humano. Para este teste, os dados de temperatura que o sensor está captando, devem ser o mais próximo do valor base dito como normal em algumas partes do corpo humano. A temperatura central cutânea média do ser humano, medido normalmente a partir da axila, varia entre 35,5°C e 37.5°C, ou seja, cada parte do corpo tem suas variâncias de temperatura.

Então foram feitos 2 testes em 2 pontos do corpo humano, primeiramente medindo a temperatura durante 5 minutos em cada ponto, obtendo um valor por minuto e feito uma média aritmética para comparar se está dentro do valor médio, antes da calibração, e depois fazer os mesmos passos só com os valores da biblioteca calibrados.

Os dois pontos são o peito e pulso, foram escolhidos estes pontos pois oferecem maior facilidade na captação e na usabilidade do sensor. Os resultados estão nas tabelas 2 e 3 a seguir. Os valores foram obtidos do próprio autor.

Tabela 2 – Valores obtidos antes da calibração

Minuto	Pulso	Peito
1	36.85°C	37.31C°
2	37.40°C	37.01C°
3	37.40C°	37.01C°
4	37.04C°	36.88C°
5	37.59C°	36.76C°
Média	37.26°C	36.99°C

Fonte: Do autor.

Tabela 3 – Valores obtidos depois da calibração

Minuto	Pulso	Peito
1	39.88°C	40.22C°
2	39.88°C	40.25C°
3	40.73C°	40.22C°
4	40.45C°	40.32C°
5	40.42C°	40.35C°
Média	40.27°C	40.27°C

Seguindo os resultados, percebe-se que não haveria necessidade de calibragem do *shield* no código fonte, pois o mesmo vem com valores bem maiores e que não está entre os valores da média normal e também não condiz com o estado do ser humano monitorado.

Já no teste com os valores *default*, o resultado foi satisfatório, pois a temperatura muda dependendo da hora do dia, se a pessoa recém fez atividades físicas ou até em estado emocional alterado.

Outra observação foi que, a diferença do valor do pulso para o peito, apresentou resultados curiosos, onde a temperatura do pulso, que fica longe do centro do corpo, apresentou valores maiores do que os recebidos do peito, que fica mais perto do centro.

O próximo teste foi sobre o sensor de aceleração, o acelerômetro, que reproduz a posição que o corpo se encontra no momento, de pé, deitado de bruços ou supino, deitado de lado esquerdo ou direito.

A resposta deste teste se deu muito satisfatória, obtendo resultados instantâneos e de altíssima precisão.

Os resultados adquiridos com implementação desta rede são muito satisfatórios, pois mesmo com algumas inconsistências, os dados são transmitidos e visualizados, sendo tratado na aplicação do *smartphone* os dados recebidos, caso receba algum pacote defeituoso, os dados na interface não são atualizados.

4. Conclusão

Ao longo deste projeto financiado pela instituição foram estudados e abordados vários tópicos e conceitos computacionais, como, redes de sensores sem fio, protocolos de comunicação sem fio, comunicação serial e linguagem de programação para dispositivos móveis com sistema operacional Android e programação desktop, que foram necessários para realização deste projeto.

Este trabalho atingiu os seus objetivos iniciais, onde o conhecimento adquirido durante a pesquisa foi utilizado para montar uma rede de sensores sem fio que monitora sinais vitais do corpo humano, visualizando os dados a partir de um *smartphone* podendo ser utilizado em diversas situações, como monitorar idosos em um asilo facilitando para os enfermeiros caso alguma situação de risco ocorra. Obviamente para uso médico outras questões precisam ser levadas em consideração na prática, como segurança, confiabilidade dos dados e qualidade de serviço. Este trabalho demonstrou a possibilidade de construir uma rede de sensores sem fio simples e de baixo custo para aplicação médica, o que proporcionou experimentação de diversos conceitos de software e hardware na prática, ampliando os conhecimentos adquiridos.

A visualização dos dados com o *smartphone* permite maior mobilidade e facilidade no monitoramento dos dados, sendo que muitas pessoas já possuem um dispositivo móvel com Bluetooth e acesso à Internet, melhorando com isso sua qualidade de vida.

A maior dificuldade encontrada para este trabalho foi a aquisição dos produtos, pois não há empresa no Brasil que venda os sensores utilizados, dificultando na hora da compra e na chegada dos produtos, devido à demora da entrega muitos problemas foram descobertos e com pouco tempo para resolvê-los.

A utilização de RSSF abre muitas portas para novas aplicações e soluções, como sugestão de trabalhos futuros pode-se empregar as seguintes extensões:

- d) adicionar novos sensores como ECG e EMG;
- e) possibilidade de não utilizar o arduino, somente os sensores e as antenas de forma que melhore a usabilidade e o conforto no seu uso;

f) criar uma aplicação web para guardar os dados e poder consultá-los futuramente;

Referências

Akyildiz, Ian F. et al. "A Survey on Sensor Networks". 2003. Tradução nossa. Disponível em: <<http://www.ece.gatech.edu/research/labs/bwn/papers/2002/j4.pdf>>. Acesso em: 03 maio. 2014.

ALEMDAR, Hande; ERSOY, Cem. Wireless sensor networks for healthcare: A survey. Elsevier. Istanbul, 11 May 2010. Computer Networks, p. 2688-2710.

HONORATO, Leandro. Desenvolvimento de uma Rede de Sensores Sem Fio Utilizando ZigBee para Aplicações Diversas. 2011. 96 f. TCC (Graduação) - Curso de Engenharia da Computação, Universidade de Pernambuco, Recife, 2011.

LOUREIRO, Antonio A.f. et al. Redes de Sensores Sem Fio. In: SIMPÓSIO BRASILEIRO DE REDES DE COMPUTADORES, 21., 2003, Minas Gerais. Anais... . Natal: Índice, 2003. p. 179 - 226.

MOREIRA, Jaílson. Um Exemplo de Computação Ubíqua em Serviços de Saúde Orientados ao Utente. 2011. 77 f. Dissertação (Mestrado) - Curso de Engenharia Eletro Técnica de Computadores, Faculdade de Engenharia da Universidade do Porto, Porto, 2011.

WAGNER, Matthias et al. Android Based Body Area Network for the Evaluation of Medical Parameters. In: INTERNATIONAL WORKSHOP ON INTELLIGENT SOLUTIONS IN EMBEDDED SYSTEMS, 10., 2012, Pilsen. Anais... . Pilsen: Wisen, 2013. p. 33 - 38.